

# **ApplinX 9.7.0.0000 API**

## **Specification**

(c) Copyright 2014 Software AG. All rights reserved. Use is subject to license terms.

---

**Package**

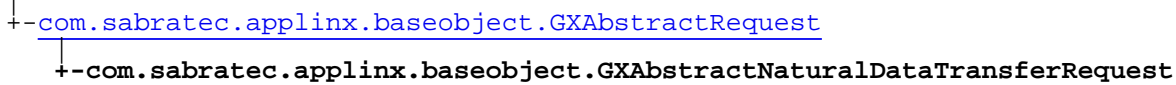
# **com.sabratec.applinx.baseobject**

Contains classes and interfaces which are the core of the ABO API.

## com.sabratec.applinx.baseobject

### Class GXAbstractNaturalDataTransferRequest

java.lang.Object



#### All Implemented Interfaces:

Cloneable, java.io.Externalizable

#### Direct Known Subclasses:

[GXNaturalDataTransferUploadRequest](#), [GXNaturalDataTransferDownloadRequest](#)

public abstract class **GXAbstractNaturalDataTransferRequest**  
 extends [GXAbstractRequest](#)

Common request for Natural Data Transfer messages.

### Constructor Summary

public	<a href="#">GXAbstractNaturalDataTransferRequest()</a>
--------	--

### Method Summary

int	<a href="#">getFormatType()</a> Return the transfer data type.
void	<a href="#">readExternal()</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal()</a> (java.io.ObjectOutput out)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

### Constructors

#### GXAbstractNaturalDataTransferRequest

public **GXAbstractNaturalDataTransferRequest()**

(continued on next page)

(continued from last page)

## Methods

### **getFormatType**

```
public int getFormatType()
```

Return the transfer data type.

---

### **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject Class GXAbstractNaturalDataTransferResponse

java.lang.Object

└─com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferResponse

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXNaturalDataTransferUploadResponse](#), [GXNaturalDataTransferDownloadResponse](#)

public abstract class **GXAbstractNaturalDataTransferResponse**  
extends Object  
implements java.io.Serializable

Response data message for Natural Data Transfer download/upload requests.

## Constructor Summary

public	<a href="#">GXAbstractNaturalDataTransferResponse</a> ( )
--------	---

## Method Summary

int	<a href="#">getNumProcessedRecords</a> ( ) The number of records that were already processed.
boolean	<a href="#">isComplete</a> ( ) Whether the transfer has been completed or not.
void	<a href="#">setComplete</a> (boolean complete) Set whether the Natural data transfer session has been completed or not.
void	<a href="#">setNumProcessedRecords</a> (int numProcessedRecords) Set the number of processed record.

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXAbstractNaturalDataTransferResponse

public **GXAbstractNaturalDataTransferResponse**( )

## Methods

(continued from last page)

## **isComplete**

```
public boolean isComplete()
```

Whether the transfer has been completed or not.

---

## **getNumProcessedRecords**

```
public int getNumProcessedRecords()
```

The number of records that were already processed.

---

## **setNumProcessedRecords**

```
public void setNumProcessedRecords(int numProcessedRecords)
```

Set the number of processed record.

---

## **setComplete**

```
public void setComplete(boolean complete)
```

Set whether the Natural data transfer session has been completed or not.

## com.sabratec.applinx.baseobject Class GXAbstractRequest

java.lang.Object

└─com.sabratec.applinx.baseobject.GXAbstractRequest

### All Implemented Interfaces:

Cloneable, java.io.Externalizable

### Direct Known Subclasses:

[GXSetVariablesRequest](#), [GXGetScreenRequest](#), [GXBinaryRequest](#), [GXAbstractSetToHostRequest](#),  
[GXAbstractSessionConnectionRequest](#), [GXAbstractNaturalDataTransferRequest](#)

public abstract class **GXAbstractRequest**  
extends Object  
implements java.io.Externalizable, Cloneable

Common settings for all requests.

## Method Summary

void	<a href="#">addVariable</a> ( <a href="#">GXIVariable</a> var)
void	<a href="#">addVariable</a> (String varName, String varValue)
void	<a href="#">addVariableArray</a> (String varName, String[] varValues)
Object	<a href="#">clone</a> ()
boolean	<a href="#">equals</a> (Object obj)
Hashtable	<a href="#">getVariables</a> ()
int	<a href="#">hashCode</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setVariables</a> (Hashtable variables)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Methods inherited from interface java.io.Externalizable

readExternal, writeExternal

## Methods

### addVariable

```
public void addVariable(GXIVariable var)
```

**Parameters:**

var - a variable object

**See Also:**

[GXBaseObjectConstants](#)

---

### addVariable

```
public void addVariable(String varName,  
    String varValue)
```

**Parameters:**

varName - the variable name

varValue - the variable value

**See Also:**

[GXBaseObjectConstants](#)

---

### addVariableArray

```
public void addVariableArray(String varName,  
    String[] varValues)
```

**Parameters:**

varName - the variable name

varValues - a variable values array

---

### getVariables

```
public Hashtable getVariables()
```

**Returns:**

a variables Hashtable of: variable name(key) -> [GXIVariable](#)(value)

---

### setVariables

```
public void setVariables(Hashtable variables)
```

---

(continued from last page)

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)
    throws java.io.IOException,
           ClassNotFoundException
```

---

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

---

## **hashCode**

```
public int hashCode()
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **clone**

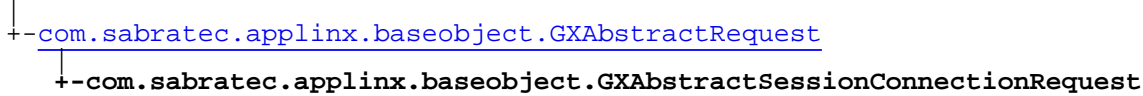
```
public Object clone()
```

---

## com.sabratec.applinx.baseobject

### Class GXAbstractSessionConnectionRequest

java.lang.Object



#### All Implemented Interfaces:

Cloneable, java.io.Externalizable

#### Direct Known Subclasses:

[GXEndSessionRequest](#), [GXCreateSessionRequest](#), [GXAttachToSessionRequest](#)

public class **GXAbstractSessionConnectionRequest**  
 extends [GXAbstractRequest](#)

Abstract connection request to a specific session.

### Constructor Summary

public	<a href="#">GXAbstractSessionConnectionRequest()</a>
--------	--

### Method Summary

String	<a href="#">getApplicationName()</a>
String	<a href="#">getClientId()</a>
GXIApplicationServerContext	<a href="#">getServerContext()</a>
String	<a href="#">getServerURL()</a>
String	<a href="#">getSessionID()</a>
String	<a href="#">getSessionPassword()</a>
void	<a href="#">readExternal()</a> (java.io.ObjectInput in)
void	<a href="#">setApplicationName()</a> (String applicationName)
void	<a href="#">setClientId()</a> (String clientId) Set the client unique id to use.
void	<a href="#">setServerContext()</a> (GXIApplicationServerContext serverContext) Sets the ApplinX server context for the session.
void	<a href="#">setServerURL()</a> (String serverURL) Sets the URL of ApplinX Server to which the <a href="#">GXIClientBaseObject</a> will be connected.

void	<a href="#"><u>setSessionID</u></a> (String sessionID)
void	<a href="#"><u>setSessionPassword</u></a> (String sessionPassword)
void	<a href="#"><u>writeExternal</u></a> (java.io.ObjectOutput out)

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

## Constructors

### GXAbstractSessionConnectionRequest

```
public GXAbstractSessionConnectionRequest()
```

## Methods

### getSessionID

```
public String getSessionID()
```

**Returns:**

the session ID.

### setSessionID

```
public void setSessionID(String sessionID)
```

**Parameters:**

sessionID - the ID of the session to which the [GXIClientBaseObject](#) will be attached.

### setApplicationName

```
public void setApplicationName(String applicationName)
```

**Parameters:**

(continued from last page)

applicationName - the name of the application on ApplinX Server to which the session should be connected.

---

## getApplicationName

```
public String getApplicationName()
```

### Returns:

the name of the ApplinX application.

---

## setSessionPassword

```
public void setSessionPassword(String sessionPassword)
```

### Parameters:

sessionPassword - the session password to use for the connection. In [GXCreateSessionRequest](#), this password is optional. When choosing to set the password it must also be set in any [GXAttachToSessionRequest](#).

---

## getSessionPassword

```
public String getSessionPassword()
```

### Returns:

the session password.

---

## setServerURL

```
public void setServerURL(String serverURL)
```

Sets the URL of ApplinX Server to which the [GXIClientBaseObject](#) will be connected.

### Parameters:

serverURL - the URL of ApplinX server.

---

## getServerURL

```
public String getServerURL()
```

### Returns:

the ApplinX Server URL.

---

## setServerContext

```
public void setServerContext(GXIApplicationServerContext serverContext)
```

Sets the ApplinX server context for the session. Used in "Local mode".

### Parameters:

serverContext - the context to set.

---

(continued from last page)

## getServerContext

```
public GXIApplicationServerContext getServerContext()
```

### Returns:

the server context used for the session.

---

## setClientId

```
public void setClientId(String clientId)
```

Set the client unique id to use. This option is for advanced users only.

### Parameters:

`clientId` - the id of all this client's connections.

---

## getClientId

```
public String getClientId()
```

### Returns:

the client unique id (all connections from same client will have same id).

---

## readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

## writeExternal

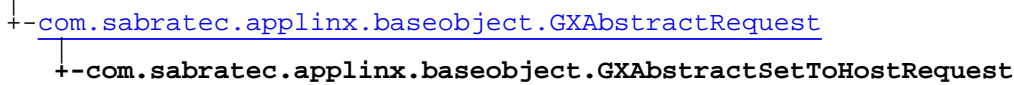
```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

## com.sabratec.applinx.baseobject

### Class GXAbstractSetToHostRequest

java.lang.Object



#### All Implemented Interfaces:

Cloneable, java.io.Externalizable

#### Direct Known Subclasses:

[GXTableRequest](#), [GXSendKeysRequest](#), [GXPathRequest](#), [GXNavigateRequest](#)

public abstract class **GXAbstractSetToHostRequest**

extends [GXAbstractRequest](#)

Abstract class for SendKeys and Execute Path requests.

### Method Summary

void	<a href="#">addInputField</a> ( <a href="#">GXIField</a> field, String fieldValue)
void	<a href="#">addInputField</a> ( <a href="#">GXInputField</a> field)
void	<a href="#">addInputField</a> ( <a href="#">GXPosition</a> position, String fieldValue)
void	<a href="#">addInputField</a> (String fieldName, <a href="#">GXPosition</a> position, String fieldValue)
void	<a href="#">addInputField</a> (String fieldName, int fieldIndex, String fieldValue)
void	<a href="#">addInputField</a> (String fieldName, String fieldValue)
void	<a href="#">addInputFields</a> ( <a href="#">GXInputField[]</a> fields)
void	<a href="#">changeInputField</a> ( <a href="#">GXInputField</a> field)
<a href="#">GXInputField[]</a>	<a href="#">getInputFields</a> ()
int	<a href="#">getSeqScreenNumber</a> () Get the sequence screen number which the send request should send against.
boolean	<a href="#">isWindowRelativePositioning</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">removeInputField</a> ( <a href="#">GXInputField</a> field)

void	<a href="#"><u>setSeqScreenNumber</u></a> (int seqScreenNumber) Defines the screen sequence number in which the send request should be send against If the sequence doesn't match the current screen sequence in the host, an exception is thrown during the send.
void	<a href="#"><u>setWindowRelativePositioning</u></a> (boolean positionsIgnoreWindow)
String	<a href="#"><u>toString</u></a> ()
void	<a href="#"><u>writeExternal</u></a> (java.io.ObjectOutput out)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

## Methods

### getInputFields

```
public GXInputField[] getInputFields()
```

### addInputField

```
public void addInputField(GXInputField field)
```

### addInputField

```
public void addInputField(GXIField field,  
String fieldValue)
```

### addInputField

```
public void addInputField(String fieldName,  
String fieldValue)
```

(continued from last page)

---

## addInputField

```
public void addInputField(String fieldName,  
    int fieldIndex,  
    String fieldValue)
```

---

## addInputField

```
public void addInputField(String fieldName,  
    GXPosition position,  
    String fieldValue)
```

---

## addInputField

```
public void addInputField(GXPosition position,  
    String fieldValue)
```

---

## removeInputField

```
public void removeInputField(GXInputField field)
```

---

## changeInputField

```
public void changeInputField(GXInputField field)
```

---

## addInputFields

```
public void addInputFields(GXInputField\[\] fields)
```

---

## setWindowRelativePositioning

```
public void setWindowRelativePositioning(boolean positionsIgnoreWindow)
```

---

## isWindowRelativePositioning

```
public boolean isWindowRelativePositioning()
```

---

## readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
    ClassNotFoundException
```

(continued from last page)

---

## writeExternal

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

---

## toString

```
public String toString()
```

---

## getSeqScreenNumber

```
public int getSeqScreenNumber()
```

Get the sequence screen number which the send request should send against. If the sequence doesn't match the current screen sequence in the host, an exception is thrown during the send.

**Returns:**

screen sequence number

---

## setSeqScreenNumber

```
public void setSeqScreenNumber(int seqScreenNumber)
```

Defines the screen sequence number in which the send request should be send against If the sequence doesn't match the current screen sequence in the host, an exception is thrown during the send.

**Parameters:**

seqScreenNumber - a screen sequence number which the data should be send against

---

## com.sabratec.applinx.baseobject Class GXAbstractWaitCondition

java.lang.Object

└─com.sabratec.applinx.baseobject.GXAbstractWaitCondition

All Implemented Interfaces:

[GXWaitCondition](#)

Direct Known Subclasses:

[GXWaitHostQuiet](#), [GXAbstractWaitString](#), [GXAbstractWaitScreen](#), [GXAbstractWaitCursor](#)

public abstract class **GXAbstractWaitCondition**

extends Object

implements [GXWaitCondition](#)

Abstract class for all wait conditions.

### Method Summary

long	<a href="#">getFlicker()</a>
long	<a href="#">getTimeout()</a>
void	<a href="#">setFlicker</a> (long flicker)
void	<a href="#">setTimeout</a> (long timeout)

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Methods

#### getTimeout

public long **getTimeout**()

#### setTimeout

public void **setTimeout**(long timeout)

#### getFlicker

public long **getFlicker**()

(continued from last page)

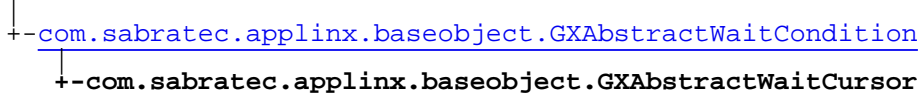
---

## **setFlicker**

```
public void setFlicker(long flicker)
```

## com.sabratec.applinx.baseobject Class GXAbstractWaitCursor

java.lang.Object



All Implemented Interfaces:

[GXWaitCondition](#)

Direct Known Subclasses:

[GXWaitWhileCursor](#), [GXWaitForCursor](#)

public abstract class **GXAbstractWaitCursor**  
extends [GXAbstractWaitCondition](#)

Abstract class for wait conditions involving the cursor.

### Method Summary

<a href="#">GXPosition</a>	<a href="#">getLowerRightPosition()</a>
<a href="#">GXPosition</a>	<a href="#">getUpperLeftPosition()</a>
void	<a href="#">setLowerRightPosition(<a href="#">GXPosition</a> lowerRight)</a>
void	<a href="#">setUpperLeftPosition(<a href="#">GXPosition</a> upperLeft)</a>

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Methods

#### getUpperLeftPosition

public [GXPosition](#) [getUpperLeftPosition\(\)](#)

#### getLowerRightPosition

public [GXPosition](#) [getLowerRightPosition\(\)](#)

---

## **setUpperLeftPosition**

```
public void setUpperLeftPosition(GXPosition upperLeft)
```

---

## **setLowerRightPosition**

```
public void setLowerRightPosition(GXPosition lowerRight)
```

## com.sabratec.applinx.baseobject Class GXAbstractWaitScreen

```
java.lang.Object
├--com.sabratec.applinx.baseobject.GXAbstractWaitCondition
│   └--com.sabratec.applinx.baseobject.GXAbstractWaitScreen
```

### All Implemented Interfaces:

[GXWaitCondition](#)

### Direct Known Subclasses:

[GXWaitWhileScreen](#), [GXWaitForScreen](#)

public abstract class **GXAbstractWaitScreen**  
extends [GXAbstractWaitCondition](#)

Abstract class for wait conditions involving a screen.

## Method Summary

String	<a href="#">getScreenName()</a>
void	<a href="#">setScreenName</a> (String screenName) Sets the screen name.

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Methods

### getScreenName

```
public String getScreenName()
```

#### Returns:

the screen name

### setScreenName

```
public void setScreenName(String screenName)
```

Sets the screen name.

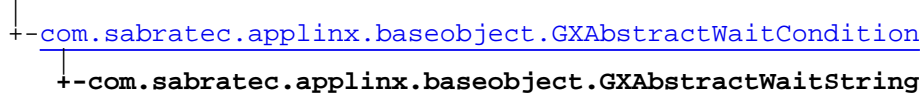
#### Parameters:

(continued from last page)

screenName - the screen name to set

## com.sabratec.applinx.baseobject Class GXAbstractWaitString

java.lang.Object



All Implemented Interfaces:

[GXWaitCondition](#)

Direct Known Subclasses:

[GXWaitWhileString](#), [GXWaitForString](#)

public abstract class **GXAbstractWaitString**  
extends [GXAbstractWaitCondition](#)

Abstract class for wait conditions involving String on the host screen.

### Method Summary

<a href="#">GXPosition</a>	<a href="#">getLowerRightPosition()</a>
String	<a href="#">getString()</a>
<a href="#">GXPosition</a>	<a href="#">getUpperLeftPosition()</a>
boolean	<a href="#">isCaseSensitive()</a>
void	<a href="#">setIsCaseSensitive(boolean isSensitive)</a>
void	<a href="#">setLowerRightPosition(<a href="#">GXPosition</a> lowerRight)</a>
void	<a href="#">setString(String string)</a>
void	<a href="#">setUpperLeftPosition(<a href="#">GXPosition</a> upperLeft)</a>

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Methods

(continued from last page)

## getString

```
public String getString()
```

---

## setString

```
public void setString(String string)
```

---

## getUpperLeftPosition

```
public GXPosition getUpperLeftPosition()
```

---

## setUpperLeftPosition

```
public void setUpperLeftPosition(GXPosition upperLeft)
```

---

## getLowerRightPosition

```
public GXPosition getLowerRightPosition()
```

---

## setLowerRightPosition

```
public void setLowerRightPosition(GXPosition lowerRight)
```

---

## isCaseSensitive

```
public boolean isCaseSensitive()
```

---

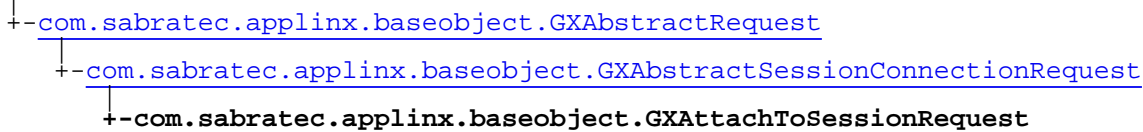
## setIsCaseSensitive

```
public void setIsCaseSensitive(boolean isSensitive)
```

---

## com.sabratec.applinx.baseobject Class GXAttachToSessionRequest

java.lang.Object



### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXAttachToSessionRequest**  
extends [GXAbstractSessionConnectionRequest](#)

Settings for attach request.

## Constructor Summary

public	<a href="#">GXAttachToSessionRequest</a> ()
public	<a href="#">GXAttachToSessionRequest</a> (GXIApplicationServerContext serverContext, String sessionId) Constructor
public	<a href="#">GXAttachToSessionRequest</a> (String serverURL, String sessionId, String sessionPassword) Constructor

## Method Summary

void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

### Methods inherited from class

[com.sabratec.applinx.baseobject.GXAbstractSessionConnectionRequest](#)

[getApplicationName](#), [getClientId](#), [getServerContext](#), [getServerURL](#), [getSessionID](#), [getSessionPassword](#), [readExternal](#), [setApplicationName](#), [setClientId](#), [setServerContext](#), [setServerURL](#), [setSessionID](#), [setSessionPassword](#), [writeExternal](#)

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Methods inherited from interface java.io.Externalizable

```
readExternal, writeExternal
```

## Constructors

### **GXAttachToSessionRequest**

```
public GXAttachToSessionRequest()
```

### **GXAttachToSessionRequest**

```
public GXAttachToSessionRequest(GXIApplicationServerContext serverContext,  
                                String sessionId)
```

Constructor

**Parameters:**

serverContext - ApplinX Server context.

sessionId - ID of the session to attach to.

### **GXAttachToSessionRequest**

```
public GXAttachToSessionRequest(String serverURL,  
                                String sessionId,  
                                String sessionPassword)
```

Constructor

**Parameters:**

serverURL - the URL of ApplinX server.

sessionId - ID to which the session will be attached.

sessionPassword - password used to attach to the existing session on ApplinX server. Should be set ONLY if a password was set in the [GXCreateSessionRequest](#).

## Methods

### **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject Class GXBaseObjectConstants

```
java.lang.Object
```

```
└─com.sabratec.applinx.baseobject.GXBaseObjectConstants
```

```
public class GXBaseObjectConstants  
extends Object
```

This class holds all the constants of the base object.

### Field Summary

public static final	<a href="#">ANY_IDENTIFIED_SCREEN</a> This constant should be used when adding a wait condition which should wait for any identified screen Sample code: if <pre>(gx_session.getScreen().getName().equals("UNKNOWN")){     GXGetScreenRequest req = new GXGetScreenRequest();     req.addWaitCondition(new GXWaitForScreen(GXBaseObjectConstants.ANY_IDENTIFIED_SCREEN,5000,0)) ;    gx_session.getScreen(req); }</pre>
public static final	<a href="#">GX_FIELD_ADJUST_LEFT</a>
public static final	<a href="#">GX_FIELD_ADJUST_NONE</a>
public static final	<a href="#">GX_FIELD_ADJUST_RIGHT</a>
public static final	<a href="#">GX_FIELD_ADJUST_RIGHT_BLANK_FILL</a>
public static final	<a href="#">GX_FIELD_ADJUST_RIGHT_ZERO_FILL</a>
public static final	<a href="#">GX_FIELD_COLOR_AQUA</a> Value: 3
public static final	<a href="#">GX_FIELD_COLOR_BLACK</a> Value: 0
public static final	<a href="#">GX_FIELD_COLOR_BLUE</a> Value: 1
public static final	<a href="#">GX_FIELD_COLOR_BROWN</a> Value: 6
public static final	<a href="#">GX_FIELD_COLOR_GRAY</a> Value: 8
public static final	<a href="#">GX_FIELD_COLOR_GREEN</a> Value: 2

public static final	<a href="#"><u>GX_FIELD_COLOR_LIGHT_AQUA</u></a> Value: <b>11</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_LIGHT_BLUE</u></a> Value: <b>9</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_LIGHT_GREEN</u></a> Value: <b>10</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_LIGHT_PURPLE</u></a> Value: <b>13</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_LIGHT_RED</u></a> Value: <b>12</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_LIGHT_WHITE</u></a> Value: <b>15</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_NONE</u></a> Value: <b>-1</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_PURPLE</u></a> Value: <b>5</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_RED</u></a> Value: <b>4</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_WHITE</u></a> Value: <b>7</b>
public static final	<a href="#"><u>GX_FIELD_COLOR_YELLOW</u></a> Value: <b>14</b>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_ALPHA_ONLY</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_ALPHANUMERIC</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_DBCS_CAN_CREATE_SISO</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_DBCS_ONLY</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_DIGITS_ONLY</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_KATAKANA_SHIFT</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_NUMERIC</u></a>
public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_REVERSED</u></a>

public static final	<a href="#"><u>GX_FIELD_DATA_TYPE_SIGNED_NUMERIC</u></a>
public static final	<a href="#"><u>GX_VAR_APPLICATION_LANGUAGE</u></a> The language configured for the application used by the current session. Value: <b>com.sabratec.applinx.config.application_language</b>
public static final	<a href="#"><u>GX_VAR_APPLY_TRANSFORMATIONS</u></a> Indicates if transformations should be applied to the next screen requested. Value: <b>com.sabratec.applinx.apply.transformations</b>
public static final	<a href="#"><u>GX_VAR_DEVICE_NAME</u></a> The device name to be used in the current session. Value: <b>com.sabratec.applinx.emulation.device_name</b>
public static final	<a href="#"><u>GX_VAR_DEVICE_TYPE</u></a> The device type to be used in the current session. Value: <b>com.sabratec.applinx.emulation.device_type</b>
public static final	<a href="#"><u>GX_VAR_DISPLAY_HIDDEN_FIELDS</u></a> Indicates whether or not to retrieve the contents of hidden fields. Value: <b>com.sabratec.applinx.screen.content.displayHiddenFields</b>
public static final	<a href="#"><u>GX_VAR_ENCODING_LOGICAL</u></a> Indicates that data should be converted from Visual to Logical. Value: <b>com.sabratec.applinx.screen.content.encoding.logical</b>
public static final	<a href="#"><u>GX_VAR_FIELDS_WITH_SUGGESTED_NAMES</u></a> Value: <b>com.sabratec.applinx.fields.with.suggested.names</b>
public static final	<a href="#"><u>GX_VAR_HOST_ADDRESS</u></a> The host IP address. Value: <b>com.sabratec.applinx.emulation.host_address</b>
public static final	<a href="#"><u>GX_VAR_HOST_PORT</u></a> The host's port address. Value: <b>com.sabratec.applinx.emulation.host_port</b>
public static final	<a href="#"><u>GX_VAR_HOST_WINDOW_ENABLED</u></a> Indicates that the host window should be ignored in the current screen. Value: <b>com.sabratec.applinx.screen.window.enabled</b>
public static final	<a href="#"><u>GX_VAR_INCLUDE_HOST_FIELDS</u></a> Indicates whether to retrieve host fields as well as application fields or just application fields. Value: <b>com.sabratec.applinx.screen.content.hostFields</b>
public static final	<a href="#"><u>GX_VAR_NATURAL_DATA_TRANSFER_DELIMITER</u></a> Relevant when downloading CSV files, determines the delimiter to be used. Value: <b>com.softwareag.applinx.ndt.delimiter</b>
public static final	<a href="#"><u>GX_VAR_NATURAL_DATA_TRANSFER_HEADERS_USED</u></a> Determines whether to print the name of the field as the column header. Value: <b>com.softwareag.applinx.ndt.headers.used</b>

public static final	<a href="#"><u>GX_VAR_NATURAL_DATA_TRANSFER_STREAM</u></a> Determines whether to stream the download file from the host through the server to the client or download the file as a whole. Value: <b>com.softwareag.applinx.ndt.stream</b>
public static final	<a href="#"><u>GX_VAR_REMOVE_CHARACTER</u></a> Indicates that the character configured to be removed should be removed. Value: <b>com.sabratec.applinx.screen.content.remove_underline</b>
public static final	<a href="#"><u>GX_VAR_REMOVE_CHARACTER_BOTH_SIDES</u></a> Indicates that the character configured to be removed should be removed from both sides of fields. Value: <b>com.sabratec.applinx.screen.content.remove_underline.sides</b>
public static final	<a href="#"><u>GX_VAR_REPLAY_DELAY</u></a> Replay delay. Value: <b>com.sabratec.applinx.config.replay_delay</b>
public static final	<a href="#"><u>GX_VAR_REPLAY_FILE</u></a> The replay file configured for the current session. Value: <b>com.sabratec.applinx.config.replay_file</b>
public static final	<a href="#"><u>GX_VAR_RETURN_FIELDS</u></a> Indicates whether the fields should be returned Used with the ACTION_SESSION_GET_SCREEN request. Value: <b>com.softwareag.applinx.return.fields</b>
public static final	<a href="#"><u>GX_VAR_RETURN_SCREEN_BUFFERS</u></a> Indicates which buffer types should be returned in the screen object. Value: <b>com.sabratec.applinx.screen.buffers</b>
public static final	<a href="#"><u>GX_VAR_SCREEN_DIRECTION</u></a> The current screen direction of the application. Value: <b>com.sabratec.applinx.screen.direction</b>
public static final	<a href="#"><u>GX_VAR_THROW_EXCEPTION_ON_INVALID_INPUT</u></a> Indicates whether or not to throw an exception if an invalid input was sent to a field Value: <b>com.sabratec.applinx.screen.content.invalidInputThrowsException</b>
public static final	<a href="#"><u>GX_VAR_TRACE_FILE</u></a> The trace file configured for the current session. Value: <b>com.sabratec.applinx.config.trace_file</b>
public static final	<a href="#"><u>GX_VAR_TRANSLATION_LANGUAGE</u></a> The translation language used by the current session. Value: <b>com.sabratec.applinx.localizer.translation_language</b>
public static final	<a href="#"><u>GX_VAR_TRIM_FIELDS</u></a> Variable indicating which fields should be trimmed. Value: <b>com.sabratec.applinx.screen.content.trim.fields</b>
public static final	<a href="#"><u>GX_VAR_TRIM_MODE</u></a> Variable indicating the trimming mode of fields. Value: <b>com.sabratec.applinx.screen.content.trim</b>

public static final	<a href="#"><u>GX_VAR_UPPERCASE_INPUT</u></a> Flag to turn all data sent to the host to uppercase. Value: <b>com.sabratec.applinx.emulation.uppercase_input</b>
public static final	<a href="#"><u>GX_VAR_VAL_BUFFERS_FIELD_ATTRIBUTES</u></a> Three field attribute buffers are returned: field attributes, extended field attributes and field color attributes. Value: <b>com.sabratec.applinx.screen.buffers.attributes</b>
public static final	<a href="#"><u>GX_VAR_VAL_BUFFERS_NONE</u></a> No screen buffers are returned. Value: <b>com.sabratec.applinx.screen.buffers.none</b>
public static final	<a href="#"><u>GX_VAR_VAL_BUFFERS_SCREEN_DATA</u></a> Screen data buffer (the fields' content) is returned. Value: <b>com.sabratec.applinx.screen.buffers.screen_data</b>
public static final	<a href="#"><u>GX_VAR_VAL_REMOVE_CHARACTER_ALL</u></a> The character to be removed should be removed from all fields including input fields and labels. Value: <b>all</b>
public static final	<a href="#"><u>GX_VAR_VAL_REMOVE_CHARACTER_BOTH_SIDES_ALL_FIELDS</u></a> Removes character from both sides of all fields. Value: <b>all</b>
public static final	<a href="#"><u>GX_VAR_VAL_REMOVE_CHARACTER_BOTH_SIDES_NUMERIC_FIELDS_ONLY</u></a> Removes character from both sides of numeric fields only. Value: <b>numeric</b>
public static final	<a href="#"><u>GX_VAR_VAL_REMOVE_CHARACTER_INPUT</u></a> The character to be removed should be removed only from input fields. Value: <b>input</b>
public static final	<a href="#"><u>GX_VAR_VAL_REMOVE_CHARACTER_NONE</u></a> The character to be removed should not be removed. Value: <b>none</b>
public static final	<a href="#"><u>GX_VAR_VAL_REPLAY_DELAY_NO_DELAY</u></a> Run replay as fast as possible. Value: <b>0</b>
public static final	<a href="#"><u>GX_VAR_VAL_REPLAY_DELAY_SIMULATE_HOST</u></a> Run replay as recorded by host. Value: <b>-1</b>
public static final	<a href="#"><u>GX_VAR_VAL_SCREEN_DIRECTION_LTR</u></a> Screen direction is left to right. Value: <b>LTR</b>
public static final	<a href="#"><u>GX_VAR_VAL_SCREEN_DIRECTION_RTL</u></a> Screen direction is right to left. Value: <b>RTL</b>
public static final	<a href="#"><u>GX_VAR_VAL_TRIM_FIELDS_ALL</u></a> Indicates that data of all fields should be trimmed. Value: <b>all</b>

public static final	<a href="#">GX_VAR_VAL_TRIM_FIELDS_INPUT_AND_BOTH</a> Indicates that data of input fields and application fields defined as "both" should be trimmed. Value: <b>input_both</b>
public static final	<a href="#">GX_VAR_VAL_TRIM_FIELDS_INPUT_ONLY</a> Indicates that data of input fields only should be trimmed. Value: <b>input_only</b>
public static final	<a href="#">GX_VAR_VAL_TRIM_FIELDS_NONE</a> Indicates that field data should not be trimmed. Value: <b>none</b>
public static final	<a href="#">GX_VAR_VAL_TRIM_MODE_BOTH_SIDES</a> Indicates trimming should be done on both sides. Value: <b>bothSides</b>
public static final	<a href="#">GX_VAR_VAL_TRIM_MODE_BY_FIELD_DIRECTION</a> Indicates trimming should be done according to field direction. Value: <b>fieldDirection</b>
public static final	<a href="#">GX_VAR_VAL_VT_SEND_CURSOR_ARROWS</a> Determines that the cursor should be set using the arrow keys. Value: <b>arrows</b>
public static final	<a href="#">GX_VAR_VAL_VT_SEND_CURSOR_NONE</a> Determines that the cursor should NOT be set to the host. Value: <b>none</b>
public static final	<a href="#">GX_VAR_VAL_VT_SEND_CURSOR_TAB</a> Determines that the cursor should be set using a tab key. Value: <b>tab_key</b>
public static final	<a href="#">GX_VAR_VT_AUTO_SKIP_WHEN_FIELD_IS_FULL</a> Flag that states whether or not the cursor will jump automatically from the current field to the next input field when the current input field is filled with content. Value: <b>com.sabratec.applinx.vt.blockmode.input_fields.auto_skip</b>
public static final	<a href="#">GX_VAR_VT_DYNAMIC_INPUT_CHECK</a> Flag that invokes the dynamic input check. Value: <b>com.sabratec.applinx.vt.blockmode.input_fields.dynamicCheck</b>
public static final	<a href="#">GX_VAR_VT_KEY_JUMP_BETWEEN_INPUT_FIELDS</a> The key ApplinX server should use in order to move between input fields in the host screen. Value: <b>com.sabratec.applinx.vt.blockmode.input_fields.jumpkey</b>
public static final	<a href="#">GX_VAR_VT_SEND_CURSOR</a> Determines the way the cursor should be set to a character mode host. Value: <b>com.sabratec.applinx.vt.blockmode.send_cursor</b>
public static final	<a href="#">GX_VAR_WORKING_FOLDER</a> The working folder for the current session. Value: <b>com.sabratec.applinx.folder.working_folder</b>

## Constructor Summary

public	<a href="#">GXBaseObjectConstants()</a>
--------	---

---

**Methods inherited from class** `java.lang.Object``equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

---

## Fields

### **GX\_FIELD\_COLOR\_NONE**

```
public static final int GX_FIELD_COLOR_NONE
```

Constant value: **-1**

---

### **GX\_FIELD\_COLOR\_BLACK**

```
public static final int GX_FIELD_COLOR_BLACK
```

Constant value: **0**

---

### **GX\_FIELD\_COLOR\_BLUE**

```
public static final int GX_FIELD_COLOR_BLUE
```

Constant value: **1**

---

### **GX\_FIELD\_COLOR\_GREEN**

```
public static final int GX_FIELD_COLOR_GREEN
```

Constant value: **2**

---

### **GX\_FIELD\_COLOR\_AQUA**

```
public static final int GX_FIELD_COLOR_AQUA
```

Constant value: **3**

---

### **GX\_FIELD\_COLOR\_RED**

```
public static final int GX_FIELD_COLOR_RED
```

Constant value: **4**

---

### **GX\_FIELD\_COLOR\_PURPLE**

```
public static final int GX_FIELD_COLOR_PURPLE
```

Constant value: **5**

---

---

## **GX\_FIELD\_COLOR\_BROWN**

```
public static final int GX_FIELD_COLOR_BROWN
```

Constant value: 6

---

## **GX\_FIELD\_COLOR\_WHITE**

```
public static final int GX_FIELD_COLOR_WHITE
```

Constant value: 7

---

## **GX\_FIELD\_COLOR\_GRAY**

```
public static final int GX_FIELD_COLOR_GRAY
```

Constant value: 8

---

## **GX\_FIELD\_COLOR\_LIGHT\_BLUE**

```
public static final int GX_FIELD_COLOR_LIGHT_BLUE
```

Constant value: 9

---

## **GX\_FIELD\_COLOR\_LIGHT\_GREEN**

```
public static final int GX_FIELD_COLOR_LIGHT_GREEN
```

Constant value: 10

---

## **GX\_FIELD\_COLOR\_LIGHT\_AQUA**

```
public static final int GX_FIELD_COLOR_LIGHT_AQUA
```

Constant value: 11

---

## **GX\_FIELD\_COLOR\_LIGHT\_RED**

```
public static final int GX_FIELD_COLOR_LIGHT_RED
```

Constant value: 12

---

## **GX\_FIELD\_COLOR\_LIGHT\_PURPLE**

```
public static final int GX_FIELD_COLOR_LIGHT_PURPLE
```

Constant value: 13

---

(continued from last page)

## GX\_FIELD\_COLOR\_YELLOW

```
public static final int GX_FIELD_COLOR_YELLOW
```

Constant value: 14

---

## GX\_FIELD\_COLOR\_LIGHT\_WHITE

```
public static final int GX_FIELD_COLOR_LIGHT_WHITE
```

Constant value: 15

---

## GX\_VAR\_DEVICE\_NAME

```
public static final java.lang.String GX_VAR_DEVICE_NAME
```

The device name to be used in the current session.

Constant value: **com.sabratec.applinx.emulation.device\_name**

---

## GX\_VAR\_DEVICE\_TYPE

```
public static final java.lang.String GX_VAR_DEVICE_TYPE
```

The device type to be used in the current session.

Constant value: **com.sabratec.applinx.emulation.device\_type**

---

## GX\_VAR\_HOST\_ADDRESS

```
public static final java.lang.String GX_VAR_HOST_ADDRESS
```

The host IP address.

Constant value: **com.sabratec.applinx.emulation.host\_address**

---

## GX\_VAR\_HOST\_PORT

```
public static final java.lang.String GX_VAR_HOST_PORT
```

The host's port address.

Constant value: **com.sabratec.applinx.emulation.host\_port**

---

## GX\_VAR\_UPPERCASE\_INPUT

```
public static final java.lang.String GX_VAR_UPPERCASE_INPUT
```

Flag to turn all data sent to the host to uppercase. Possible values: true or false.

Constant value: **com.sabratec.applinx.emulation.uppercase\_input**

---

## GX\_VAR\_VT\_SEND\_CURSOR

```
public static final java.lang.String GX_VAR_VT_SEND_CURSOR
```

Determines the way the cursor should be set to a character mode host.

Constant value: **com.sabratec.applinx.vt.blockmode.send\_cursor**

### See Also:

[GX\\_VAR\\_VAL\\_VT\\_SEND\\_CURSOR\\_TAB](#)

[GX\\_VAR\\_VAL\\_VT\\_SEND\\_CURSOR\\_ARROWS](#)

[GX\\_VAR\\_VAL\\_VT\\_SEND\\_CURSOR\\_NONE](#)

## GX\_VAR\_VAL\_VT\_SEND\_CURSOR\_TAB

```
public static final java.lang.String GX_VAR_VAL_VT_SEND_CURSOR_TAB
```

Determines that the cursor should be set using a tab key. Possible value of GX\_VAR\_VT\_SEND\_CURSOR.  
Constant value: **tab\_key**

**See Also:**

[GX\\_VAR\\_VT\\_SEND\\_CURSOR](#)

[GX\\_VAR\\_VT\\_KEY\\_JUMP\\_BETWEEN\\_INPUT\\_FIELDS](#)

---

## GX\_VAR\_VAL\_VT\_SEND\_CURSOR\_ARROWS

```
public static final java.lang.String GX_VAR_VAL_VT_SEND_CURSOR_ARROWS
```

Determines that the cursor should be set using the arrow keys. Possible value of GX\_VAR\_VT\_SEND\_CURSOR.  
Constant value: **arrows**

**See Also:**

[GX\\_VAR\\_VT\\_SEND\\_CURSOR](#)

---

## GX\_VAR\_VAL\_VT\_SEND\_CURSOR\_NONE

```
public static final java.lang.String GX_VAR_VAL_VT_SEND_CURSOR_NONE
```

Determines that the cursor should NOT be set to the host. Possible value of GX\_VAR\_VT\_SEND\_CURSOR.  
Constant value: **none**

**See Also:**

[GX\\_VAR\\_VT\\_SEND\\_CURSOR](#)

---

## GX\_VAR\_VT\_KEY\_JUMP\_BETWEEN\_INPUT\_FIELDS

```
public static final java.lang.String GX_VAR_VT_KEY_JUMP_BETWEEN_INPUT_FIELDS
```

The key ApplinX server should use in order to move between input fields in the host screen. (Example value: "[tab]").  
Constant value: **com.sabratec.applinx.vt.blockmode.input\_fields.jumpkey**

---

## GX\_VAR\_VT\_AUTO\_SKIP\_WHEN\_FIELD\_IS\_FULL

```
public static final java.lang.String GX_VAR_VT_AUTO_SKIP_WHEN_FIELD_IS_FULL
```

Flag that states whether or not the cursor will jump automatically from the current field to the next input field when the current input field is filled with content. Possible values: true, false.  
Constant value: **com.sabratec.applinx.vt.blockmode.input\_fields.auto\_skip**

---

## GX\_VAR\_VT\_DYNAMIC\_INPUT\_CHECK

```
public static final java.lang.String GX_VAR_VT_DYNAMIC_INPUT_CHECK
```

Flag that invokes the dynamic input check. The dynamic input check sends the key used to move between input fields in order to determine in runtime where there are input fields on the host screen. Possible values: true, false.  
Constant value: **com.sabratec.applinx.vt.blockmode.input\_fields.dynamicCheck**

---

## GX\_VAR\_APPLICATION\_LANGUAGE

```
public static final java.lang.String GX_VAR_APPLICATION_LANGUAGE
```

The language configured for the application used by the current session.

---

(continued from last page)

Constant value: **com.sabratec.applinx.config.application\_language**

---

## GX\_VAR\_REPLAY\_FILE

```
public static final java.lang.String GX_VAR_REPLAY_FILE
```

The replay file configured for the current session.

Constant value: **com.sabratec.applinx.config.replay\_file**

---

## GX\_VAR\_REPLAY\_DELAY

```
public static final java.lang.String GX_VAR_REPLAY_DELAY
```

Replay delay. Either a number as string, or [GX\\_VAR\\_VAL\\_REPLAY\\_DELAY\\_NO\\_DELAY](#) or [GX\\_VAR\\_VAL\\_REPLAY\\_DELAY\\_SIMULATE\\_HOST](#).

Constant value: **com.sabratec.applinx.config.replay\_delay**

---

## GX\_VAR\_VAL\_REPLAY\_DELAY\_NO\_DELAY

```
public static final java.lang.String GX_VAR_VAL_REPLAY_DELAY_NO_DELAY
```

Run replay as fast as possible. Possible value of GX\_VAR\_REPLAY\_DELAY.

Constant value: **0**

See Also:

[GX\\_VAR\\_REPLAY\\_DELAY](#)

---

## GX\_VAR\_VAL\_REPLAY\_DELAY\_SIMULATE\_HOST

```
public static final java.lang.String GX_VAR_VAL_REPLAY_DELAY_SIMULATE_HOST
```

Run replay as recorded by host. Possible value of GX\_VAR\_REPLAY\_DELAY.

Constant value: **-1**

See Also:

[GX\\_VAR\\_REPLAY\\_DELAY](#)

---

## GX\_VAR\_TRACE\_FILE

```
public static final java.lang.String GX_VAR_TRACE_FILE
```

The trace file configured for the current session.

Constant value: **com.sabratec.applinx.config.trace\_file**

---

## GX\_VAR\_TRANSLATION\_LANGUAGE

```
public static final java.lang.String GX_VAR_TRANSLATION_LANGUAGE
```

The translation language used by the current session. These are the possible values: Afrikaans Albanian Arabic Basque Belarusian Bulgarian Catalan Chinese Croatian Czech Danish Dutch English Estonian Faeroese Farsi Finnish French German Greek Hebrew Hungarian Icelandic Indonesian Italian Japanese Korean Latvian Lithuanian Norwegian Polish Portuguese Romanian Russian Serbian Slovak Slovenian Spanish Swedish Thai Turkish Ukrainian Vietnamese

Constant value: **com.sabratec.applinx.localizer.translation\_language**

---

## GX\_VAR\_WORKING\_FOLDER

```
public static final java.lang.String GX_VAR_WORKING_FOLDER
```

The working folder for the current session.

Constant value: **com.sabratec.applinx.folder.working\_folder**

---

## GX\_VAR\_SCREEN\_DIRECTION

```
public static final java.lang.String GX_VAR_SCREEN_DIRECTION
```

The current screen direction of the application.

Constant value: **com.sabratec.applinx.screen.direction**

See Also:

[GX\\_VAR\\_VAL\\_SCREEN\\_DIRECTION\\_RTL](#)

[GX\\_VAR\\_VAL\\_SCREEN\\_DIRECTION\\_LTR](#)

---

## GX\_VAR\_VAL\_SCREEN\_DIRECTION\_RTL

```
public static final java.lang.String GX_VAR_VAL_SCREEN_DIRECTION_RTL
```

Screen direction is right to left. Possible value of GX\_VAR\_SCREEN\_DIRECTION.

Constant value: **RTL**

See Also:

[GX\\_VAR\\_SCREEN\\_DIRECTION](#)

---

## GX\_VAR\_VAL\_SCREEN\_DIRECTION\_LTR

```
public static final java.lang.String GX_VAR_VAL_SCREEN_DIRECTION_LTR
```

Screen direction is left to right. Possible value of GX\_VAR\_SCREEN\_DIRECTION.

Constant value: **LTR**

See Also:

[GX\\_VAR\\_SCREEN\\_DIRECTION](#)

---

## GX\_VAR\_RETURN\_SCREEN\_BUFFERS

```
public static final java.lang.String GX_VAR_RETURN_SCREEN_BUFFERS
```

Indicates which buffer types should be returned in the screen object.

Constant value: **com.sabratec.applinx.screen.buffer**

See Also:

[GX\\_VAR\\_VAL\\_BUFFERS\\_NONE](#)

[GX\\_VAR\\_VAL\\_BUFFERS\\_SCREEN\\_DATA](#)

[GX\\_VAR\\_VAL\\_BUFFERS\\_FIELD\\_ATTRIBUTES](#)

---

## GX\_VAR\_VAL\_BUFFERS\_NONE

```
public static final java.lang.String GX_VAR_VAL_BUFFERS_NONE
```

No screen buffers are returned. Possible value of GX\_VAR\_RETURN\_SCREEN\_BUFFERS.

Constant value: **com.sabratec.applinx.screen.buffer.none**

See Also:

[GX\\_VAR\\_RETURN\\_SCREEN\\_BUFFERS](#)

---

## GX\_VAR\_VAL\_BUFFERS\_SCREEN\_DATA

```
public static final java.lang.String GX_VAR_VAL_BUFFERS_SCREEN_DATA
```

Screen data buffer (the fields' content) is returned. Possible value of GX\_VAR\_RETURN\_SCREEN\_BUFFERS.

---

(continued from last page)

Constant value: **com.sabratec.applinx.screen.buffer.screen\_data**

See Also:

[GX\\_VAR\\_RETURN\\_SCREEN\\_BUFFERS](#)

---

## GX\_VAR\_VAL\_BUFFERS\_FIELD\_ATTRIBUTES

public static final java.lang.String **GX\_VAR\_VAL\_BUFFERS\_FIELD\_ATTRIBUTES**

Three field attribute buffers are returned: field attributes, extended field attributes and field color attributes. Possible value of GX\_VAR\_RETURN\_SCREEN\_BUFFERS.

Constant value: **com.sabratec.applinx.screen.buffer.attributes**

See Also:

[GX\\_VAR\\_RETURN\\_SCREEN\\_BUFFERS](#)

---

## GX\_VAR\_REMOVE\_CHARACTER

public static final java.lang.String **GX\_VAR\_REMOVE\_CHARACTER**

Indicates that the character configured to be removed should be removed.

Constant value: **com.sabratec.applinx.screen.content.remove\_underline**

See Also:

[GX\\_VAR\\_VAL\\_REMOVE\\_CHARACTER\\_ALL](#)

[GX\\_VAR\\_VAL\\_REMOVE\\_CHARACTER\\_INPUT](#)

[GX\\_VAR\\_VAL\\_REMOVE\\_CHARACTER\\_NONE](#)

---

## GX\_VAR\_VAL\_REMOVE\_CHARACTER\_ALL

public static final java.lang.String **GX\_VAR\_VAL\_REMOVE\_CHARACTER\_ALL**

The character to be removed should be removed from all fields including input fields and labels. Possible value of GX\_VAR\_REMOVE\_CHARACTER.

Constant value: **all**

See Also:

[GX\\_VAR\\_REMOVE\\_CHARACTER](#)

---

## GX\_VAR\_VAL\_REMOVE\_CHARACTER\_INPUT

public static final java.lang.String **GX\_VAR\_VAL\_REMOVE\_CHARACTER\_INPUT**

The character to be removed should be removed only from input fields. Possible value of GX\_VAR\_REMOVE\_CHARACTER.

Constant value: **input**

See Also:

[GX\\_VAR\\_REMOVE\\_CHARACTER](#)

---

## GX\_VAR\_VAL\_REMOVE\_CHARACTER\_NONE

public static final java.lang.String **GX\_VAR\_VAL\_REMOVE\_CHARACTER\_NONE**

The character to be removed should not be removed. Possible value of GX\_VAR\_REMOVE\_CHARACTER.

Constant value: **none**

See Also:

[GX\\_VAR\\_REMOVE\\_CHARACTER](#)

## **GX\_VAR\_REMOVE\_CHARACTER\_BOTH\_SIDES**

```
public static final java.lang.String GX_VAR_REMOVE_CHARACTER_BOTH_SIDES
```

Indicates that the character configured to be removed should be removed from both sides of fields.

Constant value: **com.sabratec.applinx.screen.content.remove\_underline.sides**

### **See Also:**

[GX\\_VAR\\_REMOVE\\_CHARACTER\\_BOTH\\_SIDES](#)

[GX\\_VAR\\_VAL\\_REMOVE\\_CHARACTER\\_BOTH\\_SIDES\\_ALL\\_FIELDS](#)

---

## **GX\_VAR\_VAL\_REMOVE\_CHARACTER\_BOTH\_SIDES\_NUMERIC\_FIELDS\_ONLY**

```
public static final java.lang.String  
GX_VAR_VAL_REMOVE_CHARACTER_BOTH_SIDES_NUMERIC_FIELDS_ONLY
```

Removes character from both sides of numeric fields only. Possible value of GX\_VAR\_REMOVE\_CHARACTER\_BOTH\_SIDES.

Constant value: **numeric**

### **See Also:**

[GX\\_VAR\\_REMOVE\\_CHARACTER\\_BOTH\\_SIDES](#)

---

## **GX\_VAR\_VAL\_REMOVE\_CHARACTER\_BOTH\_SIDES\_ALL\_FIELDS**

```
public static final java.lang.String GX_VAR_VAL_REMOVE_CHARACTER_BOTH_SIDES_ALL_FIELDS
```

Removes character from both sides of all fields. Possible value of GX\_VAR\_REMOVE\_CHARACTER\_BOTH\_SIDES.

Constant value: **all**

### **See Also:**

[GX\\_VAR\\_REMOVE\\_CHARACTER\\_BOTH\\_SIDES](#)

---

## **GX\_VAR\_ENCODING\_LOGICAL**

```
public static final java.lang.String GX_VAR_ENCODING_LOGICAL
```

Indicates that data should be converted from Visual to Logical. Used by right to left languages (such as Hebrew). Possible values: true, false.

Constant value: **com.sabratec.applinx.screen.content.encoding.logical**

---

## **GX\_VAR\_TRIM\_FIELDS**

```
public static final java.lang.String GX_VAR_TRIM_FIELDS
```

Variable indicating which fields should be trimmed. It is useful when displaying the fields' data.

Constant value: **com.sabratec.applinx.screen.content.trim.fields**

### **See Also:**

[GX\\_VAR\\_VAL\\_TRIM\\_FIELDS\\_ALL](#)

[GX\\_VAR\\_VAL\\_TRIM\\_FIELDS\\_NONE](#)

[GX\\_VAR\\_VAL\\_TRIM\\_FIELDS\\_INPUT\\_ONLY](#)

[GX\\_VAR\\_VAL\\_TRIM\\_FIELDS\\_INPUT\\_AND\\_BOTH](#)

---

## **GX\_VAR\_VAL\_TRIM\_FIELDS\_INPUT\_ONLY**

```
public static final java.lang.String GX_VAR_VAL_TRIM_FIELDS_INPUT_ONLY
```

---

(continued from last page)

Indicates that data of input fields only should be trimmed. Possible value of GX\_VAR\_TRIM\_FIELDS.  
Constant value: **input\_only**

See Also:

[GX\\_VAR\\_TRIM\\_FIELDS](#)

---

## GX\_VAR\_VAL\_TRIM\_FIELDS\_INPUT\_AND\_BOTH

```
public static final java.lang.String GX_VAR_VAL_TRIM_FIELDS_INPUT_AND_BOTH
```

Indicates that data of input fields and application fields defined as "both" should be trimmed. Possible value of GX\_VAR\_TRIM\_FIELDS.  
Constant value: **input\_both**

See Also:

[GX\\_VAR\\_TRIM\\_FIELDS](#)

---

## GX\_VAR\_VAL\_TRIM\_FIELDS\_NONE

```
public static final java.lang.String GX_VAR_VAL_TRIM_FIELDS_NONE
```

Indicates that field data should not be trimmed. Possible value of GX\_VAR\_TRIM\_FIELDS.  
Constant value: **none**

See Also:

[GX\\_VAR\\_TRIM\\_FIELDS](#)

---

## GX\_VAR\_VAL\_TRIM\_FIELDS\_ALL

```
public static final java.lang.String GX_VAR_VAL_TRIM_FIELDS_ALL
```

Indicates that data of all fields should be trimmed. Possible value of GX\_VAR\_TRIM\_FIELDS.  
Constant value: **all**

See Also:

[GX\\_VAR\\_TRIM\\_FIELDS](#)

---

## GX\_VAR\_TRIM\_MODE

```
public static final java.lang.String GX_VAR_TRIM_MODE
```

Variable indicating the trimming mode of fields.  
Constant value: **com.sabratec.applinx.screen.content.trim**

See Also:

[GX\\_VAR\\_VAL\\_TRIM\\_MODE\\_BOTH\\_SIDES](#)

[GX\\_VAR\\_VAL\\_TRIM\\_MODE\\_BY\\_FIELD\\_DIRECTION](#)

---

## GX\_VAR\_VAL\_TRIM\_MODE\_BOTH\_SIDES

```
public static final java.lang.String GX_VAR_VAL_TRIM_MODE_BOTH_SIDES
```

Indicates trimming should be done on both sides.  
Constant value: **bothSides**

---

## GX\_VAR\_VAL\_TRIM\_MODE\_BY\_FIELD\_DIRECTION

```
public static final java.lang.String GX_VAR_VAL_TRIM_MODE_BY_FIELD_DIRECTION
```

(continued from last page)

Indicates trimming should be done according to field direction.  
Constant value: **fieldDirection**

---

## GX\_VAR\_INCLUDE\_HOST\_FIELDS

```
public static final java.lang.String GX_VAR_INCLUDE_HOST_FIELDS
```

Indicates whether to retrieve host fields as well as application fields or just application fields.  
Constant value: **com.sabratec.applinx.screen.content.hostFields**

---

## GX\_VAR\_DISPLAY\_HIDDEN\_FIELDS

```
public static final java.lang.String GX_VAR_DISPLAY_HIDDEN_FIELDS
```

Indicates whether or not to retrieve the contents of hidden fields.  
Constant value: **com.sabratec.applinx.screen.content.displayHiddenFields**

---

## GX\_VAR\_THROW\_EXCEPTION\_ON\_INVALID\_INPUT

```
public static final java.lang.String GX_VAR_THROW_EXCEPTION_ON_INVALID_INPUT
```

Indicates whether or not to throw an exception if an invalid input was sent to a field  
Constant value: **com.sabratec.applinx.screen.content.invalidInputThrowsException**

---

## GX\_VAR\_HOST\_WINDOW\_ENABLED

```
public static final java.lang.String GX_VAR_HOST_WINDOW_ENABLED
```

Indicates that the host window should be ignored in the current screen. This parameter affects retrieving the content of the screen (all the positions of the fields are absolute), but it does not affect the identification of the screen. Possible values: true, false.  
Constant value: **com.sabratec.applinx.screen.window.enabled**

---

## GX\_VAR\_FIELDS\_WITH\_SUGGESTED\_NAMES

```
public static final java.lang.String GX_VAR_FIELDS_WITH_SUGGESTED_NAMES
```

Constant value: **com.sabratec.applinx.fields.with.suggested.names**

---

## GX\_VAR\_APPLY\_TRANSFORMATIONS

```
public static final java.lang.String GX_VAR_APPLY_TRANSFORMATIONS
```

Indicates if transformations should be applied to the next screen requested. Used with the ACTION\_SESSION\_GET\_SCREEN request. Possible values: true, false.  
Constant value: **com.sabratec.applinx.apply.transformations**

---

## GX\_VAR\_RETURN\_FIELDS

```
public static final java.lang.String GX_VAR_RETURN_FIELDS
```

Indicates whether the fields should be returned Used with the ACTION\_SESSION\_GET\_SCREEN request. Possible values: true, false.  
Constant value: **com.softwareag.applinx.return.fields**

---

## GX\_VAR\_NATURAL\_DATA\_TRANSFER\_DELIMITER

```
public static final java.lang.String GX_VAR_NATURAL_DATA_TRANSFER_DELIMITER
```

(continued from last page)

Relevant when downloading CSV files, determines the delimiter to be used. By default "," is used.

Constant value: **com.softwareag.applinx.ndt.delimiter**

---

## **GX\_VAR\_NATURAL\_DATA\_TRANSFER\_HEADERS\_USED**

```
public static final java.lang.String GX_VAR_NATURAL_DATA_TRANSFER_HEADERS_USED
```

Determines whether to print the name of the field as the column header. Possible values: true, false. By default: false.

Constant value: **com.softwareag.applinx.ndt.headers.used**

---

## **GX\_VAR\_NATURAL\_DATA\_TRANSFER\_STREAM**

```
public static final java.lang.String GX_VAR_NATURAL_DATA_TRANSFER_STREAM
```

Determines whether to stream the download file from the host through the server to the client or download the file as a whole.

Constant value: **com.softwareag.applinx.ndt.stream**

---

## **GX\_FIELD\_DATA\_TYPE\_ALPHANUMERIC**

```
public static final int GX_FIELD_DATA_TYPE_ALPHANUMERIC
```

---

## **GX\_FIELD\_DATA\_TYPE\_NUMERIC**

```
public static final int GX_FIELD_DATA_TYPE_NUMERIC
```

---

## **GX\_FIELD\_DATA\_TYPE\_ALPHA\_ONLY**

```
public static final int GX_FIELD_DATA_TYPE_ALPHA_ONLY
```

---

## **GX\_FIELD\_DATA\_TYPE\_DIGITS\_ONLY**

```
public static final int GX_FIELD_DATA_TYPE_DIGITS_ONLY
```

---

## **GX\_FIELD\_DATA\_TYPE\_SIGNED\_NUMERIC**

```
public static final int GX_FIELD_DATA_TYPE_SIGNED_NUMERIC
```

---

## **GX\_FIELD\_DATA\_TYPE\_KATAKANA\_SHIFT**

```
public static final int GX_FIELD_DATA_TYPE_KATAKANA_SHIFT
```

---

## **GX\_FIELD\_DATA\_TYPE\_DBCS\_ONLY**

```
public static final int GX_FIELD_DATA_TYPE_DBCS_ONLY
```

---

---

## **GX\_FIELD\_DATA\_TYPE\_DBCS\_CAN\_CREATE\_SISO**

```
public static final int GX_FIELD_DATA_TYPE_DBCS_CAN_CREATE_SISO
```

---

## **GX\_FIELD\_DATA\_TYPE\_REVERSED**

```
public static final int GX_FIELD_DATA_TYPE_REVERSED
```

---

## **GX\_FIELD\_ADJUST\_NONE**

```
public static final int GX_FIELD_ADJUST_NONE
```

---

## **GX\_FIELD\_ADJUST\_RIGHT\_ZERO\_FILL**

```
public static final int GX_FIELD_ADJUST_RIGHT_ZERO_FILL
```

---

## **GX\_FIELD\_ADJUST\_RIGHT\_BLANK\_FILL**

```
public static final int GX_FIELD_ADJUST_RIGHT_BLANK_FILL
```

---

## **GX\_FIELD\_ADJUST\_RIGHT**

```
public static final int GX_FIELD_ADJUST_RIGHT
```

---

## **GX\_FIELD\_ADJUST\_LEFT**

```
public static final int GX_FIELD_ADJUST_LEFT
```

---

## **ANY\_IDENTIFIED\_SCREEN**

```
public static final java.lang.String ANY_IDENTIFIED_SCREEN
```

This constant should be used when adding a wait condition which should wait for any identified screen

Sample code:

```
if (gx_session.getScreen().getName().equals("UNKNOWN")){  
    GXGetScreenRequest req = new GXGetScreenRequest();  
    req.addWaitCondition(new  
GXWaitForScreen(GXBaseObjectConstants.ANY_IDENTIFIED_SCREEN, 5000, 0));  
    gx_session.getScreen(req);  
}
```

## Constructors

(continued from last page)

## **GXBaseObjectConstants**

```
public GXBaseObjectConstants()
```

## com.sabratec.applinx.baseobject

### Class GXBinaryRequest

java.lang.Object

└- [com.sabratec.applinx.baseobject.GXAbstractRequest](#)  
   └- **com.sabratec.applinx.baseobject.GXBinaryRequest**

#### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXBinaryRequest**  
 extends [GXAbstractRequest](#)

### Constructor Summary

public	<a href="#">GXBinaryRequest</a> ()
public	<a href="#">GXBinaryRequest</a> (byte[] data)

### Method Summary

byte[]	<a href="#">getData</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setData</a> (byte[] data)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

### Constructors

#### GXBinaryRequest

public **GXBinaryRequest**()

(continued from last page)

---

## GXBinaryRequest

```
public GXBinaryRequest(byte[] data)
```

## Methods

### getData

```
public byte[] getData()
```

---

### setData

```
public void setData(byte[] data)
```

---

### writeExternal

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

### readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

## com.sabratec.applinx.baseobject Class GXBinaryResponse

java.lang.Object

└─com.sabratec.applinx.baseobject.GXBinaryResponse

### All Implemented Interfaces:

java.io.Serializable

public class **GXBinaryResponse**  
extends Object  
implements java.io.Serializable

## Constructor Summary

public	<a href="#">GXBinaryResponse</a> (byte[] data)
--------	--

## Method Summary

byte[]	<a href="#">getData</a> ()
--------	----------------------------

void	<a href="#">setData</a> (byte[] data)
------	---------------------------------------

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXBinaryResponse

public **GXBinaryResponse**(byte[] data)

## Methods

### getData

public byte[] **getData**()

### setData

public void **setData**(byte[] data)

## com.sabratec.applinx.baseobject Class GXClientApplicationConfig

java.lang.Object

└─com.sabratec.applinx.baseobject.GXClientApplicationConfig

### All Implemented Interfaces:

java.io.Serializable

public class **GXClientApplicationConfig**  
extends Object  
implements java.io.Serializable

Data structure that contains the application configuration retrieved from the server.

### See Also:

[GXReadOnlyBaseObject.getApplicationConfig\(\)](#)

## Constructor Summary

public	<a href="#">GXClientApplicationConfig()</a>
public	<a href="#">GXClientApplicationConfig</a> (String applicationName, String hostDeviceType, String hostAddress, int language, boolean trimFields, boolean naturalDataTransferSupported, String gctFileName, int nonActivityTimeout, String deviceName, String keyboardMappingXML, String langConvType) Create a new client application configuration instance.

## Method Summary

String	<a href="#">getApplicationName()</a>
String	<a href="#">getHostAddress()</a>
String	<a href="#">getHostDeviceName()</a>
String	<a href="#">getHostDeviceType()</a>
String	<a href="#">getKeyboardMappingXML()</a>
String	<a href="#">getLangConvType()</a>
int	<a href="#">getLanguage()</a>
int	<a href="#">getNonAtivityTimeOut()</a>
boolean	<a href="#">isNaturalDataTransferSupported()</a>
boolean	<a href="#">isTrimFields()</a>

**Methods inherited from class** `java.lang.Object``equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### **GXClientApplicationConfig**

```
public GXClientApplicationConfig()
```

### **GXClientApplicationConfig**

```
public GXClientApplicationConfig(String applicationName,  
                                String hostDeviceType,  
                                String hostAddress,  
                                int language,  
                                boolean trimFields,  
                                boolean naturalDataTransferSupported,  
                                String gctFileName,  
                                int nonActivityTimeout,  
                                String deviceName,  
                                String keyboardMappingXML,  
                                String langConvType)
```

Create a new client application configuration instance.

## Methods

### **getApplicationName**

```
public String getApplicationName()
```

**Returns:**

the application name

### **getHostDeviceType**

```
public String getHostDeviceType()
```

**Returns:**

the host's device type

### **getHostDeviceName**

```
public String getHostDeviceName()
```

**Returns:**

the host's device name

---

## getHostAddress

```
public String getHostAddress()
```

### Returns:

the host's address

---

## getLanguage

```
public int getLanguage()
```

### Returns:

the application language

---

## isTrimFields

```
public boolean isTrimFields()
```

### Returns:

whether the configuration is defined to trim fields.

---

## isNaturalDataTransferSupported

```
public boolean isNaturalDataTransferSupported()
```

### Returns:

whether the host supports Natural Data Transfer.

---

## getNonAtivityTimeOut

```
public int getNonAtivityTimeOut()
```

### Returns:

the Non Ativity Time Out (milisecends)

---

## getKeyboardMappingXML

```
public String getKeyboardMappingXML()
```

---

## getLangConvType

```
public String getLangConvType()
```

---

## com.sabratec.applinx.baseobject Class GXClientBaseObjectFactory

java.lang.Object

└─com.sabratec.applinx.baseobject.GXClientBaseObjectFactory

public class **GXClientBaseObjectFactory**  
extends Object

### Constructor Summary

public	<a href="#">GXClientBaseObjectFactory()</a>
--------	---

### Method Summary

static void	<a href="#">detachSession</a> ( <a href="#">GXReadOnlyBaseObject</a> baseObject)
static void	<a href="#">endSession</a> ( <a href="#">GXEndSessionRequest</a> endSessionRequest)
static void	<a href="#">endSession</a> ( <a href="#">GXIClientBaseObject</a> baseObject)
static <a href="#">GXIClientBaseObject</a>	<a href="#">getBaseObject</a> ( <a href="#">GXAttachToSessionRequest</a> attachRequest)
static <a href="#">GXIClientBaseObject</a>	<a href="#">getBaseObject</a> ( <a href="#">GXCreateSessionRequest</a> connectRequest)
static <a href="#">GXReadOnlyBaseObject</a>	<a href="#">getReadOnlyBaseObject</a> ( <a href="#">GXAttachToSessionRequest</a> attachRequest)

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXClientBaseObjectFactory

public **GXClientBaseObjectFactory**()

## Methods

### getBaseObject

public static [GXIClientBaseObject](#) **getBaseObject**([GXCreateSessionRequest](#) connectRequest)  
throws [GXConnectionException](#),  
[GXGeneralException](#)

(continued from last page)

---

## getBaseObject

```
public static GXIClientBaseObject getBaseObject(GXAttachToSessionRequest
attachRequest)
    throws GXConnectionException,
           GXGeneralException
```

---

## getReadOnlyBaseObject

```
public static GXIReadOnlyBaseObject getReadOnlyBaseObject(GXAttachToSessionRequest
attachRequest)
    throws GXConnectionException,
           GXGeneralException
```

---

## detachSession

```
public static void detachSession(GXIReadOnlyBaseObject baseObject)
    throws GXGeneralException,
           GXConnectionException
```

---

## endSession

```
public static void endSession(GXIClientBaseObject baseObject)
    throws GXGeneralException,
           GXConnectionException
```

---

## endSession

```
public static void endSession(GXEndSessionRequest endSessionRequest)
    throws GXGeneralException,
           GXConnectionException
```

# com.sabratec.applinx.baseobject

## Class GXConnectionException



```
public class GXConnectionException
extends GXGeneralException
```

Exception indicating that a connection problems occurred between the client and ApplinX server or between ApplinX server and the host.

Constructor Summary	
public	<a href="#">GXConnectionException</a> (String message, Throwable e, int errorCode)

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

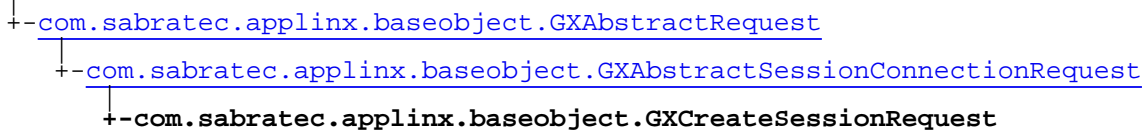
## Constructors

### GXConnectionException

```
public GXConnectionException(String message,
                             Throwable e,
                             int errorCode)
```

## com.sabratec.applinx.baseobject Class GXCreateSessionRequest

java.lang.Object



### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXCreateSessionRequest**  
extends [GXAbstractSessionConnectionRequest](#)

This class holds all the parameters needed for the getBaseObject() method of GXClientBaseObjectFactory.

### Constructor Summary

public	<a href="#">GXCreateSessionRequest()</a>
public	<a href="#">GXCreateSessionRequest</a> (GXIApplicationServerContext serverContext, String applicationName) Constructor for local connection to ApplinX Server.
public	<a href="#">GXCreateSessionRequest</a> (String serverURL, String applicationName, String sessionId, String password, String sessionDescription, String hostSessionId, String hostSessionPassword) Constructor for all parameters besides host service.
public	<a href="#">GXCreateSessionRequest</a> (String serverURL, String applicationName, String sessionId, String password, String sessionDescription, String hostService) Constructor for all parameters.

### Method Summary

String	<a href="#">getConnectionPool()</a>
String	<a href="#">getHostNewPassword()</a>
String	<a href="#">getHostPassword()</a>
String	<a href="#">getHostService()</a> <b>Deprecated.</b> @see <a href="#">getConnectionPool()</a>
String	<a href="#">getHostUserName()</a>
String	<a href="#">getSessionDescription()</a>
boolean	<a href="#">isFailOnWarning()</a>

void	<a href="#"><u>readExternal</u></a> (java.io.ObjectInput in)
void	<a href="#"><u>setConnectionPool</u></a> (String pool) /** Set the session to use a predefined connection pool.
void	<a href="#"><u>setFailOnWarning</u></a> (boolean failOnWarning) Set whether to throw an exception as for error on connection warnings.
void	<a href="#"><u>setHostNewPassword</u></a> (String pwd) Sets a password that will be used to replace the current host password.
void	<a href="#"><u>setHostPassword</u></a> (String pwd) Sets a password that will be used to connect to the host.
void	<a href="#"><u>setHostUserName</u></a> (String id) Sets a user ID that will be used to connect to the host.
void	<a href="#"><u>setSessionDescription</u></a> (String description) Set the session description.
void	<a href="#"><u>setUseHostService</u></a> (String serviceName) <b>Deprecated.</b> @see <a href="#"><u>setConnectionPool(String)</u></a>
void	<a href="#"><u>writeExternal</u></a> (java.io.ObjectOutput out)

**Methods inherited from class**[com.sabratec.applinx.baseobject.GXAbstractSessionConnectionRequest](#)

[getApplicationName](#), [getClientId](#), [getServerContext](#), [getServerURL](#), [getSessionID](#),  
[getSessionPassword](#), [readExternal](#), [setApplicationName](#), [setClientId](#), [setServerContext](#),  
[setServerURL](#), [setSessionID](#), [setSessionPassword](#), [writeExternal](#)

**Methods inherited from class** [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#),  
[readExternal](#), [setVariables](#), [writeExternal](#)

**Methods inherited from class** java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

**Methods inherited from interface** java.io.Externalizable

[readExternal](#), [writeExternal](#)

## Constructors

### GXCreateSessionRequest

```
public GXCreateSessionRequest()
```

(continued from last page)

## GXCreateSessionRequest

```
public GXCreateSessionRequest(GXIApplicationServerContext serverContext,
                             String applicationName)
```

Constructor for local connection to ApplinX Server.

### Parameters:

serverContext - context of ApplinX server.

applicationName - name of a valid application in ApplinX server (at the given URL).

---

## GXCreateSessionRequest

```
public GXCreateSessionRequest(String serverURL,
                             String applicationName,
                             String sessionId,
                             String password,
                             String sessionDescription,
                             String hostSessionId,
                             String hostSessionPassword)
```

Constructor for all parameters besides host service.

### Parameters:

serverURL - URL of ApplinX server.

applicationName - name of a valid application in ApplinX server (at the given URL).

sessionId - ID for the session that will be created on ApplinX server.

password - password for the connection between the base object and ApplinX server.

sessionDescription - description string for the session. It will appear in the sessions list in the Administrator.

hostSessionId - host connection username. Relevant for SSH or Natural UNIX hosts.

hostSessionPassword - host connection password. Relevant for SSH or Natural UNIX hosts.

---

## GXCreateSessionRequest

```
public GXCreateSessionRequest(String serverURL,
                             String applicationName,
                             String sessionId,
                             String password,
                             String sessionDescription,
                             String hostService)
```

Constructor for all parameters.

### Parameters:

serverURL - URL of ApplinX server.

applicationName - name of a valid application in ApplinX server (at the given URL).

sessionId - ID for the session that will be created on ApplinX server.

password - password for the connection between the base object and ApplinX server.

sessionDescription - description string for the session. It will appear in the sessions list in the Administrator.

hostService - connect to a connection from a pool of a host service.

## Methods

### setSessionDescription

```
public void setSessionDescription(String description)
```

Set the session description.

### Parameters:

(continued from last page)

description - the description of this session

---

## getSessionDescription

```
public String getSessionDescription()
```

### Returns:

The session description.

---

## setUseHostService

```
public void setUseHostService(String serviceName)
```

**Deprecated.** @see [setConnectionPool\(String\)](#)

Set the session to use a predefined host service.

### Parameters:

serviceName - the host service to use

---

## setConnectionPool

```
public void setConnectionPool(String pool)
```

/\*\* Set the session to use a predefined connection pool.

### Parameters:

pool - the connection pool to use

---

## getHostService

```
public String getHostService()
```

**Deprecated.** @see [getConnectionPool\(\)](#)

### Returns:

the configured host service of this session. null if no service was defined.

---

## getConnectionPool

```
public String getConnectionPool()
```

### Returns:

the configured host service of this session. null if no service was defined.

---

## getHostUserName

```
public String getHostUserName()
```

### Returns:

The user ID configured for the connection to the host that will be created by this request.

---

(continued from last page)

**See Also:**

[setHostUserName\(String\)](#)

---

## getHostPassword

```
public String getHostPassword()
```

**Returns:**

The password configured for the connection to the host that will be created by this request.

**See Also:**

[setHostUserName\(String\)](#)

---

## getHostNewPassword

```
public String getHostNewPassword()
```

**Returns:**

The password configured for the to replace the current host password.

**See Also:**

[setHostNewPassword\(String\)](#)

---

## setHostUserName

```
public void setHostUserName(String id)
```

Sets a user ID that will be used to connect to the host. Setting host user ID and host user password is relevant only for hosts with **SSH** or host of type **Natural UNIX**.

**Parameters:**

id - The user ID for the connection.

---

## setHostPassword

```
public void setHostPassword(String pwd)
```

Sets a password that will be used to connect to the host.

**Parameters:**

pwd - The password for the connection

**See Also:**

[setHostUserName\(String\)](#)

---

## setHostNewPassword

```
public void setHostNewPassword(String pwd)
```

Sets a password that will be used to replace the current host password.

**Parameters:**

pwd - The new password

---

---

(continued from last page)

**See Also:**

[setHostUserName\(String\)](#)

---

## **isFailOnWarning**

```
public boolean isFailOnWarning()
```

**Returns:**

Whether to throw an exception in case of warnings, as in errors.

---

## **setFailOnWarning**

```
public void setFailOnWarning(boolean failOnWarning)
```

Set whether to throw an exception as for error on connection warnings. For example: if the password is about to expire in a few days.  
By default false.

---

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject Class GXCreateSessionResponse

java.lang.Object

└─com.sabratec.applinx.baseobject.GXCreateSessionResponse

### All Implemented Interfaces:

java.io.Serializable

```
public class GXCreateSessionResponse
    extends Object
    implements java.io.Serializable
```

Response data message for the connect request

### See Also:

ACTION\_SESSION\_CONNECT

## Constructor Summary

public	<a href="#">GXCreateSessionResponse</a> (String userId) Create a new create session response message.
--------	--

## Method Summary

String	<a href="#">getUserId</a> () Return the actual ID of the user.
--------	---

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXCreateSessionResponse

```
public GXCreateSessionResponse(String userId)
```

Create a new create session response message.

#### Parameters:

userId - the actual ID of the connected user.

## Methods

### getUserId

```
public String getUserId()
```

Return the actual ID of the user.

## com.sabratec.applinx.baseobject Class GXCursor

java.lang.Object

└─com.sabratec.applinx.baseobject.GXCursor

### All Implemented Interfaces:

java.io.Serializable

```
public class GXCursor
  extends Object
  implements java.io.Serializable
```

This class represents the cursor position to be set to the host screen. The cursor is set according to the field it is in.

### Constructor Summary

public	<a href="#">GXCursor</a> ()
public	<a href="#">GXCursor</a> ( <a href="#">GXPosition</a> pos) Sets the cursor at the specified position.
public	<a href="#">GXCursor</a> (String fieldName) Sets the cursor to the specified appfield.
public	<a href="#">GXCursor</a> (String fieldName, int index) Sets the cursor to the specified appfield in the specified index.

### Method Summary

int	<a href="#">getFieldIndex</a> ()
String	<a href="#">getFieldName</a> ()
<a href="#">GXPosition</a>	<a href="#">getPosition</a> ()
void	<a href="#">setFieldIndex</a> (int index)
void	<a href="#">setFieldName</a> (String fieldName)
void	<a href="#">setPosition</a> ( <a href="#">GXPosition</a> pos)
String	<a href="#">toString</a> ()

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Constructors

### GXCursor

```
public GXCursor()
```

### GXCursor

```
public GXCursor(GXPosition pos)
```

Sets the cursor at the specified position. If the field is an input field, the cursor will be set at the end of the text in it.

**Parameters:**

pos - the position where the cursor should be set.

### GXCursor

```
public GXCursor(String fieldName)
```

Sets the cursor to the specified appfield. If the field is a label, the cursor will be set at the start position of the field. If the field is an input, the cursor will be set at the end of the text in the field. If the field is a multiple appfield, the cursor will be set to the first index of the appfield.

**Parameters:**

fieldName - the name of the appfield in which the cursor should be set.

### GXCursor

```
public GXCursor(String fieldName,  
                int index)
```

Sets the cursor to the specified appfield in the specified index. If the field is a label, the cursor will be set at the start position of the field. If the field is an input, the cursor will be set at the end of the text in the field.

**Parameters:**

fieldName - the name of the appfield in which the cursor should be set.

index - The index of the multiple appfield in which the cursor should be set.

## Methods

### setPosition

```
public void setPosition(GXPosition pos)
```

**Parameters:**

pos - the position to which to set the cursor position.

**See Also:**

[GXCursor\(GXPosition\)](#)

### getPosition

```
public GXPosition getPosition()
```

(continued from last page)

**Returns:**

the position set for the cursor.

---

**setFieldName**

```
public void setFieldName(String fieldName)
```

**Parameters:**

fieldName - the name of the appfield in which the cursor should be set.

**See Also:**

[GXCursor\(String\)](#)

---

**getFieldName**

```
public String getFieldName()
```

**Returns:**

the field name set for the cursor.

---

**setFieldIndex**

```
public void setFieldIndex(int index)
```

**Parameters:**

index - the index of the multiple appfield in which the cursor should be set.

**See Also:**

[GXCursor\(String, int\)](#)

---

**getFieldIndex**

```
public int getFieldIndex()
```

**Returns:**

the multiple appfield index set for the cursor.

---

**toString**

```
public String toString()
```

## com.sabratec.applinx.baseobject Interface GXDebuggableRequest

All Known Implementing Classes:

[GXTableRequest](#), [GXPathRequest](#), [GXNavigateRequest](#)

public interface **GXDebuggableRequest**

Generic interface for requests that perform an action that can be debugged.

### Method Summary

boolean	<a href="#">isDebugMode</a> ( )
void	<a href="#">setDebugMode</a> ( ) Sets the request to be executed in debug mode.

### Methods

#### setDebugMode

public void **setDebugMode**( )

Sets the request to be executed in debug mode.

#### isDebugMode

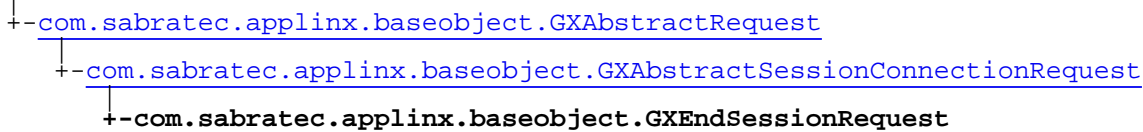
public boolean **isDebugMode**( )

**Returns:**

the status of the debug mode flag.

## com.sabratec.applinx.baseobject Class GXEndSessionRequest

java.lang.Object



### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXEndSessionRequest**  
extends [GXAbstractSessionConnectionRequest](#)

Settings for disconnect request.

### Constructor Summary

public	<a href="#">GXEndSessionRequest()</a>
public	<a href="#">GXEndSessionRequest</a> (GXIApplicationServerContext serverContext)

### Method Summary

void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class

[com.sabratec.applinx.baseobject.GXAbstractSessionConnectionRequest](#)

[getApplicationName](#), [getClientId](#), [getServerContext](#), [getServerURL](#), [getSessionID](#), [getSessionPassword](#), [readExternal](#), [setApplicationName](#), [setClientId](#), [setServerContext](#), [setServerURL](#), [setSessionID](#), [setSessionPassword](#), [writeExternal](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

(continued from last page)

## Constructors

### **GXEndSessionRequest**

```
public GXEndSessionRequest()
```

---

### **GXEndSessionRequest**

```
public GXEndSessionRequest(GXIApplicationServerContext serverContext)
```

#### **Parameters:**

serverContext - the server context to disconnect from.

## Methods

### **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject

### Class GXField

java.lang.Object

└─com.sabratec.applinx.baseobject.GXField

All Implemented Interfaces:

java.io.Serializable, [GXIField](#)

public class **GXField**  
 extends Object  
 implements [GXIField](#), java.io.Serializable

This class represents one field on the host screen. This field can be a host field or an application field, or an instance within a multiple appfield.

#### Field Summary

public static final	<a href="#">FIELD_TYPE_APPFIELD</a> Value: 0
public static final	<a href="#">FIELD_TYPE_HOSTFIELD</a> Value: 1

Fields inherited from interface [com.sabratec.applinx.baseobject.GXIField](#)

[TYPE\\_APPFIELD](#), [TYPE\\_HOSTFIELD](#), [TYPE\\_HOSTKEY\\_FIELD](#), [TYPE\\_MULTIPLE\\_APPFIELD](#)

#### Constructor Summary

public	<a href="#">GXField()</a>
--------	---------------------------

#### Method Summary

boolean	<a href="#">equals</a> (Object obj)
int	<a href="#">getBGColor</a> ()
String	<a href="#">getContent</a> ()
int	<a href="#">getDataType</a> ()
int	<a href="#">getFGColor</a> ()
int	<a href="#">getIndex</a> ()
String	<a href="#">getInputMask</a> ()

int	<a href="#">getLength()</a>
String	<a href="#">getName()</a>
<a href="#">GXPosition</a>	<a href="#">getPosition()</a>
String	<a href="#">getScreenName()</a>
int	<a href="#">getScreenSequenceNumber()</a>
String	<a href="#">getVisualContent()</a>
boolean	<a href="#">isAppField()</a>
boolean	<a href="#">isAutoCursorJump()</a>
boolean	<a href="#">isAutoEnter()</a>
boolean	<a href="#">isBlinking()</a>
boolean	<a href="#">isHighIntensified()</a>
boolean	<a href="#">isHostKeyField()</a>
boolean	<a href="#">isMultiple()</a>
boolean	<a href="#">isProtected()</a>
boolean	<a href="#">isReversedVideo()</a>
boolean	<a href="#">isUnderlined()</a>
boolean	<a href="#">isVisible()</a>

**Methods inherited from class** `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface** `com.sabratec.applinx.baseobject.GXIField`

[getBGColor](#), [getContent](#), [getDataType](#), [getFGColor](#), [getIndex](#), [getInputMask](#), [getLength](#), [getName](#), [getPosition](#), [getScreenName](#), [getScreenSequenceNumber](#), [getVisualContent](#), [isAppField](#), [isAutoCursorJump](#), [isAutoEnter](#), [isBlinking](#), [isHighIntensified](#), [isMultiple](#), [isProtected](#), [isReversedVideo](#), [isUnderlined](#), [isVisible](#)

**Methods inherited from interface** `com.sabratec.applinx.baseobject.table.GXICellable`

[getContent](#)

(continued from last page)

## Fields

### FIELD\_TYPE\_APPFIELD

```
public static final int FIELD_TYPE_APPFIELD
```

Constant value: 0

### FIELD\_TYPE\_HOSTFIELD

```
public static final int FIELD_TYPE_HOSTFIELD
```

Constant value: 1

## Constructors

### GXField

```
public GXField()
```

## Methods

### getScreenName

```
public String getScreenName()
```

### getScreenSequenceNumber

```
public int getScreenSequenceNumber()
```

### getName

```
public String getName()
```

### getIndex

```
public int getIndex()
```

### getPosition

```
public GXPosition getPosition()
```

(continued from last page)

---

## getLength

```
public int getLength()
```

---

---

## getContent

```
public String getContent()
```

---

---

## isProtected

```
public boolean isProtected()
```

---

---

## isVisible

```
public boolean isVisible()
```

---

---

## isHighIntensified

```
public boolean isHighIntensified()
```

---

---

## isAutoEnter

```
public boolean isAutoEnter()
```

---

---

## getBGColor

```
public int getBGColor()
```

---

---

## getFGColor

```
public int getFGColor()
```

---

---

## isAutoCursorJump

```
public boolean isAutoCursorJump()
```

---

---

## isAppField

```
public boolean isAppField()
```

---

(continued from last page)

---

## isHostKeyField

```
public boolean isHostKeyField()
```

---

## isMultiple

```
public boolean isMultiple()
```

---

## isBlinking

```
public boolean isBlinking()
```

---

## getDataType

```
public int getDataType()
```

---

## getInputMask

```
public String getInputMask()
```

---

## equals

```
public boolean equals(Object obj)
```

---

## getVisualContent

```
public String getVisualContent()
```

---

## isUnderlined

```
public boolean isUnderlined()
```

---

## isReversedVideo

```
public boolean isReversedVideo()
```

## com.sabratec.applinx.baseobject Class GXFieldCollection

java.lang.Object

└─com.sabratec.applinx.baseobject.GXFieldCollection

All Implemented Interfaces:

java.io.Serializable, [GXFieldCollection](#)

public class **GXFieldCollection**  
extends Object  
implements [GXFieldCollection](#), java.io.Serializable

This class is a collection of fields. It analyzes the fields array and allows retrieval of fields according to name, multiple index, etc.

### Constructor Summary

public	<a href="#">GXFieldCollection</a> ( <a href="#">GXIField[]</a> fields)
public	<a href="#">GXFieldCollection</a> (Vector fields)

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
<a href="#">GXFieldCollection</a>	<a href="#">getAppFields</a> ()
<a href="#">GXIField</a>	<a href="#">getFieldAt</a> (int index)
<a href="#">GXIField</a>	<a href="#">getFieldByPosition</a> ( <a href="#">GXPosition</a> position)
int	<a href="#">getFieldCount</a> ()
<a href="#">GXIField</a>	<a href="#">getFieldInstance</a> (String name, int fieldIndex)
<a href="#">GXIField[]</a>	<a href="#">getFields</a> ()
<a href="#">GXFieldCollection</a>	<a href="#">getFields</a> (String name)
<a href="#">GXIField</a>	<a href="#">getFirstFieldInstance</a> (String name)
<a href="#">GXIField</a>	<a href="#">getLastFieldInstance</a> (String name)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.GXFieldCollection](#)

---

```
getAppFields, getFieldAt, getFieldByPosition, getFieldCount, getFieldInstance,  
getFields, getFields, getFirstFieldInstance, getLastFieldInstance
```

---

## Constructors

### **GXFieldCollection**

```
public GXFieldCollection(GXIField\[\] fields)
```

---

### **GXFieldCollection**

```
public GXFieldCollection(Vector fields)
```

---

## Methods

### **getFields**

```
public GXIField\[\] getFields()
```

---

### **getFieldAt**

```
public GXIField getFieldAt(int index)
```

---

### **getFieldCount**

```
public int getFieldCount()
```

---

### **getFields**

```
public GXIFieldCollection getFields(String name)
```

---

### **getAppFields**

```
public GXIFieldCollection getAppFields()
```

#### **Returns:**

all appfields in current collection.

---

(continued from last page)

## getFirstFieldInstance

```
public GXIField getFirstFieldInstance(String name)
```

**Returns:**

first instance for this appfield in current collection.

---

## getLastFieldInstance

```
public GXIField getLastFieldInstance(String name)
```

**Returns:**

last instance for this appfield in current collection.

---

## getFieldInstance

```
public GXIField getFieldInstance(String name,  
                                int fieldIndex)
```

**Returns:**

the instance at the given index of this appfield.

---

## getFieldByPosition

```
public GXIField getFieldByPosition(GXPosition position)
```

**Returns:**

hostfield/appfield by position.

---

## equals

```
public boolean equals(Object obj)
```

## com.sabratec.applinx.baseobject Class GXGeneralException

java.lang.Object

└-GXExceptionWithErrorCode

└-com.sabratec.applinx.baseobject.GXGeneralException

Direct Known Subclasses:

[GXColumnNotFoundException](#), [GXPathException](#), [GXNavigateException](#), [GXNaturalDataTransferException](#), [GXConnectionException](#)

```
public class GXGeneralException
extends GXExceptionWithErrorCode
```

Exception indicating general error.

### Constructor Summary

public	<a href="#">GXGeneralException</a> (String message, Throwable e, int errorCode)
--------	---

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

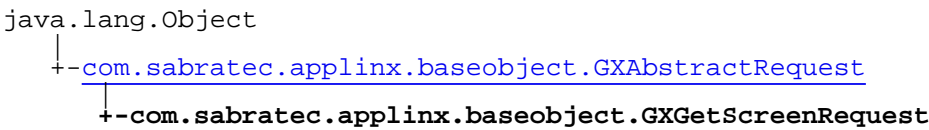
## Constructors

### GXGeneralException

```
public GXGeneralException(String message,
                          Throwable e,
                          int errorCode)
```

# com.sabratec.applinx.baseobject

## Class GXGetScreenRequest



All Implemented Interfaces:  
Cloneable, java.io.Externalizable

public class **GXGetScreenRequest**  
extends [GXAbstractRequest](#)

Settings for the getScreen request.

### Constructor Summary

public	<a href="#">GXGetScreenRequest</a> ()
--------	---------------------------------------

### Method Summary

void	<a href="#">addWaitCondition</a> ( <a href="#">GXWaitCondition</a> waitCondition) Adds a wait condition to the getScreen request.
<a href="#">GXWaitCondition[]</a>	<a href="#">getWaitCoditions</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

Methods inherited from class <a href="#">com.sabratec.applinx.baseobject.GXAbstractRequest</a>
<a href="#">addVariable</a> , <a href="#">addVariable</a> , <a href="#">addVariableArray</a> , <a href="#">clone</a> , <a href="#">equals</a> , <a href="#">getVariables</a> , <a href="#">hashCode</a> , <a href="#">readExternal</a> , <a href="#">setVariables</a> , <a href="#">writeExternal</a>

Methods inherited from class java.lang.Object
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>

Methods inherited from interface java.io.Externalizable
<a href="#">readExternal</a> , <a href="#">writeExternal</a>

### Constructors

**GXGetScreenRequest**  
public **GXGetScreenRequest**()

(continued from last page)

## Methods

### addWaitCondition

```
public void addWaitCondition(GXWaitCondition waitCondition)
```

Adds a wait condition to the getScreen request. The wait conditions will be executed with a logical AND between them according to the order they were added to the request. Each type of wait condition has its own class, which implements GXWaitCondition.

**Parameters:**

waitCondition - the wait condition to add to the request.

### getWaitCodings

```
public GXWaitCondition\[\] getWaitCodings()
```

### readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

### writeExternal

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

# com.sabratec.applinx.baseobject

## Interface GXIClientBaseObject

All Superinterfaces:

[GXIReadOnlyBaseObject](#)

All Subinterfaces:

[GXINaturalDataTransferClientBaseObject](#)

public interface **GXIClientBaseObject**  
 extends [GXIReadOnlyBaseObject](#)

This object represents a live session against the ApplinX server and the host. The different methods in the base object session connect to ApplinX server, execute an action on the host and return a response.

Method Summary	
<a href="#">GXPathResponse</a>	<a href="#">executePath</a> ( <a href="#">GXPathRequest</a> request) Executes a path on the host, and returns the response of the path (screens, variables).
<a href="#">GXPathResponse</a>	<a href="#">executePath</a> (String pathName) Executes a path on the host, and returns the response of the path (screens, variables).
GXIProcedureResponse	<a href="#">executeProcedure</a> (GXIExtendedProcedureRequest request) Use this method to execute Procedure Clients generated in the current ApplinX version
GXIProcedureResponse	<a href="#">executeProcedure</a> (GXIProcedureRequest request) <b>Deprecated.</b> Execute the Procedure client with the <code>GXIClientBaseObject.executeProcedure(GXIProcedureRequest, String, String)</code> method, or with the <code>GXIClientBaseObject.executeProcedure(GXIExtendedProcedureRequest)</code> method.
GXIProcedureResponse	<a href="#">executeProcedure</a> (GXIProcedureRequest request, String serviceName, String methodName) Use this method in order to execute Procedure Clients generated in ApplinX version 5.x, 8.0 and 8.1
<a href="#">GXITable</a>	<a href="#">getTable</a> () <b>Deprecated.</b> Replaced with <code>GXIClientBaseObject.getScreen().getTable()</code> which returns a table associated with the current screen
<a href="#">GXITable</a>	<a href="#">getTable</a> ( <a href="#">GXTableRequest</a> request) <b>Deprecated.</b> Replaced with <code>GXIClientBaseObject.executeProcedure</code> which returns structured data. Refer to the documentation for further details.
<a href="#">GXITable</a>	<a href="#">getTable</a> (String tableName) <b>Deprecated.</b> Replaced with <code>GXIClientBaseObject.executeProcedure</code> which returns structured data.
void	<a href="#">navigateTo</a> ( <a href="#">GXNavigateRequest</a> request) Navigates from the current screen to the destination screen using a given map.
void	<a href="#">refreshScreen</a> () Refreshes the current local screen that <code>getScreen()</code> returns with contents from the host.

void	<a href="#">sendKeys</a> ( <a href="#">GXSendKeysRequest</a> request) A request that sets input fields, cursor position and sends a key to the host.
void	<a href="#">sendKeys</a> (String key) A request that sends a key to the host.

Methods inherited from interface [com.sabratec.applinx.baseobject.GXIClientBaseObject](#)

[getApplicationConfig](#), [getScreen](#), [getScreen](#), [getSeqScreenNumber](#), [getSessionID](#), [isReadOnly](#)

## Methods

### sendKeys

```
public void sendKeys(GXSendKeysRequest request)
    throws GXConnectionException,
           GXGeneralException
```

A request that sets input fields, cursor position and sends a key to the host. If not host key is specified [enter] is sent to the host.

#### Parameters:

request - the sendKeys request.

#### Throws:

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.  
[GXGeneralException](#) - Other errors.

### sendKeys

```
public void sendKeys(String key)
    throws GXConnectionException,
           GXGeneralException
```

A request that sends a key to the host.

#### Parameters:

key - the host key to send.

#### Throws:

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.  
[GXGeneralException](#) - Other errors.

### executePath

```
public GXPathResponse executePath(GXPathRequest request)
    throws GXConnectionException,
           GXPathException,
           GXGeneralException
```

Executes a path on the host, and returns the response of the path (screens, variables).

#### Parameters:

request - the path request.

(continued from last page)

**Returns:**

the path's response.

**Throws:**[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.[GXPathException](#) - Errors in path execution.[GXGeneralException](#) - Other errors.

---

## executePath

```
public GXPathResponse executePath(String pathName)
    throws GXConnectionException,
           GXPathException,
           GXGeneralException
```

Executes a path on the host, and returns the response of the path (screens, variables).

**Parameters:**

pathName - the name of the path to be executed.

**Returns:**

the path's response.

**Throws:**[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.[GXPathException](#) - Errors in path execution.[GXGeneralException](#) - Other errors.

---

## executeProcedure

```
public GXIPProcedureResponse executeProcedure(GXIPProcedureRequest request,
        String serviceName,
        String methodName)
    throws GXConnectionException,
           GXGeneralException
```

Use this method in order to execute Procedure Clients generated in ApplinX version 5.x, 8.0 and 8.1

**Parameters:**

request - the request object of the method. The class is a generated class of procedure client, e.g: <ProcedureName>Request

**Returns:**

the response object. The class is a generated class of procedure client, e.g: <ProcedureName>Response

**Throws:**[GXConnectionException](#)[GXGeneralException](#)

---

## executeProcedure

```
public GXIPProcedureResponse executeProcedure(GXIPProcedureRequest request)
    throws GXConnectionException,
           GXGeneralException
```

**Deprecated.** *Execute the Procedure client with the*

*`GXIClientBaseObject.executeProcedure(GXIPProcedureRequest, String, String)` method, or with the `GXIClientBaseObject.executeProcedure(GXIPExtendedProcedureRequest)` method.*

(continued from last page)

**Parameters:**

request - the request object of the method. The class is a generated class of procedure client, e.g: <ProcedureName>Request

**Returns:**

the response object. The class is a generated class of procedure client, e.g: <ProcedureName>Response

**Throws:**

[GXConnectionException](#)

[GXGeneralException](#)

---

## executeProcedure

```
public GXIPProcedureResponse executeProcedure(GXIExtendedProcedureRequest request)
    throws GXConnectionException,
           GXGeneralException
```

Use this method to execute Procedure Clients generated in the current ApplinX version

**Parameters:**

request - the request object of the method. The class is a generated class of procedure client, e.g: <ProcedureName>Request

**Returns:**

the response object. The class is a generated class of procedure client, e.g: <ProcedureName>Response

**Throws:**

[GXConnectionException](#)

[GXGeneralException](#)

---

## getTable

```
public GXITable getTable(GXTableRequest request)
    throws GXConnectionException,
           GXGeneralException
```

**Deprecated.** Replaced with `GXIClientBaseObject.executeProcedure` which returns structured data. Refer to the documentation for further details.

This method retrieves a table from ApplinX server according to the GXTableRequest.

**Parameters:**

request - the table request.

**Returns:**

the table.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - Other errors.

---

## getTable

```
public GXITable getTable(String tableName)
    throws GXConnectionException,
           GXGeneralException
```

**Deprecated.** Replaced with `GXIClientBaseObject.executeProcedure` which returns structured data.

(continued from last page)

This method retrieves the path or screen associated table of the specified name from the ApplinX server.

**Parameters:**

tableName - the table's name as defined in the ApplinX repository.

**Returns:**

the table.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - Other errors.

## getTable

```
public GXITable getTable()
    throws GXConnectionException,
           GXGeneralException
```

**Deprecated.** Replaced with `GXIClientBaseObject.getScreen().getTable()` which returns a table associated with the current screen

This method retrieves a table for the current host screen, if such is defined, including its full details. This is assuming the screen contains only one table in it. If more than one table exists for the screen, no table is returned.

**Returns:**

the screen's table.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - Other errors.

## navigateTo

```
public void navigateTo(GXNavigateRequest request)
    throws GXConnectionException,
           GXNavigateException,
           GXGeneralException
```

Navigates from the current screen to the destination screen using a given map.

**Parameters:**

request - the navigation request.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

[GXNavigateException](#) - Errors in navigation process.

[GXGeneralException](#) - Other errors.

## refreshScreen

```
public void refreshScreen()
    throws GXConnectionException,
           GXGeneralException
```

Refreshes the current local screen that `getScreen()` returns with contents from the host.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

(continued from last page)

[GXGeneralException](#) - Other errors.

**See Also:**

[GXIReadOnlyBaseObject.getScreen\(\)](#)

## com.sabratec.applinx.baseobject Interface GXIField

All Superinterfaces:

[GXICellable](#)

All Known Implementing Classes:

[GXFieldTableCell](#), [GXField](#)

public interface **GXIField**  
extends [GXICellable](#)

This class represents one field in the host screen. When there are multiple application fields, it represents one index of the multiple application field.

Note: The field object is a snapshot of a field as it was taken from a host screen. You cannot set properties to the field. In order to set content to fields on the host screen, use the session object and the request objects.

### Field Summary

public static final	<a href="#">TYPE_APPFIELD</a> Value: 1
public static final	<a href="#">TYPE_HOSTFIELD</a> Value: 0
public static final	<a href="#">TYPE_HOSTKEY_FIELD</a> Value: 3
public static final	<a href="#">TYPE_MULTIPLE_APPFIELD</a> Value: 2

### Method Summary

int	<a href="#">getBGColor()</a>
String	<a href="#">getContent()</a>
int	<a href="#">getDataType()</a>
int	<a href="#">getFGColor()</a>
int	<a href="#">getIndex()</a>
String	<a href="#">getInputMask()</a>
int	<a href="#">getLength()</a>

String	<a href="#">getName()</a>
<a href="#">GXPosition</a>	<a href="#">getPosition()</a>
String	<a href="#">getScreenName()</a>
int	<a href="#">getScreenSequenceNumber()</a>
String	<a href="#">getVisualContent()</a>
boolean	<a href="#">isAppField()</a>
boolean	<a href="#">isAutoCursorJump()</a>
boolean	<a href="#">isAutoEnter()</a>
boolean	<a href="#">isBlinking()</a>
boolean	<a href="#">isHighIntensified()</a>
boolean	<a href="#">isMultiple()</a>
boolean	<a href="#">isProtected()</a>
boolean	<a href="#">isReversedVideo()</a>
boolean	<a href="#">isUnderlined()</a>
boolean	<a href="#">isVisible()</a>

Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXICellable](#)

[getContent](#)

## Fields

### TYPE\_HOSTFIELD

```
public static final int TYPE_HOSTFIELD
```

Constant value: 0

### TYPE\_APPFIELD

```
public static final int TYPE_APPFIELD
```

Constant value: 1

---

## TYPE\_MULTIPLE\_APPFIELD

```
public static final int TYPE_MULTIPLE_APPFIELD
```

Constant value: 2

---

## TYPE\_HOSTKEY\_FIELD

```
public static final int TYPE_HOSTKEY_FIELD
```

Constant value: 3

## Methods

### getScreenName

```
public String getScreenName()
```

**Returns:**

the name (ID) of the screen, from which this field was taken.

---

### getScreenSequenceNumber

```
public int getScreenSequenceNumber()
```

**Returns:**

the sequential number of the screen, from which this field was taken.

---

### getName

```
public String getName()
```

**Returns:**

Returns the name of the field (for application fields). If the field is a host field, it returns null.

---

### getIndex

```
public int getIndex()
```

**Returns:**

the index of the field instance (if it is part of a multiple application field).

---

### getPosition

```
public GXPosition getPosition()
```

(continued from last page)

**Returns:**

the position of the field in the host screen.

---

**getLength**

```
public int getLength()
```

**Returns:**

the length of the field.

---

**getContent**

```
public String getContent()
```

**Returns:**

the content of the field (as it was taken from the host screen).

---

**isProtected**

```
public boolean isProtected()
```

**Returns:**

whether this field is a protected field.

---

**isVisible**

```
public boolean isVisible()
```

**Returns:**

whether this field is visible on the host screen.

---

**isHighIntensified**

```
public boolean isHighIntensified()
```

**Returns:**

whether this field is an intensified field.

---

**isAutoEnter**

```
public boolean isAutoEnter()
```

**Returns:**

whether this field is an auto enter field (once the field content completely fills the field, it is sent to the host screen).

---

## getBGColor

```
public int getBGColor()
```

**Returns:**

the background color of this field.

---

## getFGColor

```
public int getFGColor()
```

**Returns:**

the foreground color of this field.

---

## isAutoCursorJump

```
public boolean isAutoCursorJump()
```

**Returns:**

whether this field is an auto cursor jump field. This means that the cursor should jump automatically to the next field when this field is completely filled with content.

---

## isAppField

```
public boolean isAppField()
```

**Returns:**

whether this field is an application field.

---

## isMultiple

```
public boolean isMultiple()
```

**Returns:**

whether this field is part of a multiple application field.

---

## isBlinking

```
public boolean isBlinking()
```

**Returns:**

whether this field is blinking.

---

---

(continued from last page)

## getDataType

```
public int getDataType()
```

### Returns:

the field's data type

### See Also:

[GXBaseObjectConstants for valid data types](#)

---

## getInputMask

```
public String getInputMask()
```

### Returns:

the field's input mask. Valid only for unprotected application fields.

---

## getVisualContent

```
public String getVisualContent()
```

### Returns:

the field's content in its visual form, without a conversion to logical.

---

## isUnderlined

```
public boolean isUnderlined()
```

### Returns:

whether this field is underlined.

---

## isReversedVideo

```
public boolean isReversedVideo()
```

### Returns:

whether this field is reversed video.

---

## com.sabratec.applinx.baseobject Interface GXIFieldCollection

All Known Implementing Classes:  
[GXIFieldCollection](#)

public interface **GXIFieldCollection**

This class represents a collection of fields as they were gathered from a screen or many screens. It should be used to perform queries on the fields.

### Method Summary

<a href="#">GXIFieldCollection</a>	<a href="#">getAppFields()</a>
<a href="#">GXIField</a>	<a href="#">getFieldAt(int index)</a>
<a href="#">GXIField</a>	<a href="#">getFieldByPosition(GXPosition position)</a>
int	<a href="#">getFieldCount()</a>
<a href="#">GXIField</a>	<a href="#">getFieldInstance(String name, int fieldIndex)</a>
<a href="#">GXIField[]</a>	<a href="#">getFields()</a>
<a href="#">GXIFieldCollection</a>	<a href="#">getFields(String name)</a>
<a href="#">GXIField</a>	<a href="#">getFirstFieldInstance(String name)</a>
<a href="#">GXIField</a>	<a href="#">getLastFieldInstance(String name)</a>

### Methods

#### getFieldCount

```
public int getFieldCount()
```

**Returns:**

the number of fields in the collection.

#### getFieldAt

```
public GXIField getFieldAt(int index)
```

**Parameters:**

---

(continued from last page)

index - the index of the field in the collection.

**Returns:**

the field at the given index from the collection.

---

## getFields

```
public GXIField\[\] getFields()
```

**Returns:**

the field array on which the collection was constructed.

---

## getAppFields

```
public GXIFieldCollection getAppFields()
```

**Returns:**

all the application fields in the current collection.

---

## getFieldByPosition

```
public GXIField getFieldByPosition(GXPosition position)
```

**Parameters:**

position - the position of the first character of the field.

**Returns:**

the field that starts at the given position. This method should be used when the field collection is from one screen. If the collection collects more than one screen, the positions may collide.

---

## getFieldInstance

```
public GXIField getFieldInstance(String name,  
                                int fieldIndex)
```

**Parameters:**

name - multiple application field name

fieldIndex - index within the multiple application field.

**Returns:**

the instance of the field at the given index.

---

## getFirstFieldInstance

```
public GXIField getFirstFieldInstance(String name)
```

**Parameters:**

name - the name of an application field in the collection

---

(continued from last page)

**Returns:**

the first instance of this application field that was added to the collection.

---

## getLastFieldInstance

```
public GXIField getLastFieldInstance(String name)
```

**Parameters:**

name - application field name

**Returns:**

the last instance (currently available in the collection) of the given application field.

---

## getFields

```
public GXIFieldCollection getFields(String name)
```

**Parameters:**

name - name of an application field

**Returns:**

a collection holding all the instances of the application field.

---

## com.sabratec.applinx.baseobject Interface GXIHostKey

---

public interface **GXIHostKey**

---

### Method Summary

String	<a href="#">getAction()</a>
String	<a href="#">getCaption()</a>
<a href="#">GXPosition</a>	<a href="#">getDisplayPosition()</a>
String	<a href="#">getDisplayText()</a>

---

### Methods

#### **getCaption**

public String **getCaption()**

---

#### **getAction**

public String **getAction()**

---

#### **getDisplayText**

public String **getDisplayText()**

---

#### **getDisplayPosition**

public [GXPosition](#) **getDisplayPosition()**

## com.sabratec.applinx.baseobject Interface GXINaturalDataTransferClientBaseObject

All Superinterfaces:

[GXIClientBaseObject](#), [GXIReadOnlyBaseObject](#)

public interface **GXINaturalDataTransferClientBaseObject**  
extends [GXIClientBaseObject](#)

This object is an extension of the [GXIClientBaseObject](#) that provides the option to perform Natural Data Transfer (NDT) actions.

### Method Summary

void	<a href="#">confirmCommand()</a> Confirm the execution of a Natural Data Transfer command.
<a href="#">GXINaturalDataTransferDownloadResponse</a>	<a href="#">continueDownload()</a> Continues the download process.
<a href="#">GXINaturalDataTransferUploadResponse</a>	<a href="#">continueUpload()</a> Continues the process of uploading a file.
void	<a href="#">rejectTransfer()</a> Sends an indication to the host that a transfer process should be rejected.
<a href="#">GXINaturalDataTransferDownloadResponse</a>	<a href="#">startDownload(<a href="#">GXINaturalDataTransferDownloadRequest</a> request)</a> Starts the download process.
<a href="#">GXINaturalDataTransferUploadResponse</a>	<a href="#">startUpload(<a href="#">GXINaturalDataTransferUploadRequest</a> request)</a> Starts the process of uploading a file.

#### Methods inherited from interface [com.sabratec.applinx.baseobject.GXIClientBaseObject](#)

[executePath](#), [executePath](#), [executeProcedure](#), [executeProcedure](#), [executeProcedure](#),  
[getTable](#), [getTable](#), [getTable](#), [navigateTo](#), [refreshScreen](#), [sendKeys](#), [sendKeys](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.GXIReadOnlyBaseObject](#)

[getApplicationConfig](#), [getScreen](#), [getScreen](#), [getSeqScreenNumber](#), [getSessionID](#),  
[isReadOnly](#)

### Methods

#### startDownload

```
public GXINaturalDataTransferDownloadResponse
startDownload(GXINaturalDataTransferDownloadRequest request)
    throws GXConnectionException,
           GXINaturalDataTransferException,
           GXGeneralException
```

(continued from last page)

Starts the download process. The server will download a file from the host, and will return it when it has been fully downloaded.

Note that this method may return before the download is completed.

**Parameters:**

fileType - Type of downloaded file. Supported types:

- `FORMAT_TYPE_TXT`
- `FORMAT_TYPE_CSV`

**Throws:**

[GXNaturalDataTransferException](#) - when there was a problem with the data transfer

[GXConnectionException](#) - When connection problems occur between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - when there is a general problem

---

## continueDownload

```
public GXNaturalDataTransferDownloadResponse continueDownload()  
throws GXConnectionException,  
       GXNaturalDataTransferException,  
       GXGeneralException
```

Continues the download process. The server will continue a previous download from the host, and will return the download data only if the download process was completed. This method may be called only after calling `startDownload()`.

Note that this method may return before the download is completed.

**Throws:**

[GXNaturalDataTransferException](#) - if there is a problem with the data transfer

[GXConnectionException](#) - When connection problems occur between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - if there is a general problem

---

## startUpload

```
public GXNaturalDataTransferUploadResponse  
startUpload(GXNaturalDataTransferUploadRequest request)  
throws GXConnectionException,  
       GXNaturalDataTransferException,  
       GXGeneralException
```

Starts the process of uploading a file. The server will upload a file to the host.

Note that this method may return before the upload is completed.

**Parameters:**

request - the upload request details

**Throws:**

[GXNaturalDataTransferException](#) - if there is a problem with the data transfer

[GXConnectionException](#) - When connection problems occur between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - if there is a general problem

---

## continueUpload

```
public GXNaturalDataTransferUploadResponse continueUpload()  
throws GXConnectionException,  
       GXNaturalDataTransferException,  
       GXGeneralException
```

(continued from last page)

Continues the process of uploading a file. The server will upload a previously given file to the host. This method may be called only after calling `startUpload(InputStream)`.

Note that this method may return before the upload is completed.

**Throws:**

[GXNaturalDataTransferException](#) - if there is a problem with the data transfer

[GXConnectionException](#) - When connection problems occur between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - if there is a general problem

---

## rejectTransfer

```
public void rejectTransfer()  
    throws GXConnectionException,  
           GXNaturalDataTransferException,  
           GXGeneralException
```

Sends an indication to the host that a transfer process should be rejected. It applies for both upload, download and a command.

**Throws:**

[GXNaturalDataTransferException](#) - if there is a problem with the data transfer

[GXConnectionException](#) - When connection problems occur between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - if there is a general problem

---

## confirmCommand

```
public void confirmCommand()  
    throws GXConnectionException,  
           GXNaturalDataTransferException,  
           GXGeneralException
```

Confirm the execution of a Natural Data Transfer command.

When the command is an asynchronous command, the confirmation should be sent after the command execution started successfully.

When the command is a synchronous command, the confirmation should be sent only after the command execution finished successfully.

**Throws:**

[GXNaturalDataTransferException](#) - if there is a problem with the data transfer

[GXConnectionException](#) - When connection problems occur between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - if there is a general problem

## com.sabratec.applinx.baseobject Interface GXINaturalDataTransferScreen

All Superinterfaces:

[GXIScreen](#)

public interface **GXINaturalDataTransferScreen**

extends [GXIScreen](#)

Interface for a screen that contain Natural Data Transfer (NDT) information.

### Method Summary

String	<a href="#">getDataTransferCommand()</a>
String	<a href="#">getDataTransferFilename()</a>
int	<a href="#">getDataTransferState()</a> Returns the state of the Natural Data Transfer screen.
String	<a href="#">getDataTransferWorkfileCode()</a>
boolean	<a href="#">isDataTransferCommandSynchronous()</a>

#### Methods inherited from interface [com.sabratec.applinx.baseobject.GXIScreen](#)

[getBuffers](#), [getBuffers](#), [getColumns](#), [getCursorPosition](#), [getFieldByCursorPosition](#), [getFieldContent](#), [getFieldContent](#), [getFieldInstance](#), [getFields](#), [getFields](#), [getFields](#), [getFields](#), [getFolderPath](#), [getHostKeys](#), [getName](#), [getRows](#), [getScreenGroupNames](#), [getSeqNumber](#), [getTable](#), [getTables](#), [getWindow](#), [getWindowLowerRight](#), [getWindows](#), [getWindowUpperLeft](#), [hasAlarm](#), [isDirectionRTL](#), [isDisplayChanged](#), [isLocked](#), [isMemberOf](#), [isTabbingRTL](#), [isWindow](#)

### Methods

#### **getDataTransferState**

public int **getDataTransferState()**

Returns the state of the Natural Data Transfer screen.

**Returns:**

(continued from last page)

type of the Natural Data Transfer screen. If this is a normal screen, STATE\_NOT\_NDT\_SCREEN will be returned.

Can be one of the following:

- STATE\_UNKNOWN
- STATE\_NOT\_NDT\_SCREEN
- STATE\_DOWNLOAD\_BEGINNING
- STATE\_DOWNLOAD\_IN\_PROCESS
- STATE\_UPLOAD\_BEGINNING
- STATE\_UPLOAD\_IN\_PROCESS
- STATE\_COMMAND
- STATE\_ERROR

---

## getDataTransferFilename

```
public String getDataTransferFilename()
```

### Returns:

the file name suggested by the host. null if the host did not suggest one (most common case).

---

## getDataTransferWorkfileCode

```
public String getDataTransferWorkfileCode()
```

### Returns:

the workfile code, which is the unique file identifier on the host.

---

## getDataTransferCommand

```
public String getDataTransferCommand()
```

### Returns:

the command that was sent to the client thru Natural Data Transfer

---

## isDataTransferCommandSynchronous

```
public boolean isDataTransferCommandSynchronous()
```

### Returns:

Whether the Natural Data Transfer command is synchronous. If the command is synchronous the client should reply only after performing the command. If the command is asynchronous, the client should reply right after validating it.

## com.sabratec.applinx.baseobject Class GXInnerClientBaseObjectFactory

java.lang.Object

└─com.sabratec.applinx.baseobject.GXInnerClientBaseObjectFactory

public class **GXInnerClientBaseObjectFactory**  
extends Object

General factory to create connected / attached base objects, detach and disconnect them from ApplinX server.

There are three acceptable protocols, which dictate different communication modes between the base object and the server.

- ApplinX - Used to communicate with the server over a socket.
- Local (direct) - Used to communicate with an in-process server. All base objects communicate with a single server object. Currently used when the server is loaded as a servlet.

**This class was created for internal use, please don't use it directly , use [GXClientBaseObjectFactory](#) instead**

### Constructor Summary

public	<a href="#">GXInnerClientBaseObjectFactory</a> ()
--------	---

### Method Summary

static void	<a href="#">detachSession</a> ( <a href="#">GXReadOnlyBaseObject</a> baseObject) Detaches the given base object from ApplinX server.
static void	<a href="#">endSession</a> ( <a href="#">GXEndSessionRequest</a> endSessionRequest) Ends a session on ApplinX server according to the given end session request.
static void	<a href="#">endSession</a> ( <a href="#">GXIClientBaseObject</a> baseObject) Ends the session of the given base object.
static <a href="#">GXIClientBaseObject</a>	<a href="#">getBaseObject</a> ( <a href="#">GXAttachToSessionRequest</a> attachRequest) Creates a base object, and attaches it to an existing session on ApplinX server.
static <a href="#">GXIClientBaseObject</a>	<a href="#">getBaseObject</a> ( <a href="#">GXCreateSessionRequest</a> connectRequest) Creates a base object, and connects it to ApplinX server and the host.
static <a href="#">GXReadOnlyBaseObject</a>	<a href="#">getReadOnlyBaseObject</a> ( <a href="#">GXAttachToSessionRequest</a> attachRequest) Creates a read-only base object, and attaches it to an existing session on ApplinX server.

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### GXInnerClientBaseObjectFactory

public **GXInnerClientBaseObjectFactory**()

(continued from last page)

## Methods

### getBaseObject

```
public static GXIClientBaseObject getBaseObject(GXCreateSessionRequest connectRequest)  
    throws GXConnectionException,  
           GXGeneralException
```

Creates a base object, and connects it to ApplinX server and the host.

**Parameters:**

connectRequest - the connection parameters.

**Returns:**

connected base object.

**Throws:**

[GXConnectionException](#) - when the connection to the host / ApplinX server was not successful.

[GXGeneralException](#) - other exception thrown in the process.

### getBaseObject

```
public static GXIClientBaseObject getBaseObject(GXAttachToSessionRequest  
attachRequest)  
    throws GXConnectionException,  
           GXGeneralException
```

Creates a base object, and attaches it to an existing session on ApplinX server. There could be only one attached BaseObject to the same session at a time. This method is meaningless when using localserver:// protocol.

**Parameters:**

attachRequest - the connection parameters.

**Returns:**

attached base object.

**Throws:**

[GXConnectionException](#) - when the connection to the host / ApplinX server was not successful.

[GXGeneralException](#) - other exception thrown in the process.

### getReadOnlyBaseObject

```
public static GXIRedOnlyBaseObject getReadOnlyBaseObject(GXAttachToSessionRequest  
attachRequest)  
    throws GXConnectionException,  
           GXGeneralException
```

Creates a read-only base object, and attaches it to an existing session on ApplinX server. There could be several read-only BaseObject attached to the server at the same time. This method is meaningless when using localserver:// protocol.

**Parameters:**

attachRequest - the connection parameters.

**Returns:**

attached base object.

**Throws:**

[GXConnectionException](#) - when the connection to ApplinX server was not successful.

(continued from last page)

[GXGeneralException](#) - other exception thrown in the process.

---

## detachSession

```
public static void detachSession(GXReadOnlyBaseObject baseObject)
    throws GXGeneralException,
           GXConnectionException
```

Detaches the given base object from ApplinX server. This method is meaningless when using localserver:// protocol.

---

## endSession

```
public static void endSession(GXIClientBaseObject baseObject)
    throws GXGeneralException,
           GXConnectionException
```

Ends the session of the given base object.

### Parameters:

baseObject - the session object.

### Throws:

[GXConnectionException](#) - when the connection to the host / ApplinX server was not successful.  
[GXGeneralException](#) - other exception thrown in the process.

---

## endSession

```
public static void endSession(GXEndSessionRequest endSessionRequest)
    throws GXGeneralException,
           GXConnectionException
```

Ends a session on ApplinX server according to the given end session request.

### Parameters:

endSessionRequest - the session parameters.

### Throws:

[GXConnectionException](#) - when the connection to the host / ApplinX server was not successful.  
[GXGeneralException](#) - other exception thrown in the process.

## com.sabratec.applinx.baseobject

### Class GXInputField

java.lang.Object

└─com.sabratec.applinx.baseobject.GXInputField

#### All Implemented Interfaces:

java.io.Serializable

```
public class GXInputField
extends Object
implements java.io.Serializable
```

This class represents an input field whose data will be sent to the host. The field can be determined by position (for host field), by name (for application field) or by name and index (for an instance of a multiple application field). The GXInputField instance should be added to a request object of type GXSendKeysRequest or GXPathRequest.

### Constructor Summary

public	<a href="#">GXInputField()</a>
public	<a href="#">GXInputField</a> (String name, String value)
public	<a href="#">GXInputField</a> ( <a href="#">GXPosition</a> pos, String value)
public	<a href="#">GXInputField</a> (String name, int index, String value)

### Method Summary

boolean	<a href="#">equals</a> (Object inputField)
int	<a href="#">getIndex</a> ()
String	<a href="#">getName</a> ()
<a href="#">GXPosition</a>	<a href="#">getPosition</a> ()
String	<a href="#">getValue</a> ()
boolean	<a href="#">isAppField</a> ()
void	<a href="#">setIndex</a> (int index)
void	<a href="#">setName</a> (String name) Sets the name of the field to update.
void	<a href="#">setPosition</a> ( <a href="#">GXPosition</a> position) Sets the position of the field to update.

void	<a href="#">setValue</a> (String value)
String	<a href="#">toString</a> ()

**Methods inherited from class `java.lang.Object`**

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### **GXInputField**

```
public GXInputField()
```

### **GXInputField**

```
public GXInputField(String name,  
                    String value)
```

### **GXInputField**

```
public GXInputField(GXPosition pos,  
                  String value)
```

### **GXInputField**

```
public GXInputField(String name,  
                  int index,  
                  String value)
```

## Methods

### **getPosition**

```
public GXPosition getPosition()
```

### **setPosition**

```
public void setPosition(GXPosition position)
```

Sets the position of the field to update.

**Parameters:**

`position` - position of the field to set.

---

## getName

```
public String getName()
```

### Returns:

the name that was set to this field.

---

## setName

```
public void setName(String name)
```

Sets the name of the field to update.

### Parameters:

name - the name of the application field to set.

---

## isAppField

```
public boolean isAppField()
```

---

## getIndex

```
public int getIndex()
```

### Returns:

the multiple index set for the input field.

---

## setIndex

```
public void setIndex(int index)
```

### Parameters:

index - index of an instance of a multiple application field to set. The name of the field must be set using setName().

---

## getValue

```
public String getValue()
```

### Returns:

the value set for the input field.

---

## setValue

```
public void setValue(String value)
```

(continued from last page)

**Parameters:**

value - the value to set to this field on the host.

---

**equals**

```
public boolean equals(Object inputField)
```

---

**toString**

```
public String toString()
```

## com.sabratec.applinx.baseobject Interface GXIReadOnlyBaseObject

All Subinterfaces:

[GXIClientBaseObject](#), [GXINaturalDataTransferClientBaseObject](#)

public interface **GXIReadOnlyBaseObject**

This object represents a live session against the ApplinX server and the host, in Read-Only mode, i.e. only monitoring the session is permitted.

See Also:

[GXIClientBaseObject](#)

### Method Summary

<a href="#">GXIClientApplicationConfiguration</a>	<a href="#">getApplicationConfig()</a> Returns the current application configuration.
<a href="#">GXIScreen</a>	<a href="#">getScreen()</a> Returns the current screen from the host.
<a href="#">GXIScreen</a>	<a href="#">getScreen(GXGetScreenRequest request)</a> Returns the current screen from the host.
int	<a href="#">getSeqScreenNumber()</a> returns the sequential number of the host screen in the server (counted since the session started).
String	<a href="#">getSessionID()</a>
boolean	<a href="#">isReadOnly()</a> Return whether this client has a write access, or only a read-only access.

### Methods

#### getScreen

```
public GXIScreen getScreen(GXGetScreenRequest request)  
    throws GXConnectionException,  
           GXGeneralException
```

Returns the current screen from the host. May also execute wait conditions before retrieving it. If it is called without a request object and if no other request was sent to the host since the last time getScreen was called, it returns a local copy of the last retrieved screen.

**Parameters:**

request - the getScreen request.

**Returns:**

the current screen from the host.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

(continued from last page)

[GXGeneralException](#) - Other errors.**See Also:**[getScreen\(\)](#)

---

## getScreen

```
public GXIScreen getScreen()  
    throws GXConnectionException,  
           GXGeneralException
```

Returns the current screen from the host. May also execute wait conditions before retrieving it. When called without a request object and if no other request was sent to the host since the last time getScreen was called, it returns a local copy of the last retrieved screen.

**Returns:**

a local copy of the last retrieved screen.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.  
[GXGeneralException](#) - Other errors.

---

## getSeqScreenNumber

```
public int getSeqScreenNumber()  
    throws GXConnectionException,  
           GXGeneralException
```

returns the sequential number of the host screen in the server (counted since the session started).

**Returns:**

the sequential screen number.

**Throws:**

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.  
[GXGeneralException](#) - Other errors.

---

## getSessionID

```
public String getSessionID()
```

**Returns:**

the ID of the session on ApplinX Server.

---

## getApplicationConfig

```
public GXClientApplicationConfig getApplicationConfig()  
    throws GXConnectionException,  
           GXGeneralException
```

Returns the current application configuration. The first time this method is called, a request will be sent to the server to retrieve the information. Sequential calls to this method will return the same configuration (additional requests will not be sent to the server).

Note that when the application configuration is changed on the server, the way to retrieve the updated information is to create a new BaseObject and call this method again.

**Throws:**

(continued from last page)

[GXConnectionException](#) - When connection problems occurred between the client and ApplinX server or the ApplinX server and the host.

[GXGeneralException](#) - Other errors.

---

## **isReadOnly**

```
public boolean isReadOnly()
```

Return whether this client has a write access, or only a read-only access.

## com.sabratec.applinx.baseobject Interface GXIReplayerClientBaseObject

public interface **GXIReplayerClientBaseObject**

### Method Summary

GXReplayerInfo	<a href="#">getReplayerInfo()</a>
GXReplayerStatus	<a href="#">getReplayerStatus()</a>
GXReplayerSendData	<a href="#">getReplayScreenData()</a>
void	<a href="#">setReplayScreen</a> (GXSetReplayScreen request)

### Methods

#### setReplayScreen

```
public void setReplayScreen(GXSetReplayScreen request)  
    throws GXConnectionException,  
           GXGeneralException
```

#### getReplayerStatus

```
public GXReplayerStatus getReplayerStatus()  
    throws GXConnectionException,  
           GXGeneralException
```

#### getReplayerInfo

```
public GXReplayerInfo getReplayerInfo()  
    throws GXConnectionException,  
           GXGeneralException
```

#### getReplayScreenData

```
public GXReplayerSendData getReplayScreenData()  
    throws GXConnectionException,  
           GXGeneralException
```

## com.sabratec.applinx.baseobject Interface GXIScreen

All Subinterfaces:

[GXIScreenWithPrintData](#), [GXINaturalDataTransferScreen](#)

public interface **GXIScreen**  
extends java.io.Serializable

Interface for objects representing a screen entity that wraps an emulation screen. This class represents one instance of a host screen. It is a snapshot of a screen as it was received from the host. You cannot set contents to it. Setting contents to the host is done through the session object (using requests).

### Method Summary

<a href="#">GXScreenBuffers</a>	<a href="#">getBuffers()</a> <b>Deprecated.</b> use <a href="#">getBuffers(boolean)</a> instead.
<a href="#">GXScreenBuffers</a>	<a href="#">getBuffers</a> (boolean onlyWindowBuffer) Returns the buffers from the current screen.
int	<a href="#">getColumns()</a>
<a href="#">GXPosition</a>	<a href="#">getCursorPosition()</a>
<a href="#">GXIField</a>	<a href="#">getFieldByCursorPosition()</a> Gets the field which the cursor position is located at
String	<a href="#">getFieldContent</a> (String fieldName)
String	<a href="#">getFieldContent</a> (String fieldName, int fieldIndex)
<a href="#">GXIField</a>	<a href="#">getFieldInstance</a> (String fieldName, int fieldIndex)
<a href="#">GXIFieldCollection</a>	<a href="#">getFields()</a>
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> ( <a href="#">GXIFieldCollectionQuery</a> query)
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> ( <a href="#">GXIScreenCollectionFieldsQuery</a> query) <b>Deprecated.</b> this method is deprecated and is replaced by <a href="#">getFields(GXIFieldCollectionQuery query)</a>
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> (String fieldName)
String	<a href="#">getFolderPath()</a>
<a href="#">GXIScreenKey[]</a>	<a href="#">getHostKeys()</a> Gets the host keys associated with this screen.
String	<a href="#">getName()</a>

int	<a href="#">getRows()</a>
String[]	<a href="#">getScreenGroupNames()</a>
int	<a href="#">getSeqNumber()</a>
<a href="#">GXITable</a>	<a href="#">getTable()</a> <b>Deprecated.</b> As of release 8.2.2 HF3, replaced by <a href="#">getTables()</a>
<a href="#">GXITable[]</a>	<a href="#">getTables()</a> Gets the tables associated with this screen.
<a href="#">GXIWindow</a>	<a href="#">getWindow()</a> Gets the window associated with this screen.
<a href="#">GXPosition</a>	<a href="#">getWindowLowerRight()</a> <b>Deprecated.</b> use @see <a href="#">GXIScreen#getWindow()</a> , @see <a href="#">GXIWindow#getBounds()</a>
<a href="#">GXIWindow[]</a>	<a href="#">getWindows()</a> Gets an array of the windows associated with this screen, or null if none exist.
<a href="#">GXPosition</a>	<a href="#">getWindowUpperLeft()</a> <b>Deprecated.</b> use @see <a href="#">GXIScreen#getWindow()</a> , @see <a href="#">GXIWindow#getBounds()</a>
boolean	<a href="#">hasAlarm()</a>
boolean	<a href="#">isDirectionRTL()</a>
boolean	<a href="#">isDisplayChanged()</a>
boolean	<a href="#">isLocked()</a>
boolean	<a href="#">isMemberOf(String screenGroupName)</a> Checks whether the current screen belongs to the specified screen group.
boolean	<a href="#">isTabbingRTL()</a>
boolean	<a href="#">isWindow()</a>

## Methods

### getName

```
public String getName()
```

#### Returns:

the name (ID) of the screen as it was identified by ApplinX.

(continued from last page)

---

## getScreenGroupNames

```
public String[] getScreenGroupNames()
```

**Returns:**

a name array of the screens screen groups.

---

## isWindow

```
public boolean isWindow()
```

**Returns:**

whether the current screen was identified by ApplinX as a host window.

---

## getWindowUpperLeft

```
public GXPosition getWindowUpperLeft()
```

**Deprecated.** *use @see GXIScreen#getWindow(), @see GXIWindow#getBounds()*

**Returns:**

the upper left coordinate of the identified host window.

---

## getWindowLowerRight

```
public GXPosition getWindowLowerRight()
```

**Deprecated.** *use @see GXIScreen#getWindow(), @see GXIWindow#getBounds()*

**Returns:**

the lower right coordinate of the identified host window.

---

## getSeqNumber

```
public int getSeqNumber()
```

**Returns:**

the sequential number of the screen (since the session started).

---

## isDisplayChanged

```
public boolean isDisplayChanged()
```

**Returns:**

whether the current screen reflects display changes that were received from the host since the last time ApplinX Server sent content.

---

## getCursorPosition

```
public GXPosition getCursorPosition()
```

**Returns:**

the position of the cursor on the screen.

---

## getFolderPath

```
public String getFolderPath()
```

**Returns:**

the screen ID's folder path.

---

## getRows

```
public int getRows()
```

**Returns:**

the number of rows in the current screen. If the current screen is a host window the number of rows is the height of the window.

---

## getColumns

```
public int getColumns()
```

**Returns:**

the number of columns in the current screen. If the current screen is a host window, the number of columns is the width of the window.

---

## getBuffers

```
public GXScreenBuffers getBuffers(boolean onlyWindowBuffer)
```

Returns the buffers from the current screen. Buffers will be returned only if the variable [GXBaseObjectConstants.GX\\_VAR\\_RETURN\\_SCREEN\\_BUFFERS](#) was set in the [GXGetScreenRequest](#) to [GXBaseObjectConstants.GX\\_VAR\\_VAL\\_BUFFERS\\_SCREEN\\_DATA](#). By default buffers will not be available.

**Parameters:**

onlyWindowBuffer - when there is a window, only returns the window buffer and not the entire screen buffer.

**Returns:**

the screen buffers object.

---

## getBuffers

```
public GXScreenBuffers getBuffers()
```

**Deprecated.** use [getBuffers\(boolean\)](#) instead.

Backward compatability. Equivalent to `getBuffers(true)`.

---

---

(continued from last page)

**Returns:**

the screen buffers object.

---

## isDirectionRTL

```
public boolean isDirectionRTL()
```

**Returns:**

whether the direction of the current screen is RTL.

---

## isTabbingRTL

```
public boolean isTabbingRTL()
```

**Returns:**

whether the tabbing direction of the current screen is RTL.

---

## isLocked

```
public boolean isLocked()
```

**Returns:**

whether the system was locked upon the creation of the screen or not. In normal situations, the system is NOT locked upon screen creation.

**See Also:**

[for more details on this feature.](#)

---

## hasAlarm

```
public boolean hasAlarm()
```

**Returns:**

whether the screen should sound an alarm

---

## getFields

```
public GXIFieldCollection getFields()
```

**Returns:**

a field collection holding application fields in the screen. To collect both the host fields and the application fields set the [GXBaseObjectConstants.GX\\_VAR\\_INCLUDE\\_HOST\\_FIELDS](#) variable in the [GXGetScreenRequest](#) to "true".

---

## getFields

```
public GXIFieldCollection getFields(GXIScreenCollectionFieldsQuery query)
```

---

(continued from last page)

**Deprecated.** *this method is deprecated and is replaced by `getFields(GXIFieldCollectionQuery query)`*

**Parameters:**

query - a query that creates GXIFieldCollection from GXIScreenCollection

**Returns:**

the collection returned by the query

---

## getFields

```
public GXIFieldCollection getFields(GXIFieldCollectionQuery query)
```

**Parameters:**

query - a query that creates GXIFieldCollection from a field collection query

**Returns:**

the collection returned by the query

---

## getFields

```
public GXIFieldCollection getFields(String fieldName)
```

**Parameters:**

fieldName - name of an application field

**Returns:**

a collection holding all the instances of the application field.

---

## getFieldInstance

```
public GXIField getFieldInstance(String fieldName,  
    int fieldIndex)
```

**Parameters:**

fieldName - multiple application field name

fieldIndex - index within the multiple application field.

**Returns:**

the instance of the field at the given index.

---

## getFieldContent

```
public String getFieldContent(String fieldName,  
    int fieldIndex)
```

**Parameters:**

fieldName - a name of an application field.

fieldIndex - an index of a multiple application field.

(continued from last page)

**Returns:**

the content of the given instance of the application field.

---

## getFieldContent

```
public String getFieldContent(String fieldName)
```

**Parameters:**

fieldName - a name of an application field.

**Returns:**

the content of the first instance of the application field.

---

## isMemberOf

```
public boolean isMemberOf(String screenGroupName)
```

Checks whether the current screen belongs to the specified screen group.

The specified screen group name may contain a full name including folder path, or the screen group name only.

The function is not case sensitive.

**Parameters:**

screenGroupName - the screen group name

**Returns:**

whether the current screen is a member of the specified screen group.

---

## getWindow

```
public GXIScreen getWindow()
```

Gets the window associated with this screen.

---

## getTable

```
public GXIScreen getTable()
```

**Deprecated.** As of release 8.2.2 HF3, replaced by [getTables\(\)](#)

Gets the table associated with this screen.

---

## getTables

```
public GXIScreen[] getTables()
```

Gets the tables associated with this screen.

---

## getWindows

```
public GXIScreen[] getWindows()
```

Gets an array of the windows associated with this screen, or null if none exist.

The windows in this array are sorted according to the order of arrival from the host. The last element in the array is the last window received.

---

## getHostKeys

```
public GXIScreenKey\[\] getHostKeys()
```

Gets the host keys associated with this screen.

---

## getFieldByCursorPosition

```
public GXIScreenField getFieldByCursorPosition()
```

Gets the field which the cursor position is located at

**Returns:**

the field which the cursor position is located at

## com.sabratec.applinx.baseobject Interface GXIScreenCollection

public interface **GXIScreenCollection**

This class represents a collection of screens. It is returned from a path response, and can also be created by you from an array of screens.

### Method Summary

<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> ( <a href="#">GXIScreenCollectionFieldsQuery</a> query)
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> (String fieldName) The method returns from all the screens in this collection all the fields that have the specified name.
<a href="#">GXIScreen</a>	<a href="#">getScreenAt</a> (int index)
int	<a href="#">getScreenCount</a> ()
<a href="#">GXIScreenCollection</a>	<a href="#">getScreens</a> (String screenName)

### Methods

#### getScreenCount

```
public int getScreenCount()
```

**Returns:**

the number of screens in the collection.

#### getScreenAt

```
public GXIScreen getScreenAt(int index)
```

**Parameters:**

index - the index of the screen in the screens collection.

**Returns:**

the screen at the given index.

#### getFields

```
public GXIFieldCollection getFields(GXIScreenCollectionFieldsQuery query)
```

(continued from last page)

**Parameters:**

query - the query for filtering the collection's data.

**Returns:**

the fields gathered by the given query, returned in the form of a field collection.

---

## getFields

```
public GXIFieldCollection getFields(String fieldName)
```

The method returns from all the screens in this collection all the fields that have the specified name. When the field is a multiple application field, each instance of the application field will be returned as a field in the collection.

**Parameters:**

fieldName - the name of the application field to return.

**Returns:**

all the instances of the given field name from ALL the screens in the collection.

---

## getScreens

```
public GXIScreenCollection getScreens(String screenName)
```

**Parameters:**

screenName - the name of the screen to return.

**Returns:**

a collection constructed from all the instances of the given screen name in the current collection.

# com.sabratec.applinx.baseobject Interface GXIScreenWithPrintData

All Superinterfaces:  
[GXIScreen](#)

public interface **GXIScreenWithPrintData**  
extends [GXIScreen](#)

Extension for the [GXIScreen](#) class that adds printing data buffer to the screen. Used by some unisys hosts.

## Method Summary

byte[]	<a href="#">getPrintData()</a>
--------	--------------------------------

### Methods inherited from interface [com.sabratec.applinx.baseobject.GXIScreen](#)

[getBuffers](#), [getBuffers](#), [getColumns](#), [getCursorPosition](#), [getFieldByCursorPosition](#), [getFieldContent](#), [getFieldContent](#), [getFieldInstance](#), [getFields](#), [getFields](#), [getFields](#), [getFields](#), [getFolderPath](#), [getHostKeys](#), [getName](#), [getRows](#), [getScreenGroupNames](#), [getSeqNumber](#), [getTable](#), [getTables](#), [getWindow](#), [getWindowLowerRight](#), [getWindows](#), [getWindowUpperLeft](#), [hasAlarm](#), [isDirectionRTL](#), [isDisplayChanged](#), [isLocked](#), [isMemberOf](#), [isTabbingRTL](#), [isWindow](#)

## Methods

### getPrintData

public byte[] **getPrintData()**

**Returns:**  
the print data associated with this screen.

## com.sabratec.applinx.baseobject Interface GXIVariable

All Known Implementing Classes:

[GXVariableArray](#), [GXVariable](#)

```
public interface GXIVariable
extends java.io.Serializable
```

This interface represents a variable, and allows setting it to a session request. It is implemented by [GXVariable](#), for single value variables, and by [GXVariableArray](#) for multiple values. The common methods in this interface are the variable's name methods.

### Field Summary

<code>public static final</code>	<a href="#">NULL</a> This constant can be used to override a variable with a null value. Value: -
<code>public static final</code>	<a href="#">OVERRIDE</a> this const is used to override the recorder configuration Value: -1

### Method Summary

<code>String</code>	<a href="#">getName()</a>
<code>void</code>	<a href="#">setName(String name)</a>

### Fields

#### NULL

```
public static final java.lang.String NULL
```

This constant can be used to override a variable with a null value.

For example: request.addVariable(GXBaseObjectConstants.GX\_VAR\_REPLAY\_FILE, GXIVariable.NULL) will define that the session will not use the predefined replay file.

Constant value: -

#### OVERRIDE

```
public static final java.lang.String OVERRIDE
```

this const is used to override the recorder configuration

Constant value: -1

### Methods

#### getName

```
public String getName()
```

(continued from last page)

**Returns:**

The name of the variable or variable array.

---

**setName**

```
public void setName(String name)
```

**Parameters:**

name - The name of the variable or variable array.

## com.sabratec.applinx.baseobject Interface GXIWaitCondition

All Known Implementing Classes:  
[GXAbstractWaitCondition](#)

---

```
public interface GXIWaitCondition  
extends java.io.Serializable
```

Interface for all wait conditions.

---

---

## com.sabratec.applinx.baseobject Interface GXIWindow

public interface **GXIWindow**

This class represents a window.

### Method Summary

GXRectangle	<a href="#">getBounds()</a>
String	<a href="#">getTitle()</a>
boolean	<a href="#">isBoundsIncludeBorder()</a>

### Methods

#### getTitle

```
public String getTitle()
```

**Returns:**

the window title

#### getBounds

```
public GXRectangle getBounds()
```

**Returns:**

the window boundaries

#### isBoundsIncludeBorder

```
public boolean isBoundsIncludeBorder()
```

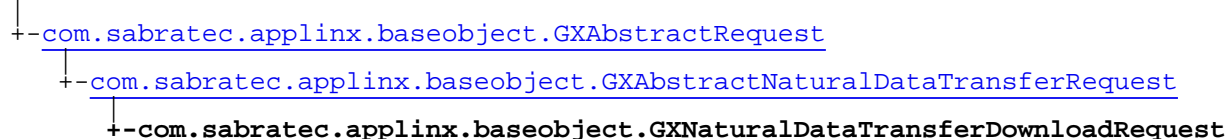
**Returns:**

whether the boundaries returned by the [getBounds\(\)](#) method include a border

## com.sabratec.applinx.baseobject

### Class GXNaturalDataTransferDownloadRequest

java.lang.Object



#### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXNaturalDataTransferDownloadRequest**

extends [GXAbstractNaturalDataTransferRequest](#)

Setting request for Natural Data Transfer download messages.

### Constructor Summary

public	<a href="#">GXNaturalDataTransferDownloadRequest()</a>
public	<a href="#">GXNaturalDataTransferDownloadRequest(int type)</a>

### Method Summary

void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class

[com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferRequest](#)

[getFormatType](#), [readExternal](#), [writeExternal](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

### Constructors

(continued from last page)

---

## GXNaturalDataTransferDownloadRequest

```
public GXNaturalDataTransferDownloadRequest()
```

---

## GXNaturalDataTransferDownloadRequest

```
public GXNaturalDataTransferDownloadRequest(int type)
```

### See Also:

```
com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferRequest(int)
```

## Methods

### readExternal

```
public void readExternal(java.io.ObjectInput in)
    throws java.io.IOException,
           ClassNotFoundException
```

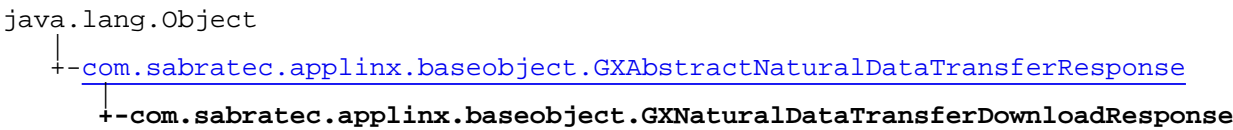
---

### writeExternal

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

com.sabratec.applinx.baseobject

Class GXNaturalDataTransferDownloadResponse



All Implemented Interfaces:

java.io.Serializable

public class **GXNaturalDataTransferDownloadResponse**  
extends [GXAbstractNaturalDataTransferResponse](#)

Response data message for [GXNaturalDataTransferDownloadRequest](#)  
**See Also:**  
ACTION\_NATURAL\_DATA\_DOWNLOAD

Constructor Summary

public	<a href="#">GXNaturalDataTransferDownloadResponse</a> ( )
--------	---

Method Summary

byte[]	<a href="#">getData</a> ( ) The actual download data.
boolean	<a href="#">isFirstChunk</a> ( ) flag for marking the first chunk of the stream, due to override the existing file (if any)
void	<a href="#">setData</a> (byte[] data) Set the data included in this response.
void	<a href="#">setFirstChunk</a> (boolean firstChunk) flag for marking the first chunk of the stream, due to override the existing file (if any)

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferResponse](#)

[getNumProcessedRecords](#), [isComplete](#), [setComplete](#), [setNumProcessedRecords](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructors

**GXNaturalDataTransferDownloadResponse**

public **GXNaturalDataTransferDownloadResponse**( )

## Methods

### getData

```
public byte[] getData()
```

The actual download data. null as long as the download was not completed.

---

### setData

```
public void setData(byte[] data)
```

Set the data included in this response. This method is for internal use.

**Parameters:**

data - the downloaded data

---

### isFirstChunk

```
public boolean isFirstChunk()
```

flag for marking the first chunk of the stream, due to override the existing file (if any)

**Returns:**

---

### setFirstChunk

```
public void setFirstChunk(boolean firstChunk)
```

flag for marking the first chunk of the stream, due to override the existing file (if any)

**Parameters:**

firstChunk

---

## com.sabratec.applinx.baseobject Class GXNaturalDataTransferException

```
java.lang.Object
  |
  +--GXExceptionWithErrorCode
        |
        +--com.sabratec.applinx.baseobject.GXGeneralException
              |
              +--com.sabratec.applinx.baseobject.GXNaturalDataTransferException
```

```
public class GXNaturalDataTransferException
extends GXGeneralException
```

Exception indicating that there was a problem with a Natural data transfer transaction.

### Constructor Summary

public	<a href="#">GXNaturalDataTransferException</a> (String message, Throwable e, int errorCode)
--------	---

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### **GXNaturalDataTransferException**

```
public GXNaturalDataTransferException(String message,
                                     Throwable e,
                                     int errorCode)
```

## com.sabratec.applinx.baseobject

### Class GXNaturalDataTransferUploadRequest

```

java.lang.Object
  |
  +- com.sabratec.applinx.baseobject.GXAbstractRequest
      |
      +- com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferRequest
          |
          +- com.sabratec.applinx.baseobject.GXNaturalDataTransferUploadRequest
  
```

#### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXNaturalDataTransferUploadRequest**  
 extends [GXAbstractNaturalDataTransferRequest](#)

Setting request for Natural Data Transfer upload messages.

### Constructor Summary

public	<a href="#">GXNaturalDataTransferUploadRequest()</a>
public	<a href="#">GXNaturalDataTransferUploadRequest</a> (int type, byte[] data)

### Method Summary

byte[]	<a href="#">getData()</a> Returns the data to upload.
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class

[com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferRequest](#)

[getFormatType](#), [readExternal](#), [writeExternal](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

(continued from last page)

## Constructors

### **GXNaturalDataTransferUploadRequest**

```
public GXNaturalDataTransferUploadRequest()
```

---

### **GXNaturalDataTransferUploadRequest**

```
public GXNaturalDataTransferUploadRequest(int type,  
                                           byte[] data)
```

**Parameters:**

data - the data to upload

**See Also:**

`com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferRequest(int)`

## Methods

### **getData**

```
public byte[] getData()
```

Returns the data to upload.

---

### **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject Class GXNaturalDataTransferUploadResponse

java.lang.Object

└- [com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferResponse](#)  
└- [com.sabratec.applinx.baseobject.GXNaturalDataTransferUploadResponse](#)

### All Implemented Interfaces:

java.io.Serializable

public class **GXNaturalDataTransferUploadResponse**  
extends [GXAbstractNaturalDataTransferResponse](#)

Response data message for [GXNaturalDataTransferUploadRequest](#)

### See Also:

ACTION\_NATURAL\_DATA\_UPLOAD

## Constructor Summary

public	<a href="#">GXNaturalDataTransferUploadResponse()</a>
--------	---

### Methods inherited from class

[com.sabratec.applinx.baseobject.GXAbstractNaturalDataTransferResponse](#)

[getNumProcessedRecords](#), [isComplete](#), [setComplete](#), [setNumProcessedRecords](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

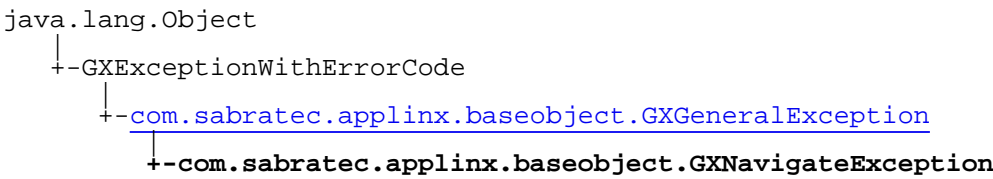
## Constructors

### GXNaturalDataTransferUploadResponse

public **GXNaturalDataTransferUploadResponse()**

# com.sabratec.applinx.baseobject

## Class GXNavigateException



```
public class GXNavigateException
extends GXGeneralException
```

Exception indicating an error occurred in a navigation process.

### Constructor Summary

public	<a href="#">GXNavigateException</a> (String message, Throwable e, int errorCode)
--------	--

### Methods inherited from class java.lang.Object

<code>equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait</code>
--

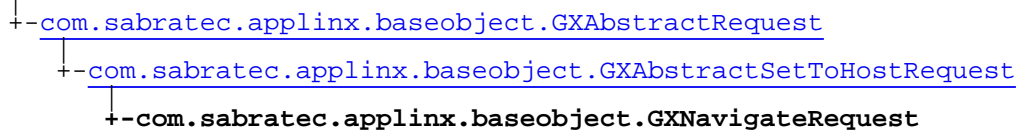
### Constructors

#### GXNavigateException

```
public GXNavigateException(String message,
                           Throwable e,
                           int errorCode)
```

## com.sabratec.applinx.baseobject Class GXNavigateRequest

java.lang.Object



### All Implemented Interfaces:

[GXDebuggableRequest](#), Cloneable, java.io.Externalizable

public class **GXNavigateRequest**

extends [GXAbstractSetToHostRequest](#)

implements java.io.Externalizable, Cloneable, [GXDebuggableRequest](#)

Settings for map type paths execution request.

### Constructor Summary

public	<a href="#">GXNavigateRequest</a> ( )
public	<a href="#">GXNavigateRequest</a> (String destinationScreenName) Sets the destination screen for the navigation process.
public	<a href="#">GXNavigateRequest</a> (String destinationScreenName, String mapName) <b>Deprecated.</b> Application map are defined globally for each application. Map name will be ignored

### Method Summary

String	<a href="#">getDestinationScreenName</a> ( )
String	<a href="#">getMapName</a> ( ) <b>Deprecated.</b> Application map are defined globally for each application. Map name will be ignored
boolean	<a href="#">isDebugMode</a> ( )
void	<a href="#">readExternal</a> ( java.io.ObjectInput in)
void	<a href="#">setDebugMode</a> ( ) Sets the path to be executed in debug mode.
void	<a href="#">setDestinationScreenName</a> (String destScreen)
void	<a href="#">setMapName</a> (String mapName) <b>Deprecated.</b> Application map are defined globally for each application. Map name will be ignored
void	<a href="#">writeExternal</a> ( java.io.ObjectOutput out)

**Methods inherited from class** [com.sabratec.applinx.baseobject.GXAbstractSetToHostRequest](#)

[addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#),  
[addInputField](#), [addInputFields](#), [changeInputField](#), [getInputFields](#), [getSeqScreenNumber](#),  
[isWindowRelativePositioning](#), [readExternal](#), [removeInputField](#), [setSeqScreenNumber](#),  
[setWindowRelativePositioning](#), [toString](#), [writeExternal](#)

**Methods inherited from class** [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#),  
[readExternal](#), [setVariables](#), [writeExternal](#)

**Methods inherited from class** [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

**Methods inherited from interface** [java.io.Externalizable](#)

[readExternal](#), [writeExternal](#)

**Methods inherited from interface** [com.sabratec.applinx.baseobject.GXDebuggableRequest](#)

[isDebugMode](#), [setDebugMode](#)

## Constructors

### GXNavigateRequest

```
public GXNavigateRequest()
```

### GXNavigateRequest

```
public GXNavigateRequest(String destinationScreenName)
```

Sets the destination screen for the navigation process.

**Parameters:**

`destinationScreenName` - the name of the destination screen.

### GXNavigateRequest

```
public GXNavigateRequest(String destinationScreenName,  
                        String mapName)
```

**Deprecated.** *Application map are defined globally for each application. Map name will be ignored*

Sets the destination screen and the map name for the navigation process.

**Parameters:**

`destinationScreenName` - the name of the destination screen.

`mapName` - the name of the map.

## Methods

---

(continued from last page)

## getDestinationScreenName

```
public String getDestinationScreenName()
```

**Returns:**

the path's destination screen name

---

## setDestinationScreenName

```
public void setDestinationScreenName(String destScreen)
```

**Parameters:**

destScreen - the path's destination screen name to set

---

## setMapName

```
public void setMapName(String mapName)
```

**Deprecated.** *Application map are defined globally for each application. Map name will be ignored*

Sets the name of the map path to navigate with.

**Parameters:**

mapName - the name of the map

---

## getMapName

```
public String getMapName()
```

**Deprecated.** *Application map are defined globally for each application. Map name will be ignored*

**Returns:**

the map path name to navigate with

---

## setDebugMode

```
public void setDebugMode()
```

Sets the path to be executed in debug mode.

---

## isDebugMode

```
public boolean isDebugMode()
```

**Returns:**

the status of the debug mode flag

---

(continued from last page)

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)
    throws java.io.IOException,
           ClassNotFoundException
```

---

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject Class GXPathException

```

java.lang.Object
  |
  +--GXExceptionWithErrorCode
        |
        +--com.sabratec.applinx.baseobject.GXGeneralException
              |
              +--com.sabratec.applinx.baseobject.GXPathException

```

```

public class GPXPathException
extends GXGeneralException

```

Exception indicating an error occurred in a path process.

### Constructor Summary

public	<a href="#">GPXPathException</a> (String message, Throwable e, int errorCode)
--------	---

### Method Summary

<a href="#">GPXPathResponse</a>	<a href="#">getPathResponse</a> () Returns the path response.
---------------------------------	--

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GPXPathException

```

public GPXPathException(String message,
                        Throwable e,
                        int errorCode)

```

## Methods

### getPathResponse

```

public GPXPathResponse getPathResponse()

```

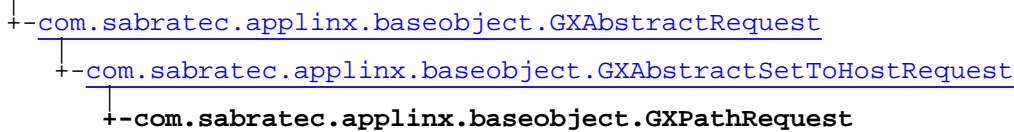
Returns the path response. All screens that were gathered during path execution (until the error occurred) may be retrieved from the path response.

#### Returns:

the path's response object.

## com.sabratec.applinx.baseobject Class GXPathRequest

java.lang.Object



### All Implemented Interfaces:

[GXDebuggableRequest](#), Cloneable, java.io.Externalizable

public class **GPXPathRequest**  
 extends [GXAbstractSetToHostRequest](#)  
 implements java.io.Externalizable, Cloneable, [GXDebuggableRequest](#)

Settings for path execution request.

### Constructor Summary

public	<a href="#">GPXPathRequest</a> ()
public	<a href="#">GPXPathRequest</a> (String pathName)
public	<a href="#">GPXPathRequest</a> (String pathName, boolean isDebugMode)

### Method Summary

String	<a href="#">getPathName</a> ()
boolean	<a href="#">isDebugMode</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setDebugMode</a> () Sets the path to be executed in debug mode.
void	<a href="#">setPathName</a> (String pathName) Sets the name of the path that will be executed.
String	<a href="#">toString</a> ()
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractSetToHostRequest](#)

[addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#),  
[addInputField](#), [addInputFields](#), [changeInputField](#), [getInputFields](#), [getSeqScreenNumber](#),  
[isWindowRelativePositioning](#), [readExternal](#), [removeInputField](#), [setSeqScreenNumber](#),  
[setWindowRelativePositioning](#), [toString](#), [writeExternal](#)

**Methods inherited from class** [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

**Methods inherited from class** java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

**Methods inherited from interface** java.io.Externalizable

[readExternal](#), [writeExternal](#)

**Methods inherited from interface** [com.sabratec.applinx.baseobject.GXDebuggableRequest](#)

[isDebugMode](#), [setDebugMode](#)

## Constructors

### GPXPathRequest

```
public GPXPathRequest()
```

### GPXPathRequest

```
public GPXPathRequest(String pathName)
```

**Parameters:**

pathName - name of the path to execute.

### GPXPathRequest

```
public GPXPathRequest(String pathName,
                      boolean isDebugMode)
```

**Parameters:**

pathName - name of the path to execute.

isDebugMode - flag to execute the path in debug mode. In this mode, a very detailed log of the path execution process is returned.

## Methods

### getPathName

```
public String getPathName()
```

**Returns:**

the path name set to the request.

---

## setPathName

```
public void setPathName(String pathName)
```

Sets the name of the path that will be executed.

### Parameters:

pathName - the name of the path.

---

## setDebugMode

```
public void setDebugMode()
```

Sets the path to be executed in debug mode.

---

## isDebugMode

```
public boolean isDebugMode()
```

### Returns:

the status of the debug mode flag.

---

## readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

## writeExternal

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

## toString

```
public String toString()
```

## com.sabratec.applinx.baseobject Class GXPathResponse

java.lang.Object

└─com.sabratec.applinx.baseobject.GXPathResponse

### All Implemented Interfaces:

java.io.Externalizable

public class **GXPathResponse**  
extends Object  
implements java.io.Externalizable

This class is returned from GXIBaseObject.executePath() method. It contains all the contents of the response of the path.

### Constructor Summary

public	<a href="#">GXPathResponse</a> ( <a href="#">GXIScreen[]</a> screens, GXVariables variables) Create a new path response
public	<a href="#">GXPathResponse</a> ()

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getPathExecutionLog</a> ()
<a href="#">GXIScreenCollection</a>	<a href="#">getScreens</a> ()
String	<a href="#">getVariable</a> (String varName)
String[]	<a href="#">getVariableArray</a> (String varName)
String	<a href="#">getVariableAt</a> (String varName, int index)
int	<a href="#">getVariableCount</a> (String varName)
String[]	<a href="#">getVariableNames</a> ()
boolean	<a href="#">isVariableArray</a> (String varName)
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setPathExecutionLog</a> (String pathExecutionLog) Set the execution log of the path.
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

**Methods inherited from class `java.lang.Object`**

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface `java.io.Externalizable`**

`readExternal`, `writeExternal`

## Constructors

### **GPXPathResponse**

```
public GPXPathResponse(GXIScreen\[\] screens,  
                       GXVariables variables)
```

Create a new path response

**Parameters:**

`screens` - the path collected screens

`variables` - the path returned variables

### **GPXPathResponse**

```
public GPXPathResponse()
```

## Methods

### **getScreens**

```
public GXIScreenCollection getScreens()
```

**Returns:**

the screen collection that was gathered during path execution.

### **getVariable**

```
public String getVariable(String varName)
```

**Parameters:**

`varName` - the name of the variable.

**Returns:**

the value of the variable as was set during path execution.

### **isVariableArray**

```
public boolean isVariableArray(String varName)
```

---

(continued from last page)

**Parameters:**

varName - the name of the variable.

**Returns:**

true if there is a variable array with the input name in the response.

---

## getVariableNames

```
public String[] getVariableNames()
```

**Returns:**

an array of the names of all the variables in the response.

---

## getVariableArray

```
public String[] getVariableArray(String varName)
```

**Parameters:**

varName - the name of the variable array.

**Returns:**

an array of values set to the variable during path execution.

---

## getVariableAt

```
public String getVariableAt(String varName,  
    int index)
```

**Parameters:**

varName - the name of the variable array.

index - the index in the array.

**Returns:**

one of the values from a variable array, according to index.

---

## getVariableCount

```
public int getVariableCount(String varName)
```

**Parameters:**

varName - the name of the variable array.

**Returns:**

the size of a variable array. If no such array exists, -1 is returned. If the variable is not an array, 1 is returned.

---

## getPathExecutionLog

```
public String getPathExecutionLog()
```

---

---

(continued from last page)

**Returns:**

the execution log of the path. might be null.

---

**setPathExecutionLog**

```
public void setPathExecutionLog(String pathExecutionLog)
```

Set the execution log of the path. This method is for internal use.

**Parameters:**

pathExecutionLog - the log to set

---

**equals**

```
public boolean equals(Object obj)
```

---

**writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

**readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

## com.sabratec.applinx.baseobject Class GXScreenBuffers

java.lang.Object

└─com.sabratec.applinx.baseobject.GXScreenBuffers

### All Implemented Interfaces:

java.io.Serializable

public class **GXScreenBuffers**  
extends Object  
implements java.io.Serializable

This class wraps the buffers that are returned from the host for a screen.

### Constructor Summary

public	<a href="#">GXScreenBuffers()</a>
public	<a href="#">GXScreenBuffers</a> (String text, byte[] attributes, byte[] eab, byte[] eabColor)

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
byte[]	<a href="#">getEABBuffer</a> ()
byte[]	<a href="#">getEABColor</a> ()
byte[]	<a href="#">getFieldAttributeBuffer</a> ()
String	<a href="#">getScreenBuffer</a> ()

### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

### Constructors

#### GXScreenBuffers

public **GXScreenBuffers**()

(continued from last page)

## GXScreenBuffers

```
public GXScreenBuffers(String text,
                        byte[] attributes,
                        byte[] eab,
                        byte[] eabColor)
```

## Methods

### getScreenBuffer

```
public String getScreenBuffer()
```

**Returns:**

the buffer that holds the content of fields in the host screen.

---

### getFieldAttributeBuffer

```
public byte[] getFieldAttributeBuffer()
```

**Returns:**

the buffer that holds the attributes of fields in the host screen.

---

### getEABBuffer

```
public byte[] getEABBuffer()
```

**Returns:**

the buffer that holds the extended field attributes for fields in the host screen.

---

### getEABColor

```
public byte[] getEABColor()
```

**Returns:**

the buffer that holds the colors of fields in the host screen.

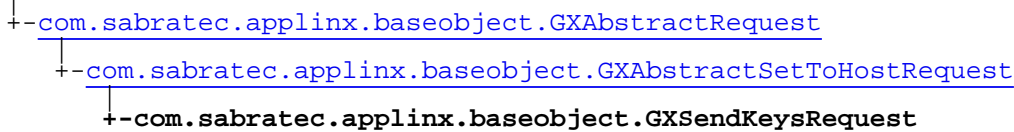
---

### equals

```
public boolean equals(Object obj)
```

## com.sabratec.applinx.baseobject Class GXSendKeysRequest

java.lang.Object



### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXSendKeysRequest**  
extends [GXAbstractSetToHostRequest](#)

Settings for send keys request. If no key is specified [enter] is the default key.

### Constructor Summary

public	<a href="#">GXSendKeysRequest</a> ()
public	<a href="#">GXSendKeysRequest</a> (String keys)
public	<a href="#">GXSendKeysRequest</a> (String keys, <a href="#">GXIScreen</a> screen)

### Method Summary

Object	<a href="#">clone</a> ()
<a href="#">GXCursor</a>	<a href="#">getCursor</a> ()
String	<a href="#">getKeys</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setCursor</a> ( <a href="#">GXCursor</a> cursor) Sets the cursor position for the request.
void	<a href="#">setKeys</a> (String keys) Sets the key sequence that will be set to the host.
String	<a href="#">toString</a> ()
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractSetToHostRequest](#)

[addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#),  
[addInputField](#), [addInputFields](#), [changeInputField](#), [getInputFields](#), [getSeqScreenNumber](#),  
[isWindowRelativePositioning](#), [readExternal](#), [removeInputField](#), [setSeqScreenNumber](#),  
[setWindowRelativePositioning](#), [toString](#), [writeExternal](#)

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#),  
[readExternal](#), [setVariables](#), [writeExternal](#)

Methods inherited from class [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [java.io.Externalizable](#)

[readExternal](#), [writeExternal](#)

## Constructors

### GXSendKeysRequest

```
public GXSendKeysRequest()
```

### GXSendKeysRequest

```
public GXSendKeysRequest(String keys)
```

#### Parameters:

keys - the key (or sequence of keys and data) to send.

### GXSendKeysRequest

```
public GXSendKeysRequest(String keys,  
                           GXIScreen screen)
```

#### Parameters:

keys - the key (or sequence of keys and data) to send.

screen - a screen in which the request is initialized from. Set the sequence number and window state from the screen

## Methods

### getKeys

```
public String getKeys()
```

#### Returns:

(continued from last page)

the key (or sequence of keys and data) set to be sent to the host.

---

## setKeys

```
public void setKeys(String keys)
```

Sets the key sequence that will be set to the host.

### Parameters:

keys - the key (or sequence of keys and data) to send.

---

## setCursor

```
public void setCursor(GXCursor cursor)
```

Sets the cursor position for the request.

### Parameters:

cursor - the cursor position to be sent to the host.

---

## getCursor

```
public GXCursor getCursor()
```

### Returns:

the cursor position set in the request.

---

## readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

## writeExternal

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

## toString

```
public String toString()
```

---

## clone

```
public Object clone()
```

---

## com.sabratec.applinx.baseobject Class GXSessionStatusResponse

java.lang.Object

└─com.sabratec.applinx.baseobject.GXSessionStatusResponse

### All Implemented Interfaces:

java.io.Serializable

public class **GXSessionStatusResponse**  
extends Object  
implements java.io.Serializable

Response data message for the get status request.

## Constructor Summary

public	<a href="#">GXSessionStatusResponse</a> (int seqScreenNumber) Create a new session status response message.
--------	--

## Method Summary

int	<a href="#">getSeqScreenNumber</a> () Returns the current session sequential screen number.
-----	--

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXSessionStatusResponse

public **GXSessionStatusResponse**(int seqScreenNumber)

Create a new session status response message.

#### Parameters:

seqScreenNumber - the current session sequential screen number.

## Methods

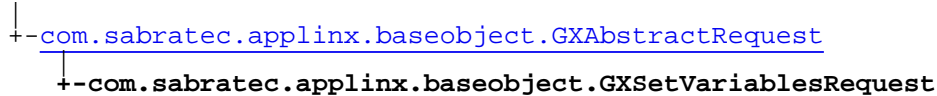
### getSeqScreenNumber

public int **getSeqScreenNumber**()

Returns the current session sequential screen number.

## com.sabratec.applinx.baseobject Class GXSetVariablesRequest

java.lang.Object



### All Implemented Interfaces:

Cloneable, java.io.Externalizable

public class **GXSetVariablesRequest**  
extends [GXAbstractRequest](#)

Set Variables Request.

## Constructor Summary

public	<a href="#">GXSetVariablesRequest()</a>
--------	---

## Method Summary

void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#), [readExternal](#), [setVariables](#), [writeExternal](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

## Constructors

### GXSetVariablesRequest

public **GXSetVariablesRequest**()

## Methods

(continued from last page)

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)
    throws java.io.IOException,
           ClassNotFoundException
```

---

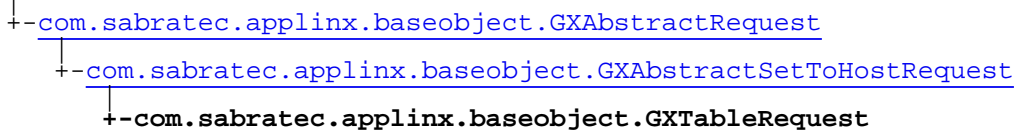
## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject

### Class GXTableRequest

java.lang.Object



#### All Implemented Interfaces:

[GXDebuggableRequest](#), Cloneable, java.io.Externalizable

public class **GXTableRequest**

extends [GXAbstractSetToHostRequest](#)

implements java.io.Externalizable, Cloneable, [GXDebuggableRequest](#)

Settings for get table execution request.

### Constructor Summary

public	<a href="#">GXTableRequest</a> () Constructs a new Table Request for a table associated with the current host screen.
public	<a href="#">GXTableRequest</a> (String tableName)
public	<a href="#">GXTableRequest</a> (String tableName, boolean isFullDetails, boolean isDebugMode)

### Method Summary

String	<a href="#">getTable</a> ()
boolean	<a href="#">isDebugMode</a> ()
boolean	<a href="#">isFullDetails</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setDebugMode</a> () Sets the path, that is executed to retrieve the table, to be executed in debug mode.
void	<a href="#">setFullDetails</a> (boolean isFullDetails)
void	<a href="#">setTableName</a> (String tableName)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractSetToHostRequest](#)

[addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#), [addInputField](#),  
[addInputField](#), [addInputFields](#), [changeInputField](#), [getInputFields](#), [getSeqScreenNumber](#),  
[isWindowRelativePositioning](#), [readExternal](#), [removeInputField](#), [setSeqScreenNumber](#),  
[setWindowRelativePositioning](#), [toString](#), [writeExternal](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractRequest](#)

[addVariable](#), [addVariable](#), [addVariableArray](#), [clone](#), [equals](#), [getVariables](#), [hashCode](#),  
[readExternal](#), [setVariables](#), [writeExternal](#)

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface `java.io.Externalizable`

`readExternal`, `writeExternal`

#### Methods inherited from interface [com.sabratec.applinx.baseobject.GXDebuggableRequest](#)

[isDebugMode](#), [setDebugMode](#)

## Constructors

### **GXTableRequest**

```
public GXTableRequest()
```

Constructs a new Table Request for a table associated with the current host screen.

### **GXTableRequest**

```
public GXTableRequest(String tableName)
```

#### Parameters:

`tableName` - name of table to create.

### **GXTableRequest**

```
public GXTableRequest(String tableName,  
                      boolean isFullDetails,  
                      boolean isDebugMode)
```

#### Parameters:

`tableName` - name of table to create.

`isFullDetails` - whether to return all the host field properties of the table cells (colors, attributes, etc.). Otherwise, only cells' contents are returned. By default this parameter is true.

`isDebugMode` - flag to execute the table's path in debug mode (relevant only for a path table). In this mode, a very detailed log of the path execution process is written to the log.

## Methods

(continued from last page)

---

## getTableName

```
public String getTableName()
```

### Returns:

the table name

---

## setTableName

```
public void setTableName(String tableName)
```

### Parameters:

tableName - the table name

---

## setDebugMode

```
public void setDebugMode()
```

Sets the path, that is executed to retrieve the table, to be executed in debug mode.

---

## isDebugMode

```
public boolean isDebugMode()
```

### Returns:

the status of the debug mode flag.

---

## isFullDetails

```
public boolean isFullDetails()
```

### Returns:

whether the table should be returned with field attributes

---

## setFullDetails

```
public void setFullDetails(boolean isFullDetails)
```

### Parameters:

isFullDetails - whether the table should be returned with field attributes

---

## readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

---

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

## com.sabratec.applinx.baseobject

### Class GXVariable

```

java.lang.Object
  |
  +--GXKeyValue
      |
      +--com.sabratec.applinx.baseobject.GXVariable
  
```

All Implemented Interfaces:  
[GXIVariable](#)

public class **GXVariable**  
 extends [GXKeyValue](#)  
 implements [GXIVariable](#)

This class represents a single variable to be set to a session request. The variables that can be set include application parameters, screen properties and path variables.

All the constants for the names of the variables can be found in [GXBaseObjectConstants](#).

Fields inherited from interface [com.sabratec.applinx.baseobject.GXIVariable](#)

[NULL](#), [OVERRIDE](#)

### Constructor Summary

public	<a href="#">GXVariable</a> () empty ctor - to be used only by serialization
public	<a href="#">GXVariable</a> (String name)
public	<a href="#">GXVariable</a> (String name, String value)

### Method Summary

Object	<a href="#">clone</a> ()
boolean	<a href="#">equals</a> (Object obj)
int	<a href="#">hashCode</a> ()
boolean	<a href="#">isVariable</a> (String varName)

Methods inherited from class [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.sabratec.applinx.baseobject.GXIVariable](#)

[getName](#), [setName](#)

## Constructors

### GXVariable

```
public GXVariable()
```

empty ctor - to be used only by serialization

---

### GXVariable

```
public GXVariable(String name)
```

#### Parameters:

name - the name of the variable to set.

---

### GXVariable

```
public GXVariable(String name,  
                  String value)
```

#### Parameters:

name - the name of the application field to set.

value - value to set to the variable.

---

## Methods

### isVariable

```
public boolean isVariable(String varName)
```

---

### hashCode

```
public int hashCode()
```

---

### equals

```
public boolean equals(Object obj)
```

---

### clone

```
public Object clone()
```

---

## com.sabratec.applinx.baseobject Class GXVariableArray

java.lang.Object

└─com.sabratec.applinx.baseobject.GXVariableArray

All Implemented Interfaces:

[GXIVariable](#)

public class **GXVariableArray**  
extends Object  
implements [GXIVariable](#)

This class represents a variable with multiple values to be set to a session request. Variables that can be set are path variables.

Fields inherited from interface [com.sabratec.applinx.baseobject.GXIVariable](#)

[NULL](#), [OVERRIDE](#)

### Constructor Summary

public	<a href="#">GXVariableArray</a> (String name, String[] values)
--------	--

### Method Summary

String	<a href="#">getName</a> ()
--------	----------------------------

String[]	<a href="#">getValues</a> ()
----------	------------------------------

void	<a href="#">setName</a> (String name)
------	---------------------------------------

void	<a href="#">setValues</a> (String[] values)
------	---

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Methods inherited from interface [com.sabratec.applinx.baseobject.GXIVariable](#)

[getName](#), [setName](#)

### Constructors

#### GXVariableArray

```
public GXVariableArray(String name,
                       String[] values)
```

(continued from last page)

**Parameters:**

name - the name of the variable to set.

values - array of String values to set to the variable.

## Methods

**getName**

```
public String getName()
```

**See Also:**[GXIVariable.getName\(\)](#)

---

**setName**

```
public void setName(String name)
```

**See Also:**[GXIVariable.setName\(String\)](#)

---

**setValues**

```
public void setValues(String[] values)
```

**Parameters:**

values - the values to set in the variable array

---

**getValues**

```
public String[] getValues()
```

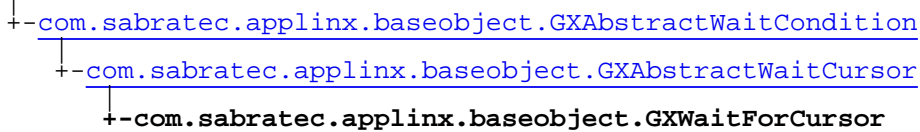
**Returns:**

the variable array values

## com.sabratec.applinx.baseobject

### Class GXWaitForCursor

java.lang.Object



All Implemented Interfaces:

[GXWaitCondition](#)

```
public class GXWaitForCursor
extends GXAbstractWaitCursor
```

Wait condition that instructs the ApplinX server to wait for the cursor to reach a specified location.

### Constructor Summary

public	<a href="#">GXWaitForCursor</a> ( <a href="#">GXPosition</a> cursorAt, long timeout, long flicker) Sets a wait condition that instructs the ApplinX server to wait for the cursor to reach the specified position.
public	<a href="#">GXWaitForCursor</a> ( <a href="#">GXPosition</a> upperLeft, <a href="#">GXPosition</a> lowerRight, long timeout, long flicker) Sets a wait condition that instructs the ApplinX server to wait for the cursor to reach a position within the specified rectangle.

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCursor](#)

[getLowerRightPosition](#), [getUpperLeftPosition](#), [setLowerRightPosition](#), [setUpperLeftPosition](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### GXWaitForCursor

```
public GXWaitForCursor(GXPosition cursorAt,
                      long timeout,
                      long flicker)
```

Sets a wait condition that instructs the ApplinX server to wait for the cursor to reach the specified position.

##### Parameters:

cursorAt - Position where the cursor should be.

(continued from last page)

timeout - Expiration time of the wait condition.

flicker - Time to wait after the wait condition was met.

---

## GXWaitForCursor

```
public GXWaitForCursor(GXPosition upperLeft,  
                       GXPosition lowerRight,  
                       long timeout,  
                       long flicker)
```

Sets a wait condition that instructs the ApplinX server to wait for the cursor to reach a position within the specified rectangle.

### Parameters:

upperLeft - Upper left position of the rectangle where the cursor should be.

lowerRight - Lower right position of the rectangle where the cursor should be.

timeout - Expiration time of the wait condition.

flicker - Time to wait after the wait condition was met.

## com.sabratec.applinx.baseobject Class GXWaitForScreen

```

java.lang.Object
  +- com.sabratec.applinx.baseobject.GXAbstractWaitCondition
      +- com.sabratec.applinx.baseobject.GXAbstractWaitScreen
          +- com.sabratec.applinx.baseobject.GXWaitForScreen
  
```

All Implemented Interfaces:

[GXWaitCondition](#)

```

public class GXWaitForScreen
extends GXAbstractWaitScreen
  
```

Wait condition that instructs the ApplinX server to wait for a specific screen.

### Constructor Summary

public	<a href="#">GXWaitForScreen</a> (String screenName, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait until the screen from the host is identified as the screen with the given name (ID).
--------	---

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitScreen](#)

[getScreenName](#), [setScreenName](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXWaitForScreen

```

public GXWaitForScreen(String screenName,
                       long timeout,
                       long flicker)
  
```

Sets a wait condition that instructs ApplinX server to wait until the screen from the host is identified as the screen with the given name (ID).

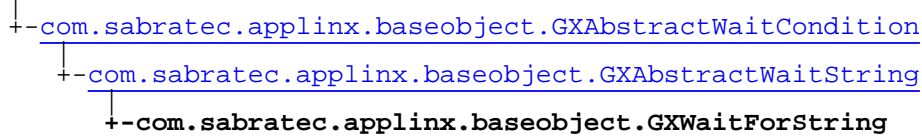
#### Parameters:

screenName - Name of the expected screen.  
 timeout - Expiration time of the wait condition.  
 flicker - Time to wait after the wait condition was met.

## com.sabratec.applinx.baseobject

### Class GXWaitForString

java.lang.Object



All Implemented Interfaces:

[GXWaitCondition](#)

public class **GXWaitForString**  
 extends [GXAbstractWaitString](#)

Wait condition that instructs the ApplinX server to wait for a specific string in a specific location on the screen.

### Constructor Summary

public	<a href="#">GXWaitForString</a> (String string, boolean isCaseSensitive, <a href="#">GXPosition</a> stringAt, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait until the given string appears at the given position.
public	<a href="#">GXWaitForString</a> (String string, boolean isCaseSensitive, <a href="#">GXPosition</a> upperLeft, <a href="#">GXPosition</a> lowerRight, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait until the given string appears in the rectangle specified by upper left and lower right positions.

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitString](#)

[getLowerRightPosition](#), [getString](#), [getUpperLeftPosition](#), [isCaseSensitive](#), [setIsCaseSensitive](#), [setLowerRightPosition](#), [setString](#), [setUpperLeftPosition](#)

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXWaitForString

```

public GXWaitForString(String string,
                      boolean isCaseSensitive,
                      GXPosition stringAt,
                      long timeout,
                      long flicker)
  
```

Sets a wait condition that instructs ApplinX server to wait until the given string appears at the given position.

(continued from last page)

**Parameters:**

`string` - The expected string.  
`isCaseSensitive` - Whether the condition is case sensitive  
`stringAt` - Position where the string should appear.  
`timeout` - Expiration time of the wait condition.  
`flicker` - Time to wait after the wait condition was met.

---

## GXWaitForString

```
public GXWaitForString(String string,  
                        boolean isCaseSensitive,  
                        GXPosition upperLeft,  
                        GXPosition lowerRight,  
                        long timeout,  
                        long flicker)
```

Sets a wait condition that instructs ApplinX server to wait until the given string appears in the rectangle specified by upper left and lower right positions.

**Parameters:**

`string` - The expected string.  
`isCaseSensitive` - Whether the condition is case sensitive  
`upperLeft` - Upper left position of the rectangle where the string should appear.  
`lowerRight` - Lower right position of the rectangle where the string should appear.  
`timeout` - Expiration time of the wait condition.  
`flicker` - Time to wait after the wait condition was met.

## com.sabratec.applinx.baseobject

### Class GXWaitHostQuiet

java.lang.Object

├─ [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)  
 └─ **com.sabratec.applinx.baseobject.GXWaitHostQuiet**

All Implemented Interfaces:

[GXWaitCondition](#)

public class **GXWaitHostQuiet**  
 extends [GXAbstractWaitCondition](#)

Wait condition that instructs the ApplinX server to wait a specified time before returning a host screen.

### Constructor Summary

public	<a href="#">GXWaitHostQuiet</a> (long timeout, long flicker) Sets a wait condition that instructs the ApplinX server to wait the specified time before returning the host screen.
public	<a href="#">GXWaitHostQuiet</a> (long timeout, long flicker, boolean waitForSystemUnlock)

### Method Summary

boolean	<a href="#">isWaitForSystemUnlock</a> () Determine Whether to wait for keyboard unlock or not (also called "set-ctrl-n"), default is true.
---------	---

Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### GXWaitHostQuiet

```
public GXWaitHostQuiet(long timeout,
                       long flicker)
```

Sets a wait condition that instructs the ApplinX server to wait the specified time before returning the host screen. This condition is less recommended because it will always wait the specified time before returning the host screen even if the screen has arrived fully before the expiration time.

#### Parameters:

**timeout** - Expiration time of the wait condition.  
**flicker** - Time to wait after the wait condition was met.

## GXWaitHostQuiet

```
public GXWaitHostQuiet(long timeout,  
                        long flicker,  
                        boolean waitForSystemUnlock)
```

### Parameters:

timeout - Expiration time of the wait condition.

flicker - Time to wait after the wait condition was met.

waitForSystemUnlock - Whether to wait for keyboard unlock or not (also called "set-ctrl-n"), default is true.

### See Also:

[GXWaitHostQuiet\(long, long\)](#)

## Methods

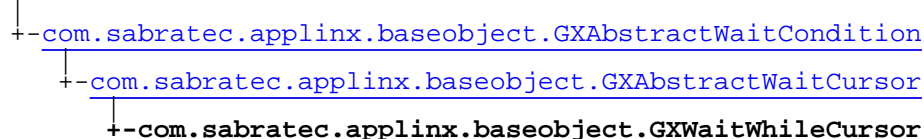
### isWaitForSystemUnlock

```
public boolean isWaitForSystemUnlock()
```

Determine Whether to wait for keyboard unlock or not (also called "set-ctrl-n"), default is true.

## com.sabratec.applinx.baseobject Class GXWaitWhileCursor

java.lang.Object



All Implemented Interfaces:

[GXWaitCondition](#)

public class **GXWaitWhileCursor**

extends [GXAbstractWaitCursor](#)

Wait condition that instructs the ApplinX server to wait as long as the cursor is in a specified location.

### Constructor Summary

public	<a href="#">GXWaitWhileCursor</a> ( <a href="#">GXPosition</a> cursorAt, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait while the cursor is at the specified position.
public	<a href="#">GXWaitWhileCursor</a> ( <a href="#">GXPosition</a> upperLeft, <a href="#">GXPosition</a> lowerRight, long timeout, long flicker) Sets a wait condition that instructs ApplinX Server to wait while the cursor is in the rectangle specified by upper left and lower right positions.

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCursor](#)

[getLowerRightPosition](#), [getUpperLeftPosition](#), [setLowerRightPosition](#), [setUpperLeftPosition](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### GXWaitWhileCursor

```
public GXWaitWhileCursor(GXPosition cursorAt,
                        long timeout,
                        long flicker)
```

Sets a wait condition that instructs ApplinX server to wait while the cursor is at the specified position.

##### Parameters:

cursorAt - Position where the cursor should be.

(continued from last page)

timeout - Expiration time of the wait condition.

flicker - Time to wait after the wait condition was met.

---

## **GXWaitWhileCursor**

```
public GXWaitWhileCursor(GXPosition upperLeft,  
                        GXPosition lowerRight,  
                        long timeout,  
                        long flicker)
```

Sets a wait condition that instructs ApplinX Server to wait while the cursor is in the rectangle specified by upper left and lower right positions.

### **Parameters:**

upperLeft - Upper left position of the rectangle where the cursor should be.

lowerRight - Lower right position of the rectangle where the cursor should be.

timeout - Expiration time of the wait condition.

flicker - Time to wait after the wait condition was met.

## com.sabratec.applinx.baseobject Class GXWaitWhileScreen

```

java.lang.Object
  |
  +--com.sabratec.applinx.baseobject.GXAbstractWaitCondition
        |
        +--com.sabratec.applinx.baseobject.GXAbstractWaitScreen
              |
              +--com.sabratec.applinx.baseobject.GXWaitWhileScreen
  
```

All Implemented Interfaces:

[GXWaitCondition](#)

```

public class GXWaitWhileScreen
extends GXAbstractWaitScreen
  
```

Wait condition that instructs the ApplinX server to wait until the screen is different from a specific screen.

### Constructor Summary

public	<a href="#">GXWaitWhileScreen</a> (String screenName, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait while the screen from the host is identified as the screen with the given name (ID).
--------	---

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitScreen](#)

[getScreenName](#), [setScreenName](#)

#### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXWaitWhileScreen

```

public GXWaitWhileScreen(String screenName,
                          long timeout,
                          long flicker)
  
```

Sets a wait condition that instructs ApplinX server to wait while the screen from the host is identified as the screen with the given name (ID).

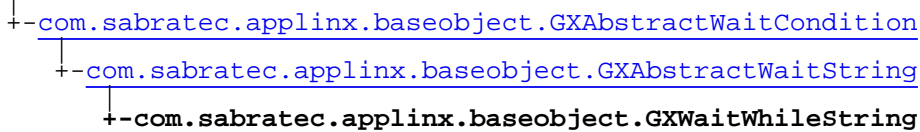
#### Parameters:

screenName - Name of the screen.  
 timeout - Expiration time of the wait condition.  
 flicker - Time to wait after the wait condition was met.

## com.sabratec.applinx.baseobject

### Class GXWaitWhileString

java.lang.Object



All Implemented Interfaces:

[GXWaitCondition](#)

public class **GXWaitWhileString**  
 extends [GXAbstractWaitString](#)

Wait condition that instructs the ApplinX server to wait as long as a specific string in a specific location on the screen is present.

### Constructor Summary

public	<a href="#">GXWaitWhileString</a> (String string, boolean isCaseSensitive, <a href="#">GXPosition</a> stringAt, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait while the given string appears at the given position.
public	<a href="#">GXWaitWhileString</a> (String string, boolean isCaseSensitive, <a href="#">GXPosition</a> upperLeft, <a href="#">GXPosition</a> lowerRight, long timeout, long flicker) Sets a wait condition that instructs ApplinX server to wait while the given string appears in the rectangle specified by upper left and lower right positions.

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitString](#)

[getLowerRightPosition](#), [getString](#), [getUpperLeftPosition](#), [isCaseSensitive](#), [setIsCaseSensitive](#), [setLowerRightPosition](#), [setString](#), [setUpperLeftPosition](#)

### Methods inherited from class [com.sabratec.applinx.baseobject.GXAbstractWaitCondition](#)

[getFlicker](#), [getTimeout](#), [setFlicker](#), [setTimeout](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXWaitWhileString

```

public GXWaitWhileString(String string,
                        boolean isCaseSensitive,
                        GXPosition stringAt,
                        long timeout,
                        long flicker)
  
```

Sets a wait condition that instructs ApplinX server to wait while the given string appears at the given position.

(continued from last page)

**Parameters:**

`string` - The expected string.  
`isCaseSensitive` - Whether the condition is case sensitive  
`stringAt` - Position where the string should appear.  
`timeout` - Expiration time of the wait condition.  
`flicker` - Time to wait after the wait condition was met.

---

## GXWaitWhileString

```
public GXWaitWhileString(String string,  
                          boolean isCaseSensitive,  
                          GXPosition upperLeft,  
                          GXPosition lowerRight,  
                          long timeout,  
                          long flicker)
```

Sets a wait condition that instructs ApplinX server to wait while the given string appears in the rectangle specified by upper left and lower right positions.

**Parameters:**

`string` - The expected string.  
`isCaseSensitive` - Whether the condition is case sensitive  
`upperLeft` - Upper left position of the rectangle where the string should appear.  
`lowerRight` - Lower right position of the rectangle where the string should appear.  
`timeout` - Expiration time of the wait condition.  
`flicker` - Time to wait after the wait condition was met.

---

## Package

# **com.sabratec.applinx.baseobject.query**

Contains classes and interfaces for executing data queries on screens and fields.

## com.sabratec.applinx.baseobject.query Class GXAbstractFieldMatchQuery

java.lang.Object

└─com.sabratec.applinx.baseobject.query.GXAbstractFieldMatchQuery

All Implemented Interfaces:

[GXIScreenCollectionFieldsQuery](#)

public abstract class **GXAbstractFieldMatchQuery**  
extends Object  
implements [GXIScreenCollectionFieldsQuery](#)

Description: This abstract query collects all the instances of fields that meet the abstract criteria from the first screen **only**

### Constructor Summary

public	<a href="#">GXAbstractFieldMatchQuery()</a>
--------	---

### Method Summary

abstract boolean	<a href="#">getFieldMatchQuery</a> ( <a href="#">GXIField</a> field)
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> ( <a href="#">GXIScreenCollection</a> screenCollection)

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface

[com.sabratec.applinx.baseobject.query.GXIScreenCollectionFieldsQuery](#)

[getFields](#)

## Constructors

### GXAbstractFieldMatchQuery

public **GXAbstractFieldMatchQuery**()

## Methods

### getFieldMatchQuery

public abstract boolean **getFieldMatchQuery**([GXIField](#) field)

---

## getFields

```
public GXIFieldCollection getFields(GXIScreenCollection screenCollection)
```

# com.sabratec.applinx.baseobject.query Class GXFieldCollectionQueryAppField

java.lang.Object  
└--com.sabratec.applinx.baseobject.query.GXFieldCollectionQueryAppField

All Implemented Interfaces:  
[GXIFieldCollectionQuery](#)

public class **GXFieldCollectionQueryAppField**  
extends Object  
implements [GXIFieldCollectionQuery](#)

Constructor Summary	
public	<a href="#">GXFieldCollectionQueryAppField</a> (String name, int index) Constructor
public	<a href="#">GXFieldCollectionQueryAppField</a> (String name) Constructor

Method Summary	
boolean	<a href="#">getFieldMatchQuery</a> ( <a href="#">GXIField</a> field) GXIField uses full name

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

Methods inherited from interface <a href="#">com.sabratec.applinx.baseobject.query.GXIFieldCollectionQuery</a>	
<a href="#">getFieldMatchQuery</a>	

## Constructors

**GXFieldCollectionQueryAppField**

public **GXFieldCollectionQueryAppField**(String name,  
int index)

Constructor

Parameters:

name - the full or short name of the application field

index - the instance of a multiple application field

**GXFieldCollectionQueryAppField**

public **GXFieldCollectionQueryAppField**(String name)

---

(continued from last page)

Constructor

**Parameters:**

name - the full or short name of the application field

## Methods

### **getFieldMatchQuery**

```
public boolean getFieldMatchQuery(GXField field)
```

GXField uses full name

## com.sabratec.applinx.baseobject.query Class GXFieldCollectionQueryProtected

java.lang.Object

└-com.sabratec.applinx.baseobject.query.GXFieldCollectionQueryProtected

All Implemented Interfaces:

[GXFieldCollectionQuery](#)

public class **GXFieldCollectionQueryProtected**  
extends Object  
implements [GXFieldCollectionQuery](#)

### Constructor Summary

public	<a href="#">GXFieldCollectionQueryProtected</a> (boolean isProtected) Constructor
--------	--

### Method Summary

boolean	<a href="#">getFieldMatchQuery</a> ( <a href="#">GXIField</a> field)
---------	--

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.baseobject.query.GXFieldCollectionQuery](#)

[getFieldMatchQuery](#)

### Constructors

#### **GXFieldCollectionQueryProtected**

public **GXFieldCollectionQueryProtected**(boolean isProtected)

Constructor

### Methods

#### **getFieldMatchQuery**

public boolean **getFieldMatchQuery**([GXIField](#) field)

## com.sabratec.applinx.baseobject.query Interface GXIFieldCollectionQuery

All Known Implementing Classes:

[GXFieldCollectionQueryProtected](#), [GXFieldCollectionQueryAppField](#)

public interface **GXIFieldCollectionQuery**

### Method Summary

boolean	<a href="#">getFieldMatchQuery</a> ( <a href="#">GXIField</a> field)
---------	--

### Methods

#### **getFieldMatchQuery**

public boolean **getFieldMatchQuery**([GXIField](#) field)

# com.sabratec.applinx.baseobject.query Interface GXIScreenCollectionFieldsQuery

All Known Implementing Classes:  
[GXScreenCollectionFieldNameQuery](#) , [GXAbstractFieldMatchQuery](#)

public interface **GXIScreenCollectionFieldsQuery**

This is the interface for all queries that are performed on GXIScreenCollection, and return GXIFieldCollection.

Method Summary	
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> ( <a href="#">GXIScreenCollection</a> screenCollection)

## Methods

### getFields

public [GXIFieldCollection](#) **getFields**([GXIScreenCollection](#) screenCollection)

# com.sabratec.applinx.baseobject.query

## Class GXScreenCollectionFieldNameQuery

java.lang.Object  
└--com.sabratec.applinx.baseobject.query.GXScreenCollectionFieldNameQuery

All Implemented Interfaces:  
[GXIScreenCollectionFieldsQuery](#)

public class **GXScreenCollectionFieldNameQuery**  
extends Object  
implements [GXIScreenCollectionFieldsQuery](#)

Description: This query collects all the instances of fields that have the name given to it in the constructor. If the field is a multiple appfield in one screen, all its instances will be collected.

Constructor Summary	
public	<a href="#">GXScreenCollectionFieldNameQuery</a> (String fieldName)

Method Summary	
<a href="#">GXIFieldCollection</a>	<a href="#">getFields</a> ( <a href="#">GXIScreenCollection</a> screenCollection)

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

Methods inherited from interface <a href="#">com.sabratec.applinx.baseobject.query.GXIScreenCollectionFieldsQuery</a>	
<a href="#">getFields</a>	

## Constructors

**GXScreenCollectionFieldNameQuery**  
public **GXScreenCollectionFieldNameQuery**(String fieldName)

## Methods

**getFields**  
public [GXIFieldCollection](#) **getFields**([GXIScreenCollection](#) screenCollection)

---

## Package

# **com.sabratec.applinx.baseobject.table**

Contains classes and interfaces for retrieving and manipulating ApplinX Tables.

## com.sabratec.applinx.baseobject.table Class GXAbstractTable

java.lang.Object

└─com.sabratec.applinx.baseobject.table.GXAbstractTable

### All Implemented Interfaces:

java.io.Externalizable, [GXITable](#)

### Direct Known Subclasses:

[GXPartialRowsTable](#), [GXMultipleRecordsTable](#), [GXMasterTable](#), [GXDefaultTable](#)

public abstract class **GXAbstractTable**

extends Object

implements [GXITable](#), java.io.Externalizable

## Constructor Summary

public	<a href="#">GXAbstractTable</a> (String name)
public	<a href="#">GXAbstractTable</a> ()
public	<a href="#">GXAbstractTable</a> (String name, String[] columnsNames)
public	<a href="#">GXAbstractTable</a> (String name, <a href="#">GXIScreenCollection</a> screens) Creates a new table with all the multiple application fields in all the screens in the screen collection as the table's columns.
public	<a href="#">GXAbstractTable</a> (String name, <a href="#">GXIScreen</a> screen) Creates a new table with all the multiple application fields in the screen as the table's columns.

## Method Summary

void	<a href="#">addColumn</a> ( <a href="#">GXITableColumn</a> column)
void	<a href="#">addColumn</a> (String column)
void	<a href="#">addColumnns</a> (String[] columns)
void	<a href="#">createTable</a> ( <a href="#">GXIScreen</a> screen)
void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screenCollection, GXVariables variables)
boolean	<a href="#">equals</a> (Object obj)
<a href="#">GXITableColumnCollection</a>	<a href="#">getColumns</a> ()

GXRectangle	<a href="#">getCornersCoordinates()</a>
GXRectangle	<a href="#">getHeaderCoordinates()</a>
String	<a href="#">getName()</a>
<a href="#">GXTableRow</a>	<a href="#">getNewRow()</a>
<a href="#">GXTableColumn[]</a>	<a href="#">getPrimaryKey()</a>
<a href="#">GXTableRowCollection</a>	<a href="#">getRows()</a>
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">removeColumn</a> (int columnIndex)
void	<a href="#">removeColumn</a> (String columnName)
void	<a href="#">setCornersCoordinates</a> (GXRectangle corners)
void	<a href="#">setHeaderCoordinates</a> (GXRectangle header)
void	<a href="#">setName</a> (String name)
void	<a href="#">setPrimaryKey</a> ( <a href="#">GXTableColumn</a> primaryKey)
void	<a href="#">setPrimaryKey</a> ( <a href="#">GXTableColumn[]</a> primaryKey)
void	<a href="#">sort</a> (int[] columnIndices, boolean[] isAscending)
void	<a href="#">sort</a> (int columnIndex, boolean isAscending)
void	<a href="#">sort</a> (String[] columnNames, boolean[] isAscending)
void	<a href="#">sort</a> (String columnName, boolean isAscending)
String	<a href="#">toString()</a>
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

**Methods inherited from class** java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface** [com.sabratec.applinx.baseobject.table.GXTable](#)
[addColumn](#), [addColumn](#), [addColumnns](#), [createTable](#), [createTable](#), [createTable](#), [getColumns](#),  
[getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [removeColumn](#), [removeColumn](#),  
[setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#)

**Methods inherited from interface** `java.io.Externalizable``readExternal, writeExternal`

## Constructors

### **GXAbstractTable**

```
public GXAbstractTable(String name)
```

### **GXAbstractTable**

```
public GXAbstractTable()
```

### **GXAbstractTable**

```
public GXAbstractTable(String name,  
                        String[] columnsNames)
```

### **GXAbstractTable**

```
public GXAbstractTable(String name,  
                        GXIScreenCollection screens)
```

Creates a new table with all the multiple application fields in all the screens in the screen collection as the table's columns.

### **GXAbstractTable**

```
public GXAbstractTable(String name,  
                        GXIScreen screen)
```

Creates a new table with all the multiple application fields in the screen as the table's columns.

## Methods

### **createTable**

```
public void createTable(GXIScreen screen)  
    throws GXColumnNotFoundException
```

### **createTable**

```
public void createTable(GXIScreenCollection screenCollection,  
                        GXVariables variables)  
    throws GXColumnNotFoundException
```

---

## removeColumn

```
public void removeColumn(int columnIndex)
```

---

---

## removeColumn

```
public void removeColumn(String columnName)
```

---

---

## sort

```
public void sort(String[] columnNames,  
                boolean[] isAscending)
```

---

---

## sort

```
public void sort(String columnName,  
                boolean isAscending)
```

---

---

## sort

```
public void sort(int[] columnIndices,  
                boolean[] isAscending)
```

---

---

## sort

```
public void sort(int columnIndex,  
                boolean isAscending)
```

---

---

## getName

```
public String getName()
```

---

---

## setName

```
public void setName(String name)
```

---

---

## getColumns

```
public GXITableColumnCollection getColumns()
```

---

---

## addColumn

```
public void addColumn(GXTableColumn column)
```

---

## addColumn

```
public void addColumn(String column)
```

---

## addColumnns

```
public void addColumnns(String[] columns)
```

---

## getRows

```
public GXTableRowCollection getRows()
```

---

## getNewRow

```
public GXTableRow getNewRow()
```

---

## getPrimaryKey

```
public GXTableColumn\[\] getPrimaryKey()
```

---

## setPrimaryKey

```
public void setPrimaryKey(GXTableColumn\[\] primaryKey)
```

---

## setPrimaryKey

```
public void setPrimaryKey(GXTableColumn primaryKey)
```

---

## setCornersCoordinates

```
public void setCornersCoordinates(GXRectangle corners)
```

---

## setHeaderCoordinates

```
public void setHeaderCoordinates(GXRectangle header)
```

---

(continued from last page)

---

## **getCornersCoordinates**

```
public GXRectangle getCornersCoordinates()
```

---

## **getHeaderCoordinates**

```
public GXRectangle getHeaderCoordinates()
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

## **toString**

```
public String toString()
```

## com.sabratec.applinx.baseobject.table Class GXCellable

java.lang.Object

└─com.sabratec.applinx.baseobject.table.GXCellable

All Implemented Interfaces:

[GXICellable](#)

public class **GXCellable**  
extends Object  
implements [GXICellable](#)

### Constructor Summary

public	<a href="#">GXCellable</a> (String content)
--------	---

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getContent</a> ()

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXICellable](#)

[getContent](#)

### Constructors

#### **GXCellable**

public **GXCellable**(String content)

### Methods

#### **getContent**

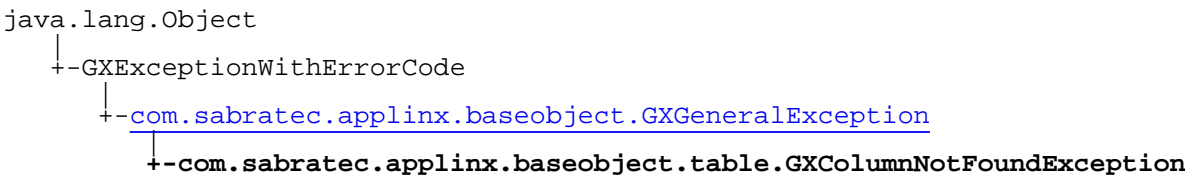
public String **getContent**()

(continued from last page)

**equals**

```
public boolean equals(Object obj)
```

# com.sabratec.applinx.baseobject.table Class GXColumnNotFoundException



public class **GXColumnNotFoundException**  
extends [GXGeneralException](#)

Constructor Summary	
public	<a href="#">GXColumnNotFoundException</a> (String columnName)

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

## Constructors

### GXColumnNotFoundException

public **GXColumnNotFoundException**(String columnName)

## com.sabratec.applinx.baseobject.table Class GXColumnsComparator

java.lang.Object

└-com.sabratec.applinx.baseobject.table.GXColumnsComparator

public class **GXColumnsComparator**  
extends Object

### Constructor Summary

public	<a href="#">GXColumnsComparator</a> ( <a href="#">GXITableColumn[]</a> columns, boolean isCaseSensitive, boolean[] isAscending)
--------	---

### Method Summary

int	<a href="#">compare</a> (Object a, Object b)
-----	--

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### GXColumnsComparator

```
public GXColumnsComparator(GXITableColumn\[\] columns,  
                           boolean isCaseSensitive,  
                           boolean[] isAscending)
```

##### Parameters:

columns - the table columns. Should be sorted according to the desired sorting order.

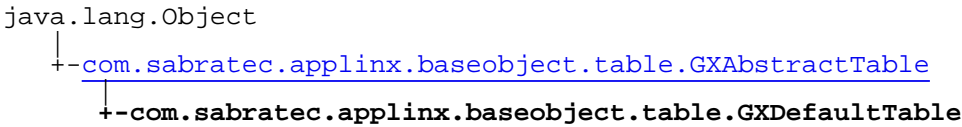
### Methods

#### compare

```
public int compare(Object a,  
                  Object b)
```

# com.sabratec.applinx.baseobject.table

## Class GXDefaultTable



All Implemented Interfaces:  
java.io.Externalizable, [GXITable](#)

public class **GXDefaultTable**  
extends [GXAbstractTable](#)

This table should be used in the following cases: - A table based on one host screen. - A table with vertical scrolling only (one host screen being scrolled) - A table with vertical and horizontal scrolling, but with the same records in each instance of scrolling over the screens.

Constructor Summary	
public	<a href="#">GXDefaultTable</a> ()
public	<a href="#">GXDefaultTable</a> (String name)
public	<a href="#">GXDefaultTable</a> (String name, String[] columnsNames)
public	<a href="#">GXDefaultTable</a> (String name, <a href="#">GXIScreenCollection</a> screens) Creates a new table with all the multiple application fields in all the screens in the screen collection as the table's columns.
public	<a href="#">GXDefaultTable</a> (String name, <a href="#">GXIScreen</a> screen) Creates a new table with all the multiple application fields in the screen as the table's columns.

Method Summary	
void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screens)

Methods inherited from class <a href="#">com.sabratec.applinx.baseobject.table.GXAbstractTable</a>
<a href="#">addColumn</a> , <a href="#">addColumn</a> , <a href="#">addColumns</a> , <a href="#">createTable</a> , <a href="#">createTable</a> , <a href="#">equals</a> , <a href="#">getColumns</a> , <a href="#">getCornersCoordinates</a> , <a href="#">getHeaderCoordinates</a> , <a href="#">getName</a> , <a href="#">getNewRow</a> , <a href="#">getPrimaryKey</a> , <a href="#">getRows</a> , <a href="#">readExternal</a> , <a href="#">removeColumn</a> , <a href="#">removeColumn</a> , <a href="#">setCornersCoordinates</a> , <a href="#">setHeaderCoordinates</a> , <a href="#">setName</a> , <a href="#">setPrimaryKey</a> , <a href="#">setPrimaryKey</a> , <a href="#">sort</a> , <a href="#">sort</a> , <a href="#">sort</a> , <a href="#">sort</a> , <a href="#">toString</a> , <a href="#">writeExternal</a>

Methods inherited from class java.lang.Object
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>

Methods inherited from interface <a href="#">com.sabratec.applinx.baseobject.table.GXITable</a>
---

---

```
addColumn, addColumn, addColumns, createTable, createTable, createTable, getColumns,  
getName, getNewRow, getPrimaryKey, getRows, removeColumn, removeColumn,  
setPrimaryKey, setPrimaryKey, sort, sort, sort, sort
```

---

Methods inherited from interface `java.io.Externalizable`

```
readExternal, writeExternal
```

---

## Constructors

### **GXDefaultTable**

```
public GXDefaultTable()
```

---

### **GXDefaultTable**

```
public GXDefaultTable(String name)
```

---

### **GXDefaultTable**

```
public GXDefaultTable(String name,  
                      String[] columnsNames)
```

---

### **GXDefaultTable**

```
public GXDefaultTable(String name,  
                     GXIScreenCollection screens)
```

Creates a new table with all the multiple application fields in all the screens in the screen collection as the table's columns.

---

### **GXDefaultTable**

```
public GXDefaultTable(String name,  
                     GXIScreen screen)
```

Creates a new table with all the multiple application fields in the screen as the table's columns.

---

## Methods

### **createTable**

```
public void createTable(GXIScreenCollection screens)  
    throws GXColumnNotFoundException
```

## com.sabratec.applinx.baseobject.table Class GXFieldTableCell

java.lang.Object

```

  |
+-com.sabratec.applinx.baseobject.table.GXTableCell
  |
+-com.sabratec.applinx.baseobject.table.GXFieldTableCell

```

All Implemented Interfaces:

[GXIField](#), java.io.Serializable, [GXITableCell](#)

public class **GXFieldTableCell**  
 extends [GXTableCell](#)  
 implements [GXITableCell](#), java.io.Serializable, [GXIField](#)

Host field based cell.

A table cell based on a host application field. All the properties of a host field are available (colors, attributes, etc.).

Fields inherited from interface [com.sabratec.applinx.baseobject.GXIField](#)

[TYPE\\_APPFIELD](#), [TYPE\\_HOSTFIELD](#), [TYPE\\_HOSTKEY\\_FIELD](#), [TYPE\\_MULTIPLE\\_APPFIELD](#)

### Constructor Summary

public	<a href="#">GXFieldTableCell</a> ( <a href="#">GXIField</a> hostField)
--------	--

### Method Summary

int	<a href="#">getBGColor</a> ()
int	<a href="#">getDataType</a> ()
int	<a href="#">getFGColor</a> ()
int	<a href="#">getIndex</a> ()
String	<a href="#">getInputMask</a> ()
int	<a href="#">getLength</a> ()
String	<a href="#">getName</a> ()
<a href="#">GXPosition</a>	<a href="#">getPosition</a> ()
String	<a href="#">getScreenName</a> ()
int	<a href="#">getScreenSequenceNumber</a> ()

String	<a href="#">getVisualContent()</a>
boolean	<a href="#">isAppField()</a>
boolean	<a href="#">isAutoCursorJump()</a>
boolean	<a href="#">isAutoEnter()</a>
boolean	<a href="#">isBlinking()</a>
boolean	<a href="#">isHighIntensified()</a>
boolean	<a href="#">isMultiple()</a>
boolean	<a href="#">isProtected()</a>
boolean	<a href="#">isReversedVideo()</a>
boolean	<a href="#">isUnderlined()</a>
boolean	<a href="#">isVisible()</a>
String	<a href="#">toString()</a>

#### Methods inherited from class [com.sabratec.applinx.baseobject.table.GXTableCell](#)

[equals](#), [getContent](#), [setContent](#)

#### Methods inherited from class [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITableCell](#)

[getContent](#), [setContent](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.GXIField](#)

[getBGColor](#), [getContent](#), [getDataType](#), [getFGColor](#), [getIndex](#), [getInputMask](#), [getLength](#), [getName](#), [getPosition](#), [getScreenName](#), [getScreenSequenceNumber](#), [getVisualContent](#), [isAppField](#), [isAutoCursorJump](#), [isAutoEnter](#), [isBlinking](#), [isHighIntensified](#), [isMultiple](#), [isProtected](#), [isReversedVideo](#), [isUnderlined](#), [isVisible](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXICellable](#)

[getContent](#)

## Constructors

(continued from last page)

## GXFieldTableCell

```
public GXFieldTableCell(GXIField hostField)
```

## Methods

### getName

```
public String getName()
```

### getIndex

```
public int getIndex()
```

### getScreenName

```
public String getScreenName()
```

### isMultiple

```
public boolean isMultiple()
```

### getScreenSequenceNumber

```
public int getScreenSequenceNumber()
```

### getPosition

```
public GXPosition getPosition()
```

### getLength

```
public int getLength()
```

### isProtected

```
public boolean isProtected()
```

(continued from last page)

---

## isVisible

```
public boolean isVisible()
```

---

## isHighIntensified

```
public boolean isHighIntensified()
```

---

## isAutoEnter

```
public boolean isAutoEnter()
```

---

## isAutoCursorJump

```
public boolean isAutoCursorJump()
```

---

## isBlinking

```
public boolean isBlinking()
```

---

## getFGColor

```
public int getFGColor()
```

---

## getBGColor

```
public int getBGColor()
```

---

## getDataType

```
public int getDataType()
```

---

## getInputMask

```
public String getInputMask()
```

---

## getVisualContent

```
public String getVisualContent()
```

---

(continued from last page)

---

---

## **isAppField**

```
public boolean isAppField()
```

---

---

## **isUnderlined**

```
public boolean isUnderlined()
```

---

---

## **toString**

```
public String toString()
```

---

---

## **isReversedVideo**

```
public boolean isReversedVideo()
```

---

## com.sabratec.applinx.baseobject.table Class GXHostTableUtils

java.lang.Object

└─com.sabratec.applinx.baseobject.table.GXHostTableUtils

public class **GXHostTableUtils**  
extends Object

Title: GXHostTableUtils

Description: contains table utility functions, used mainly for display.

### Constructor Summary

public	<a href="#">GXHostTableUtils()</a>
--------	------------------------------------

### Method Summary

static <a href="#">GXPosition</a>	<a href="#">getDefinedLowerRightCorner</a> ( <a href="#">GXITable</a> table)
static <a href="#">GXPosition</a>	<a href="#">getDefinedUpperLeftCorner</a> ( <a href="#">GXITable</a> table)
static int	<a href="#">getFirstAppfieldColumnIndex</a> ( <a href="#">GXITable</a> table)
static <a href="#">GXPosition</a>	<a href="#">getHeaderLowerRight</a> ( <a href="#">GXITable</a> table)
static <a href="#">GXPosition</a>	<a href="#">getHeaderUpperLeft</a> ( <a href="#">GXITable</a> table)
static int	<a href="#">getLastAppfieldColumnIndex</a> ( <a href="#">GXITable</a> table)
static <a href="#">GXPosition</a>	<a href="#">getLowerRightCorner</a> ( <a href="#">GXITable</a> table)
static <a href="#">GXPosition</a>	<a href="#">getUpperLeftCorner</a> ( <a href="#">GXITable</a> table)
static boolean	<a href="#">isColumnAppfieldColumn</a> ( <a href="#">GXITable</a> table, int column)
static boolean	<a href="#">isPositionInTable</a> ( <a href="#">GXITable</a> table, int row, int col)

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

(continued from last page)

## GXHostTableUtils

```
public GXHostTableUtils()
```

## Methods

### getHeaderUpperLeft

```
public static GXPosition getHeaderUpperLeft(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

the table header's upper left corner, as defined in the Table entity in ApplinX Composer.

### getHeaderLowerRight

```
public static GXPosition getHeaderLowerRight(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

the table header's lower right corner, as defined in the Table entity in ApplinX Composer.

### getDefinedUpperLeftCorner

```
public static GXPosition getDefinedUpperLeftCorner(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

the upper left corner of the table area, as defined in the Table entity in ApplinX Composer. May be different than the actual data corner, because of filters.

**See Also:**

[getHeaderUpperLeft\(GXITable\)](#)

### getDefinedLowerRightCorner

```
public static GXPosition getDefinedLowerRightCorner(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

---

(continued from last page)

the lower right corner of the table area, as defined in the Table entity in ApplinX Composer. May be different than the actual data corner, because of filters.

**See Also:**

[getLowerRightCorner\(GXITable\)](#)

---

## isPositionInTable

```
public static boolean isPositionInTable(GXITable table,
    int row,
    int col)
```

**Parameters:**

table - the table.

row - should range between 0 to rowCount - 1.

col - should range between 0 to columnCount - 1.

**Returns:**

whether the position in the specified row / column coordinates is inside the specified table, in its defined area (the actual area can be smaller in case of filters).

---

## getUpperLeftCorner

```
public static GXPosition getUpperLeftCorner(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

the actual upper left corner of the table.

**See Also:**

[getDefinedUpperLeftCorner\(GXITable\)](#)

---

## isColumnAppfieldColumn

```
public static boolean isColumnAppfieldColumn(GXITable table,
    int column)
```

**Parameters:**

table - the table.

column - column's index, should range between 0 to columnCount - 1.

**Returns:**

whether the column at the specified index is of type Application Field.

---

## getLowerRightCorner

```
public static GXPosition getLowerRightCorner(GXITable table)
```

**Parameters:**

table - the table.

---

---

(continued from last page)

**Returns:**

the actual lower right corner of the table.

**See Also:**

[getDefinedLowerRightCorner\(GXITable\)](#)

---

## getFirstAppfieldColumnIndex

```
public static int getFirstAppfieldColumnIndex(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

the index of the first table column which is of type Application Field, -1 if no such column is found.

---

## getLastAppfieldColumnIndex

```
public static int getLastAppfieldColumnIndex(GXITable table)
```

**Parameters:**

table - the table.

**Returns:**

the index of the last table column which is of type Application Field, -1 if no such column is found.

# com.sabratec.applinx.baseobject.table Interface GXICellable

All Subinterfaces:  
[GXIField](#)

All Known Implementing Classes:  
[GXCellable](#)

public interface **GXICellable**

Description: This interface represents a cell in a host table.

Method Summary	
String	<a href="#">getContent()</a>

## Methods

### getContent

public String **getContent()**

**Returns:**  
the content of the cell.

---

## com.sabratec.applinx.baseobject.table Interface GXIColumnIdentifier

---

public interface **GXIColumnIdentifier**

Used by GXAddFieldToPreviousRowFilter, in order to determine if a row contains a field to be copied to a previous row.

---

### Method Summary

String	<a href="#">getColumnName()</a>
boolean	<a href="#">isCell(<a href="#">GXITableRow</a> row)</a>

---

### Methods

#### isCell

public boolean **isCell**([GXITableRow](#) row)

---

#### getColumnName

public String **getColumnName**()

---

## com.sabratec.applinx.baseobject.table Interface GXIMasterRowIdentifier

---

public interface **GXIMasterRowIdentifier**

---

### Method Summary

boolean	<a href="#">isMasterRow</a> ( <a href="#">GXITableRow</a> row)
---------	--

---

### Methods

#### **isMasterRow**

public boolean **isMasterRow**([GXITableRow](#) row)

## com.sabratec.applinx.baseobject.table Interface GXIRowsFilter

All Known Implementing Classes:

[GXNonEmptyRowsFilter](#), [GXNonDuplicateRowsFilter](#), [GXAddFieldToPreviousRowFilter](#)

public interface **GXIRowsFilter**  
extends java.io.Serializable

Row filtering object.

This class represents a filter for excluding table rows, based on certain conditions. Currently, when defining a table entity in ApplinX Composer, two types of filters can be defined: filtering empty rows and filtering duplicate rows. Using this interface, custom filters can be added to the table's RowCollection.

**See Also:**

### Method Summary

<a href="#">GXITableRowCollection</a>	<a href="#">filter</a> ( <a href="#">GXITableRowCollection</a> rows) Used to filter an existing row collection.
boolean	<a href="#">isPrimaryKeyValid</a> (String primaryKey, <a href="#">GXITableRowCollection</a> rowsSoFar) This method is used, by ApplinX Server, when building a table.
boolean	<a href="#">isRowValid</a> ( <a href="#">GXITableRow</a> row, <a href="#">GXITableRowCollection</a> rowsSoFar) This method is used, by ApplinX Server, when building a table, to decide whether to add a row to the table.

### Methods

#### filter

public [GXITableRowCollection](#) **filter**([GXITableRowCollection](#) rows)

Used to filter an existing row collection.

**Parameters:**

rows - the rows to filter.

**Returns:**

the rows after filtering.

#### isRowValid

public boolean **isRowValid**([GXITableRow](#) row,  
[GXITableRowCollection](#) rowsSoFar)

This method is used, by ApplinX Server, when building a table, to decide whether to add a row to the table. Can also be used by filter(), for each row.

**Parameters:**

(continued from last page)

`row` - the row to check

`rowsSoFar` - the rows added to the table so far, for conditions that depend on previous rows (for example - duplicate rows).

**Returns:**

whether the row is valid or not.

**See Also:**

[filter\(GXITableRowCollection\)](#)

---

## isPrimaryKeyValid

```
public boolean isPrimaryKeyValid(String primaryKey,  
    GXITableRowCollection rowsSoFar)
```

This method is used, by ApplinX Server, when building a table. Before starting to build a row, an initial check validates the primary key. If it isn't valid, the row is skipped.

**Parameters:**

`primaryKey` - the row's primary key.

`rowsSoFar` - the rows added to the table so far, for conditions that depend on previous rows (for example - duplicate rows).

**Returns:**

whether the row's primary key is valid or not.

## com.sabratec.applinx.baseobject.table Interface GXITable

All Known Implementing Classes:

[GXAbstractTable](#)

public interface **GXITable**  
extends java.io.Serializable

General table entity.

The external table interface.

### Method Summary

void	<a href="#">addColumn</a> ( <a href="#">GXITableColumn</a> column) Adds a column at the end of the table.
void	<a href="#">addColumn</a> (String columnName) Creates a new table column of the specified name and adds it at the end of the table.
void	<a href="#">addColumnns</a> (String[] columnNames) Creates new table columns of the specified names and adds them at the end of the table, according to their order in the names array.
void	<a href="#">createTable</a> ( <a href="#">GXIScreen</a> screen)
void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screenCollection)
void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screenCollection, <a href="#">GXVariables</a> variables) Creates a table based on Base Object screens and variables and on the table's columns schema.
<a href="#">GXITableColumnCollection</a>	<a href="#">getColumns</a> ()
String	<a href="#">getName</a> ()
<a href="#">GXITableRow</a>	<a href="#">getNewRow</a> () Used to programmatically add a new row to the table.
<a href="#">GXITableColumn[]</a>	<a href="#">getPrimaryKey</a> () The table's primary key is based on one or more columns.
<a href="#">GXITableRowCollection</a>	<a href="#">getRows</a> ()
void	<a href="#">removeColumn</a> (int columnIndex) Removes the column at the specified index.
void	<a href="#">removeColumn</a> (String columnName) Removes the column of the specified name.

void	<a href="#">setPrimaryKey</a> ( <a href="#">GXITableColumn</a> primaryKey) Sets one table column as the primary key.
void	<a href="#">setPrimaryKey</a> ( <a href="#">GXITableColumn[]</a> primaryKey) Sets the columns that compose the table's primary key.
void	<a href="#">sort</a> (int[] columnIndices, boolean[] isAscending) Sorts table rows according to specified columns indices.
void	<a href="#">sort</a> (int columnIndex, boolean isAscending) Sorts table rows according to specified column index.
void	<a href="#">sort</a> (String[] columnNames, boolean[] isAscending) Sorts table rows according to specified column names.
void	<a href="#">sort</a> (String columnName, boolean isAscending) Sorts table rows according to specified column name.

Methods

getName

```
public String getName()
```

**Returns:**  
table's name, defined in the table entity definition.

getColumns

```
public GXITableColumnCollection getColumns()
```

**Returns:**  
table's defined columns.

addColumn

```
public void addColumn(GXITableColumn column)
```

Adds a column at the end of the table.

**Parameters:**  
column - the column to add.

addColumn

```
public void addColumn(String columnName)
```

Creates a new table column of the specified name and adds it at the end of the table.

**Parameters:**  
columnName - name of the column to create and add.

---

## addColumnns

```
public void addColumnns(String[] columnNames)
```

Creates new table columns of the specified names and adds them at the end of the table, according to their order in the names array.

### Parameters:

columnNames - names of the column to create and add.

---

## removeColumn

```
public void removeColumn(int columnIndex)
```

Removes the column at the specified index.

### Parameters:

columnIndex - column index to remove, should range between 0 to columnCount - 1.

---

## removeColumn

```
public void removeColumn(String columnName)
```

Removes the column of the specified name.

### Parameters:

columnName - column name to remove.

---

## getRows

```
public GXITableRowCollection getRows()
```

### Returns:

table's data rows.

---

## getNewRow

```
public GXITableRow getNewRow()
```

Used to programmatically add a new row to the table. Done by getting a new row, filling in its cells and then adding it to the table.

### Returns:

a new empty row, containing the table's defined columns.

---

## getPrimaryKey

```
public GXITableColumn\[\] getPrimaryKey()
```

The table's primary key is based on one or more columns. The key should yield a unique record. If two records have the same key, one is chosen arbitrarily when searching a row according to the primary key. If no primary key column is specified, the default key is the screen index + row index (in the host screen).

### Returns:

table's columns that compose the primary key definition.

---

## setPrimaryKey

```
public void setPrimaryKey(GXITableColumn\[\] primaryKey)
```

Sets the columns that compose the table's primary key.

### Parameters:

primaryKey - the columns that compose the primary key.

### See Also:

[getPrimaryKey\(\)](#)

---

## setPrimaryKey

```
public void setPrimaryKey(GXITableColumn primaryKey)
```

Sets one table column as the primary key.

### Parameters:

primaryKey - the table column to serve as the primary key.

### See Also:

[getPrimaryKey\(\)](#)

---

## createTable

```
public void createTable(GXIScreenCollection screenCollection,  
    GXVariables variables)  
throws GXColumnNotFoundException
```

Creates a table based on Base Object screens and variables and on the table's columns schema.

### Parameters:

screenCollection - a collection of screen(s).

variables - path variables.

### Throws:

[GXColumnNotFoundException](#) - when the field or variable associated to the column is not found in the input.

---

## createTable

```
public void createTable(GXIScreenCollection screenCollection)  
throws GXColumnNotFoundException
```

### See Also:

[createTable\(GXIScreenCollection, GXVariables\)](#)

---

## createTable

```
public void createTable(GXIScreen screen)  
throws GXColumnNotFoundException
```

### See Also:

[createTable\(GXIScreenCollection, GXVariables\)](#)

---

---

## sort

```
public void sort(String[] columnNames,  
                boolean[] isAscending)
```

Sorts table rows according to specified column names. Sort priority is determined by the order of the columns in the names array.

### Parameters:

columnNames - sorting columns.

isAscending - whether they should be sorted in ascending or descending order. Note: the sort is case-sensitive.

---

## sort

```
public void sort(String columnName,  
                boolean isAscending)
```

Sorts table rows according to specified column name.

### Parameters:

columnName - sorting column.

isAscending - whether they should be sorted in ascending or descending order. Note: the sort is case-sensitive.

---

## sort

```
public void sort(int[] columnIndices,  
                boolean[] isAscending)
```

Sorts table rows according to specified columns indices. Sort priority is determined by the order of columns in the names array.

### Parameters:

columnIndices - sorting columns.

isAscending - whether they should be sorted in ascending or descending order. Note: the sort is case-sensitive.

---

## sort

```
public void sort(int columnIndex,  
                boolean isAscending)
```

Sorts table rows according to specified column index.

### Parameters:

columnIndex - sorting column.

isAscending - whether they should be sorted in ascending or descending order. Note: the sort is case-sensitive.

---

## com.sabratec.applinx.baseobject.table Interface GXITableCell

All Known Implementing Classes:

[GXTableCell](#)

---

public interface **GXITableCell**

Basic table cell.

a basic cell in a table.

**See Also:**

[GXFieldTableCell](#), [GXVariableTableCell](#)

---

### Method Summary

String	<a href="#">getContent()</a>
void	<a href="#">setContent(String content)</a> Sets the content of the cell.

---

### Methods

#### **getContent**

public String **getContent**()

**Returns:**

cell's content.

---

#### **setContent**

public void **setContent**(String content)

Sets the content of the cell. Note this method only affects the cell (the contents are not sent to the host).

**Parameters:**

content - new cell's content.

## com.sabratec.applinx.baseobject.table Interface GXTableColumn

All Known Implementing Classes:

[GXTableColumn](#)

public interface **GXTableColumn**

Table column.

Represents a column definition in the table.

### Field Summary

public static final	<a href="#">GX_TABLE_COLUMN_DATA_TYPE_ALPHANUMERIC</a> Value: <b>0</b>
public static final	<a href="#">GX_TABLE_COLUMN_DATA_TYPE_NUMERIC</a> Value: <b>1</b>
public static final	<a href="#">GX_TABLE_COLUMN_TYPE_APPFIELD</a>
public static final	<a href="#">GX_TABLE_COLUMN_TYPE_OTHER</a>
public static final	<a href="#">GX_TABLE_COLUMN_TYPE_VARIABLE</a>

### Method Summary

String	<a href="#">getAppFieldFullName()</a>
String	<a href="#">getAppFieldName()</a>
String	<a href="#">getCaption()</a>
int	<a href="#">getCopyCount()</a>
int	<a href="#">getDataType()</a>
String	<a href="#">getName()</a>
String[]	<a href="#">getScreenNames()</a> Used for horizontal scrolling, for tables spanning over more than one identified host screen.
int	<a href="#">getType()</a> Returns the type of the column's mapped object.
String	<a href="#">getVariableName()</a>

boolean	<a href="#"><u>isMasterRow()</u></a>
boolean	<a href="#"><u>isToCopy()</u></a>
void	<a href="#"><u>setAppFieldName</u></a> (String name) Sets the name of the application field that is to be associated with this column.
void	<a href="#"><u>setCaption</u></a> (String caption) Sets the column's caption.
void	<a href="#"><u>setCopyCount</u></a> (int copyCount)
void	<a href="#"><u>setDataType</u></a> (int dataType) sets the table's data type.
void	<a href="#"><u>setIsMasterRow</u></a> (boolean isMasterRow)
void	<a href="#"><u>setIsToCopy</u></a> (boolean isToCopy)
void	<a href="#"><u>setName</u></a> (String name) Sets the name of the column.
void	<a href="#"><u>setScreenNames</u></a> (String[] names)
void	<a href="#"><u>setVariableName</u></a> (String name) Sets the name of the variable that is to be associated with this column.

## Fields

### **GX\_TABLE\_COLUMN\_DATA\_TYPE\_ALPHANUMERIC**

```
public static final int GX_TABLE_COLUMN_DATA_TYPE_ALPHANUMERIC
```

Constant value: 0

### **GX\_TABLE\_COLUMN\_DATA\_TYPE\_NUMERIC**

```
public static final int GX_TABLE_COLUMN_DATA_TYPE_NUMERIC
```

Constant value: 1

### **GX\_TABLE\_COLUMN\_TYPE\_APPFIELD**

```
public static final int GX_TABLE_COLUMN_TYPE_APPFIELD
```

### **GX\_TABLE\_COLUMN\_TYPE\_VARIABLE**

```
public static final int GX_TABLE_COLUMN_TYPE_VARIABLE
```

---

## GX\_TABLE\_COLUMN\_TYPE\_OTHER

```
public static final int GX_TABLE_COLUMN_TYPE_OTHER
```

## Methods

### getDataType

```
public int getDataType()
```

**Returns:**

the column's data type, as defined in the data type constants.

---

### setDataType

```
public void setDataType(int dataType)
```

sets the table's data type.

**Parameters:**

dataType - table's data type, should be one of the defined types.

---

### getType

```
public int getType()
```

Returns the type of the column's mapped object. Current possible types are application field and variable array, as defined in the column type constants.

**Returns:**

the type of the column's mapped object.

---

### getName

```
public String getName()
```

**Returns:**

column's name.

---

### setName

```
public void setName(String name)
```

Sets the name of the column. A column name should be unique, setting a column's name to a name that already exists will override the existing column.

**Parameters:**

name - new column's name.

(continued from last page)

---

## getAppFieldName

```
public String getAppFieldName()
```

### Returns:

the name of the application field associated with this column, when this column is an application field column, otherwise null.

---

## getAppFieldFullName

```
public String getAppFieldFullName()
```

### Returns:

the full name of the application field associated with this column, when this column is an application field column, otherwise null.

---

## setAppFieldName

```
public void setAppFieldName(String name)
```

Sets the name of the application field that is to be associated with this column. Setting the name also makes the column's type be Application Field based.

### Parameters:

name - the application field's full name (including the folder structure).

---

## getVariableName

```
public String getVariableName()
```

### Returns:

the name of the variable associated with this column, when this column is a variable column, otherwise null.

---

## setVariableName

```
public void setVariableName(String name)
```

Sets the name of the variable that is to be associated with this column. Setting the name also makes the column's type Variable based.

### Parameters:

name - the variable's name.

---

## getCaption

```
public String getCaption()
```

### Returns:

the column's caption.

---

(continued from last page)

---

## setCaption

```
public void setCaption(String caption)
```

Sets the column's caption.

### Parameters:

caption - the new caption.

---

## getScreenNames

```
public String[] getScreenNames()
```

Used for horizontal scrolling, for tables spanning over more than one identified host screen. Multiple screen names are used if the same application field is mapped in more than one screen, but should be collected for the table only from some of the screens.

### Returns:

the full names of the screens.

---

## setScreenNames

```
public void setScreenNames(String[] names)
```

### Parameters:

names - full screen names (including folder structure).

### See Also:

[getScreenNames\(\)](#)

---

## setIsMasterRow

```
public void setIsMasterRow(boolean isMasterRow)
```

---

## isMasterRow

```
public boolean isMasterRow()
```

---

## setIsToCopy

```
public void setIsToCopy(boolean isToCopy)
```

---

## isToCopy

```
public boolean isToCopy()
```

---

(continued from last page)

## **setCopyCount**

```
public void setCopyCount(int copyCount)
```

---

## **getCopyCount**

```
public int getCopyCount()
```

## com.sabratec.applinx.baseobject.table Interface GXITableColumnCollection

All Known Implementing Classes:  
[GXITableColumnCollection](#)

public interface **GXITableColumnCollection**

Table's column schema.

Represents the defined columns schema of this table. The collection is composed of GXITableColumn.

### Method Summary

void	<a href="#">addColumn</a> ( <a href="#">GXITableColumn</a> column) Adds a new column to the schema.
<a href="#">GXITableColumn</a>	<a href="#">getColumn</a> (int columnIndex)
<a href="#">GXITableColumn</a>	<a href="#">getColumn</a> (String columnName)
int	<a href="#">getColumnCount</a> ()
<a href="#">GXITableColumn[]</a>	<a href="#">getColumns</a> ()
void	<a href="#">removeColumn</a> (int columnIndex) Removes a column from the column's schema only, and not the actual column's data.
void	<a href="#">removeColumn</a> (String columnName) Removes a column from the column's schema only, and not the actual column's data.

### Methods

#### getColumns

```
public GXITableColumn\[\] getColumns()
```

**Returns:**

the column's collection as an array.

#### getColumnCount

```
public int getColumnCount()
```

**Returns:**

the number of columns defined for this table.

## getColumn

```
public GXITableColumn getColumn(int columnIndex)
```

**Parameters:**

columnIndex - the index of the column to retrieve. Should range between 0 and columnCount.

**Returns:**

the table's column at the specified index.

**See Also:**

[getColumnCount\(\)](#)

---

## getColumn

```
public GXITableColumn getColumn(String columnName)
```

**Parameters:**

columnName - the name of the column to retrieve.

**Returns:**

the table's column of the specified name.

---

## addColumn

```
public void addColumn(GXITableColumn column)
```

Adds a new column to the schema. The new column will be added at the end of the column's list.

**Parameters:**

column - the new column to add.

---

## removeColumn

```
public void removeColumn(int columnIndex)
```

Removes a column from the column's schema only, and not the actual column's data. To remove columns and their data, use `GXITable.removeColumn`

**Parameters:**

columnIndex - the index of the column to remove. Should range between 0 and columnCount.

**See Also:**

[GXITable.removeColumn\(int\)](#)  
[getColumnCount\(\)](#)

---

## removeColumn

```
public void removeColumn(String columnName)
```

Removes a column from the column's schema only, and not the actual column's data. To remove columns and their data, use `GXITable.removeColumn`

**Parameters:**

(continued from last page)

columnName - the name of the column to remove.

**See Also:**

[GXITable.removeColumn\(String\)](#)

## com.sabratec.applinx.baseobject.table Interface GXITableRow

All Known Implementing Classes:

[GXITableRow](#)

public interface **GXITableRow**

Table row.

Represents a data row in the table. Each row consists of GXITableCell. The cells structure in the row is determined according to the column schema.

### Method Summary

<a href="#">GXITableCell</a>	<a href="#">getItem</a> (int columnIndex)
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (String columnName)
String	<a href="#">getItemContent</a> (int columnIndex)
String	<a href="#">getItemContent</a> (String columnName)
<a href="#">GXITableCell[]</a>	<a href="#">getItems</a> ()
String[]	<a href="#">getItemsContents</a> ()
String	<a href="#">getPrimaryKey</a> ()
void	<a href="#">removeItem</a> (int columnIndex) Removes the cell at the specified column index.
void	<a href="#">removeItem</a> (String columnName) Removes the cell of the specified column name.
void	<a href="#">setItem</a> (int columnIndex, <a href="#">GXITableCell</a> item) Sets a table cell at specified column index, and replaces the previous cell (if such existed).
void	<a href="#">setItem</a> (String columnName, <a href="#">GXITableCell</a> item) Sets a table cell of the specified column name, and replaces the previous cell (if such existed).
void	<a href="#">setItemContent</a> (int columnIndex, String content) Sets the content of the cell in the specified column's index.
void	<a href="#">setItemContent</a> (String columnName, String content) Sets the content of the cell of the specified column's name.

### Methods

(continued from last page)

---

## getItems

```
public GXITableCell\[\] getItems()
```

**Returns:**

the row's cells as an array.

---

## getItemsContents

```
public String[] getItemsContents()
```

**Returns:**

the contents of the row's cells.

---

## getItem

```
public GXITableCell getItem(int columnIndex)
```

**Parameters:**

columnIndex - should range between 0 to columnCount - 1.

**Returns:**

the cell at the specified column index.

---

## getItem

```
public GXITableCell getItem(String columnName)
```

**Parameters:**

columnName - column's name.

**Returns:**

the cell of the specified column name.

---

## setItem

```
public void setItem(int columnIndex,  
    GXITableCell item)
```

Sets a table cell at specified column index, and replaces the previous cell (if such existed).

**Parameters:**

columnIndex - the column's index. Should range between 0 to columnCount - 1.  
item - the new cell.

---

## setItem

```
public void setItem(String columnName,  
    GXITableCell item)
```

(continued from last page)

Sets a table cell of the specified column name, and replaces the previous cell (if such existed).

**Parameters:**

columnName - column's name.  
item - the new cell.

---

## getItemContent

```
public String getItemContent(int columnIndex)
```

**Parameters:**

columnIndex - the column's index. Should range between 0 to columnCount - 1.

**Returns:**

the content of the cell at the specified column's index.

---

## getItemContent

```
public String getItemContent(String columnName)
```

**Parameters:**

columnName - column's name.

**Returns:**

the content of the cell at the specified column's name.

---

## setItemContent

```
public void setItemContent(int columnIndex,  
    String content)
```

Sets the content of the cell in the specified column's index.

**Parameters:**

columnIndex - the column's index. Should range between 0 to columnCount - 1.  
content - the new cell's content.

---

## setItemContent

```
public void setItemContent(String columnName,  
    String content)
```

Sets the content of the cell of the specified column's name.

**Parameters:**

columnName - column's name.  
content - the new cell's content.

---

## getPrimaryKey

```
public String getPrimaryKey()
```

**Returns:**

---

(continued from last page)

the value of the primary key for this row.

**See Also:**

[`GXITable.getPrimaryKey\(\)`](#)

---

## removeItem

```
public void removeItem(int columnIndex)
```

Removes the cell at the specified column index. The cell value becomes null.

**Parameters:**

`columnIndex` - the column's index. Should range between 0 to `columnCount` - 1.

---

## removeItem

```
public void removeItem(String columnName)
```

Removes the cell of the specified column name. The cell value becomes null.

**Parameters:**

`columnName` - column's name.

## com.sabratec.applinx.baseobject.table Interface GXITableRowCollection

All Known Implementing Classes:

[GXITableRowCollection](#)

public interface **GXITableRowCollection**

Table's rows' data.

Represents the table's rows' data. The collection is composed of GXITableRow objects. Can be used to directly access a row or a cell, and to filter rows.

### Method Summary

void	<a href="#">addFilter</a> ( <a href="#">GXIRowsFilter</a> filter) Adds a new filter to the table.
void	<a href="#">addRow</a> ( <a href="#">GXITableRow</a> row) Adds a new row as the last row in the table.
<a href="#">GXITableRowCollection</a>	<a href="#">filterRows</a> () Filters the table's rows according to the filters added to the table.
<a href="#">GXIRowsFilter[]</a>	<a href="#">getFilters</a> ()
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (int rowIndex, int columnIndex)
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (int rowIndex, String columnName)
String	<a href="#">getItemContent</a> (int rowIndex, int columnIndex)
String	<a href="#">getItemContent</a> (int rowIndex, String columnName)
<a href="#">GXITableRow</a>	<a href="#">getRow</a> (int rowIndex)
<a href="#">GXITableRow</a>	<a href="#">getRowByKey</a> (String primaryKey) Finds a row according to the specified primary key.
<a href="#">GXITableRow</a>	<a href="#">getRowByKey</a> (String[] primaryKey) Finds a row with the specified primary key values.
int	<a href="#">getRowCount</a> ()
<a href="#">GXITableRow[]</a>	<a href="#">getRows</a> ()
void	<a href="#">removeFilter</a> ( <a href="#">GXIRowsFilter</a> filter) Removes the specified filter.
void	<a href="#">removeRow</a> ( <a href="#">GXITableRow</a> row) removes the specified row.

void	<a href="#">removeRow</a> (int rowIndex) Removes the row at the specified index.
void	<a href="#">setItemContent</a> (int rowIndex, int columnIndex, String content) Sets the contents of the cell at the specified row / column coordinates.
void	<a href="#">setItemContent</a> (int rowIndex, String columnName, String content) Sets the contents of the cell at the specified row / column coordinates.
void	<a href="#">sort</a> (GXAbstractSorter sorter, GXIComparator comparator) Sorts the row's data using the specified sorter and comparator.

## Methods

### getRows

public [GXTableRow\[\]](#) **getRows**()

**Returns:**  
the rows' data as an array.

### getRowCount

public int **getRowCount**()

**Returns:**  
the number of the table's rows.

### getRow

public [GXTableRow](#) **getRow**(int rowIndex)

**Parameters:**  
rowIndex - the row's index. Should range between 0 to rowCount - 1.

**Returns:**  
the row at the specified index.

**See Also:**  
[getRowCount\(\)](#)

### addRow

public void **addRow**([GXTableRow](#) row)

Adds a new row as the last row in the table.

**Parameters:**  
row - the row to add.

## removeRow

```
public void removeRow(int rowIndex)
```

Removes the row at the specified index.

**Parameters:**

rowIndex - row index to be removed. Should range between 0 to rowCount - 1.

---

## removeRow

```
public void removeRow(GXITableRow row)
```

removes the specified row.

**Parameters:**

row - the row to remove.

---

## addFilter

```
public void addFilter(GXIRowsFilter filter)
```

Adds a new filter to the table.

**Parameters:**

filter - the filter to be added.

**See Also:**

[GXIRowsFilter](#)  
[filterRows\(\)](#)

---

## getFilters

```
public GXIRowsFilter\[\] getFilters()
```

**Returns:**

table's associated filters.

---

## removeFilter

```
public void removeFilter(GXIRowsFilter filter)
```

Removes the specified filter.

**Parameters:**

filter - the filter to remove.

---

## filterRows

```
public GXITableRowCollection filterRows()
```

Filters the table's rows according to the filters added to the table.

**Returns:**

the filtered rows.

---

(continued from last page)

**See Also:**

[GXIRowsFilter.filter\(GXITableRowCollection\)](#)  
[addFilter\(GXIRowsFilter\)](#)

---

## getItem

```
public GXITableCell getItem(int rowIndex,  
                             int columnIndex)
```

**Parameters:**

rowIndex - should range between 0 - rowCount - 1.  
columnIndex - should range between 0 - columnsCount - 1.

**Returns:**

the cell at the specified row / column coordinates.

---

## getItem

```
public GXITableCell getItem(int rowIndex,  
                             String columnName)
```

**Parameters:**

rowIndex - should range between 0 - rowCount - 1.  
columnName - the name of the column.

**Returns:**

the cell at the specified row / column coordinates.

---

## getItemContent

```
public String getItemContent(int rowIndex,  
                             int columnIndex)
```

**Parameters:**

rowIndex - should range between 0 - rowCount - 1.  
columnIndex - should range between 0 - columnsCount - 1.

**Returns:**

the content of the cell at the specified row / column coordinates.

---

## getItemContent

```
public String getItemContent(int rowIndex,  
                             String columnName)
```

**Parameters:**

rowIndex - should range between 0 - rowCount - 1.  
columnName - the name of the column.

**Returns:**

(continued from last page)

the content of the cell at the specified row / column coordinates.

---

## setItemContent

```
public void setItemContent(int rowIndex,  
    int columnIndex,  
    String content)
```

Sets the contents of the cell at the specified row / column coordinates.

### Parameters:

rowIndex - should range between 0 - rowCount - 1.  
columnIndex - should range between 0 - columnsCount - 1.  
content - the cell's new content.

### See Also:

[GXITableCell.setContent\(String\)](#)

---

## setItemContent

```
public void setItemContent(int rowIndex,  
    String columnName,  
    String content)
```

Sets the contents of the cell at the specified row / column coordinates.

### Parameters:

rowIndex - should range between 0 - rowCount - 1.  
columnName - the name of the column.  
content - the cell's new content.

### See Also:

[GXITableCell.setContent\(String\)](#)

---

## getRowByKey

```
public GXITableRow getRowByKey(String[] primaryKey)
```

Finds a row with the specified primary key values. Should be used for tables with a combination of several columns as a primary key.

### Parameters:

primaryKey - the values of the primary key's columns.

### Returns:

the found row. Returns null when a row with the specified values as the primary key was not found.

Example: If the primary key was defined to be the columns "first name" and "last name", the primaryKey parameter should hold the String array "John", "Doe".

### See Also:

[GXITable.getPrimaryKey\(\)](#)

---

## getRowByKey

```
public GXITableRow getRowByKey(String primaryKey)
```

Finds a row according to the specified primary key.

### Parameters:

(continued from last page)

primaryKey - the requested primary key.

**Returns:**

the found row. Null when a row was not found.

**See Also:**

[GXITable.getPrimaryKey\(\)](#)

---

## sort

```
public void sort(GXAbstractSorter sorter,  
                 GXIComparator comparator)
```

Sorts the row's data using the specified sorter and comparator. It is recommended to use the sort methods in GXITable and not this method directly.

**Parameters:**

sorter - the sorter.

comparator - the comparator used for sorting.

**See Also:**

[GXITable.sort\(int\[\], boolean\[\]\)](#)

[GXITable.sort\(int, boolean\)](#)

[GXITable.sort\(String, boolean\)](#)

[GXITable.sort\(String\[\], boolean\[\]\)](#)

## com.sabratec.applinx.baseobject.table Class GXMasterTable

java.lang.Object

```

  |
  +-- com.sabratec.applinx.baseobject.table.GXAbstractTable
      |
      +-- com.sabratec.applinx.baseobject.table.GXMasterTable
  
```

All Implemented Interfaces:

java.io.Externalizable, [GXITable](#)

public class **GXMasterTable**  
extends [GXAbstractTable](#)

This table represents a host table that its width is more than one host screen, (horizontal scrolling), and not all records appear in all of the host screens. In this table, it is necessary to specify the screen name for each table column, and the Master Screen: the screen containing all of the records. In addition, the primary key must appear in every screen.

### Constructor Summary

public	<a href="#">GXMasterTable</a> (String name, String[] screenNames, String masterScreenName)
public	<a href="#">GXMasterTable</a> (String name)

### Method Summary

void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screens)
boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getMasterScreenName</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class [com.sabratec.applinx.baseobject.table.GXAbstractTable](#)

[addColumn](#), [addColumn](#), [addColumns](#), [createTable](#), [createTable](#), [equals](#), [getColumns](#), [getCornersCoordinates](#), [getHeaderCoordinates](#), [getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [readExternal](#), [removeColumn](#), [removeColumn](#), [setCornersCoordinates](#), [setHeaderCoordinates](#), [setName](#), [setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#), [toString](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITable](#)

[addColumn](#), [addColumn](#), [addColumns](#), [createTable](#), [createTable](#), [createTable](#), [getColumns](#), [getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [removeColumn](#), [removeColumn](#), [setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#)

Methods inherited from interface `java.io.Externalizable`

`readExternal`, `writeExternal`

## Constructors

### **GXMasterTable**

```
public GXMasterTable(String name,
                    String[] screenNames,
                    String masterScreenName)
```

### **GXMasterTable**

```
public GXMasterTable(String name)
```

## Methods

### **createTable**

```
public void createTable(GXIScreenCollection screens)
    throws GXColumnNotFoundException
```

### **getMasterScreenName**

```
public String getMasterScreenName()
```

### **equals**

```
public boolean equals(Object obj)
```

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

(continued from last page)

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)
    throws java.io.IOException,
           ClassNotFoundException
```

## com.sabratec.applinx.baseobject.table Class GXMultipleRecordsTable

java.lang.Object

```

+--com.sabratec.applinx.baseobject.table.GXAbstractTable
    +--com.sabratec.applinx.baseobject.table.GXMultipleRecordsTable
  
```

All Implemented Interfaces:

java.io.Externalizable, [GXITable](#)

public class **GXMultipleRecordsTable**  
extends [GXAbstractTable](#)

This table represents a host table with "conditional" application fields, or more than one columns schema: A row with schema 1 (master row) appears, and afterwards all of its "child" rows with schema 2, until another row of schema 1 appears. In this table, the fields from the master row are copied to its child rows.

### Constructor Summary

public	<a href="#">GXMultipleRecordsTable</a> (String name, <a href="#">GXIMasterRowIdentifier</a> idetifier)
--------	--

### Method Summary

void	<a href="#">addColumn</a> ( <a href="#">GXITableColumn</a> column)
void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screens)
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">setShowFirstRowsWithoutMaster</a> (boolean show)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class [com.sabratec.applinx.baseobject.table.GXAbstractTable](#)

[addColumn](#), [addColumn](#), [addColumns](#), [createTable](#), [createTable](#), [createTable](#), [equals](#), [getColumns](#), [getCornersCoordinates](#), [getHeaderCoordinates](#), [getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [readExternal](#), [removeColumn](#), [removeColumn](#), [setCornersCoordinates](#), [setHeaderCoordinates](#), [setName](#), [setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#), [toString](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITable](#)

[addColumn](#), [addColumn](#), [addColumns](#), [createTable](#), [createTable](#), [createTable](#), [getColumns](#), [getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [removeColumn](#), [removeColumn](#), [setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#)

**Methods inherited from interface** `java.io.Externalizable``readExternal, writeExternal`

## Constructors

### **GXMultipleRecordsTable**

```
public GXMultipleRecordsTable(String name,  
                             GXIMasterRowIdentifier idetifier)
```

## Methods

### **addColumn**

```
public void addColumn(GXITableColumn column)
```

### **setShowFirstRowsWithoutMaster**

```
public void setShowFirstRowsWithoutMaster(boolean show)
```

### **createTable**

```
public void createTable(GXIScreenCollection screens)  
    throws GXColumnNotFoundException
```

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

### **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

## com.sabratec.applinx.baseobject.table Class GXPartialRowsTable

java.lang.Object

```

+--com.sabratec.applinx.baseobject.table.GXAbstractTable
    |
    +--com.sabratec.applinx.baseobject.table.GXPartialRowsTable
  
```

All Implemented Interfaces:

java.io.Externalizable, [GXITable](#)

public class **GXPartialRowsTable**  
extends [GXAbstractTable](#)

This table represents a host table that its width is more than one host screen, (horizontal scrolling), and not all records appear in all of the host screens. In this table, it is necessary to specify the screen name for each table column, and the Master Screen: the screen containing all of the records. In addition, the primary key must appear in every screen.

### Constructor Summary

public	<a href="#">GXPartialRowsTable</a> (String name, String[] screenNames, String masterScreenName)
--------	---

### Method Summary

void	<a href="#">addColumn</a> ( <a href="#">GXITableColumn</a> column)
void	<a href="#">createTable</a> ( <a href="#">GXIScreenCollection</a> screens)
String	<a href="#">getMasterScreenName</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class [com.sabratec.applinx.baseobject.table.GXAbstractTable](#)

[addColumn](#), [addColumn](#), [addColumnns](#), [createTable](#), [createTable](#), [equals](#), [getColumns](#), [getCornersCoordinates](#), [getHeaderCoordinates](#), [getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [readExternal](#), [removeColumn](#), [removeColumn](#), [setCornersCoordinates](#), [setHeaderCoordinates](#), [setName](#), [setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#), [toString](#), [writeExternal](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITable](#)

[addColumn](#), [addColumn](#), [addColumnns](#), [createTable](#), [createTable](#), [createTable](#), [getColumns](#), [getName](#), [getNewRow](#), [getPrimaryKey](#), [getRows](#), [removeColumn](#), [removeColumn](#), [setPrimaryKey](#), [setPrimaryKey](#), [sort](#), [sort](#), [sort](#), [sort](#)

---

**Methods inherited from interface** `java.io.Externalizable``readExternal, writeExternal`

---

---

## Constructors

### **GXPartialRowsTable**

```
public GXPartialRowsTable(String name,  
                          String[] screenNames,  
                          String masterScreenName)
```

---

## Methods

### **createTable**

```
public void createTable(GXIScreenCollection screens)  
    throws GXColumnNotFoundException
```

---

### **addColumn**

```
public void addColumn(GXITableColumn column)
```

---

### **getMasterScreenName**

```
public String getMasterScreenName()
```

---

### **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

### **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

# com.sabratec.applinx.baseobject.table

## Class GXScreenNameUndefinedException



public class **GXScreenNameUndefinedException**  
extends GXGeneralRuntimeException

Constructor Summary	
public	<a href="#">GXScreenNameUndefinedException</a> (String columnName)

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

## Constructors

### GXScreenNameUndefinedException

public **GXScreenNameUndefinedException**(String columnName)

# com.sabratec.applinx.baseobject.table

## Class GXTableCell

java.lang.Object  
└─com.sabratec.applinx.baseobject.table.GXTableCell

All Implemented Interfaces:  
java.io.Serializable, [GXITableCell](#)

Direct Known Subclasses:  
[GXVariableTableCell](#), [GXFieldTableCell](#)

public class **GXTableCell**  
extends Object  
implements [GXITableCell](#), java.io.Serializable

### Constructor Summary

public	<a href="#">GXTableCell</a> ( <a href="#">GXICellable</a> cellable)
--------	---

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getContent</a> ()
void	<a href="#">setContent</a> (String content)

### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITableCell](#)

[getContent](#), [setContent](#)

## Constructors

**GXTableCell**  
public **GXTableCell**([GXICellable](#) cellable)

## Methods

(continued from last page)

## **getContent**

```
public String getContent()
```

---

## **setContent**

```
public void setContent(String content)
```

---

## **equals**

```
public boolean equals(Object obj)
```

## com.sabratec.applinx.baseobject.table Class GXTableColumn

java.lang.Object

└─com.sabratec.applinx.baseobject.table.GXTableColumn

### All Implemented Interfaces:

java.io.Serializable, [GXITableColumn](#)

public class **GXTableColumn**  
 extends Object  
 implements [GXITableColumn](#), java.io.Serializable

Fields inherited from interface [com.sabratec.applinx.baseobject.table.GXITableColumn](#)

[GX\\_TABLE\\_COLUMN\\_DATA\\_TYPE\\_ALPHANUMERIC](#), [GX\\_TABLE\\_COLUMN\\_DATA\\_TYPE\\_NUMERIC](#),  
[GX\\_TABLE\\_COLUMN\\_TYPE\\_APPFIELD](#), [GX\\_TABLE\\_COLUMN\\_TYPE\\_OTHER](#), [GX\\_TABLE\\_COLUMN\\_TYPE\\_VARIABLE](#)

## Constructor Summary

public	<a href="#">GXTableColumn</a> (String name, String appFieldName, String[] screenNames, String caption, int dataType)
public	<a href="#">GXTableColumn</a> (String name)
public	<a href="#">GXTableColumn</a> (String name, String[] screenNames)
public	<a href="#">GXTableColumn</a> (String name, String caption, int dataType)
public	<a href="#">GXTableColumn</a> (String name, String variableName, String caption, int dataType)

## Method Summary

boolean	<a href="#">equals</a> (Object obj)
String	<a href="#">getAppFieldFullName</a> ()
String	<a href="#">getAppFieldName</a> ()
String	<a href="#">getCaption</a> ()
int	<a href="#">getCopyCount</a> ()
int	<a href="#">getDataType</a> ()
String	<a href="#">getName</a> ()

String[]	<a href="#"><u>getScreenNames()</u></a>
int	<a href="#"><u>getType()</u></a>
String	<a href="#"><u>getVariableName()</u></a>
boolean	<a href="#"><u>isMasterRow()</u></a>
boolean	<a href="#"><u>isToCopy()</u></a>
void	<a href="#"><u>setAppFieldFullName()</u></a> (String appFieldName)
void	<a href="#"><u>setAppFieldName()</u></a> (String appFieldName)
void	<a href="#"><u>setCaption()</u></a> (String caption)
void	<a href="#"><u>setCopyCount()</u></a> (int copyCount)
void	<a href="#"><u>setDataType()</u></a> (int dataType)
void	<a href="#"><u>setIsMasterRow()</u></a> (boolean isMasterRow)
void	<a href="#"><u>setIsToCopy()</u></a> (boolean isToCopy)
void	<a href="#"><u>setName()</u></a> (String name)
void	<a href="#"><u>setScreenNames()</u></a> (String[] names)
void	<a href="#"><u>setVariableName()</u></a> (String name)
String	<a href="#"><u>toString()</u></a>

**Methods inherited from class java.lang.Object**

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITableColumn](#)**

[getAppFieldFullName](#), [getAppFieldName](#), [getCaption](#), [getCopyCount](#), [getDataType](#), [getName](#), [getScreenNames](#), [getType](#), [getVariableName](#), [isMasterRow](#), [isToCopy](#), [setAppFieldName](#), [setCaption](#), [setCopyCount](#), [setDataType](#), [setIsMasterRow](#), [setIsToCopy](#), [setName](#), [setScreenNames](#), [setVariableName](#)

## Constructors

(continued from last page)

---

## GXTableColumn

```
public GXTableColumn(String name,  
                      String appFieldName,  
                      String[] screenNames,  
                      String caption,  
                      int dataType)
```

---

## GXTableColumn

```
public GXTableColumn(String name)
```

---

## GXTableColumn

```
public GXTableColumn(String name,  
                      String[] screenNames)
```

---

## GXTableColumn

```
public GXTableColumn(String name,  
                      String caption,  
                      int dataType)
```

---

## GXTableColumn

```
public GXTableColumn(String name,  
                      String variableName,  
                      String caption,  
                      int dataType)
```

## Methods

### getDataType

```
public int getDataType()
```

---

### setDataType

```
public void setDataType(int dataType)
```

---

### getName

```
public String getName()
```

---

## setName

```
public void setName(String name)
```

---

## getAppFieldName

```
public String getAppFieldName()
```

---

## setAppFieldName

```
public void setAppFieldName(String appFieldName)
```

---

## getAppFieldFullName

```
public String getAppFieldFullName()
```

---

## setAppFieldFullName

```
public void setAppFieldFullName(String appFieldName)
```

---

## getVariableName

```
public String getVariableName()
```

---

## setVariableName

```
public void setVariableName(String name)
```

---

## getCaption

```
public String getCaption()
```

---

## setCaption

```
public void setCaption(String caption)
```

---

## getScreenNames

```
public String[] getScreenNames()
```

---

(continued from last page)

---

## setScreenNames

```
public void setScreenNames(String[] names)
```

---

---

## setIsMasterRow

```
public void setIsMasterRow(boolean isMasterRow)
```

---

---

## isMasterRow

```
public boolean isMasterRow()
```

---

---

## setIsToCopy

```
public void setIsToCopy(boolean isToCopy)
```

---

---

## isToCopy

```
public boolean isToCopy()
```

---

---

## setCopyCount

```
public void setCopyCount(int copyCount)
```

---

---

## getCopyCount

```
public int getCopyCount()
```

---

---

## getType

```
public int getType()
```

---

---

## equals

```
public boolean equals(Object obj)
```

---

(continued from last page)

## **toString**

```
public String toString()
```

com.sabratec.applinx.baseobject.table  
Class GXTableColumnCollection

java.lang.Object  
└─com.sabratec.applinx.baseobject.table.GXTableColumnCollection

All Implemented Interfaces:  
java.io.Externalizable, [GXITableColumnCollection](#)

public class **GXTableColumnCollection**  
extends Object  
implements [GXITableColumnCollection](#), java.io.Externalizable

Constructor Summary	
public	<a href="#">GXTableColumnCollection</a> ()

Method Summary	
void	<a href="#">addColumn</a> ( <a href="#">GXITableColumn</a> column)
boolean	<a href="#">equals</a> (Object obj)
<a href="#">GXITableColumn</a>	<a href="#">getColumn</a> (int columnIndex)
<a href="#">GXITableColumn</a>	<a href="#">getColumn</a> (String columnName)
int	<a href="#">getColumnCount</a> ()
<a href="#">GXITableColumn[]</a>	<a href="#">getColumns</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">removeColumn</a> (int columnIndex)
void	<a href="#">removeColumn</a> (String columnName)
String	<a href="#">toString</a> ()
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	
Methods inherited from interface <a href="#">com.sabratec.applinx.baseobject.table.GXITableColumnCollection</a>	

---

```
addColumn, getColumn, getColumn, getColumnCount, getColumns, removeColumn,  
removeColumn
```

---

Methods inherited from interface `java.io.Externalizable`

```
readExternal, writeExternal
```

---

## Constructors

### **GXTableColumnCollection**

```
public GXTableColumnCollection()
```

## Methods

### **getColumns**

```
public GXTableColumn[] getColumns()
```

---

### **getColumnCount**

```
public int getColumnCount()
```

---

### **getColumn**

```
public GXTableColumn getColumn(int columnIndex)
```

---

### **getColumn**

```
public GXTableColumn getColumn(String columnName)
```

---

### **addColumn**

```
public void addColumn(GXTableColumn column)
```

---

### **removeColumn**

```
public void removeColumn(int columnIndex)
```

---

(continued from last page)

---

## removeColumn

```
public void removeColumn(String columnName)
```

---

## equals

```
public boolean equals(Object obj)
```

---

## writeExternal

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

## readExternal

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

## toString

```
public String toString()
```

---

## com.sabratec.applinx.baseobject.table Class GXTableRow

java.lang.Object

└-com.sabratec.applinx.baseobject.table.GXTableRow

All Implemented Interfaces:

java.io.Externalizable, [GXTableRow](#)

public class **GXTableRow**  
 extends Object  
 implements [GXTableRow](#), java.io.Externalizable

### Constructor Summary

public	<a href="#">GXTableRow()</a>
--------	------------------------------

### Method Summary

boolean	<a href="#">equals</a> (Object obj)
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (int columnIndex)
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (String columnName)
String	<a href="#">getItemContent</a> (int columnIndex)
String	<a href="#">getItemContent</a> (String columnName)
<a href="#">GXITableCell[]</a>	<a href="#">getItems</a> ()
String[]	<a href="#">getItemsContents</a> ()
String	<a href="#">getPrimaryKey</a> ()
void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">removeItem</a> (int columnIndex)
void	<a href="#">removeItem</a> (String columnName)
void	<a href="#">setItem</a> (int columnIndex, <a href="#">GXITableCell</a> item)
void	<a href="#">setItem</a> (String columnName, <a href="#">GXITableCell</a> item)
void	<a href="#">setItemContent</a> (int columnIndex, String content)

void	<a href="#">setItemContent</a> (String columnName, String content)
void	<a href="#">setPrimaryKey</a> (String primaryKey)
String	<a href="#">toString</a> ()
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXTableRow](#)

[getItem](#), [getItem](#), [getItemContent](#), [getItemContent](#), [getItems](#), [getItemsContents](#), [getPrimaryKey](#), [removeItem](#), [removeItem](#), [setItem](#), [setItem](#), [setItemContent](#), [setItemContent](#)

#### Methods inherited from interface java.io.Externalizable

`readExternal`, `writeExternal`

## Constructors

### GXTableRow

```
public GXTableRow()
```

## Methods

### getItems

```
public GXTableCell[] getItems()
```

### getItemsContents

```
public String[] getItemsContents()
```

### getItem

```
public GXTableCell getItem(int columnIndex)
```

### getItem

```
public GXTableCell getItem(String columnName)
```

(continued from last page)

---

## setItem

```
public void setItem(int columnIndex,  
    GXTableCell item)
```

---

## setItem

```
public void setItem(String columnName,  
    GXTableCell item)
```

---

## getItemContent

```
public String getItemContent(int columnIndex)
```

---

## getItemContent

```
public String getItemContent(String columnName)
```

---

## setItemContent

```
public void setItemContent(int columnIndex,  
    String content)
```

---

## setItemContent

```
public void setItemContent(String columnName,  
    String content)
```

---

## getPrimaryKey

```
public String getPrimaryKey()
```

---

## setPrimaryKey

```
public void setPrimaryKey(String primaryKey)
```

---

## removeItem

```
public void removeItem(int columnIndex)
```

(continued from last page)

---

## **removeItem**

```
public void removeItem(String columnName)
```

---

## **equals**

```
public boolean equals(Object obj)
```

---

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)  
    throws java.io.IOException,  
           ClassNotFoundException
```

---

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)  
    throws java.io.IOException
```

---

## **toString**

```
public String toString()
```

## com.sabratec.applinx.baseobject.table Class GXTableRowCollection

java.lang.Object

└-com.sabratec.applinx.baseobject.table.GXTableRowCollection

All Implemented Interfaces:

java.io.Externalizable, [GXITableRowCollection](#)

public class **GXTableRowCollection**  
 extends Object  
 implements [GXITableRowCollection](#), java.io.Externalizable

### Constructor Summary

public	<a href="#">GXTableRowCollection()</a>
--------	--

### Method Summary

void	<a href="#">addFilter</a> ( <a href="#">GXIRowsFilter</a> filter)
void	<a href="#">addRow</a> ( <a href="#">GXITableRow</a> row)
boolean	<a href="#">equals</a> (Object obj)
<a href="#">GXITableRowCollection</a>	<a href="#">filterRows</a> ()
<a href="#">GXIRowsFilter[]</a>	<a href="#">getFilters</a> ()
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (int rowIndex, int columnIndex)
<a href="#">GXITableCell</a>	<a href="#">getItem</a> (int rowIndex, String columnName)
String	<a href="#">getItemContent</a> (int rowIndex, int columnIndex)
String	<a href="#">getItemContent</a> (int rowIndex, String columnName)
<a href="#">GXITableRow</a>	<a href="#">getRow</a> (int rowIndex)
<a href="#">GXITableRow</a>	<a href="#">getRowByKey</a> (String primaryKey)
<a href="#">GXITableRow</a>	<a href="#">getRowByKey</a> (String[] primaryKey)
int	<a href="#">getRowCount</a> ()
<a href="#">GXITableRow[]</a>	<a href="#">getRows</a> ()

void	<a href="#">readExternal</a> (java.io.ObjectInput in)
void	<a href="#">removeFilter</a> ( <a href="#">GXIRowsFilter</a> filter)
void	<a href="#">removeRow</a> ( <a href="#">GXTableRow</a> row)
void	<a href="#">removeRow</a> (int rowIndex)
void	<a href="#">setItemContent</a> (int rowIndex, int columnIndex, String content)
void	<a href="#">setItemContent</a> (int rowIndex, String columnName, String content)
void	<a href="#">sort</a> (GXAbstractSorter sorter, GXIComparator comparator)
String	<a href="#">toString</a> ()
void	<a href="#">writeExternal</a> (java.io.ObjectOutput out)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXTableRowCollection](#)

[addFilter](#), [addRow](#), [filterRows](#), [getFilters](#), [getItem](#), [getItem](#), [getItemContent](#), [getItemContent](#), [getRow](#), [getRowByKey](#), [getRowByKey](#), [getRowCount](#), [getRows](#), [removeFilter](#), [removeRow](#), [removeRow](#), [setItemContent](#), [setItemContent](#), [sort](#)

#### Methods inherited from interface java.io.Externalizable

[readExternal](#), [writeExternal](#)

## Constructors

### **GXTableRowCollection**

```
public GXTableRowCollection()
```

## Methods

### **getRows**

```
public GXTableRow[] getRows()
```

### **getRowCount**

```
public int getRowCount()
```

(continued from last page)

---

## getRow

```
public GXTableRow getRow(int rowIndex)
```

---

---

## addRow

```
public void addRow(GXTableRow row)
```

---

---

## removeRow

```
public void removeRow(int rowIndex)
```

---

---

## removeRow

```
public void removeRow(GXTableRow row)
```

---

---

## addFilter

```
public void addFilter(GXIRowsFilter filter)
```

---

---

## getFilters

```
public GXIRowsFilter\[\] getFilters()
```

---

---

## removeFilter

```
public void removeFilter(GXIRowsFilter filter)
```

---

---

## filterRows

```
public GXTableRowCollection filterRows()
```

---

---

## getItem

```
public GXTableCell getItem(int rowIndex,  
                           int columnIndex)
```

---

(continued from last page)

---

## getItem

```
public GXTableCell getItem(int rowIndex,  
    String columnName)
```

---

## getItemContent

```
public String getItemContent(int rowIndex,  
    int columnIndex)
```

---

## getItemContent

```
public String getItemContent(int rowIndex,  
    String columnName)
```

---

## setItemContent

```
public void setItemContent(int rowIndex,  
    int columnIndex,  
    String content)
```

---

## setItemContent

```
public void setItemContent(int rowIndex,  
    String columnName,  
    String content)
```

---

## getRowByKey

```
public GXTableRow getRowByKey(String[] primaryKey)
```

---

## getRowByKey

```
public GXTableRow getRowByKey(String primaryKey)
```

---

## sort

```
public void sort(GXAbstractSorter sorter,  
    GXIComparator comparator)
```

---

## equals

```
public boolean equals(Object obj)
```

(continued from last page)

---

## **writeExternal**

```
public void writeExternal(java.io.ObjectOutput out)
    throws java.io.IOException
```

---

## **readExternal**

```
public void readExternal(java.io.ObjectInput in)
    throws java.io.IOException,
           ClassNotFoundException
```

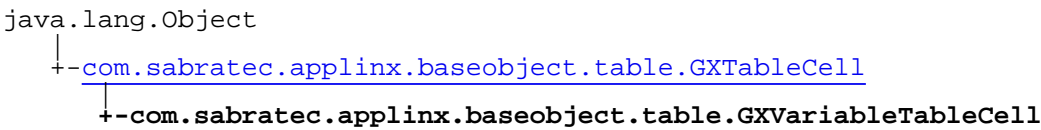
---

## **toString**

```
public String toString()
```

# com.sabratec.applinx.baseobject.table

## Class GXVariableTableCell



All Implemented Interfaces:  
java.io.Serializable, [GXITableCell](#)

public class **GXVariableTableCell**  
extends [GXTableCell](#)

Variable-based cell.  
  
A table cell based on a path variable array.

### Constructor Summary

public	<a href="#">GXVariableTableCell</a> (String content)
public	<a href="#">GXVariableTableCell</a> ( <a href="#">GXICellable</a> cellable)

### Method Summary

boolean	<a href="#">isProtected</a> ()
---------	--------------------------------

Methods inherited from class [com.sabratec.applinx.baseobject.table.GXTableCell](#)  
  
[equals](#), [getContent](#), [setContent](#)

Methods inherited from class java.lang.Object  
  
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXITableCell](#)  
  
[getContent](#), [setContent](#)

### Constructors

#### GXVariableTableCell

public **GXVariableTableCell**(String content)

(continued from last page)

## **GXVariableTableCell**

```
public GXVariableTableCell(GXICellable cellable)
```

### **Methods**

#### **isProtected**

```
public boolean isProtected()
```

---

## Package

# **com.sabratec.applinx.baseobject.table.filter**

Contains classes and interfaces for filtering table's data.

## com.sabratec.applinx.baseobject.table.filter Class GXAddFieldToPreviousRowFilter

java.lang.Object

└─com.sabratec.applinx.baseobject.table.filter.GXAddFieldToPreviousRowFilter

All Implemented Interfaces:

[GXIRowsFilter](#)

public class **GXAddFieldToPreviousRowFilter**  
extends Object  
implements [GXIRowsFilter](#)

This filter is used for horizontal scrolling of the same screen id, but with different fields (all mapped in the same screen). For each row, the filter checks if fields (identified by GXIColumnIdentifier) should be copied to a previous row. If so, it copies them and doesn't add the current row.

### Constructor Summary

public	<a href="#">GXAddFieldToPreviousRowFilter</a> ( <a href="#">GXIColumnIdentifier[]</a> fieldsToCopy)
--------	---

### Method Summary

<a href="#">GXITableRowCollection</a>	<a href="#">filter</a> ( <a href="#">GXITableRowCollection</a> rows)
boolean	<a href="#">isPrimaryKeyValid</a> (String primaryKey, <a href="#">GXITableRowCollection</a> rowsSoFar)
boolean	<a href="#">isRowValid</a> ( <a href="#">GXITableRow</a> row, <a href="#">GXITableRowCollection</a> rowsSoFar)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXIRowsFilter](#)

[filter](#), [isPrimaryKeyValid](#), [isRowValid](#)

### Constructors

#### GXAddFieldToPreviousRowFilter

public **GXAddFieldToPreviousRowFilter**([GXIColumnIdentifier\[\]](#) fieldsToCopy)

### Methods

(continued from last page)

## **filter**

```
public GXITableRowCollection filter(GXITableRowCollection rows)
```

---

## **isRowValid**

```
public boolean isRowValid(GXITableRow row,  
    GXITableRowCollection rowsSoFar)
```

---

## **isPrimaryKeyValid**

```
public boolean isPrimaryKeyValid(String primaryKey,  
    GXITableRowCollection rowsSoFar)
```

# com.sabratec.applinx.baseobject.table.filter

## Class GXNonDuplicateRowsFilter

java.lang.Object

└─com.sabratec.applinx.baseobject.table.filter.GXNonDuplicateRowsFilter

All Implemented Interfaces:

[GXIRowsFilter](#)

public class **GXNonDuplicateRowsFilter**  
extends Object  
implements [GXIRowsFilter](#)

Filters duplicate rows according to the primary key.

Constructor Summary	
public	<a href="#">GXNonDuplicateRowsFilter()</a>

Method Summary	
<a href="#">GXITableRowCollection</a>	<a href="#">filter</a> ( <a href="#">GXITableRowCollection</a> rows)
boolean	<a href="#">isPrimaryKeyValid</a> (String primaryKey, <a href="#">GXITableRowCollection</a> rowsSoFar)
boolean	<a href="#">isRowValid</a> ( <a href="#">GXITableRow</a> row, <a href="#">GXITableRowCollection</a> rowsSoFar)

Methods inherited from class java.lang.Object
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>

Methods inherited from interface <a href="#">com.sabratec.applinx.baseobject.table.GXIRowsFilter</a>
<a href="#">filter</a> , <a href="#">isPrimaryKeyValid</a> , <a href="#">isRowValid</a>

## Constructors

**GXNonDuplicateRowsFilter**

public **GXNonDuplicateRowsFilter**()

## Methods

**filter**

public [GXITableRowCollection](#) **filter**([GXITableRowCollection](#) rows)

(continued from last page)

---

## isRowValid

```
public boolean isRowValid(GXITableRow row,  
    GXITableRowCollection rowsSoFar)
```

---

## isPrimaryKeyValid

```
public boolean isPrimaryKeyValid(String primaryKey,  
    GXITableRowCollection rowsSoFar)
```

## com.sabratec.applinx.baseobject.table.filter

### Class GXNonEmptyRowsFilter

java.lang.Object

└-com.sabratec.applinx.baseobject.table.filter.GXNonEmptyRowsFilter

All Implemented Interfaces:

[GXIRowsFilter](#)

public class **GXNonEmptyRowsFilter**  
 extends Object  
 implements [GXIRowsFilter](#)

Example for filtering rows - filters rows with an empty column. If no column specified, filters rows with an empty primary key.

#### Constructor Summary

public	<a href="#">GXNonEmptyRowsFilter</a> (String[] columnNames)
public	<a href="#">GXNonEmptyRowsFilter</a> ( <a href="#">GXITableColumn[]</a> columns)
public	<a href="#">GXNonEmptyRowsFilter</a> ()

#### Method Summary

<a href="#">GXITableRowCollection</a>	<a href="#">filter</a> ( <a href="#">GXITableRowCollection</a> rows)
boolean	<a href="#">isPrimaryKeyValid</a> (String primaryKey, <a href="#">GXITableRowCollection</a> rowsSoFar)
boolean	<a href="#">isRowValid</a> ( <a href="#">GXITableRow</a> row, <a href="#">GXITableRowCollection</a> rowsSoFar)
String	<a href="#">toString</a> ()

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.baseobject.table.GXIRowsFilter](#)

[filter](#), [isPrimaryKeyValid](#), [isRowValid](#)

## Constructors

### GXNonEmptyRowsFilter

public **GXNonEmptyRowsFilter**(String[] columnNames)

---

## GXNonEmptyRowsFilter

```
public GXNonEmptyRowsFilter(GXITableColumn\[\] columns)
```

---

## GXNonEmptyRowsFilter

```
public GXNonEmptyRowsFilter()
```

## Methods

### filter

```
public GXITableRowCollection filter(GXITableRowCollection rows)
```

---

### isRowValid

```
public boolean isRowValid(GXITableRow row,  
    GXITableRowCollection rowsSoFar)
```

---

### isPrimaryKeyValid

```
public boolean isPrimaryKeyValid(String primaryKey,  
    GXITableRowCollection rowsSoFar)
```

---

### toString

```
public String toString()
```

---

**Package**

**com.sabratec.applinx.common**

## com.sabratec.applinx.common

### Class GXResponseCodes

java.lang.Object

└-com.sabratec.applinx.common.GXResponseCodes

public class **GXResponseCodes**  
extends Object

#### Field Summary

public static final	<a href="#"><u>APPLICATION_RELOAD_FAILED</u></a> Value: <b>7008</b>
public static final	<a href="#"><u>DESTINATION_FOLDER_IS_SAME</u></a> Value: <b>7011</b>
public static final	<a href="#"><u>DESTINATION_FOLDER_IS_SUBFOLDER</u></a> Value: <b>7012</b>
public static final	<a href="#"><u>ERROR_APPLICATION_NOT_DEFINED</u></a> Value: <b>5007</b>
public static final	<a href="#"><u>ERROR_CANNOT_CONNECT_HOST</u></a> Value: <b>5002</b>
public static final	<a href="#"><u>ERROR_CANNOT_CONNECT_SERVER</u></a> Value: <b>5004</b>
public static final	<a href="#"><u>ERROR_CANT_DELETE_ASSIGNED_EXCEPTION_PATH</u></a> Value: <b>5103</b>
public static final	<a href="#"><u>ERROR_CANT_DELETE_REFERENCED_ENTITY</u></a> Value: <b>5101</b>
public static final	<a href="#"><u>ERROR_CENTRASITE_REGISTRATION</u></a> Value: <b>10005</b>
public static final	<a href="#"><u>ERROR_CHANGING_WEB_SERVICE_TARGET_SERVER</u></a> Value: <b>8500</b>
public static final	<a href="#"><u>ERROR_CONCURRENT_NUMBER_OF_USERS_EXCEEDS_LICENSE_AGREEMENT</u></a> Value: <b>5060</b>
public static final	<a href="#"><u>ERROR_CONFIGURATION_SAVE_FAILED</u></a> Value: <b>6001</b>

public static final	<a href="#"><u>ERROR_CONNECTING_DEVICE_NAME</u></a> Value: <b>5031</b>
public static final	<a href="#"><u>ERROR_CONNECTION_INFO</u></a> Value: <b>5011</b>
public static final	<a href="#"><u>ERROR_CREATING_ENTITIES_FROM_WSDL</u></a> Value: <b>8001</b>
public static final	<a href="#"><u>ERROR_CREATING_TABLE</u></a> Value: <b>5080</b>
public static final	<a href="#"><u>ERROR_CURSOR_POSITION_NOT_SET</u></a> Value: <b>5030</b>
public static final	<a href="#"><u>ERROR_DB_GENERAL</u></a> Value: <b>5102</b>
public static final	<a href="#"><u>ERROR_DEPLOYING_WEB_SERVICE</u></a> Value: <b>10001</b>
public static final	<a href="#"><u>ERROR_DEPLOYMENT_GENERAL</u></a> Value: <b>10000</b>
public static final	<a href="#"><u>ERROR_DEPLOYMENT_HANDLER_NOT_AVAILABLE</u></a> Value: <b>10006</b>
public static final	<a href="#"><u>ERROR_DISPLAY_SESSION_REQUEST_FOR_PRINTER_SESSION</u></a> Value: <b>5070</b>
public static final	<a href="#"><u>ERROR_EXECUTING_MAP_BAD_DESTINATION</u></a> Value: <b>5035</b>
public static final	<a href="#"><u>ERROR_EXECUTING_MAP_NO_MAP_IN_APPL</u></a> Value: <b>5034</b>
public static final	<a href="#"><u>ERROR_EXECUTING_MAP_NO_PATH_BETWEEN_SCREEN</u></a> Value: <b>5033</b>
public static final	<a href="#"><u>ERROR_EXECUTING_MAP_SCREEN_NOT_IN_MAP</u></a> Value: <b>5032</b>
public static final	<a href="#"><u>ERROR_EXECUTING_PATH_FROMSCREENID</u></a> Value: <b>5015</b>
public static final	<a href="#"><u>ERROR_EXECUTING_PATH_GENERAL</u></a> Value: <b>5017</b>

public static final	<a href="#"><u>ERROR_EXECUTING_PATH_RESOLVE_ACTION</u></a> Value: <b>5029</b>
public static final	<a href="#"><u>ERROR_EXECUTING_PATH_TOSCREENID</u></a> Value: <b>5016</b>
public static final	<a href="#"><u>ERROR_EXECUTING_PATH_TYPE_EXCEPTION</u></a> Value: <b>5028</b>
public static final	<a href="#"><u>ERROR_EXECUTING_PROCEDURE_GENERAL</u></a> Value: <b>9000</b>
public static final	<a href="#"><u>ERROR_EXPORTING_WEB_SERVICE</u></a> Value: <b>10004</b>
public static final	<a href="#"><u>ERROR_FIELD_CONTENT_NOT_SET</u></a> Value: <b>5036</b>
public static final	<a href="#"><u>ERROR_GENERAL</u></a> Value: <b>5000</b>
public static final	<a href="#"><u>ERROR_HOST_EXISTS_IN_APPLICATION</u></a> Value: <b>6000</b>
public static final	<a href="#"><u>ERROR_HOST_SESSION_DISCONNECTED</u></a> Value: <b>5018</b>
public static final	<a href="#"><u>ERROR_IMPORT_FILE_REPOSITORY</u></a> Value: <b>5038</b>
public static final	<a href="#"><u>ERROR_INCOMPATIBLE_VERSIONS</u></a> Value: <b>6002</b>
public static final	<a href="#"><u>ERROR_INCOMPATIBLE_WSS_VERSION</u></a> Value: <b>10007</b>
public static final	<a href="#"><u>ERROR_INCORRECT_PASSWORD</u></a> Value: <b>5012</b>
public static final	<a href="#"><u>ERROR_INDEX_OUT_OF_BOUND</u></a> Value: <b>5006</b>
public static final	<a href="#"><u>ERROR_INVALID_LICENSE_COMBINATION</u></a> Value: <b>6113</b>
public static final	<a href="#"><u>ERROR_INVALID_LICENSE_KEY</u></a> Value: <b>6004</b>

public static final	<a href="#"><u>ERROR_LICENSE_FILE_NOT_FOUND</u></a> Value: <b>6005</b>
public static final	<a href="#"><u>ERROR_LICENSE_TERMS_VAIOLATION</u></a> Value: <b>5061</b>
public static final	<a href="#"><u>ERROR_LOGIN_FAILED</u></a> Value: <b>5023</b>
public static final	<a href="#"><u>ERROR_MAP_STEP_ALREADY_EXISTS</u></a> Value: <b>5040</b>
public static final	<a href="#"><u>ERROR_NATURAL_DATA_TRANSFER_FAILURE</u></a> Value: <b>5100</b>
public static final	<a href="#"><u>ERROR_NATURAL_DATA_TRANSFER_INAVLID_STATE</u></a> Value: <b>5101</b>
public static final	<a href="#"><u>ERROR_OVERWRITING</u></a> Value: <b>5014</b>
public static final	<a href="#"><u>ERROR_PATH_NOT_FOUND</u></a> Value: <b>5025</b>
public static final	<a href="#"><u>ERROR_PERMISSION_DENIED</u></a> Value: <b>5024</b>
public static final	<a href="#"><u>ERROR_PREFS_SAVE_FAILED</u></a> Value: <b>5026</b>
public static final	<a href="#"><u>ERROR_PRINTER_CONNECT_ERROR</u></a> Value: <b>5027</b>
public static final	<a href="#"><u>ERROR_PRINTER_DISABLED</u></a> Value: <b>5021</b>
public static final	<a href="#"><u>ERROR_REPLACING_LICENSE</u></a> Value: <b>6006</b>
public static final	<a href="#"><u>ERROR_REPLAY_DEVICE_INCOMPATIBILITY</u></a> Value: <b>5037</b>
public static final	<a href="#"><u>ERROR_REPLAY_FILE</u></a> Value: <b>5010</b>
public static final	<a href="#"><u>ERROR_REPLAY_FILE_NOT_FOUND</u></a> Value: <b>5009</b>

public static final	<a href="#"><u>ERROR_RETRIEVING_WSDL_SERVICES</u></a> Value: <b>8000</b>
public static final	<a href="#"><u>ERROR_RETRIEVING_WSDL_SERVICES_PERMISSION_DENIED</u></a> Value: <b>8002</b>
public static final	<a href="#"><u>ERROR_SCREEN_NOT_SYNC</u></a> Value: <b>5027</b>
public static final	<a href="#"><u>ERROR_SECURITY_SAVE_FAILED</u></a> Value: <b>6003</b>
public static final	<a href="#"><u>ERROR_SERVICE_CONNECTION_NOT_READY</u></a> Value: <b>5055</b>
public static final	<a href="#"><u>ERROR_SERVICE_CONNECTION_TIMEOUT_EXPIRED</u></a> Value: <b>5053</b>
public static final	<a href="#"><u>ERROR_SERVICE_GENERAL</u></a> Value: <b>5050</b>
public static final	<a href="#"><u>ERROR_SERVICE_NO_AVAILABLE_CONNECTION</u></a> Value: <b>5052</b>
public static final	<a href="#"><u>ERROR_SERVICE_NOT_ACTIVE</u></a> Value: <b>5051</b>
public static final	<a href="#"><u>ERROR_SERVICE_NOT_FOUND</u></a> Value: <b>5056</b>
public static final	<a href="#"><u>ERROR_SERVICE_PATH_NOT_FOUND</u></a> Value: <b>5054</b>
public static final	<a href="#"><u>ERROR_SESSION_VARIABLE_DOESNT_EXIST</u></a> Value: <b>5090</b>
public static final	<a href="#"><u>ERROR_SESSION_VARIABLE_TYPE_MISMATCH</u></a> Value: <b>5091</b>
public static final	<a href="#"><u>ERROR_SOCKET</u></a> Error communicating with ApplinX server, may occur as a result of incompatible client and server version, or network malfunctions. Value: <b>5005</b>
public static final	<a href="#"><u>ERROR_SYNCHRONIZING_REPOSITORY</u></a> Value: <b>7019</b>
public static final	<a href="#"><u>ERROR_TIMEOUT_EXPIRED</u></a> Value: <b>5022</b>

public static final	<a href="#"><u>ERROR_UNDEPLOYING_WEB_SERVICE</u></a> Value: <b>10002</b>
public static final	<a href="#"><u>ERROR_UNKNOWN_ACTION</u></a> Value: <b>5200</b>
public static final	<a href="#"><u>ERROR_UNKNOWN_SERVER_ADDRESS</u></a> Value: <b>5003</b>
public static final	<a href="#"><u>ERROR_UPDATING_WEB_SERVICE</u></a> Value: <b>10003</b>
public static final	<a href="#"><u>ERROR_USER_ALREADY_ATTACHED</u></a> Value: <b>5008</b>
public static final	<a href="#"><u>ERROR_USER_NOT_CONNECTED</u></a> Value: <b>5001</b>
public static final	<a href="#"><u>ERROR_WAIT_CONDITION_NOT_MET</u></a> Value: <b>5013</b>
public static final	<a href="#"><u>ERROR_WAIT_CONDITION_NOT_VALID</u></a> Value: <b>5039</b>
public static final	<a href="#"><u>EXPORT_APPLICATION_ENTITIES_FAILED</u></a> Value: <b>7015</b>
public static final	<a href="#"><u>FAULTY_SCREEN_IMAGE</u></a> Value: <b>7018</b>
public static final	<a href="#"><u>FIELD_EXISTS_IN_ACTIONS</u></a> Value: <b>7004</b>
public static final	<a href="#"><u>FILE_DOES_NOT_EXIST</u></a> Value: <b>5019</b>
public static final	<a href="#"><u>IMPORT_APPLICATION_CONFIG_MISSING</u></a> Value: <b>7020</b>
public static final	<a href="#"><u>IMPORT_APPLICATION_COULD_NOT_RESTORE_OVERRIDEN_APP_FROM_BACKUP</u></a> Value: <b>7022</b>
public static final	<a href="#"><u>IMPORT_APPLICATION_ENTITIES_FAILED</u></a> Value: <b>7014</b>
public static final	<a href="#"><u>IMPORT_APPLICATION_FAILED</u></a> Value: <b>7013</b>

public static final	<a href="#"><u>IMPORT_APPLICATION_HOST_CATEGORY_UNSUPPORTED</u></a> Value: <b>7021</b>
public static final	<a href="#"><u>INDEXATION_PROCESS_FAILED</u></a> Value: <b>7009</b>
public static final	<a href="#"><u>MAPPING_EXCEEDS_LINE</u></a> Value: <b>7007</b>
public static final	<a href="#"><u>MAPPING_OVERLAP</u></a> Value: <b>7005</b>
public static final	<a href="#"><u>MAPPING_UTILITY_FAILED</u></a> Value: <b>7010</b>
public static final	<a href="#"><u>MESSAGE_PATH_BY_STEP_COMPLETE</u></a> Value: <b>4001</b>
public static final	<a href="#"><u>MESSAGE_PATH_COMPLETE</u></a> Value: <b>4002</b>
public static final	<a href="#"><u>NO_ERROR</u></a> Value: <b>0</b>
public static final	<a href="#"><u>NO_WARNING</u></a> Value: <b>0</b>
public static final	<a href="#"><u>OBJECT_ALREADY_EXISTS</u></a> Value: <b>7001</b>
public static final	<a href="#"><u>OBJECT_NOT_DELETED</u></a> Value: <b>7003</b>
public static final	<a href="#"><u>OBJECT_NOT_FOUND</u></a> Value: <b>7002</b>
public static final	<a href="#"><u>OBJECT_NOT_UPDATED</u></a> Value: <b>7003</b>
public static final	<a href="#"><u>REPOSITORY_NOT_AVAILABLE</u></a> Value: <b>7000</b>
public static final	<a href="#"><u>REPOSITORY_NOT_CREATED</u></a> Value: <b>7006</b>
public static final	<a href="#"><u>REPOSITORY_NOT_SYNCHRONIZED</u></a> Value: <b>7016</b>

public static final	<a href="#"><u>RESTRICTED_ACCESS_TO_URL</u></a> Value: <b>12001</b>
public static final	<a href="#"><u>SESSION_CANCELED</u></a> Value: <b>11000</b>
public static final	<a href="#"><u>SESSION_CANCELED_BY_ADMIN</u></a> Value: <b>11001</b>
public static final	<a href="#"><u>SESSION_CANCELED_BY_INACTIVE_TIMEOUT</u></a> Value: <b>11002</b>
public static final	<a href="#"><u>UPDATE_NONE_EXISTING_ENTITY</u></a> Value: <b>7017</b>
public static final	<a href="#"><u>WARNING_30_DAYS_EVAL_VERSION</u></a> Value: <b>6100</b>
public static final	<a href="#"><u>WARNING_60_DAYS_EVAL_VERSION</u></a> Value: <b>6101</b>
public static final	<a href="#"><u>WARNING_90_DAYS_EVAL_VERSION</u></a> Value: <b>6102</b>
public static final	<a href="#"><u>WARNING_DURING_CONNECT</u></a> Value: <b>3011</b>
public static final	<a href="#"><u>WARNING_EVALUATION_EXPIRED</u></a> Value: <b>6103</b>
public static final	<a href="#"><u>WARNING_FIELD_CONTENT_ALREADY_EXIST</u></a> Value: <b>3004</b>
public static final	<a href="#"><u>WARNING_FIELD_IS_PROTECTED</u></a> Value: <b>3007</b>
public static final	<a href="#"><u>WARNING_PATH_SCREEN_NOT_ADDED_TO_RESPONSE</u></a> Value: <b>3003</b>
public static final	<a href="#"><u>WARNING_SCREEN_IMAGE_INSERT_FAILED</u></a> Value: <b>3006</b>
public static final	<a href="#"><u>WARNING_WAIT_CONDITION_FAILED</u></a> Value: <b>3001</b>

## Constructor Summary

public	<a href="#"><u>GXResponseCodes</u></a> ( )
--------	--

**Methods inherited from class** `java.lang.Object``equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Fields

### **NO\_WARNING**

```
public static final int NO_WARNING
```

Constant value: **0**

### **WARNING\_WAIT\_CONDITION\_FAILED**

```
public static final int WARNING_WAIT_CONDITION_FAILED
```

Constant value: **3001**

### **WARNING\_PATH\_SCREEN\_NOT\_ADDED\_TO\_RESPONSE**

```
public static final int WARNING_PATH_SCREEN_NOT_ADDED_TO_RESPONSE
```

Constant value: **3003**

### **WARNING\_FIELD\_CONTENT\_ALREADY\_EXIST**

```
public static final int WARNING_FIELD_CONTENT_ALREADY_EXIST
```

Constant value: **3004**

### **WARNING\_SCREEN\_IMAGE\_INSERT\_FAILED**

```
public static final int WARNING_SCREEN_IMAGE_INSERT_FAILED
```

Constant value: **3006**

### **WARNING\_FIELD\_IS\_PROTECTED**

```
public static final int WARNING_FIELD_IS_PROTECTED
```

Constant value: **3007**

### **WARNING\_DURING\_CONNECT**

```
public static final int WARNING_DURING_CONNECT
```

Constant value: **3011**

(continued from last page)

---

## MESSAGE\_PATH\_BY\_STEP\_COMPLETE

```
public static final int MESSAGE_PATH_BY_STEP_COMPLETE
```

Constant value: **4001**

---

## MESSAGE\_PATH\_COMPLETE

```
public static final int MESSAGE_PATH_COMPLETE
```

Constant value: **4002**

---

## NO\_ERROR

```
public static final int NO_ERROR
```

Constant value: **0**

---

## ERROR\_GENERAL

```
public static final int ERROR_GENERAL
```

Constant value: **5000**

---

## ERROR\_USER\_NOT\_CONNECTED

```
public static final int ERROR_USER_NOT_CONNECTED
```

Constant value: **5001**

---

## ERROR\_CANNOT\_CONNECT\_HOST

```
public static final int ERROR_CANNOT_CONNECT_HOST
```

Constant value: **5002**

---

## ERROR\_UNKNOWN\_SERVER\_ADDRESS

```
public static final int ERROR_UNKNOWN_SERVER_ADDRESS
```

Constant value: **5003**

---

## ERROR\_CANNOT\_CONNECT\_SERVER

```
public static final int ERROR_CANNOT_CONNECT_SERVER
```

Constant value: **5004**

---

## ERROR\_SOCKET

```
public static final int ERROR_SOCKET
```

---

(continued from last page)

Error communicating with ApplinX server, may occur as a result of incompatible client and server version, or network mailfunctions.

Constant value: **5005**

---

## **ERROR\_INDEX\_OUT\_OF\_BOUND**

```
public static final int ERROR_INDEX_OUT_OF_BOUND
```

Constant value: **5006**

---

## **ERROR\_APPLICATION\_NOT\_DEFINED**

```
public static final int ERROR_APPLICATION_NOT_DEFINED
```

Constant value: **5007**

---

## **ERROR\_USER\_ALREADY\_ATTACHED**

```
public static final int ERROR_USER_ALREADY_ATTACHED
```

Constant value: **5008**

---

## **ERROR\_REPLAY\_FILE\_NOT\_FOUND**

```
public static final int ERROR_REPLAY_FILE_NOT_FOUND
```

Constant value: **5009**

---

## **ERROR\_REPLAY\_FILE**

```
public static final int ERROR_REPLAY_FILE
```

Constant value: **5010**

---

## **ERROR\_CONNECTION\_INFO**

```
public static final int ERROR_CONNECTION_INFO
```

Constant value: **5011**

---

## **ERROR\_INCORRECT\_PASSWORD**

```
public static final int ERROR_INCORRECT_PASSWORD
```

Constant value: **5012**

---

## **ERROR\_WAIT\_CONDITION\_NOT\_MET**

```
public static final int ERROR_WAIT_CONDITION_NOT_MET
```

Constant value: **5013**

---

---

## **ERROR\_OVERWRITING**

public static final int **ERROR\_OVERWRITING**

Constant value: **5014**

---

## **ERROR\_EXECUTING\_PATH\_FROMSCREENID**

public static final int **ERROR\_EXECUTING\_PATH\_FROMSCREENID**

Constant value: **5015**

---

## **ERROR\_EXECUTING\_PATH\_TOSCREENID**

public static final int **ERROR\_EXECUTING\_PATH\_TOSCREENID**

Constant value: **5016**

---

## **ERROR\_EXECUTING\_PATH\_GENERAL**

public static final int **ERROR\_EXECUTING\_PATH\_GENERAL**

Constant value: **5017**

---

## **ERROR\_HOST\_SESSION\_DISCONNECTED**

public static final int **ERROR\_HOST\_SESSION\_DISCONNECTED**

Constant value: **5018**

---

## **FILE\_DOES\_NOT\_EXIST**

public static final int **FILE\_DOES\_NOT\_EXIST**

Constant value: **5019**

---

## **ERROR\_PRINTER\_DISABLED**

public static final int **ERROR\_PRINTER\_DISABLED**

Constant value: **5021**

---

## **ERROR\_TIMEOUT\_EXPIRED**

public static final int **ERROR\_TIMEOUT\_EXPIRED**

Constant value: **5022**

---

(continued from last page)

---

## ERROR\_LOGIN\_FAILED

```
public static final int ERROR_LOGIN_FAILED
```

Constant value: 5023

---

## ERROR\_PERMISSION\_DENIED

```
public static final int ERROR_PERMISSION_DENIED
```

Constant value: 5024

---

## ERROR\_PATH\_NOT\_FOUND

```
public static final int ERROR_PATH_NOT_FOUND
```

Constant value: 5025

---

## ERROR\_PREFS\_SAVE\_FAILED

```
public static final int ERROR_PREFS_SAVE_FAILED
```

Constant value: 5026

---

## ERROR\_SCREEN\_NOT\_SYNC

```
public static final int ERROR_SCREEN_NOT_SYNC
```

Constant value: 5027

---

## ERROR\_WAIT\_CONDITION\_NOT\_VALID

```
public static final int ERROR_WAIT_CONDITION_NOT_VALID
```

Constant value: 5039

---

## ERROR\_UNKNOWN\_ACTION

```
public static final int ERROR_UNKNOWN_ACTION
```

Constant value: 5200

---

## ERROR\_CANT\_DELETE\_REFERENCED\_ENTITY

```
public static final int ERROR_CANT_DELETE_REFERENCED_ENTITY
```

Constant value: 5101

---

## ERROR\_DB\_GENERAL

```
public static final int ERROR_DB_GENERAL
```

---

(continued from last page)

Constant value: 5102

---

## **ERROR\_CANT\_DELETE\_ASSIGNED\_EXCEPTION\_PATH**

```
public static final int ERROR_CANT_DELETE_ASSIGNED_EXCEPTION_PATH
```

Constant value: 5103

---

## **ERROR\_PRINTER\_CONNECT\_ERROR**

```
public static final int ERROR_PRINTER_CONNECT_ERROR
```

Constant value: 5027

---

## **ERROR\_EXECUTING\_PATH\_TYPE\_EXCEPTION**

```
public static final int ERROR_EXECUTING_PATH_TYPE_EXCEPTION
```

Constant value: 5028

---

## **ERROR\_EXECUTING\_PATH\_RESOLVE\_ACTION**

```
public static final int ERROR_EXECUTING_PATH_RESOLVE_ACTION
```

Constant value: 5029

---

## **ERROR\_CURSOR\_POSITION\_NOT\_SET**

```
public static final int ERROR_CURSOR_POSITION_NOT_SET
```

Constant value: 5030

---

## **ERROR\_FIELD\_CONTENT\_NOT\_SET**

```
public static final int ERROR_FIELD_CONTENT_NOT_SET
```

Constant value: 5036

---

## **ERROR\_REPLAY\_DEVICE\_INCOMPATIBILITY**

```
public static final int ERROR_REPLAY_DEVICE_INCOMPATIBILITY
```

Constant value: 5037

---

## **ERROR\_IMPORT\_FILE\_REPOSITORY**

```
public static final int ERROR_IMPORT_FILE_REPOSITORY
```

Constant value: 5038

---

## **ERROR\_CONNECTING\_DEVICE\_NAME**

```
public static final int ERROR_CONNECTING_DEVICE_NAME
```

Constant value: **5031**

---

## **ERROR\_EXECUTING\_MAP\_SCREEN\_NOT\_IN\_MAP**

```
public static final int ERROR_EXECUTING_MAP_SCREEN_NOT_IN_MAP
```

Constant value: **5032**

---

## **ERROR\_EXECUTING\_MAP\_NO\_PATH\_BETWEEN\_SCREEN**

```
public static final int ERROR_EXECUTING_MAP_NO_PATH_BETWEEN_SCREEN
```

Constant value: **5033**

---

## **ERROR\_EXECUTING\_MAP\_NO\_MAP\_IN\_APPL**

```
public static final int ERROR_EXECUTING_MAP_NO_MAP_IN_APPL
```

Constant value: **5034**

---

## **ERROR\_EXECUTING\_MAP\_BAD\_DESTINATION**

```
public static final int ERROR_EXECUTING_MAP_BAD_DESTINATION
```

Constant value: **5035**

---

## **ERROR\_MAP\_STEP\_ALREADY\_EXISTS**

```
public static final int ERROR_MAP_STEP_ALREADY_EXISTS
```

Constant value: **5040**

---

## **ERROR\_CREATING\_TABLE**

```
public static final int ERROR_CREATING_TABLE
```

Constant value: **5080**

---

## **ERROR\_DISPLAY\_SESSION\_REQUEST\_FOR\_PRINTER\_SESSION**

```
public static final int ERROR_DISPLAY_SESSION_REQUEST_FOR_PRINTER_SESSION
```

Constant value: **5070**

---

(continued from last page)

---

## **ERROR\_CONCURRENT\_NUMBER\_OF\_USERS\_EXCEEDS\_LICENSE\_AGREEMENT**

```
public static final int ERROR_CONCURRENT_NUMBER_OF_USERS_EXCEEDS_LICENSE_AGREEMENT
```

Constant value: 5060

---

## **ERROR\_LICENSE\_TERMS\_VAIOLATION**

```
public static final int ERROR_LICENSE_TERMS_VAIOLATION
```

Constant value: 5061

---

## **ERROR\_SESSION\_VARIABLE\_DOESNT\_EXIST**

```
public static final int ERROR_SESSION_VARIABLE_DOESNT_EXIST
```

Constant value: 5090

---

## **ERROR\_SESSION\_VARIABLE\_TYPE\_MISMATCH**

```
public static final int ERROR_SESSION_VARIABLE_TYPE_MISMATCH
```

Constant value: 5091

---

## **ERROR\_SERVICE\_GENERAL**

```
public static final int ERROR_SERVICE_GENERAL
```

Constant value: 5050

---

## **ERROR\_SERVICE\_NOT\_ACTIVE**

```
public static final int ERROR_SERVICE_NOT_ACTIVE
```

Constant value: 5051

---

## **ERROR\_SERVICE\_NO\_AVAILABLE\_CONNECTION**

```
public static final int ERROR_SERVICE_NO_AVAILABLE_CONNECTION
```

Constant value: 5052

---

## **ERROR\_SERVICE\_CONNECTION\_TIMEOUT\_EXPIRED**

```
public static final int ERROR_SERVICE_CONNECTION_TIMEOUT_EXPIRED
```

Constant value: 5053

---

(continued from last page)

---

## **ERROR\_SERVICE\_PATH\_NOT\_FOUND**

```
public static final int ERROR_SERVICE_PATH_NOT_FOUND
```

Constant value: **5054**

---

## **ERROR\_SERVICE\_CONNECTION\_NOT\_READY**

```
public static final int ERROR_SERVICE_CONNECTION_NOT_READY
```

Constant value: **5055**

---

## **ERROR\_SERVICE\_NOT\_FOUND**

```
public static final int ERROR_SERVICE_NOT_FOUND
```

Constant value: **5056**

---

## **ERROR\_NATURAL\_DATA\_TRANSFER\_FAILURE**

```
public static final int ERROR_NATURAL_DATA_TRANSFER_FAILURE
```

Constant value: **5100**

---

## **ERROR\_NATURAL\_DATA\_TRANSFER\_INAVLID\_STATE**

```
public static final int ERROR_NATURAL_DATA_TRANSFER_INAVLID_STATE
```

Constant value: **5101**

---

## **ERROR\_HOST\_EXISTS\_IN\_APPLICATION**

```
public static final int ERROR_HOST_EXISTS_IN_APPLICATION
```

Constant value: **6000**

---

## **ERROR\_CONFIGURATION\_SAVE\_FAILED**

```
public static final int ERROR_CONFIGURATION_SAVE_FAILED
```

Constant value: **6001**

---

## **ERROR\_INCOMPATIBLE\_VERSIONS**

```
public static final int ERROR_INCOMPATIBLE_VERSIONS
```

Constant value: **6002**

---

## **ERROR\_SECURITY\_SAVE\_FAILED**

```
public static final int ERROR_SECURITY_SAVE_FAILED
```

---

(continued from last page)

---

Constant value: **6003**

---

**ERROR\_INVALID\_LICENSE\_KEY**

```
public static final int ERROR_INVALID_LICENSE_KEY
```

Constant value: **6004**

---

**ERROR\_LICENSE\_FILE\_NOT\_FOUND**

```
public static final int ERROR_LICENSE_FILE_NOT_FOUND
```

Constant value: **6005**

---

**ERROR\_REPLACING\_LICENSE**

```
public static final int ERROR_REPLACING_LICENSE
```

Constant value: **6006**

---

**WARNING\_30\_DAYS\_EVAL\_VERSION**

```
public static final int WARNING_30_DAYS_EVAL_VERSION
```

Constant value: **6100**

---

**WARNING\_60\_DAYS\_EVAL\_VERSION**

```
public static final int WARNING_60_DAYS_EVAL_VERSION
```

Constant value: **6101**

---

**WARNING\_90\_DAYS\_EVAL\_VERSION**

```
public static final int WARNING_90_DAYS_EVAL_VERSION
```

Constant value: **6102**

---

**WARNING\_EVALUATION\_EXPIRED**

```
public static final int WARNING_EVALUATION_EXPIRED
```

Constant value: **6103**

---

**ERROR\_INVALID\_LICENSE\_COMBINATION**

```
public static final int ERROR_INVALID_LICENSE_COMBINATION
```

Constant value: **6113**

---

## REPOSITORY\_NOT\_AVAILABLE

```
public static final int REPOSITORY_NOT_AVAILABLE
```

Constant value: 7000

---

## OBJECT\_ALREADY\_EXISTS

```
public static final int OBJECT_ALREADY_EXISTS
```

Constant value: 7001

---

## OBJECT\_NOT\_FOUND

```
public static final int OBJECT_NOT_FOUND
```

Constant value: 7002

---

## OBJECT\_NOT\_UPDATED

```
public static final int OBJECT_NOT_UPDATED
```

Constant value: 7003

---

## OBJECT\_NOT\_DELETED

```
public static final int OBJECT_NOT_DELETED
```

Constant value: 7003

---

## FIELD\_EXISTS\_IN\_ACTIONS

```
public static final int FIELD_EXISTS_IN_ACTIONS
```

Constant value: 7004

---

## MAPPING\_OVERLAP

```
public static final int MAPPING_OVERLAP
```

Constant value: 7005

---

## REPOSITORY\_NOT\_CREATED

```
public static final int REPOSITORY_NOT_CREATED
```

Constant value: 7006

---

(continued from last page)

---

## MAPPING\_EXCEEDS\_LINE

```
public static final int MAPPING_EXCEEDS_LINE
```

Constant value: 7007

---

## APPLICATION\_RELOAD\_FAILED

```
public static final int APPLICATION_RELOAD_FAILED
```

Constant value: 7008

---

## INDEXATION\_PROCESS\_FAILED

```
public static final int INDEXATION_PROCESS_FAILED
```

Constant value: 7009

---

## MAPPING\_UTILITY\_FAILED

```
public static final int MAPPING_UTILITY_FAILED
```

Constant value: 7010

---

## DESTINATION\_FOLDER\_IS\_SAME

```
public static final int DESTINATION_FOLDER_IS_SAME
```

Constant value: 7011

---

## DESTINATION\_FOLDER\_IS\_SUBFOLDER

```
public static final int DESTINATION_FOLDER_IS_SUBFOLDER
```

Constant value: 7012

---

## IMPORT\_APPLICATION\_FAILED

```
public static final int IMPORT_APPLICATION_FAILED
```

Constant value: 7013

---

## IMPORT\_APPLICATION\_ENTITIES\_FAILED

```
public static final int IMPORT_APPLICATION_ENTITIES_FAILED
```

Constant value: 7014

---

## EXPORT\_APPLICATION\_ENTITIES\_FAILED

```
public static final int EXPORT_APPLICATION_ENTITIES_FAILED
```

---

(continued from last page)

---

Constant value: 7015

---

## **REPOSITORY\_NOT\_SYNCHRONIZED**

```
public static final int REPOSITORY_NOT_SYNCHRONIZED
```

---

Constant value: 7016

---

## **UPDATE\_NONE\_EXISTING\_ENTITY**

```
public static final int UPDATE_NONE_EXISTING_ENTITY
```

---

Constant value: 7017

---

## **FAULTY\_SCREEN\_IMAGE**

```
public static final int FAULTY_SCREEN_IMAGE
```

---

Constant value: 7018

---

## **ERROR\_SYNCHRONIZING\_REPOSITORY**

```
public static final int ERROR_SYNCHRONIZING_REPOSITORY
```

---

Constant value: 7019

---

## **IMPORT\_APPLICATION\_CONFIG\_MISSING**

```
public static final int IMPORT_APPLICATION_CONFIG_MISSING
```

---

Constant value: 7020

---

## **IMPORT\_APPLICATION\_HOST\_CATEGORY\_UNSUPPORTED**

```
public static final int IMPORT_APPLICATION_HOST_CATEGORY_UNSUPPORTED
```

---

Constant value: 7021

---

## **IMPORT\_APPLICATION\_COULD\_NOT\_RESTORE\_OVERRIDEN\_APP\_FROM\_BACKUP**

```
public static final int IMPORT_APPLICATION_COULD_NOT_RESTORE_OVERRIDEN_APP_FROM_BACKUP
```

---

Constant value: 7022

---

## **ERROR\_RETRIEVING\_WSDL\_SERVICES**

```
public static final int ERROR_RETRIEVING_WSDL_SERVICES
```

---

Constant value: 8000

---

---

## **ERROR\_CREATING\_ENTITIES\_FROM\_WSDL**

public static final int **ERROR\_CREATING\_ENTITIES\_FROM\_WSDL**

Constant value: **8001**

---

## **ERROR\_RETRIEVING\_WSDL\_SERVICES\_PERMISSION\_DENIED**

public static final int **ERROR\_RETRIEVING\_WSDL\_SERVICES\_PERMISSION\_DENIED**

Constant value: **8002**

---

## **ERROR\_CHANGING\_WEB\_SERVICE\_TARGET\_SERVER**

public static final int **ERROR\_CHANGING\_WEB\_SERVICE\_TARGET\_SERVER**

Constant value: **8500**

---

## **ERROR\_EXECUTING\_PROCEDURE\_GENERAL**

public static final int **ERROR\_EXECUTING\_PROCEDURE\_GENERAL**

Constant value: **9000**

---

## **ERROR\_DEPLOYMENT\_GENERAL**

public static final int **ERROR\_DEPLOYMENT\_GENERAL**

Constant value: **10000**

---

## **ERROR\_DEPLOYING\_WEB\_SERVICE**

public static final int **ERROR\_DEPLOYING\_WEB\_SERVICE**

Constant value: **10001**

---

## **ERROR\_UNDEPLOYING\_WEB\_SERVICE**

public static final int **ERROR\_UNDEPLOYING\_WEB\_SERVICE**

Constant value: **10002**

---

## **ERROR\_UPDATING\_WEB\_SERVICE**

public static final int **ERROR\_UPDATING\_WEB\_SERVICE**

Constant value: **10003**

---

(continued from last page)

---

## ERROR\_EXPORTING\_WEB\_SERVICE

```
public static final int ERROR_EXPORTING_WEB_SERVICE
```

Constant value: 10004

---

## ERROR\_CENTRASITE\_REGISTRATION

```
public static final int ERROR_CENTRASITE_REGISTRATION
```

Constant value: 10005

---

## ERROR\_DEPLOYMENT\_HANDLER\_NOT\_AVAILABLE

```
public static final int ERROR_DEPLOYMENT_HANDLER_NOT_AVAILABLE
```

Constant value: 10006

---

## ERROR\_INCOMPATIBLE\_WSS\_VERSION

```
public static final int ERROR_INCOMPATIBLE_WSS_VERSION
```

Constant value: 10007

---

## SESSION\_CANCELED

```
public static final int SESSION_CANCELED
```

Constant value: 11000

---

## SESSION\_CANCELED\_BY\_ADMIN

```
public static final int SESSION_CANCELED_BY_ADMIN
```

Constant value: 11001

---

## SESSION\_CANCELED\_BY\_INACTIVE\_TIMEOUT

```
public static final int SESSION_CANCELED_BY_INACTIVE_TIMEOUT
```

Constant value: 11002

---

## RESTRICTED\_ACCESS\_TO\_URL

```
public static final int RESTRICTED_ACCESS_TO_URL
```

Constant value: 12001

## Constructors

(continued from last page)

## **GXResponseCodes**

```
public GXResponseCodes( )
```

---

**Package**

**com.sabratec.applinx.framework**

# com.sabratec.applinx.framework

## Class GXAppConfig

java.lang.Object

└-com.sabratec.applinx.framework.GXAppConfig

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXWebAppConfig](#)

public class **GXAppConfig**  
 extends Object  
 implements java.io.Serializable

framework configurations basic class

## Constructor Summary

public	<a href="#">GXAppConfig()</a>
--------	-------------------------------

## Method Summary

String	<a href="#">get_Build()</a> For .NET framework usage.
boolean	<a href="#">get_CaptureTagsEvents()</a> <b>Deprecated.</b> Events are capture on the client side. Use <i>GX_CaptureInputTagEvents</i> in <i>_inc_applconf.js</i>
boolean	<a href="#">get_DesignMode()</a> For .NET framework usage.
String	<a href="#">get_ErrorForm()</a> For .NET framework usage.
int	<a href="#">get_FieldsModification()</a> <b>Deprecated.</b> - ApplinX determines the modified fields both from content difference, and typed fields.
int	<a href="#">get_FieldTypesInUse()</a> For .NET framework usage.
<a href="#">GXFtpConfig</a>	<a href="#">get_FtpConfig()</a> For .NET framework usage.
String	<a href="#">get_IndexForm()</a> For .NET framework usage.
boolean	<a href="#">get_Instant()</a> For .NET framework usage.

<a href="#">GXRenderConfig</a>	<a href="#">get_InstantConfig()</a> For .NET framework usage.
String	<a href="#">get_InstantForm()</a> For .NET framework usage.
String	<a href="#">get_InstantWinForm()</a> For .NET framework usage.
boolean	<a href="#">get_IsInstant()</a> For .NET framework usage.
<a href="#">GXMacroConfig</a>	<a href="#">get_MacroConfig()</a> For .NET framework usage.
String	<a href="#">get_MapPath()</a> For .NET framework usage.
boolean	<a href="#">get_PerformSyncCheckInInstant()</a> For .NET framework usage.
GXProceduresConfig	<a href="#">get_ProceduresConfig()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectBGColors()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectEmulationBehaviour()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectFGColors()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectHostCursor()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectHostProtected()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectMaxlength()</a> For .NET framework usage.
<a href="#">GXConfig</a>	<a href="#">get_SessionConfig()</a> For .NET framework usage.
<a href="#">GXSSOConfig</a>	<a href="#">get_SingleSignOnConfig()</a> For .NET framework usage.
<a href="#">GXSSOConfig</a>	<a href="#">get_SsoConfig()</a> For .NET framework usage.
boolean	<a href="#">get_SupportDupAndFieldMark()</a> For .NET framework usage.
<a href="#">GXTableBuildConfig</a>	<a href="#">get_TableBuildConfig()</a> For .NET framework usage.
boolean	<a href="#">get_TrimLabels()</a> For .NET framework usage.

boolean	<a href="#">get_UseFolders()</a> For .NET framework usage.
boolean	<a href="#">get_UseKeyboardPFKeys()</a> <b>Deprecated.</b> Please use the configuration file <code>config/gx_clientConfig.xml</code>
boolean	<a href="#">get_UseModalWindows()</a> For .NET framework usage.
boolean	<a href="#">get_UseScreenGroups()</a> For .NET framework usage.
String	<a href="#">get_Version()</a> For .NET framework usage.
String	<a href="#">getBuild()</a> Get the current ApplinX build number.
boolean	<a href="#">getCaptureTagsEvents()</a> <b>Deprecated.</b> Events are capture on the client side. Use <code>GX_CaptureInputTagEvents</code> in <code>_inc_applconf.js</code>
boolean	<a href="#">getDesignMode()</a> Get the design of the application, for example: to show the composer link.
String	<a href="#">getErrorForm()</a> The value of the error page.
int	<a href="#">getFieldsModification()</a> <b>Deprecated.</b> - ApplinX determines the modified fields both from content difference, and typed fields
int	<a href="#">getFieldTypesInUse()</a> Enumeration for type of fields should the framework read/update: Possible values: GXFrameworkHandler.GX_USE_ALL_FIELDS GXFrameworkHandler.GX_USE_APPFIELDS_AND_INPUTS GXFrameworkHandler.GX_USE_APPFIELDS GXFrameworkHandler.GX_USE_SINGLE_APPFIELDS
<a href="#">GXFtpConfig</a>	<a href="#">getFtpConfig()</a> Contains the FTP configuration
String	<a href="#">getIndexForm()</a>
<a href="#">GXRenderConfig</a>	<a href="#">getInstantConfig()</a> Contains the instant configuration
String	<a href="#">getInstantForm()</a> The value of the instant page.
String	<a href="#">getInstantWinForm()</a> The value of the page when a host window is shown and the framework is using the pop-up manager.
<a href="#">GXMacroConfig</a>	<a href="#">getMacroConfig()</a> Contains the macro configuration
String	<a href="#">getMapPath()</a> <b>Deprecated.</b> Application map are defined globally for each application. Map name will be ignored

boolean	<a href="#">getPerformSyncCheckInInstant()</a> Each page that inherits from GXDefaultLogicWebForm (.NET) / GXDefaultLogicContext (JSP) checks synchronization against the current host screen.
GXProceduresConfig	<a href="#">getProceduresConfig()</a> Contains the procedures configuration for activating ApplinX procedures
boolean	<a href="#">getReflectBGColors()</a> Get if to dynamically add a relevant back color css to the page fields when calling gx_fillForm.
boolean	<a href="#">getReflectEmulationBehaviour()</a> Determines whether to dynamically add to input fields emulation attributes when calling gx_fillForm.
boolean	<a href="#">getReflectFGColors()</a> Get if to dynamically add a relevant color css to the page fields when calling gx_fillForm.
boolean	<a href="#">getReflectHostCursor()</a> When inheriting from GXBasicWebForm (.NET) or from GXBasicContext (JSP), ApplinX full JavaScript is not added to the page.
boolean	<a href="#">getReflectHostProtected()</a> Determines whether to dynamically check the page fields when gx_fillForm is called.
boolean	<a href="#">getReflectMaxlength()</a> Determines whether to dynamically add a maxlength attribute to input fields when calling gx_fillForm in generated pages.
<a href="#">GXConfig</a>	<a href="#">getSessionConfig()</a> Contains the ApplinX session connection declaration
<a href="#">GXSSOConfig</a>	<a href="#">getSingleSignOnConfig()</a> Contains single signon configurations
boolean	<a href="#">getSupportDupAndFieldMark()</a> Is supporting Mainframe host keys Dup and Fieldmark in instant and generated pages.
<a href="#">GXTableBuildConfig</a>	<a href="#">getTableBuildConfig()</a> Contains the table configuration for generated pages
boolean	<a href="#">getTrimLabels()</a> Is trimming the protected field when gx_fillForm is called.
boolean	<a href="#">getUseFolders()</a> Determines whether to use ApplinX folders also for the Web application.
boolean	<a href="#">getUseKeyboardPFKeys()</a> <b>Deprecated.</b> Please use the configuration file config/gx_clientConfig.xml
boolean	<a href="#">getUseModalWindows()</a> Determines whether to activate ApplinX pop-up manager and transform recognized host windows for Web pop-up windows.
boolean	<a href="#">getUseScreenGroups()</a> Determines whether to load a page for the current screen group, when the current screen does not contain a matching page.
String	<a href="#">getVersion()</a> Get the current ApplinX version.

boolean	<a href="#"><u>isInstant()</u></a> For internal use
void	<a href="#"><u>set_CaptureTagsEvents</u></a> (boolean captureTagsEvents) <b>Deprecated.</b> <i>Events are capture on the client side. Use <code>GX_CaptureInputTagEvents</code> in <code>_inc_applconf.js</code></i>
void	<a href="#"><u>set_DesignMode</u></a> (boolean isDesignMode) For .NET framework usage.
void	<a href="#"><u>set_ErrorForm</u></a> (String errorForm) For .NET framework usage.
void	<a href="#"><u>set_FieldsModification</u></a> (int fieldsModification) <b>Deprecated.</b> - <i>ApplinX determines the modified fields both from content difference, and typed fields.</i>
void	<a href="#"><u>set_FieldTypesInUse</u></a> (int fieldTypesInUse) For .NET framework usage.
void	<a href="#"><u>set_FtpConfig</u></a> ( <a href="#"><u>GXFtpConfig</u></a> ftpConfig) For .NET framework usage.
void	<a href="#"><u>set_IndexForm</u></a> (String indexForm) For .NET framework usage.
void	<a href="#"><u>set_Instant</u></a> (boolean isInstant) For .NET framework usage.
void	<a href="#"><u>set_InstantConfig</u></a> ( <a href="#"><u>GXRenderConfig</u></a> instantConfig) For .NET framework usage.
void	<a href="#"><u>set_InstantForm</u></a> (String instantForm) For .NET framework usage.
void	<a href="#"><u>set_InstantWinForm</u></a> (String instantWinForm) For .NET framework usage.
void	<a href="#"><u>set_IsInstant</u></a> (boolean isInstant) For .NET framework usage.
void	<a href="#"><u>set_MacroConfig</u></a> ( <a href="#"><u>GXMacroConfig</u></a> macroConfig) For .NET framework usage.
void	<a href="#"><u>set_MapPath</u></a> (String mapPath) For .NET framework usage.
void	<a href="#"><u>set_PerformSyncCheckInInstant</u></a> (boolean performSyncCheckInInstant) For .NET framework usage.
void	<a href="#"><u>set_ProceduresConfig</u></a> ( <a href="#"><u>GXProceduresConfig</u></a> proceduresConfig) For .NET framework usage.
void	<a href="#"><u>set_ReflectBGColors</u></a> (boolean reflectBGColors) For .NET framework usage.
void	<a href="#"><u>set_ReflectEmulationBehaviour</u></a> (boolean reflectEmulationBehaviour) For .NET framework usage.

void	<a href="#"><u>set_ReflectFGColors</u></a> (boolean reflectFGColors) Set if to dynamically add a relevant color css to the page fields when calling gx_fillForm.
void	<a href="#"><u>set_ReflectHostCursor</u></a> (boolean reflectHostCursor) For .NET framework usage.
void	<a href="#"><u>set_ReflectHostProtected</u></a> (boolean reflectHostProtected) For .NET framework usage.
void	<a href="#"><u>set_ReflectMaxlength</u></a> (boolean reflectMaxlength) For .NET framework usage.
void	<a href="#"><u>set_SessionConfig</u></a> (GXConfig sessionConfig) For .NET framework usage.
void	<a href="#"><u>set_SingleSignOnConfig</u></a> (GXSSOConfig ssoConfig) For .NET framework usage.
void	<a href="#"><u>set_SsoConfig</u></a> (GXSSOConfig ssoConfig) For .NET framework usage.
void	<a href="#"><u>set_SupportDupAndFieldMark</u></a> (boolean supportDupAndFieldMark) For .NET framework usage.
void	<a href="#"><u>set_TableBuildConfig</u></a> (GXTableBuildConfig tableConfig) For .NET framework usage.
void	<a href="#"><u>set_TrimLabels</u></a> (boolean TrimLabels) For .NET framework usage.
void	<a href="#"><u>set_UseFolders</u></a> (boolean useFolders) For .NET framework usage.
void	<a href="#"><u>set_UseKeyboardPFKeys</u></a> (boolean useKeyboardPFKeys) <b>Deprecated.</b> Please use the configuration file config/gx_clientConfig.xml
void	<a href="#"><u>set_UseModalWindows</u></a> (boolean useModalWindows) For .NET framework usage.
void	<a href="#"><u>set_UseScreenGroups</u></a> (boolean useScreenGroups) For .NET framework usage.
void	<a href="#"><u>setCaptureTagsEvents</u></a> (boolean captureTagsEvents) <b>Deprecated.</b> Events are capture on the client side. Use GX_CaptureInputTagEvents in _inc_applconf.js
void	<a href="#"><u>setDesignMode</u></a> (boolean isDesignMode) Set the design of the application, for example: to show the composer link.
void	<a href="#"><u>setErrorForm</u></a> (String errorForm) The value of the error page.
void	<a href="#"><u>setFieldsModification</u></a> (int fieldsModification) <b>Deprecated.</b> - ApplinX determines the modified fields both from content difference, and typed fields

void	<a href="#"><u>setFieldTypesInUse</u></a> (int fieldTypesInUse) Enumeration for type of fields should the framework read/update: Possible values: GXFrameworkHandler.GX_USE_ALL_FIELDS GXFrameworkHandler.GX_USE_APPFIELDS_AND_INPUTS GXFrameworkHandler.GX_USE_APPFIELDS GXFrameworkHandler.GX_USE_SINGLE_APPFIELDS
void	<a href="#"><u>setFtpConfig</u></a> ( <a href="#"><u>GXFtpConfig</u></a> ftpConfig) For internal use
void	<a href="#"><u>setIndexForm</u></a> (String indexForm)
void	<a href="#"><u>setInstantConfig</u></a> ( <a href="#"><u>GXRenderConfig</u></a> instantConfig) For internal use
void	<a href="#"><u>setInstantForm</u></a> (String instantForm) Returns the value of the instant page.
void	<a href="#"><u>setInstantWinForm</u></a> (String instantWinForm) The value of the page when a host window is shown and the framework is using the pop-up manager.
void	<a href="#"><u>setIsInstant</u></a> (boolean isInstant) For internal use
void	<a href="#"><u>setMacroConfig</u></a> ( <a href="#"><u>GXMacroConfig</u></a> macroConfig) For internal use
void	<a href="#"><u>setMapPath</u></a> (String mapPath) <b>Deprecated.</b> Application map are defined globally for each application. Map name will be ignored
void	<a href="#"><u>setPerformSyncCheckInInstant</u></a> (boolean performSyncCheckInInstant) Each page that inherits from GXDefaultLogicWebForm (.NET) / GXDefaultLogicContext (JSP) checks synchronization against the current host screen.
void	<a href="#"><u>setProceduresConfig</u></a> ( <a href="#"><u>GXProceduresConfig</u></a> proceduresConfig) For internal use
void	<a href="#"><u>setReflectBGColors</u></a> (boolean reflectBGColors) Determines whether the field background color in the Web application will be the same as the field background color in the host.
void	<a href="#"><u>setReflectEmulationBehaviour</u></a> (boolean reflectEmulationBehaviour) Determines whether to add emulation attributes in text fields.
void	<a href="#"><u>setReflectFGColors</u></a> (boolean reflectFGColors) Determines whether the field foreground color in the Web application will be the same as the field foreground color in the host.
void	<a href="#"><u>setReflectHostCursor</u></a> (boolean reflectHostCursor) When inheriting from GXBasicWebForm (.NET) or from GXBasicContext (JSP), ApplinX full JavaScript is not added to the page.
void	<a href="#"><u>setReflectHostProtected</u></a> (boolean reflectHostProtected) Determines whether the protected fields in the host will be read only in the Web application (when gx_fillForm is called).
void	<a href="#"><u>setReflectMaxlength</u></a> (boolean reflectMaxlength) Determines whether the fields' maximum length attribute in the host will be the maximum length attribute in the input fields in the Web application.

void	<a href="#"><u>setSessionConfig</u></a> ( <a href="#"><u>GXConfig</u></a> sessionConfig) For internal use
void	<a href="#"><u>setSingleSignOnConfig</u></a> ( <a href="#"><u>GXSSOConfig</u></a> ssoConfig) For internal use
void	<a href="#"><u>setSupportDupAndFieldMark</u></a> (boolean supportDupAndFieldMark) Dup and FieldMark are special Mainframe characters.
void	<a href="#"><u>setTableBuildConfig</u></a> ( <a href="#"><u>GXTableBuildConfig</u></a> tableBuildConfig) For internal use
void	<a href="#"><u>setTrimLabels</u></a> (boolean TrimLabels) <b>Deprecated.</b> Use ApplinX server configuration, screen content tab
void	<a href="#"><u>setUseFolders</u></a> (boolean useFolders) Determines whether the Web application folder structure should be the same folder structure as configured in the ApplinX repository.
void	<a href="#"><u>setUseKeyboardPFKeys</u></a> (boolean useKeyboardPFKeys) <b>Deprecated.</b> Please use the configuration file config/gx_clientConfig.xml
void	<a href="#"><u>setUseModalWindows</u></a> (boolean useModalWindows) Determines whether to activate ApplinX pop-up manager and transform recognized host windows to Web pop-up windows.
void	<a href="#"><u>setUseScreenGroups</u></a> (boolean useScreenGroups) Determines whether to load a page for the current screen group, when the current screen does not contain a matching page.

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXAppConfig

```
public GXAppConfig()
```

## Methods

### get\_Version

```
public String get_Version()
```

For .NET framework usage. Expose the method as a property.

### getVersion

```
public String getVersion()
```

Get the current ApplinX version.

**Returns:**

(continued from last page)

String

---

## get\_Build

```
public String get_Build()
```

For .NET framework usage. Expose the method as a property.

---

## getBuild

```
public String getBuild()
```

Get the current ApplinX build number.

**Returns:**

String

---

## get\_DesignMode

```
public boolean get_DesignMode()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DesignMode

```
public void set_DesignMode(boolean isDesignMode)
```

For .NET framework usage. Expose the method as a property.

---

## getDesignMode

```
public boolean getDesignMode()
```

Get the design of the application, for example: to show the composer link. Default value: true

**Returns:**

String

---

## setDesignMode

```
public void setDesignMode(boolean isDesignMode)
```

Set the design of the application, for example: to show the composer link. It is recommended to change this parameter to false when deploying the application to production Default value: true

---

## get\_UseKeyboardPFKeys

```
public boolean get_UseKeyboardPFKeys()
```

**Deprecated.** Please use the configuration file *config/gx\_clientConfig.xml*

For .NET framework usage. Expose the method as a property.

---

## set\_UseKeyboardPFKeys

```
public void set_UseKeyboardPFKeys(boolean useKeyboardPFKeys)
```

**Deprecated.** Please use the configuration file *config/gx\_clientConfig.xml*

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## getUseKeyboardPFKeys

```
public boolean getUseKeyboardPFKeys()
```

**Deprecated.** Please use the configuration file `config/gx_clientConfig.xml`

Is the pf keys on the browser are activated. Default value: true.

**Returns:**

boolean

---

## setUseKeyboardPFKeys

```
public void setUseKeyboardPFKeys(boolean useKeyboardPFKeys)
```

**Deprecated.** Please use the configuration file `config/gx_clientConfig.xml`

Set if the pf keys on the browser are activated. Default value: true.

---

## get\_SupportDupAndFieldMark

```
public boolean get_SupportDupAndFieldMark()
```

For .NET framework usage. Expose the method as a property.

---

## set\_SupportDupAndFieldMark

```
public void set_SupportDupAndFieldMark(boolean supportDupAndFieldMark)
```

For .NET framework usage. Expose the method as a property.

---

## getSupportDupAndFieldMark

```
public boolean getSupportDupAndFieldMark()
```

Is supporting Mainframe host keys Dup and Fieldmark in instant and generated pages.

**Returns:**

boolean

---

## setSupportDupAndFieldMark

```
public void setSupportDupAndFieldMark(boolean supportDupAndFieldMark)
```

Dup and FieldMark are special Mainframe characters. In Instant and generated pages, a user can send to the host a non printable character in specific input fields.

---

## get\_FieldTypesInUse

```
public int get_FieldTypesInUse()
```

For .NET framework usage. Expose the method as a property.

---

## set\_FieldTypesInUse

```
public void set_FieldTypesInUse(int fieldTypesInUse)
```

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## getFieldTypesInUse

```
public int getFieldTypesInUse()
```

Enumeration for type of fields should the framework read/update: Possible values:  
GXFrameworkHandler.GX\_USE\_ALL\_FIELDS GXFrameworkHandler.GX\_USE\_APPFIELDS\_AND\_INPUTS  
GXFrameworkHandler.GX\_USE\_APPFIELDS GXFrameworkHandler.GX\_USE\_SINGLE\_APPFIELDS

---

## setFieldTypesInUse

```
public void setFieldTypesInUse(int fieldTypesInUse)
```

Enumeration for type of fields should the framework read/update: Possible values:  
GXFrameworkHandler.GX\_USE\_ALL\_FIELDS GXFrameworkHandler.GX\_USE\_APPFIELDS\_AND\_INPUTS  
GXFrameworkHandler.GX\_USE\_APPFIELDS GXFrameworkHandler.GX\_USE\_SINGLE\_APPFIELDS

---

## get\_TrimLabels

```
public boolean get_TrimLabels()
```

For .NET framework usage. Expose the method as a property.

---

## set\_TrimLabels

```
public void set_TrimLabels(boolean TrimLabels)
```

For .NET framework usage. Expose the method as a property.

---

## getTrimLabels

```
public boolean getTrimLabels()
```

Is trimming the protected field when gx\_fillForm is called.

---

## setTrimLabels

```
public void setTrimLabels(boolean TrimLabels)
```

**Deprecated.** Use ApplinX server configuration, screen content tab

---

## get\_InstantForm

```
public String get_InstantForm()
```

For .NET framework usage. Expose the method as a property.

---

## set\_InstantForm

```
public void set_InstantForm(String instantForm)
```

For .NET framework usage. Expose the method as a property.

---

## getInstantForm

```
public String getInstantForm()
```

(continued from last page)

The value of the instant page. By default: instant.jsp (JSP) / instant.aspx (.NET)

---

## setInstantForm

```
public void setInstantForm(String instantForm)
```

Returns the value of the instant page. By default: instant.jsp (JSP) / instant.aspx (.NET)

---

## get\_InstantWinForm

```
public String get_InstantWinForm()
```

For .NET framework usage. Expose the method as a property.

---

## set\_InstantWinForm

```
public void set_InstantWinForm(String instantWinForm)
```

For .NET framework usage. Expose the method as a property.

---

## getInstantWinForm

```
public String getInstantWinForm()
```

The value of the page when a host window is shown and the framework is using the pop-up manager. By default: instantWin.jsp (JSP) / instantWin.aspx (.NET)

---

## setInstantWinForm

```
public void setInstantWinForm(String instantWinForm)
```

The value of the page when a host window is shown and the framework is using the pop-up manager. By default: instantWin.jsp (JSP) / instantWin.aspx (.NET)

---

## get\_ErrorForm

```
public String get_ErrorForm()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ErrorForm

```
public void set_ErrorForm(String errorForm)
```

For .NET framework usage. Expose the method as a property.

---

## getErrorForm

```
public String getErrorForm()
```

The value of the error page. By default: error.jsp (JSP) / error.aspx (.NET)

---

## setErrorForm

```
public void setErrorForm(String errorForm)
```

The value of the error page. By default: error.jsp (JSP) / error.aspx (.NET)

---

---

## get\_ReflectFGColors

```
public boolean get_ReflectFGColors( )
```

For .NET framework usage. Expose the method as a property.

---

## set\_ReflectFGColors

```
public void set_ReflectFGColors(boolean reflectFGColors)
```

Set if to dynamically add a relevant color css to the page fields when calling gx\_fillForm. The relevant css classes are in styles\_instant.css in the new application /css\_colors folder in the emulation template.

---

## getReflectFGColors

```
public boolean getReflectFGColors( )
```

Get if to dynamically add a relevant color css to the page fields when calling gx\_fillForm. The relevant css classes are in styles\_instant.css in the new application /css\_colors folder in the emulation template.

**Returns:**

whether to dynamically add a relevant color css to the page input fields

---

## setReflectFGColors

```
public void setReflectFGColors(boolean reflectFGColors)
```

Determines whether the field foreground color in the Web application will be the same as the field foreground color in the host. The relevant css classes are in css/styles\_generated.css file.

**Parameters:**

reflectFGColor - whether to dynamically add a relevant color css to the page input fields

---

## get\_ReflectBGColors

```
public boolean get_ReflectBGColors( )
```

For .NET framework usage. Expose the method as a property.

---

## set\_ReflectBGColors

```
public void set_ReflectBGColors(boolean reflectBGColors)
```

For .NET framework usage. Expose the method as a property.

---

## getReflectBGColors

```
public boolean getReflectBGColors( )
```

Get if to dynamically add a relevant back color css to the page fields when calling gx\_fillForm. The relevant css classes are in css/styles\_generated.css file.

**Returns:**

whether to dynamically add a relevant back color css to the page input fields

---

(continued from last page)

## setReflectBGColors

```
public void setReflectBGColors(boolean reflectBGColors)
```

Determines whether the field background color in the Web application will be the same as the field background color in the host. The relevant css classes are in css/styles\_generated.css file. /css\_colors folder in the emulation template.

### Parameters:

reflectBGColor - whether to dynamically add a relevant back color css to the page fields

---

## get\_ReflectEmulationBehaviour

```
public boolean get_ReflectEmulationBehaviour()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ReflectEmulationBehaviour

```
public void set_ReflectEmulationBehaviour(boolean reflectEmulationBehaviour)
```

For .NET framework usage. Expose the method as a property.

---

## getReflectEmulationBehaviour

```
public boolean getReflectEmulationBehaviour()
```

Determines whether to dynamically add to input fields emulation attributes when calling gx\_fillForm. The emulation attributes cause the input fields to behave like host emulation unprotected fields.

### Returns:

whether to dynamically add to input fields emulation attributes

---

## setReflectEmulationBehaviour

```
public void setReflectEmulationBehaviour(boolean reflectEmulationBehaviour)
```

Determines whether to add emulation attributes in text fields. Emulation attributes are special ApplinX attributes that start with "gx\_" and are added to the rendered text field tag. These attributes provide additional information to the ApplinX JavaScript engine, which changes the behavior of the text fields.  
The attributes are:

- gx\_autoSkip - will cause the text field to skip to the neighboring text field, once it is filled with content.
- gx\_dt - The data type of the field. See [GXBaseObjectConstants](#) for possible data types.
- gx\_ra - AS/400 right adjust
- gx\_au - AS/400 automatic enter

### Parameters:

reflectEmulationBehaviour - whether to add emulation attributes in text fields

---

## get\_ReflectHostCursor

```
public boolean get_ReflectHostCursor()
```

For .NET framework usage. Expose the method as a property.

(continued from last page)

---

## set\_ReflectHostCursor

```
public void set_ReflectHostCursor(boolean reflectHostCursor)
```

For .NET framework usage. Expose the method as a property.

---

## getReflectHostCursor

```
public boolean getReflectHostCursor()
```

When inheriting from GXBasicWebForm (.NET) or from GXBasicContext (JSP), ApplinX full JavaScript is not added to the page. This causes the page not to behave according to the host screen behavior. This parameter adds the functionality of focusing on an input field according to the host.

**Returns:**

whether to add the functionality of focusing on an input field according to the host

---

## setReflectHostCursor

```
public void setReflectHostCursor(boolean reflectHostCursor)
```

When inheriting from GXBasicWebForm (.NET) or from GXBasicContext (JSP), ApplinX full JavaScript is not added to the page. This causes the page not to behave according to the host screen behavior. This parameter adds the functionality of focusing on an input field according to the host.

**Parameters:**

reflectHostCursor - whether to add the functionality of focusing on an input field according to the host

---

## get\_ReflectMaxlength

```
public boolean get_ReflectMaxlength()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ReflectMaxlength

```
public void set_ReflectMaxlength(boolean reflectMaxlength)
```

For .NET framework usage. Expose the method as a property.

---

## getReflectMaxlength

```
public boolean getReflectMaxlength()
```

Determines whether to dynamically add a maxlength attribute to input fields when calling gx\_fillForm in generated pages. This saves you the work of manually adding this attribute to each input field when designing the page.

**Returns:**

whether to dynamically add a maxlength attribute to input fields

---

## setReflectMaxlength

```
public void setReflectMaxlength(boolean reflectMaxlength)
```

Determines whether the fields' maximum length attribute in the host will be the maximum length attribute in the input fields in the Web application. This saves you the work of manually adding this attribute to each input field when designing the page.

**Parameters:**

reflectMaxlength - whether to dynamically add a maxlength attribute to input fields

---

---

## get\_ReflectHostProtected

```
public boolean get_ReflectHostProtected()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ReflectHostProtected

```
public void set_ReflectHostProtected(boolean reflectHostProtected)
```

For .NET framework usage. Expose the method as a property.

---

## getReflectHostProtected

```
public boolean getReflectHostProtected()
```

Determines whether to dynamically check the page fields when gx\_fillForm is called.

**Returns:**

whether to dynamically check the page fields when gx\_fillForm is called

---

## setReflectHostProtected

```
public void setReflectHostProtected(boolean reflectHostProtected)
```

Determines whether the protected fields in the host will be read only in the Web application (when gx\_fillForm is called).

**Parameters:**

reflectHostProtected - whether to dynamically check the page fields when gx\_fillForm is called

---

## get\_MapPath

```
public String get_MapPath()
```

For .NET framework usage. Expose the method as a property.

---

## set\_MapPath

```
public void set_MapPath(String mapPath)
```

For .NET framework usage. Expose the method as a property.

---

## getMapPath

```
public String getMapPath()
```

**Deprecated.** Application map are defined globally for each application. Map name will be ignored

Determines the map path used to synchronize the host with the page. It is used only if gx\_syncHostWithForm is called.

**Returns:**

determines the map path used to synchronize the host with the page

---

## setMapPath

```
public void setMapPath(String mapPath)
```

---

(continued from last page)

**Deprecated.** Application map are defined globally for each application. Map name will be ignored

Determines the map path used to synchronize the host with the page. It is used only if `gx_syncHostWithForm` is called.

**Parameters:**

`mapPath` - determines the map path used to synchronize the host with the page

---

## get\_SessionConfig

```
public GXConfig get_SessionConfig()
```

For .NET framework usage. Expose the method as a property.

---

## set\_SessionConfig

```
public void set_SessionConfig(GXConfig sessionConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getSessionConfig

```
public GXConfig getSessionConfig()
```

Contains the ApplinX session connection declaration

**Returns:**

the ApplinX session connection declaration

---

## setSessionConfig

```
public void setSessionConfig(GXConfig sessionConfig)
```

For internal use

---

## get\_MacroConfig

```
public GXMacroConfig get_MacroConfig()
```

For .NET framework usage. Expose the method as a property.

---

## set\_MacroConfig

```
public void set_MacroConfig(GXMacroConfig macroConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getMacroConfig

```
public GXMacroConfig getMacroConfig()
```

Contains the macro configuration

**Returns:**

the macro configuration

---

## setMacroConfig

```
public void setMacroConfig(GXMacroConfig macroConfig)
```

(continued from last page)

For internal use

---

## get\_UseModalWindows

```
public boolean get_UseModalWindows()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseModalWindows

```
public void set_UseModalWindows(boolean useModalWindows)
```

For .NET framework usage. Expose the method as a property.

---

## getUseModalWindows

```
public boolean getUseModalWindows()
```

Determines whether to activate ApplinX pop-up manager and transform recognized host windows for Web pop-up windows.

**Returns:**

whether to activate ApplinX pop-up manager

---

## setUseModalWindows

```
public void setUseModalWindows(boolean useModalWindows)
```

Determines whether to activate ApplinX pop-up manager and transform recognized host windows to Web pop-up windows.

**Parameters:**

useModalWindows - whether to activate ApplinX pop-up manager

---

## get\_UseFolders

```
public boolean get_UseFolders()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseFolders

```
public void set_UseFolders(boolean useFolders)
```

For .NET framework usage. Expose the method as a property.

---

## getUseFolders

```
public boolean getUseFolders()
```

Determines whether to use ApplinX folders also for the Web application. This way the pages that need to be loaded are searched for in the relevant path

**Returns:**

whether to use ApplinX folders also for the Web application

---

(continued from last page)

---

## setUseFolders

```
public void setUseFolders(boolean useFolders)
```

Determines whether the Web application folder structure should be the same folder structure as configured in the ApplinX repository. Relevant for screens only. This is recommended when the Web application contains a large number of designed Web pages.

### Parameters:

useFolders - whether to use ApplinX folders also for the Web application

---

## get\_UseScreenGroups

```
public boolean get_UseScreenGroups()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseScreenGroups

```
public void set_UseScreenGroups(boolean useScreenGroups)
```

For .NET framework usage. Expose the method as a property.

---

## getUseScreenGroups

```
public boolean getUseScreenGroups()
```

Determines whether to load a page for the current screen group, when the current screen does not contain a matching page. If the current screen has more than one screen group, the first screen group which has a matching page will be loaded.

### Returns:

whether to load a page for the current screen group, when the current screen does not contain a matching page

---

## setUseScreenGroups

```
public void setUseScreenGroups(boolean useScreenGroups)
```

Determines whether to load a page for the current screen group, when the current screen does not contain a matching page. If the current screen has more than one screen group, the first screen group which has a matching page will be loaded.

### Parameters:

useScreenGroups - whether to load a page for the current screen group, when the current screen does not contain a matching page  
useScreenGroups

---

## get\_CaptureTagsEvents

```
public boolean get_CaptureTagsEvents()
```

**Deprecated.** Events are capture on the client side. Use *GX\_CaptureInputTagEvents* in *\_inc\_applconf.js*

---

## set\_CaptureTagsEvents

```
public void set_CaptureTagsEvents(boolean captureTagsEvents)
```

**Deprecated.** Events are capture on the client side. Use *GX\_CaptureInputTagEvents* in *\_inc\_applconf.js*

---

## getCaptureTagsEvents

```
public boolean getCaptureTagsEvents()
```

**Deprecated.** Events are capture on the client side. Use *GX\_CaptureInputTagEvents* in *\_inc\_applconf.js*

---

## setCaptureTagsEvents

```
public void setCaptureTagsEvents(boolean captureTagsEvents)
```

**Deprecated.** Events are capture on the client side. Use *GX\_CaptureInputTagEvents* in *\_inc\_applconf.js*

---

## get\_PerformSyncCheckInInstant

```
public boolean get_PerformSyncCheckInInstant()
```

For .NET framework usage. Expose the method as a property.

---

## set\_PerformSyncCheckInInstant

```
public void set_PerformSyncCheckInInstant(boolean performSyncCheckInInstant)
```

For .NET framework usage. Expose the method as a property.

---

## getPerformSyncCheckInInstant

```
public boolean getPerformSyncCheckInInstant()
```

Each page that inherits from *GXDefaultLogicWebForm* (.NET) / *GXDefaultLogicContext* (JSP) checks synchronization against the current host screen. However instant pages, by default, do not check for synchronization, therefore when *instant.jsp* is called from the URL, the framework will not synchronize to the matching page. Setting this parameter to true: make a synchronization check also in *instant.aspx* / *instantWin.aspx* (.NET) *instant.jsp* / *instantWin.jsp* (JSP).

**Returns:**

whether a synchronization check is done in instant pages

---

## setPerformSyncCheckInInstant

```
public void setPerformSyncCheckInInstant(boolean performSyncCheckInInstant)
```

Each page that inherits from *GXDefaultLogicWebForm* (.NET) / *GXDefaultLogicContext* (JSP) checks synchronization against the current host screen. However instant pages, by default, do not check for synchronization, therefore when *instant.jsp* is called from the URL, the framework will not synchronize to the matching page. Setting this parameter to true: make a synchronization check also in *instant.aspx* / *instantWin.aspx* (.NET) *instant.jsp* / *instantWin.jsp* (JSP).

**Parameters:**

*performSyncCheckInInstant* - whether a synchronization check is done in instant pages

---

## get\_FtpConfig

```
public GXFtpConfig get_FtpConfig()
```

For .NET framework usage. Expose the method as a property.

---

## getFtpConfig

```
public GXFtpConfig getFtpConfig()
```

Contains the FTP configuration

**Returns:**  
the FTP configuration

---

## setFtpConfig

```
public void setFtpConfig(GXFtpConfig ftpConfig)
```

For internal use

---

## get\_TableBuildConfig

```
public GXTableBuildConfig get_TableBuildConfig()
```

For .NET framework usage. Expose the method as a property.

---

## set\_TableBuildConfig

```
public void set_TableBuildConfig(GXTableBuildConfig tableConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getTableBuildConfig

```
public GXTableBuildConfig getTableBuildConfig()
```

Contains the table configuration for generated pages

**Returns:**  
the table configuration for generated pages

---

## setTableBuildConfig

```
public void setTableBuildConfig(GXTableBuildConfig tableBuildConfig)
```

For internal use

**Parameters:**  
tableBuildConfig

---

## get\_ProceduresConfig

```
public GXProceduresConfig get_ProceduresConfig()
```

For .NET framework usage. Expose the method as a property.

---

## getProceduresConfig

```
public GXProceduresConfig getProceduresConfig()
```

Contains the procedures configuration for activating ApplinX procedures

**Returns:**

---

(continued from last page)

the procedures configuration for activating ApplinX procedures

---

## setProceduresConfig

```
public void setProceduresConfig(GXProceduresConfig proceduresConfig)
```

For internal use

---

## get\_InstantConfig

```
public GXRenderConfig get_InstantConfig()
```

For .NET framework usage. Expose the method as a property.

---

## set\_InstantConfig

```
public void set_InstantConfig(GXRenderConfig instantConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getInstantConfig

```
public GXRenderConfig getInstantConfig()
```

Contains the instant configuration

**Returns:**

the instant configuration

---

## setInstantConfig

```
public void setInstantConfig(GXRenderConfig instantConfig)
```

For internal use

---

## get\_FieldsModification

```
public int get_FieldsModification()
```

**Deprecated.** - ApplinX determines the modified fields both from content difference, and typed fields.

---

## set\_FieldsModification

```
public void set_FieldsModification(int fieldsModification)
```

**Deprecated.** - ApplinX determines the modified fields both from content difference, and typed fields.

---

## getFieldsModification

```
public int getFieldsModification()
```

**Deprecated.** - ApplinX determines the modified fields both from content difference, and typed fields

---

---

## setFieldsModification

```
public void setFieldsModification(int fieldsModification)
```

**Deprecated.** - *ApplinX determines the modified fields both from content difference, and typed fields*

---

---

## set\_IsInstant

```
public void set_IsInstant(boolean isInstant)
```

For .NET framework usage. Expose the method as a property.

---

---

## isInstant

```
public boolean isInstant()
```

For internal use

---

---

## setIsInstant

```
public void setIsInstant(boolean isInstant)
```

For internal use

---

---

## get\_IsInstant

```
public boolean get_IsInstant()
```

For .NET framework usage. Expose the method as a property.

---

---

## set\_Instant

```
public void set_Instant(boolean isInstant)
```

For .NET framework usage. Expose the method as a property.

---

---

## getSingleSignOnConfig

```
public GXSSOConfig getSingleSignOnConfig()
```

Contains single signon configurations

**Returns:**

the single signon configuration

---

---

## setSingleSignOnConfig

```
public void setSingleSignOnConfig(GXSSOConfig ssoConfig)
```

For internal use

---

---

## get\_SsoConfig

```
public GXSSOConfig get_SsoConfig()
```

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **set\_SsoConfig**

```
public void set_SsoConfig(GXSSOConfig ssoConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **getIndexForm**

```
public String getIndexForm()
```

---

## **setIndexForm**

```
public void setIndexForm(String indexForm)
```

---

## **set\_FtpConfig**

```
public void set_FtpConfig(GXFtpConfig ftpConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **set\_ProceduresConfig**

```
public void set_ProceduresConfig(GXProceduresConfig proceduresConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_Instant**

```
public boolean get_Instant()
```

For .NET framework usage. Expose the method as a property.

---

## **get\_SingleSignOnConfig**

```
public GXSSOConfig get_SingleSignOnConfig()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_SingleSignOnConfig**

```
public void set_SingleSignOnConfig(GXSSOConfig ssoConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_IndexForm**

```
public String get_IndexForm()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## **set\_IndexForm**

```
public void set_IndexForm(String indexForm)
```

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.framework Class GXConfig

java.lang.Object

└─com.sabratec.applinx.framework.GXConfig

**All Implemented Interfaces:**

Cloneable

public class **GXConfig**  
extends Object  
implements Cloneable

### Constructor Summary

public	<a href="#">GXConfig()</a>
--------	----------------------------

### Method Summary

void	<a href="#">addVariable(GXVariable variable)</a> Adds a variable to the framework configuration The variable will be used in the internal framework requests to the server.
void	<a href="#">addVariable(String varName, String varValue)</a>
Object	<a href="#">clone()</a>
String	<a href="#">get_ApplicationName()</a> For .NET framework usage.
String	<a href="#">get_Description()</a> For .NET framework usage.
boolean	<a href="#">get_FailOnWarning()</a> For .NET framework usage.
String	<a href="#">get_HostNewPassword()</a> For .NET framework usage.
String	<a href="#">get_HostPassword()</a> For .NET framework usage.
String	<a href="#">get_HostUserName()</a> For .NET framework usage.
String	<a href="#">get_Password()</a> For .NET framework usage.
String	<a href="#">get_ServerURL()</a> For .NET framework usage.

String	<a href="#"><u>getServiceName()</u></a> For .NET framework usage.
String	<a href="#"><u>getSessionId()</u></a> For .NET framework usage.
boolean	<a href="#"><u>getShowIntermediateScreen()</u></a> For .NET framework usage.
<a href="#"><u>GXVariable</u></a>	<a href="#"><u>getVariable()</u></a> (String varName) For .NET framework usage.
<a href="#"><u>GXVariable[]</u></a>	<a href="#"><u>getVariables()</u></a> For .NET framework usage.
boolean	<a href="#"><u>getVariablesModified()</u></a> For .NET framework usage.
String	<a href="#"><u>getApplicationName()</u></a> The name of the ApplinX application.
String	<a href="#"><u>getDescription()</u></a> The description of the session in ApplinX server.
String	<a href="#"><u>getHostNewPassword()</u></a>
String	<a href="#"><u>getHostPassword()</u></a> The host password.
String	<a href="#"><u>getHostUserName()</u></a> The host user name for SSH and Natural-Unix protocol only.
String	<a href="#"><u>getPassword()</u></a> The password to the session in ApplinX server.
GXIApplicationServerContext	<a href="#"><u>getServerContext()</u></a> For J2EE usage only.
String	<a href="#"><u>getServerURL()</u></a> The ApplinX server URL, by default it is applinx://localhost:2323.
String	<a href="#"><u>getServiceName()</u></a> Formally known as host service The connection pool used to get a session.
String	<a href="#"><u>getSessionId()</u></a> The session ID in ApplinX server.
<a href="#"><u>GXVariable</u></a>	<a href="#"><u>getVariable()</u></a> (String varName) Returns a variable used by the framework by the specified variable name.
<a href="#"><u>GXVariable[]</u></a>	<a href="#"><u>getVariables()</u></a> An array of variables to be passed to the server.
boolean	<a href="#"><u>isFailOnWarning()</u></a>
boolean	<a href="#"><u>isShowIntermediateScreen()</u></a> Determines whether to show intermediate host screens.

boolean	<a href="#"><u>isVariablesModified()</u></a> For internal usage
void	<a href="#"><u>removeVariable</u></a> (String varName)
void	<a href="#"><u>set_ApplicationName</u></a> (String applicationName) For .NET framework usage.
void	<a href="#"><u>set_Description</u></a> (String desc) For .NET framework usage.
void	<a href="#"><u>set_FailOnWarning</u></a> (boolean failOnWarning) For .NET framework usage.
void	<a href="#"><u>set_HostNewPassword</u></a> (String hostNewPassword) For .NET framework usage.
void	<a href="#"><u>set_HostPassword</u></a> (String hostPassword) For .NET framework usage.
void	<a href="#"><u>set_HostUserName</u></a> (String hostUserName) For .NET framework usage.
void	<a href="#"><u>set_Password</u></a> (String password) For .NET framework usage.
void	<a href="#"><u>set_ServerURL</u></a> (String serverURL) For .NET framework usage.
void	<a href="#"><u>set_ServiceName</u></a> (String serviceName) For .NET framework usage.
void	<a href="#"><u>set_SessionId</u></a> (String sessionId) For .NET framework usage.
void	<a href="#"><u>set_ShowIntermediateScreen</u></a> (boolean show) For .NET framework usage.
void	<a href="#"><u>set_Variables</u></a> (GXVariable[] variables) For .NET framework usage.
void	<a href="#"><u>setApplicationName</u></a> (String applicationName) The name of the ApplinX application.
void	<a href="#"><u>setDescription</u></a> (String desc) The description of the session in ApplinX server.
void	<a href="#"><u>setFailOnWarning</u></a> (boolean failOnWarning)
void	<a href="#"><u>setHostNewPassword</u></a> (String hostNewPassword)
void	<a href="#"><u>setHostPassword</u></a> (String hostPassword) The host password.
void	<a href="#"><u>setHostUserName</u></a> (String hostUserName) The host user name.

void	<a href="#"><code>setPassword</code></a> (String password) The password required to access a specific session on the ApplinX server.
void	<a href="#"><code>setServerContext</code></a> (GXIApplicationServerContext serverContext) For J2EE usage only.
void	<a href="#"><code>setServerURL</code></a> (String serverURL) The ApplinX server URL, by default it is applinx://localhost:2323.
void	<a href="#"><code>setServiceName</code></a> (String serviceName) Formally known as host service The connection pool used to get a session.
void	<a href="#"><code>setSessionId</code></a> (String sessionId) The Session ID in the web server.
void	<a href="#"><code>setShowIntermediateScreen</code></a> (boolean show) Determine whether to show intermediate host screens.
void	<a href="#"><code>setVariables</code></a> (GXVariable[] variables) An array of variables to be passed to the server.
String	<a href="#"><code>toString</code></a> ()

#### Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXConfig

```
public GXConfig()
```

## Methods

### get\_ServerURL

```
public String get_ServerURL()
```

For .NET framework usage. Expose the method as a property.

### set\_ServerURL

```
public void set_ServerURL(String serverURL)
```

For .NET framework usage. Expose the method as a property.

### getServerURL

```
public String getServerURL()
```

The ApplinX server URL, by default it is applinx://localhost:2323. For secured SSL socket: applinxs://localhost:23443

**Returns:**

(continued from last page)

---

The ApplinX server URL

---

## setServerURL

```
public void setServerURL(String serverURL)
```

The ApplinX server URL, by default it is applinx://localhost:2323. For secured SSL socket: applinxs://localhost:23443

### Parameters:

serverURL - The ApplinX server URL

---

## get\_SessionId

```
public String get_SessionId()
```

For .NET framework usage. Expose the method as a property.

---

## set\_SessionId

```
public void set_SessionId(String sessionId)
```

For .NET framework usage. Expose the method as a property.

---

## getSessionId

```
public String getSessionId()
```

The session ID in ApplinX server.

### Returns:

the session ID in ApplinX server

---

## setSessionId

```
public void setSessionId(String sessionId)
```

The Session ID in the web server. Can be set according to the connecting IP address or by an allocated session ID. It is also possible to set the Web Session ID from the code (in the GXBasicContext (JSP)/GXBasicWebForm (.NET) file), overriding the option selected here.

### Parameters:

sessionId - the session ID in ApplinX server

---

## get\_HostUserName

```
public String get_HostUserName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_HostUserName

```
public void set_HostUserName(String hostUserName)
```

For .NET framework usage. Expose the method as a property.

---

## getHostUserName

```
public String getHostUserName()
```

---

---

(continued from last page)

The host user name for SSH and Natural-Unix protocol only.

**Returns:**

The host user name for SSH and Natural-Unix protocol

---

## setHostUserName

```
public void setHostUserName(String hostUserName)
```

The host user name. Relevant for SSH and Natural-Unix protocol only.

**Parameters:**

hostUserName - the host user name for SSH and Natural-Unix protocol

---

## get\_ApplicationName

```
public String get_ApplicationName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ApplicationName

```
public void set_ApplicationName(String applicationName)
```

For .NET framework usage. Expose the method as a property.

---

## getApplicationName

```
public String getApplicationName()
```

The name of the ApplinX application.

**Returns:**

the name of the ApplinX application

---

## setApplicationName

```
public void setApplicationName(String applicationName)
```

The name of the ApplinX application.

**Parameters:**

applicationName - the name of the ApplinX application

---

## get\_Description

```
public String get_Description()
```

For .NET framework usage. Expose the method as a property.

---

## set\_Description

```
public void set_Description(String desc)
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

---

## getDescription

```
public String getDescription()
```

The description of the session in ApplinX server.

### Returns:

the description of the session in ApplinX server

---

## setDescription

```
public void setDescription(String desc)
```

The description of the session in ApplinX server. This description appears in the list of sessions in ApplinX Administrator.

The information entered can be dynamic and may include tokens such as:

\$(SESSION\_ID) Represents the Web server session ID.

\$(IP) Represents the user IP address.

\$(BY\_APPLINX) Represents the ApplinX session ID.

### Parameters:

desc - the description of the session in ApplinX server

---

## get\_Password

```
public String get_Password()
```

For .NET framework usage. Expose the method as a property.

---

## set\_Password

```
public void set_Password(String password)
```

For .NET framework usage. Expose the method as a property.

---

## getPassword

```
public String getPassword()
```

The password to the session in ApplinX server.

### Returns:

the password to the session in ApplinX server

---

## setPassword

```
public void setPassword(String password)
```

The password required to access a specific session on the ApplinX server. Can be set according to the connecting IP address or by an allocated session ID. It is also possible to set the password from the code (in the GXBasicContext (JSP)/GXBasicWebForm (.NET) file), overriding the option selected here.

### Parameters:

password - the password to the session in ApplinX server

---

## get\_HostPassword

```
public String get_HostPassword()
```

---

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## set\_HostPassword

```
public void set_HostPassword(String hostPassword)
```

For .NET framework usage. Expose the method as a property.

---

## getHostPassword

```
public String getHostPassword()
```

The host password. Relevant for SSH and Natural-Unix protocol only.

### Returns:

the host password for SSH and Natural-Unix protocol only

---

## setHostPassword

```
public void setHostPassword(String hostPassword)
```

The host password. Relevant for SSH and Natural-Unix protocol only.

### Parameters:

hostPassword - the host password for SSH and Natural-Unix protocol only

---

## get\_ServiceName

```
public String get_ServiceName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ServiceName

```
public void set_ServiceName(String serviceName)
```

For .NET framework usage. Expose the method as a property.

---

## getServiceName

```
public String getServiceName()
```

Formally known as host service

The connection pool used to get a session. When a pool is stated, the framework will connect to the pool instead of using a new session.

### Returns:

the connection pool used to get a session

---

## setServiceName

```
public void setServiceName(String serviceName)
```

Formally known as host service

The connection pool used to get a session. When a pool is stated, the framework will connect to the pool instead of using a new session.

### Parameters:

(continued from last page)

serviceName - the connection pool used to get a session

---

## getVariables

```
public GXVariable\[\] getVariables()
```

An array of variables to be passed to the server. For example, changing the host address, before creating a new session or application parameters.

**Returns:**

an array of variables to be passed to the server

**See Also:**

[GXBaseObjectConstants](#)

---

## get\_Variables

```
public GXVariable\[\] get_Variables()
```

For .NET framework usage. Expose the method as a property.

---

## set\_Variables

```
public void set_Variables(GXVariable\[\] variables)
```

For .NET framework usage. Expose the method as a property.

---

## setVariables

```
public void setVariables(GXVariable\[\] variables)
```

An array of variables to be passed to the server. For example, changing the host address, before creating a new session or application parameters. The variables will be used in the internal framework requests to the server.

**Parameters:**

variables - an array of variables to be passed to the server

**See Also:**

[GXBaseObjectConstants](#)

---

## getVariable

```
public GXVariable getVariable(String varName)
```

Returns a variable used by the framework by the specified variable name. The variable will be used in the internal framework requests to the server.

**Returns:**

a variable by the specified variable name

---

## addVariable

```
public void addVariable(GXVariable variable)
```

Adds a variable to the framework configuration The variable will be used in the internal framework requests to the server.

---

(continued from last page)

---

## addVariable

```
public void addVariable(String varName,  
                        String varValue)
```

---

## removeVariable

```
public void removeVariable(String varName)
```

---

## isVariablesModified

```
public boolean isVariablesModified()
```

For internal usage

---

## getServerContext

```
public GXIApplicationServerContext getServerContext()
```

For J2EE usage only. The ApplinX server context when running in local mode.

**Returns:**

the ApplinX server context when running in local mode

---

## setServerContext

```
public void setServerContext(GXIApplicationServerContext serverContext)
```

For J2EE usage only. The ApplinX server context when running in local mode.

**Parameters:**

serverContext - the ApplinX server context when running in local mode

---

## isShowIntermediateScreen

```
public boolean isShowIntermediateScreen()
```

Determines whether to show intermediate host screens. Relevant for Natural Set Control N command

**Returns:**

whether to show host screens

---

## setShowIntermediateScreen

```
public void setShowIntermediateScreen(boolean show)
```

Determine whether to show intermediate host screens. Relevant for Natural Set Control N command.

**Parameters:**

show - whether to show intermediate host screens

---

## get\_ShowIntermediateScreen

```
public boolean get_ShowIntermediateScreen()
```

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **set\_ShowIntermediateScreen**

```
public void set_ShowIntermediateScreen(boolean show)
```

For .NET framework usage. Expose the method as a property.

---

## **clone**

```
public Object clone()  
    throws CloneNotSupportedException
```

---

## **toString**

```
public String toString()
```

---

## **getHostNewPassword**

```
public String getHostNewPassword()
```

---

## **setHostNewPassword**

```
public void setHostNewPassword(String hostNewPassword)
```

---

## **isFailOnWarning**

```
public boolean isFailOnWarning()
```

---

## **setFailOnWarning**

```
public void setFailOnWarning(boolean failOnWarning)
```

---

## **get\_Variable**

```
public GXVariable get_Variable(String varName)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_VariablesModified**

```
public boolean get_VariablesModified()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## **get\_HostNewPassword**

```
public String get_HostNewPassword()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_HostNewPassword**

```
public void set_HostNewPassword(String hostNewPassword)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_FailOnWarning**

```
public boolean get_FailOnWarning()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_FailOnWarning**

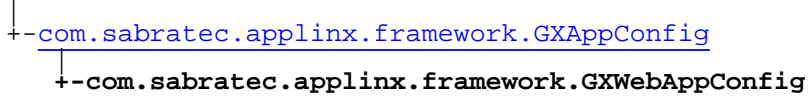
```
public void set_FailOnWarning(boolean failOnWarning)
```

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.framework

### Class GXWebAppConfig

java.lang.Object



#### All Implemented Interfaces:

Cloneable, java.io.Serializable, java.io.Serializable

public class **GXWebAppConfig**

extends [GXAppConfig](#)

implements java.io.Serializable, java.io.Serializable, Cloneable

### Constructor Summary

public	<a href="#">GXWebAppConfig()</a>
--------	----------------------------------

### Method Summary

void	<a href="#">addBindingToTemplatePart</a> (String fileId) Allows perform binding to a template part.
Object	<a href="#">clone</a> ()
boolean	<a href="#">get_addDocTypeToFrameset</a> () For .NET framework usage.
boolean	<a href="#">get_AddJavascript</a> () For .NET framework usage.
boolean	<a href="#">get_AllowTypeAhead</a> () <b>Deprecated.</b> Use <i>gx_container.aspx/jsp emulation frame instead</i>
String[]	<a href="#">get_BoundTemplateParts</a> () For .NET framework usage.
boolean	<a href="#">get_CaptureBrowserClose</a> () For .NET framework usage.
String	<a href="#">get_DocTypeStr</a> () For .NET framework usage.
String	<a href="#">get_DynamicDisableCssClass</a> () For .NET framework usage.
String	<a href="#">get_ExternalCssRequestParamName</a> () For .NET framework usage.
String	<a href="#">get_LogOffForm</a> () For .NET framework usage.

<a href="#">GXNdtConfig</a>	<a href="#">get_NaturalDataTransferConfig()</a> For .NET framework usage.
boolean	<a href="#">get_PromptBeforeSessionTimeOut()</a> For .NET framework usage.
String	<a href="#">get_ScreenLocker()</a> For .NET framework usage.
String	<a href="#">get_ScreenLockerText()</a> For .NET framework usage.
boolean	<a href="#">get_SupportPartialPageRendering()</a> <b>Deprecated.</b> <i>Partial page rendering is always supported</i>
String	<a href="#">get_TerminationPath()</a> For .NET framework usage.
boolean	<a href="#">get_UseCache()</a> <b>Deprecated.</b> <i>For .NET framework usage. Expose the method as a property.</i>
boolean	<a href="#">get_UseCalendar()</a> For .NET framework usage.
boolean	<a href="#">get_UseFramesForSubmit()</a> For .NET framework usage.
boolean	<a href="#">get_UseJavaScriptsAsResource()</a> For .NET framework usage.
boolean	<a href="#">get_UseOldJSEngine()</a> For .NET framework usage.
boolean	<a href="#">get_UseScriptsInPage()</a> For .NET framework usage.
boolean	<a href="#">get_UseStateManager()</a> For .NET framework usage.
String	<a href="#">get_VirtualDir()</a> For .NET framework usage.
String	<a href="#">get_WebWindowName()</a> For .NET framework usage.
boolean	<a href="#">get_WritePerformanceLog()</a> For .NET framework usage.
boolean	<a href="#">getAddJavascript()</a> Adds minimal JavaScript to the web page which support basic browser actions
boolean	<a href="#">getAllowTypeAhead()</a> <b>Deprecated.</b> <i>Use gx_container.aspx/jsp emulation frame instead</i>
String[]	<a href="#">getBoundTemplateParts()</a> The template parts may be added one by one by calling the method addTemplatePart or as a list in this method seperated by "," The 2nd options may be used when GXWebAppConfig is read from XML file

boolean	<a href="#"><u>getCaptureBrowserClose()</u></a> <b>Deprecated.</b> <i>Always performed</i>
String	<a href="#"><u>getDocTypeStr()</u></a> Determines the doc type parameter that should be added to pages.
String	<a href="#"><u>getDynamicDisableCssClass()</u></a> The name of the css class that is added to an input field, which changed to read only by the host according to the <code>reflectHostProtected</code> parameter.
String	<a href="#"><u>getExternalCssRequestParamName()</u></a> Defines an external css request paramater name.
GXIHostKeysTagUserExit	<a href="#"><u>getHostKeysTagUserExit()</u></a> Defines a user exit for host keys tag(JSP) / control (.NET)
String	<a href="#"><u>getLogOffForm()</u></a> Detrmine the logoff page of the web application
<a href="#"><u>GXNdtConfig</u></a>	<a href="#"><u>getNaturalDataTransferConfig()</u></a> Contains the Natural data transfer configuration
String	<a href="#"><u>getScreenLocker()</u></a> The purpose of a screen locker is to indicate to the user by means of a message, that the application is processing his request, and blocks him from interfering with the current process by repressing a button/link or keyboard PF/ENTER.
String	<a href="#"><u>getScreenLockerText()</u></a>
String	<a href="#"><u>getTerminationPath()</u></a> Defines a termination path to use when the user logs off by either closing the browser, or clicking the logoff link (activates logoff.jsp/aspx).
boolean	<a href="#"><u>getUseCache()</u></a> Determines whether to allow the back button in the browser.
boolean	<a href="#"><u>getUseStateManager()</u></a> Not internal usage
String	<a href="#"><u>getVirtualDir()</u></a> Determines the virtual directory used for the project.
String	<a href="#"><u>getWebWindowName()</u></a> Determines the ApplinX web window name.
boolean	<a href="#"><u>getWritePerformanceLog()</u></a> Determines whether to create a performance log. The performance monitor is used for analyzing framework performance.
boolean	<a href="#"><u>isAddDocTypeToFrameset()</u></a> Determines if to add the doc type parameter to the pages.
boolean	<a href="#"><u>isPromptBeforeSessionTimeOut()</u></a> Set to true to display a message indicating to the user that that the session is about to be disconnected.
boolean	<a href="#"><u>isSupportPartialPageRendering()</u></a> <b>Deprecated.</b> <i>Partial page rendering is always supported</i>

boolean	<a href="#"><u>isUseCalendar</u></a> ( )
boolean	<a href="#"><u>isUseFramesForSubmit</u></a> ( ) This feature prevents the displayed page from flickering every time the page is submitted.
boolean	<a href="#"><u>isUseJavaScriptsAsResource</u></a> ( ) For internal use
boolean	<a href="#"><u>isUseOldJSEngine</u></a> ( ) For internal use
boolean	<a href="#"><u>isUseScriptsInPage</u></a> ( ) Determines if to add framework scripts to each page.
void	<a href="#"><u>set_AddDocTypeToFrameset</u></a> (boolean addDocTypeToFrameset) For .NET framework usage.
void	<a href="#"><u>set_AddJavascript</u></a> (boolean addJavascript) For .NET framework usage.
void	<a href="#"><u>set_AllowTypeAhead</u></a> (boolean allowTypeAhead) <b>Deprecated.</b> Use <i>gx_container.aspx/jsp emulation frame instead</i>
void	<a href="#"><u>set_CaptureBrowserClose</u></a> (boolean captureBrowserClose) For .NET framework usage.
void	<a href="#"><u>set_DocTypeStr</u></a> (String docTypeStr) For .NET framework usage.
void	<a href="#"><u>set_DynamicDisableCssClass</u></a> (String dynamicDisableCssClass) For .NET framework usage.
void	<a href="#"><u>set_ExternalCssRequestParamName</u></a> (String externalCssRequestParamName) For .NET framework usage.
void	<a href="#"><u>set_LogOffForm</u></a> (String logOffForm) For .NET framework usage.
void	<a href="#"><u>set_NaturalDataTransferConfig</u></a> ( <a href="#"><u>GXNdtConfig</u></a> naturalDataTransferConfig) For internal use
void	<a href="#"><u>set_PromptBeforeSessionTimeOut</u></a> (boolean promptBeforeSessionTimeOut) For .NET framework usage.
void	<a href="#"><u>set_ScreenLocker</u></a> (String screenLockerPage) For .NET framework usage.
void	<a href="#"><u>set_ScreenLockerText</u></a> (String screenLockerPage) For .NET framework usage.
void	<a href="#"><u>set_SupportPartialPageRendering</u></a> (boolean supportPartialPageRendering1) <b>Deprecated.</b> <i>Partial page rendering is always supported</i>
void	<a href="#"><u>set_TerminationPath</u></a> (String terminationPath) For .NET framework usage.
void	<a href="#"><u>set_UseCache</u></a> (boolean useCache) For .NET framework usage.

void	<a href="#"><u>set_UseCalendar</u></a> (boolean useCalendar) For .NET framework usage.
void	<a href="#"><u>set_UseFramesForSubmit</u></a> (boolean useFramesForSubmit) For .NET framework usage.
void	<a href="#"><u>set_UseJavaScriptsAsResource</u></a> (boolean useJavaScriptsAsResource) For .NET framework usage.
void	<a href="#"><u>set_UseOldJSEngine</u></a> (boolean useOldJSEngine) For .NET framework usage.
void	<a href="#"><u>set_UseScriptsInPage</u></a> (boolean useScriptsInPage) For .NET framework usage.
void	<a href="#"><u>set_UseStateManager</u></a> (boolean useStateManager) For .NET framework usage.
void	<a href="#"><u>set_VirtualDir</u></a> (String virtualDir) For .NET framework usage.
void	<a href="#"><u>set_WebWindowName</u></a> (String webWindowName) For .NET framework usage.
void	<a href="#"><u>set_WritePerformanceLog</u></a> (boolean writePerformanceLog) For .NET framework usage.
void	<a href="#"><u>setAddDocTypeToFrameset</u></a> (boolean addDocTypeToFrameset) Determines if to add the doc type parameter to the pages.
void	<a href="#"><u>setAddJavascript</u></a> (boolean addJavascript) Adds minimal JavaScript to the web page which support basic browser actions
void	<a href="#"><u>setAllowTypeAhead</u></a> (boolean allowTypeAhead) <b>Deprecated.</b> Use <i>gx_container.aspx/jsp emulation frame instead</i>
void	<a href="#"><u>setCaptureBrowserClose</u></a> (boolean captureBrowserClose) <b>Deprecated.</b> Always performed
void	<a href="#"><u>setDocTypeStr</u></a> (String docTypeStr) Determines the doc type parameter that should be added to pages.
void	<a href="#"><u>setDynamicDisableCssClass</u></a> (String dynamicDisableCssClass) The css class that is used to give a different look to a read-only input field, which becomes read-only according to the value of the ReflectHostProtected parameter.
void	<a href="#"><u>setExternalCssRequestParamName</u></a> (String externalCssRequestParamName) Defines an external css request paramater name.
void	<a href="#"><u>setHostKeysTagUserExit</u></a> (GXIHostKeysTagUserExit hostKeysTagUserExit) Defines a user exit for host keys tag(JSP) / control (.NET)
void	<a href="#"><u>setLogOffForm</u></a> (String logOffForm) Detrmine the logoff page of the web application
void	<a href="#"><u>setNaturalDataTransferConfig</u></a> ( <a href="#"><u>GXNdtConfig</u></a> naturalDataTransferConfig) For internal use

void	<a href="#"><u>setPromptBeforeSessionTimeout</u></a> (boolean promptBeforeSessionTimeout) Set to true to display a message indicating to the user that that the session is about to be disconnected.
void	<a href="#"><u>setScreenLocker</u></a> (String screenLockerPage) The purpose of a screen locker is to indicate to the user by means of a message, that the application is processing his request, and blocks him from interfering with the current process by repressing a button/link or keyboard PF/ENTER.
void	<a href="#"><u>setScreenLockerText</u></a> (String screenLockerText)
void	<a href="#"><u>setSupportPartialPageRendering</u></a> (boolean supportPartialPageRendering1) <b>Deprecated.</b> <i>Partial page rendering is always supported</i>
void	<a href="#"><u>setTerminationPath</u></a> (String terminationPath) Defines a termination path to use when the user logs off by either closing the browser, or clicking the logoff link (activates logoff.jsp/asp).
void	<a href="#"><u>setUseCache</u></a> (boolean useCache) Determines whether to allow the back button in the browser.
void	<a href="#"><u>setUseCalendar</u></a> (boolean useCalendar)
void	<a href="#"><u>setUseFramesForSubmit</u></a> (boolean useFramesForSubmit) This feature prevents the displayed page from flickering every time the page is submitted.
void	<a href="#"><u>setUseJavaScriptsAsResource</u></a> (boolean useJavaScriptsAsResource) For internal use
void	<a href="#"><u>setUseOldJSEngine</u></a> (boolean useOldJSEngine) For internal use
void	<a href="#"><u>setUseScriptsInPage</u></a> (boolean useScriptsInPage) Determines if to add framework scripts to each page.
void	<a href="#"><u>setUseStateManager</u></a> (boolean useStateManager) Not internal usage
void	<a href="#"><u>setVirtualDir</u></a> (String virtualDir) Determines the virtual directory used for the project.
void	<a href="#"><u>setWebWindowName</u></a> (String webWindowName) Determines the ApplinX web window name.
void	<a href="#"><u>setWritePerformanceLog</u></a> (boolean writePerformanceLog) Determines whether to create a performance log. The performance monitor is used for analyzing framework performance.

Methods inherited from class [com.sabratec.applinx.framework.GXAppConfig](#)

[get\\_Build](#), [get\\_CaptureTagsEvents](#), [get\\_DesignMode](#), [get\\_ErrorForm](#),  
[get\\_FieldsModification](#), [get\\_FieldTypesInUse](#), [get\\_FtpConfig](#), [get\\_IndexForm](#),  
[get\\_Instant](#), [get\\_InstantConfig](#), [get\\_InstantForm](#), [get\\_InstantWinForm](#), [get\\_IsInstant](#),  
[get\\_MacroConfig](#), [get\\_MapPath](#), [get\\_PerformSyncCheckInInstant](#), [get\\_ProceduresConfig](#),  
[get\\_ReflectBGColors](#), [get\\_ReflectEmulationBehaviour](#), [get\\_ReflectFGColors](#),  
[get\\_ReflectHostCursor](#), [get\\_ReflectHostProtected](#), [get\\_ReflectMaxlength](#),  
[get\\_SessionConfig](#), [get\\_SingleSignOnConfig](#), [get\\_SsoConfig](#), [get\\_SupportDupAndFieldMark](#),  
[get\\_TableBuildConfig](#), [get\\_TrimLabels](#), [get\\_UseFolders](#), [get\\_UseKeyboardPFKeys](#),  
[get\\_UseModalWindows](#), [get\\_UseScreenGroups](#), [get\\_Version](#), [getBuild](#),  
[getCaptureTagsEvents](#), [getDesignMode](#), [getErrorForm](#), [getFieldsModification](#),  
[getFieldTypesInUse](#), [getFtpConfig](#), [getIndexForm](#), [getInstantConfig](#), [getInstantForm](#),  
[getInstantWinForm](#), [getMacroConfig](#), [getMapPath](#), [getPerformSyncCheckInInstant](#),  
[getProceduresConfig](#), [getReflectBGColors](#), [getReflectEmulationBehaviour](#),  
[getReflectFGColors](#), [getReflectHostCursor](#), [getReflectHostProtected](#),  
[getReflectMaxlength](#), [getSessionConfig](#), [getSingleSignOnConfig](#),  
[getSupportDupAndFieldMark](#), [getTableBuildConfig](#), [getTrimLabels](#), [getUseFolders](#),  
[getUseKeyboardPFKeys](#), [getUseModalWindows](#), [getUseScreenGroups](#), [getVersion](#), [isInstant](#),  
[set\\_CaptureTagsEvents](#), [set\\_DesignMode](#), [set\\_ErrorForm](#), [set\\_FieldsModification](#),  
[set\\_FieldTypesInUse](#), [set\\_FtpConfig](#), [set\\_IndexForm](#), [set\\_Instant](#), [set\\_InstantConfig](#),  
[set\\_InstantForm](#), [set\\_InstantWinForm](#), [set\\_IsInstant](#), [set\\_MacroConfig](#), [set\\_MapPath](#),  
[set\\_PerformSyncCheckInInstant](#), [set\\_ProceduresConfig](#), [set\\_ReflectBGColors](#),  
[set\\_ReflectEmulationBehaviour](#), [set\\_ReflectFGColors](#), [set\\_ReflectHostCursor](#),  
[set\\_ReflectHostProtected](#), [set\\_ReflectMaxlength](#), [set\\_SessionConfig](#),  
[set\\_SingleSignOnConfig](#), [set\\_SsoConfig](#), [set\\_SupportDupAndFieldMark](#),  
[set\\_TableBuildConfig](#), [set\\_TrimLabels](#), [set\\_UseFolders](#), [set\\_UseKeyboardPFKeys](#),  
[set\\_UseModalWindows](#), [set\\_UseScreenGroups](#), [setCaptureTagsEvents](#), [setDesignMode](#),  
[setErrorForm](#), [setFieldsModification](#), [setFieldTypesInUse](#), [setFtpConfig](#), [setIndexForm](#),  
[setInstantConfig](#), [setInstantForm](#), [setInstantWinForm](#), [setIsInstant](#), [setMacroConfig](#),  
[setMapPath](#), [setPerformSyncCheckInInstant](#), [setProceduresConfig](#), [setReflectBGColors](#),  
[setReflectEmulationBehaviour](#), [setReflectFGColors](#), [setReflectHostCursor](#),  
[setReflectHostProtected](#), [setReflectMaxlength](#), [setSessionConfig](#),  
[setSingleSignOnConfig](#), [setSupportDupAndFieldMark](#), [setTableBuildConfig](#), [setTrimLabels](#),  
[setUseFolders](#), [setUseKeyboardPFKeys](#), [setUseModalWindows](#), [setUseScreenGroups](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXWebAppConfig

```
public GXWebAppConfig()
```

## Methods

### get\_UseCache

```
public boolean get_UseCache()
```

**Deprecated.** For .NET framework usage. Expose the method as a property.

---

## set\_UseCache

```
public void set_UseCache(boolean useCache)
```

For .NET framework usage. Expose the method as a property.

---

## getUseCache

```
public boolean getUseCache()
```

Determines whether to allow the back button in the browser. If it is false, pressing back will get the previous page from the browser cache.

**Returns:**

whether to allow the back button in the browser

---

## setUseCache

```
public void setUseCache(boolean useCache)
```

Determines whether to allow the back button in the browser. If it is false, pressing back will get the previous page from the browser cache.

**Parameters:**

useCache - whether to allow the back button in the browser

---

## get\_VirtualDir

```
public String get_VirtualDir()
```

For .NET framework usage. Expose the method as a property.

---

## set\_VirtualDir

```
public void set_VirtualDir(String virtualDir)
```

For .NET framework usage. Expose the method as a property.

---

## getVirtualDir

```
public String getVirtualDir()
```

Determines the virtual directory used for the project. Should be without any slashes. Needs to be set only if useFolders is true. For a site the value should be empty string.

**Returns:**

the virtual directory used for the project

---

## setVirtualDir

```
public void setVirtualDir(String virtualDir)
```

Determines the virtual directory used for the project. Should be without any slashes. Needs to be set only if useFolders is true. For a site the value should be empty string.

**Parameters:**

virtualDir - the virtual directory used for the project

---

---

## get\_AllowTypeAhead

```
public boolean get_AllowTypeAhead()
```

**Deprecated.** Use *gx\_container.aspx/jsp emulation frame* instead

---

## set\_AllowTypeAhead

```
public void set_AllowTypeAhead(boolean allowTypeAhead)
```

**Deprecated.** Use *gx\_container.aspx/jsp emulation frame* instead

---

## getAllowTypeAhead

```
public boolean getAllowTypeAhead()
```

**Deprecated.** Use *gx\_container.aspx/jsp emulation frame* instead

---

## setAllowTypeAhead

```
public void setAllowTypeAhead(boolean allowTypeAhead)
```

**Deprecated.** Use *gx\_container.aspx/jsp emulation frame* instead

---

## get\_LogOffForm

```
public String get_LogOffForm()
```

For .NET framework usage. Expose the method as a property.

---

## set\_LogOffForm

```
public void set_LogOffForm(String logOffForm)
```

For .NET framework usage. Expose the method as a property.

---

## getLogOffForm

```
public String getLogOffForm()
```

Determine the logoff page of the web application

**Returns:**

the logoff page of the web application

---

## setLogOffForm

```
public void setLogOffForm(String logOffForm)
```

Determine the logoff page of the web application

---

(continued from last page)

**Parameters:**

logOffForm - the logoff page of the web application

---

**get\_UseStateManager**

```
public boolean get_UseStateManager()
```

For .NET framework usage. Expose the method as a property.

---

**set\_UseStateManager**

```
public void set_UseStateManager(boolean useStateManager)
```

For .NET framework usage. Expose the method as a property.

---

**getUseStateManager**

```
public boolean getUseStateManager()
```

Not internal usage

---

**setUseStateManager**

```
public void setUseStateManager(boolean useStateManager)
```

Not internal usage

---

**get\_ScreenLocker**

```
public String get_ScreenLocker()
```

For .NET framework usage. Expose the method as a property.

---

**set\_ScreenLocker**

```
public void set_ScreenLocker(String screenLockerPage)
```

For .NET framework usage. Expose the method as a property.

---

**getScreenLocker**

```
public String getScreenLocker()
```

The purpose of a screen locker is to indicate to the user by means of a message, that the application is processing his request, and blocks him from interfering with the current process by repressing a button/link or keyboard PF/ENTER. To activate the screen locker first select the Use screen locker check box and then access the template/screenLocker.htm file and use the width/height percentages to control the location of the message. Replace the text "Please wait" with an alternative text/image as required.

Refer to the documentation for further details.

**Returns:**

the screen locker file name

---

**setScreenLocker**

```
public void setScreenLocker(String screenLockerPage)
```

(continued from last page)

The purpose of a screen locker is to indicate to the user by means of a message, that the application is processing his request, and blocks him from interfering with the current process by repressing a button/link or keyboard PF/ENTER. To activate the screen locker first select the Use screen locker check box and then access the template/screenLocker.htm file and use the width/height percentages to control the location of the message. Replace the text "Please wait" with an alternative text/image as required.

Refer to the documentation for further details.

**Parameters:**

screenLockerPage - the screen locker file name.

---

**getScreenLockerText**

```
public String getScreenLockerText()
```

---

**setScreenLockerText**

```
public void setScreenLockerText(String screenLockerText)
```

---

**get\_DynamicDisableCssClass**

```
public String get_DynamicDisableCssClass()
```

For .NET framework usage. Expose the method as a property.

---

**set\_DynamicDisableCssClass**

```
public void set_DynamicDisableCssClass(String dynamicDisableCssClass)
```

For .NET framework usage. Expose the method as a property.

---

**getDynamicDisableCssClass**

```
public String getDynamicDisableCssClass()
```

The name of the css class that is added to an input field, which changed to read only by the host according to the reflectHostProtected parameter. Used for giving a different look to this type of input field.

**Returns:**

the name of the css class that is added to an input field which is read only in the current screen

---

**setDynamicDisableCssClass**

```
public void setDynamicDisableCssClass(String dynamicDisableCssClass)
```

The css class that is used to give a different look to a read-only input field, which becomes read-only according to the value of the ReflectHostProtected parameter.

**Parameters:**

dynamicDisableCssClass - the name of the css class that is added to an input field

---

**get\_CaptureBrowserClose**

```
public boolean get_CaptureBrowserClose()
```

For .NET framework usage. Expose the method as a property.

---

---

## set\_CaptureBrowserClose

```
public void set_CaptureBrowserClose(boolean captureBrowserClose)
```

For .NET framework usage. Expose the method as a property.

---

## getCaptureBrowserClose

```
public boolean getCaptureBrowserClose()
```

**Deprecated.** *Always performed*

When inheriting from GXBasicWebForm (.NET) / GXBasicContext (JSP), ApplinX full JavaScript is not added to the page. This causes the page not to have the ability to capture the browser close event in the client side configuration. In this case, this parameter adds the functionality of capturing the browser close and closing the host session.

**Returns:**

whether the framework is capturing the browser close and closing the host session

---

## setCaptureBrowserClose

```
public void setCaptureBrowserClose(boolean captureBrowserClose)
```

**Deprecated.** *Always performed*

When inheriting from GXBasicWebForm (.NET) / GXBasicContext (JSP), ApplinX full JavaScript is not added to the page. This causes the page not to have the ability to capture the browser close event in the client side configuration. In this case, this parameter adds the functionality of capturing the browser close and closing the host session.

**Parameters:**

captureBrowserClose - whether the framework is capturing the browser close and closing the host session

---

## get\_AddJavascript

```
public boolean get_AddJavascript()
```

For .NET framework usage. Expose the method as a property.

---

## set\_AddJavascript

```
public void set_AddJavascript(boolean addJavascript)
```

For .NET framework usage. Expose the method as a property.

---

## getAddJavascript

```
public boolean getAddJavascript()
```

Adds minimal JavaScript to the web page which support basic browser actions

**Returns:**

whether minimal JavaScript is added to the web page

---

## setAddJavascript

```
public void setAddJavascript(boolean addJavascript)
```

Adds minimal JavaScript to the web page which support basic browser actions

---

(continued from last page)

**Parameters:**

addJavascript - whether minimal JavaScript is added to the web page

---

**get\_WritePerformanceLog**

```
public boolean get_WritePerformanceLog()
```

For .NET framework usage. Expose the method as a property.

---

**set\_WritePerformanceLog**

```
public void set_WritePerformanceLog(boolean writePerformanceLog)
```

For .NET framework usage. Expose the method as a property.

---

**getWritePerformanceLog**

```
public boolean getWritePerformanceLog()
```

Determines whether to create a performance log.

The performance monitor is used for analyzing framework performance.

It writes log comments for every framework request.

A performance log line contains the following data: Session Id, Screen Name, total request time, total server side divided into: attach, sendKeys, instant generation, detach, total page client load time.

**Returns:**

whether to create a performance log

---

**setWritePerformanceLog**

```
public void setWritePerformanceLog(boolean writePerformanceLog)
```

Determines whether to create a performance log.

The performance monitor is used for analyzing framework performance.

It writes log comments for every framework request.

A performance log line contains the following data: Session Id, Screen Name, total request time, total server side divided into: attach, sendKeys, instant generation, detach, total page client load time.

**Parameters:**

writePerformanceLog - whether to create a performance log

---

**addBindingToTemplatePart**

```
public void addBindingToTemplatePart(String fileId)
```

Allows perform binding to a template part.

**Parameters:**

fileId - The name of the include file to bind to the current screen. "pageheader" for example.

---

**getBoundTemplateParts**

```
public String[] getBoundTemplateParts()
```

The template parts may be added one by one by calling the method addTemplatePart or as a list in this method seperated by "," The 2nd options may be used when GXWebAppConfig is read from XML file

**Returns:**

---

## getExternalCssRequestParamName

```
public String getExternalCssRequestParamName()
```

Defines an external css request paramater name. Used for portal integration. A portal application may provide to the framework as a query string a css URL as follows: http://?cssurl= ("cssurl" will be the value of the field in this case)

**Returns:**

an external css request paramater name

---

## setExternalCssRequestParamName

```
public void setExternalCssRequestParamName(String externalCssRequestParamName)
```

Defines an external css request paramater name. Used for portal integration. A portal application may provide to the framework as a query string a css URL as follows: http://?cssurl= ("cssurl" will be the value of the field in this case)

**Parameters:**

externalCssRequestParamName

---

## get\_ExternalCssRequestParamName

```
public String get_ExternalCssRequestParamName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ExternalCssRequestParamName

```
public void set_ExternalCssRequestParamName(String externalCssRequestParamName)
```

For .NET framework usage. Expose the method as a property.

---

## get\_NaturalDataTransferConfig

```
public GXNdtConfig get_NaturalDataTransferConfig()
```

For .NET framework usage. Expose the method as a property.

---

## getNaturalDataTransferConfig

```
public GXNdtConfig getNaturalDataTransferConfig()
```

Contains the Natural data transfer configuration

**Returns:**

the Natural data transfer configuration

---

## set\_NaturalDataTransferConfig

```
public void set_NaturalDataTransferConfig(GXNdtConfig naturalDataTransferConfig)
```

For internal use

---

## setNaturalDataTransferConfig

```
public void setNaturalDataTransferConfig(GXNdtConfig naturalDataTransferConfig)
```

For internal use

---

---

## clone

```
public Object clone()
```

---

## isUseOldJSEngine

```
public boolean isUseOldJSEngine()
```

For internal use

---

## setUseOldJSEngine

```
public void setUseOldJSEngine(boolean useOldJSEngine)
```

For internal use

---

## isUseJavaScriptsAsResource

```
public boolean isUseJavaScriptsAsResource()
```

For internal use

---

## setUseJavaScriptsAsResource

```
public void setUseJavaScriptsAsResource(boolean useJavaScriptsAsResource)
```

For internal use

---

## get\_UseJavaScriptsAsResource

```
public boolean get_UseJavaScriptsAsResource()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseJavaScriptsAsResource

```
public void set_UseJavaScriptsAsResource(boolean useJavaScriptsAsResource)
```

For .NET framework usage. Expose the method as a property.

---

## isUseScriptsInPage

```
public boolean isUseScriptsInPage()
```

Determines if to add framework scripts to each page. Otherwise the application will work from an z\_container.aspx/jsp frameset

**Returns:**

is adding framework scripts to each page

---

## setUseScriptsInPage

```
public void setUseScriptsInPage(boolean useScriptsInPage)
```

(continued from last page)

Determines if to add framework scripts to each page. Otherwise the application will work from an z\_container.aspx/jsp frameset

**Parameters:**

useScriptsInPage - if to add framework scripts to each page

---

## get\_UseScriptsInPage

```
public boolean get_UseScriptsInPage()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseScriptsInPage

```
public void set_UseScriptsInPage(boolean useScriptsInPage)
```

For .NET framework usage. Expose the method as a property.

---

## isUseFramesForSubmit

```
public boolean isUseFramesForSubmit()
```

This feature prevents the displayed page from flickering every time the page is submitted. This affect is prevented by using two separate frames, while only one is visible at each stage. These frames continuously switch between being active and passive each time a page is submitted. The active frame is displayed, and every time a page is submitted, the frame becomes the passive frame, in this way preventing the flickering affect. When the modal window feature is enabled in the Framework Configuration Editor or the 'Enable Natural-Data-transfer support' is enabled in the host, this feature is automatically enabled even if this parameter is not selected.

---

## setUseFramesForSubmit

```
public void setUseFramesForSubmit(boolean useFramesForSubmit)
```

This feature prevents the displayed page from flickering every time the page is submitted. This affect is prevented by using two separate frames, while only one is visible at each stage. These frames continuously switch between being active and passive each time a page is submitted. The active frame is displayed, and every time a page is submitted, the frame becomes the passive frame, in this way preventing the flickering affect. When the modal window feature is enabled in the Framework Configuration Editor or the 'Enable Natural-Data-transfer support' is enabled in the host, this feature is automatically enabled even if this parameter is not selected.

**Parameters:**

useFramesForSubmit

---

## isUseCalendar

```
public boolean isUseCalendar()
```

---

## setUseCalendar

```
public void setUseCalendar(boolean useCalendar)
```

---

## isPromptBeforeSessionTimeOut

```
public boolean isPromptBeforeSessionTimeOut()
```

---

(continued from last page)

Set to true to display a message indicating to the user that that the session is about to be disconnected. This message will be displayed towards the end of the Non-activity timeout period (defined in the Application Properties Host tab). The user will be able to select to resume or quit the session. When not selecting either of these options, the user will be logged off automatically. It is possible to customize the relevant prompt page (template/logoffPrompt.htm).

**Returns:**

isPromptBeforeSessionTimeOut

---

## setPromptBeforeSessionTimeOut

```
public void setPromptBeforeSessionTimeOut(boolean promptBeforeSessionTimeOut)
```

Set to true to display a message indicating to the user that that the session is about to be disconnected. This message will be displayed towards the end of the Non-activity timeout period (defined in the Application Properties Host tab). The user will be able to select to resume or quit the session. When not selecting either of these options, the user will be logged off automatically. It is possible to customize the relevant prompt page (template/logoffPrompt.htm).

**Parameters:**

promptBeforeSessionTimeOut

---

## get\_PromptBeforeSessionTimeOut

```
public boolean get_PromptBeforeSessionTimeOut()
```

For .NET framework usage. Expose the method as a property.

---

## set\_PromptBeforeSessionTimeOut

```
public void set_PromptBeforeSessionTimeOut(boolean promptBeforeSessionTimeOut)
```

For .NET framework usage. Expose the method as a property.

---

## getTerminationPath

```
public String getTerminationPath()
```

Defines a termination path to use when the user logs off by either closing the browser, or clicking the logoff link (activates logoff.jsp/aspx).

**Returns:**

termination path

---

## setTerminationPath

```
public void setTerminationPath(String terminationPath)
```

Defines a termination path to use when the user logs off by either closing the browser, or clicking the logoff link (activates logoff.jsp/aspx).

**Parameters:**

terminationPath

---

## getHostKeysTagUserExit

```
public GXIHostKeysTagUserExit getHostKeysTagUserExit()
```

Defines a user exit for host keys tag(JSP) / control (.NET)

**Returns:**

## setHostKeysTagUserExit

```
public void setHostKeysTagUserExit(GXIHostKeysTagUserExit hostKeysTagUserExit)
```

Defines a user exit for host keys tag(JSP) / control (.NET)

**Parameters:**

HostKeysTagUserExit

---

## isSupportPartialPageRendering

```
public boolean isSupportPartialPageRendering()
```

**Deprecated.** *Partial page rendering is always supported*

Partial Page Rendering (PPR) is a feature that allows a part of a page to be redrawn rather than having to reload the entire page. You can add server side buttons/links and perform logic on the target server function, and as a result update only a part of the page. Typical use: Paging, sorting, collecting aggregated data from a few screens into the same web page, and more. To implement the feature first select the Support Partial Page Rendering check box and then add a button/link which performs a server side action. Create a wrapping tag (div) to the web page, in the area you want to update and add the required JavaScript code to the link/button (Refer to the documentation for details).

**Returns:**

---

## setSupportPartialPageRendering

```
public void setSupportPartialPageRendering(boolean supportPartialPageRendering1)
```

**Deprecated.** *Partial page rendering is always supported*

Partial Page Rendering (PPR) is a feature that allows a part of a page to be redrawn rather than having to reload the entire page. You can add server side buttons/links and perform logic on the target server function, and as a result update only a part of the page. Typical use: Paging, sorting, collecting aggregated data from a few screens into the same web page, and more. To implement the feature first select the Support Partial Page Rendering check box and then add a button/link which performs a server side action. Create a wrapping tag (div) to the web page, in the area you want to update and add the required JavaScript code to the link/button (Refer to the documentation for details).

**Parameters:**

supportPartialPageRendering1

---

## getWebWindowName

```
public String getWebWindowName()
```

Determines the ApplinX web window name. Useful for activating the web application from an external link which targets to this given name, and avoids opening multiple browsers from this link

**Returns:**

window name

---

## setWebWindowName

```
public void setWebWindowName(String webWindowName)
```

Determines the ApplinX web window name. Useful for activating the web application from an external link which targets to this given name, and avoids opening multiple browsers from this link

**Parameters:**

webWindowName

---

---

## **get\_ScreenLockerText**

```
public String get_ScreenLockerText()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_ScreenLockerText**

```
public void set_ScreenLockerText(String screenLockerPage)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_BoundTemplateParts**

```
public String[] get_BoundTemplateParts()
```

For .NET framework usage. Expose the method as a property.

---

## **get\_UseOldJSEngine**

```
public boolean get_UseOldJSEngine()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_UseOldJSEngine**

```
public void set_UseOldJSEngine(boolean useOldJSEngine)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_UseFramesForSubmit**

```
public boolean get_UseFramesForSubmit()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_UseFramesForSubmit**

```
public void set_UseFramesForSubmit(boolean useFramesForSubmit)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_UseCalendar**

```
public boolean get_UseCalendar()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_UseCalendar**

```
public void set_UseCalendar(boolean useCalendar)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_TerminationPath**

```
public String get_TerminationPath()
```

---

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## set\_TerminationPath

```
public void set_TerminationPath(String terminationPath)
```

For .NET framework usage. Expose the method as a property.

---

## get\_SupportPartialPageRendering

```
public boolean get_SupportPartialPageRendering()
```

**Deprecated.** *Partial page rendering is always supported*

For .NET framework usage. Expose the method as a property.

---

## set\_SupportPartialPageRendering

```
public void set_SupportPartialPageRendering(boolean supportPartialPageRendering1)
```

**Deprecated.** *Partial page rendering is always supported*

For .NET framework usage. Expose the method as a property.

---

## get\_WebWindowName

```
public String get_WebWindowName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_WebWindowName

```
public void set_WebWindowName(String webWindowName)
```

For .NET framework usage. Expose the method as a property.

---

## setAddDocTypeToFrameset

```
public void setAddDocTypeToFrameset(boolean addDocTypeToFrameset)
```

Determines if to add the doc type parameter to the pages.

**Parameters:**

addDocTypeToFrameset

---

## isAddDocTypeToFrameset

```
public boolean isAddDocTypeToFrameset()
```

Determines if to add the doc type parameter to the pages.

**Returns:**

whether to add a doc type to frame set or not.

---

## set\_AddDocTypeToFrameset

```
public void set_AddDocTypeToFrameset(boolean addDocTypeToFrameset)
```

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **get\_addDocTypeToFrameset**

```
public boolean get_addDocTypeToFrameset()
```

For .NET framework usage. Expose the method as a property.

---

## **setDocTypeStr**

```
public void setDocTypeStr(String docTypeStr)
```

Determines the doc type parameter that should be added to pages. The value should be exactly as it should be saved in the html. example .

### **Parameters:**

docTypeStr

---

## **getDocTypeStr**

```
public String getDocTypeStr()
```

Determines the doc type parameter that should be added to pages. The value should be exactly as it should be saved in the html. example .

### **Returns:**

the doc type parameter.

---

## **set\_DocTypeStr**

```
public void set_DocTypeStr(String docTypeStr)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_DocTypeStr**

```
public String get_DocTypeStr()
```

For .NET framework usage. Expose the method as a property.

---

---

**Package**

**com.sabratec.applinx.framework.config.  
doc**

## com.sabratec.applinx.framework.config.doc Class GXLogNode

java.lang.Object

└--com.sabratec.applinx.framework.config.doc.GXLogNode

public class **GXLogNode**  
extends Object

The Framework Log lists the errors and exceptions and can then be used to debug the framework.

### Field Summary

public	<a href="#">AppendFile</a> This parameter determines whether to start the log in a new file or to append the log to the existing file.
public	<a href="#">FileName</a> The framework's log is written to this file.
public	<a href="#">LogHistory</a> The number of backups saved before overwriting the old log files.
public	<a href="#">LogLevel</a> The contents of the log file are as detailed as this property defines, where every level includes the levels above it.
public	<a href="#">MaxFileSize</a> A new log file will be started after the current file has been filled to the maximum file size.

### Constructor Summary

public	<a href="#">GXLogNode</a> ( )
--------	-------------------------------

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### FileName

public java.lang.String **FileName**

The framework's log is written to this file.

#### AppendFile

public java.lang.String **AppendFile**

This parameter determines whether to start the log in a new file or to append the log to the existing file.

---

## LogLevel

```
public java.lang.String LogLevel
```

The contents of the log file are as detailed as this property defines, where every level includes the levels above it. For example, the Debug level also logs Normal, Warnings and Errors Only levels. Available values: Normal, Warnings, Errors only, Debug (by default Normal is selected).

---

## LogHistory

```
public java.lang.String LogHistory
```

The number of backups saved before overwriting the old log files.  
For example: 10 means save the last 10 log files, in addition to the current one, then start to overwrite. Default value is 10.

---

## MaxFileSize

```
public java.lang.String MaxFileSize
```

A new log file will be started after the current file has been filled to the maximum file size.

## Constructors

### GXLogNode

```
public GXLogNode()
```

## com.sabratec.applinx.framework.config.doc

### Class GXPerformanceMonitorNode

```
java.lang.Object
```

```
└--com.sabratec.applinx.framework.config.doc.GXPerformanceMonitorNode
```

```
public class GXPerformanceMonitorNode
extends Object
```

The Performance Monitor monitors user actions and is used for analyzing framework performance. It includes log comments for every framework request. A performance log line contains the following data: Session Id, Screen Name, total request time, total server side divided into: attach, sendKeys, instant generation, detach, total page client load time. Note: Framework performance is recorded only for Instant pages and for generated pages that include PF/ENTER keys. Custom code mapped to server side buttons/links cannot be monitored. The Performance monitor is typically used in Web enabled applications.

#### Field Summary

public	<a href="#"><u>FileName</u></a> The name of the file where the performance log will be saved.
public	<a href="#"><u>PerSessionID</u></a> Determines whether separate files will be saved for each session.
public	<a href="#"><u>ShortLongDescription</u></a> Determines whether to write a short description, or a detailed description.
public	<a href="#"><u>TraceSessions</u></a> Determines whether to trace all sessions or only a specific session.
public	<a href="#"><u>TraceSessionsIDs</u></a> Enter the IDs of the specific sessions that you wish to trace.

#### Constructor Summary

public	<a href="#"><u>GXPerformanceMonitorNode()</u></a>
--------	---

#### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

#### Fields

##### FileName

```
public java.lang.String FileName
```

The name of the file where the performance log will be saved.

##### PerSessionID

```
public java.lang.String PerSessionID
```

(continued from last page)

Determines whether separate files will be saved for each session.

---

## TraceSessions

```
public java.lang.String TraceSessions
```

Determines whether to trace all sessions or only a specific session.

---

## TraceSessionsIDs

```
public java.lang.String TraceSessionsIDs
```

Enter the IDs of the specific sessions that you wish to trace. Separate the session IDs by a comma.

---

## ShortLongDescription

```
public java.lang.String ShortLongDescription
```

Determines whether to write a short description, or a detailed description. A short description means that the information is displayed in a tabular structure. A detailed description displays the information as a paragraph.

## Constructors

## GXPerformanceMonitorNode

```
public GXPerformanceMonitorNode()
```

---

**Package**

**com.sabratec.applinx.framework.emulation.ftp.dialog**

## com.sabratec.applinx.framework.emulation.ftp.dialog Class GXFtpConfig

java.lang.Object

└--com.sabratec.applinx.framework.emulation.ftp.dialog.GXFtpConfig

### All Implemented Interfaces:

java.io.Serializable

```
public class GXFtpConfig
extends Object
implements java.io.Serializable
```

In ApplinX framework, it is possible to transfer files from the client to the host or from the host to the client, using the FTP dialog screens. The HTML emulation contains a link in the footer that opens an FTP Web dialog box. To upload/download files using the FTP option, you need to configure the following parameters.

Refer to the ApplinX Frameworks Developer's Guide for details regarding how to implement this feature.

### Field Summary

public static final	<a href="#">AS400</a> Value: <b>2</b>
public static final	<a href="#">BY_APPLINX</a> Value: <b>999</b>
public static final	<a href="#">MAIN_FRAME</a> Value: <b>1</b>
public static final	<a href="#">SIMPLE</a> Value: <b>0</b>

### Constructor Summary

public	<a href="#">GXFtpConfig()</a>
--------	-------------------------------

### Method Summary

String	<a href="#">get_HostAddress()</a> For .NET framework usage.
String	<a href="#">get_HostType()</a> For .NET framework usage.
String	<a href="#">getHostAddress()</a>
String	<a href="#">getHostType()</a>

void	<a href="#"><code>set_HostAddress</code></a> (String hostAddress) For .NET framework usage.
void	<a href="#"><code>set_HostType</code></a> (String hostType) For .NET framework usage.
void	<a href="#"><code>setHostAddress</code></a> (String hostAddress) Sets the IP address of the remote FTP host.
void	<a href="#"><code>setHostType</code></a> (String hostType) Sets the type of the remote FTP host.

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### **SIMPLE**

```
public static final java.lang.String SIMPLE
```

Constant value: **0**

### **MAIN\_FRAME**

```
public static final java.lang.String MAIN_FRAME
```

Constant value: **1**

### **AS400**

```
public static final java.lang.String AS400
```

Constant value: **2**

### **BY\_APPLINX**

```
public static final java.lang.String BY_APPLINX
```

Constant value: **999**

## Constructors

### **GXFtpConfig**

```
public GXFtpConfig()
```

## Methods

(continued from last page)

---

## getHostType

```
public String getHostType()
```

---

## setHostType

```
public void setHostType(String hostType)
```

Sets the type of the remote FTP host.  
Possible values are:

- Mainframe
- AS400
- Other - for any other kind
- By Applinx - when the FTP host is the same as the host used in the current ApplinX session.

### Parameters:

hostType

---

## get\_HostType

```
public String get_HostType()
```

For .NET framework usage. Expose the method as a property.

---

## set\_HostType

```
public void set_HostType(String hostType)
```

For .NET framework usage. Expose the method as a property.

---

## getHostAddress

```
public String getHostAddress()
```

---

## setHostAddress

```
public void setHostAddress(String hostAddress)
```

Sets the IP address of the remote FTP host.

### Parameters:

hostAddress

---

## get\_HostAddress

```
public String get_HostAddress()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## **set\_HostAddress**

```
public void set_HostAddress(String hostAddress)
```

For .NET framework usage. Expose the method as a property.

---

**Package**

**com.sabratec.applinx.framework.emulation.macros**

## com.sabratec.applinx.framework.emulation.macros

### Class GXMacroConfig

java.lang.Object

└─com.sabratec.applinx.framework.emulation.macros.GXMacroConfig

#### All Implemented Interfaces:

java.io.Serializable

public class **GXMacroConfig**  
 extends Object  
 implements java.io.Serializable

Applinx emulation framework contains a built-in utility that allows end users to record macros just like in a standard emulation. The macros are saved on the Web server side in an encrypted (by default) xml file per user.

### Constructor Summary

public	<a href="#">GXMacroConfig()</a>
--------	---------------------------------

### Method Summary

boolean	<a href="#">get_EncryptData()</a> For .NET framework usage.
long	<a href="#">get_Flicker()</a> For .NET framework usage.
String	<a href="#">get_MacrosFolder()</a> For .NET framework usage.
boolean	<a href="#">get_SaveUserNameLocally()</a> For .NET framework usage.
String	<a href="#">get_UserName()</a> For .NET framework usage.
boolean	<a href="#">getEncryptData()</a>
long	<a href="#">getFlicker()</a>
String	<a href="#">getMacrosFolder()</a>
String	<a href="#">getUserName()</a>
boolean	<a href="#">saveUserNameLocally()</a>
void	<a href="#">set_EncryptData(boolean encryptData)</a> For .NET framework usage.

void	<a href="#"><u>set_Flicker</u></a> (long flicker) For .NET framework usage.
void	<a href="#"><u>set_MacrosFolder</u></a> (String folder) For .NET framework usage.
void	<a href="#"><u>set_SaveUserNameLocally</u></a> (boolean saveLocal) For .NET framework usage.
void	<a href="#"><u>set_UserName</u></a> (String userName) For .NET framework usage.
void	<a href="#"><u>setEncryptData</u></a> (boolean encryptData) Determines whether to encrypt the macro.
void	<a href="#"><u>setFlicker</u></a> (long flicker)
void	<a href="#"><u>setMacrosFolder</u></a> (String folder) Determines the folder on the web server where the macros will be saved.
void	<a href="#"><u>setSaveUserNameLocally</u></a> (boolean saveLocal) This parameter must be set to True when selecting to set the User name based on the user's cookies.
void	<a href="#"><u>setUserName</u></a> (String userName) The user macros file will be saved according to the user name, therefore it should be a unique ID of the Web user.

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXMacroConfig

```
public GXMacroConfig()
```

## Methods

### getMacrosFolder

```
public String getMacrosFolder()
```

### setMacrosFolder

```
public void setMacrosFolder(String folder)
```

Determines the folder on the web server where the macros will be saved.

(continued from last page)

---

## get\_MacrosFolder

```
public String get_MacrosFolder()
```

For .NET framework usage. Expose the method as a property.

---

## set\_MacrosFolder

```
public void set_MacrosFolder(String folder)
```

For .NET framework usage. Expose the method as a property.

---

## getUserName

```
public String getUserName()
```

---

## setUserName

```
public void setUserName(String userName)
```

The user macros file will be saved according to the user name, therefore it should be a unique ID of the Web user. The User name can be according to the IP Address, it can be cookie based, or set in the code. By default it is configured to IP Address, which is a good idea when the user's IP is constant. When selecting the cookie based option, ApplinX will assign a cookie to the end user, and will read the macro according to the cookie. Macros will be lost when the user deletes his cookies. To enable this option you should set SaveUserNameLocally to true. Select Other to use a session variable which was saved according to the login page user name, or to use the NT user name (relevant for .NET only) (Request.ServerVariables["AUTH\_USER"])

---

## get\_UserName

```
public String get_UserName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UserName

```
public void set_UserName(String userName)
```

For .NET framework usage. Expose the method as a property.

---

## saveUserNameLocally

```
public boolean saveUserNameLocally()
```

---

## get\_SaveUserNameLocally

```
public boolean get_SaveUserNameLocally()
```

For .NET framework usage. Expose the method as a property.

---

## setSaveUserNameLocally

```
public void setSaveUserNameLocally(boolean saveLocal)
```

This parameter must be set to True when selecting to set the User name based on the user's cookies.

---

---

## setEncryptData

```
public void setEncryptData(boolean encryptData)
```

Determines whether to encrypt the macro.

Recommended to use when the macro includes user names and/or passwords.

When changing this setting, existing macros will no longer function. The existing macros will be deleted from the macros folder when recording new macros.

---

## getEncryptData

```
public boolean getEncryptData()
```

---

## set\_EncryptData

```
public void set_EncryptData(boolean encryptData)
```

For .NET framework usage. Expose the method as a property.

---

## get\_EncryptData

```
public boolean get_EncryptData()
```

For .NET framework usage. Expose the method as a property.

---

## setFlicker

```
public void setFlicker(long flicker)
```

---

## getFlicker

```
public long getFlicker()
```

---

## set\_Flicker

```
public void set_Flicker(long flicker)
```

For .NET framework usage. Expose the method as a property.

---

## get\_Flicker

```
public long get_Flicker()
```

For .NET framework usage. Expose the method as a property.

---

## set\_SaveUserNameLocally

```
public void set_SaveUserNameLocally(boolean saveLocal)
```

For .NET framework usage. Expose the method as a property.

---

---

## Package

# com.sabratec.applinx.framework.sso

## Single SignOn

This package is responsible for managing Single SignOn for host applications. The process is combined from 3 parts:

### Recognizing a SignOn screen

The SignOn screen recognition can be one of the followings:

- Automatic (GXSignOnScreenAutomaticRecognizer) - When the first unprotected field of the screen is visible, and the 2nd unprotected field is invisible(password).
- By name (GXSignOnScreenByNameRecognizer) - May accepts a screen name, and optionally user/password field names.

### Retrieving the host user name/password from

The user/password retrieving part can be implemented by implementing the interface GXIUserPasswordRetriever. By default ApplinX framework provides 2 implementations:

- Http request - user name/password are retrieved as a URL parameter of posted to the ApplinX web application URL to gxfirstpage.jsp/aspx
- Http session - user name/password are retrieved as session variables saved within the ApplinX web application URL to gxfirstpage.jsp/aspx

Both retriever should must accept the paramter name of the request parameters/session variables.

### Executing a SignOn process

Determines what action to perform when a SignOn screen was recognized. The retrieved user/password will be passed to one of the following Single SignOn executor:

- Path executor (GXSSOPathExecutor) - Executes the given path with the retrieved user/password
- SendKeys executor (GXSSOSendKeysExecutor) - Performs the mentioned action key with with the retrieved user/password

All the relevant paramaters can be configured using the framework configuration editor, Single SignOn node.

## com.sabratec.applinx.framework.sso Class GXAbstractSignOnScreenRecognizer

java.lang.Object

└─com.sabratec.applinx.framework.sso.GXAbstractSignOnScreenRecognizer

All Implemented Interfaces:

[GXISignOnScreenRecognizer](#)

Direct Known Subclasses:

[GXSignOnScreenAutomaticRecognizer](#)

public abstract class **GXAbstractSignOnScreenRecognizer**

extends Object

implements [GXISignOnScreenRecognizer](#)

### Constructor Summary

public	<a href="#">GXAbstractSignOnScreenRecognizer</a> ()
--------	---

### Method Summary

<a href="#">GXIField</a>	<a href="#">get_PasswordField</a> ()
<a href="#">GXIField</a>	<a href="#">get_UserField</a> ()
<a href="#">GXIField</a>	<a href="#">getPasswordField</a> ()
<a href="#">GXIField</a>	<a href="#">getUserField</a> ()
void	<a href="#">setPasswordField</a> ( <a href="#">GXIField</a> passwordField) For internal use
void	<a href="#">setUserField</a> ( <a href="#">GXIField</a> userField) For internal use

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

#### Methods inherited from interface [com.sabratec.applinx.framework.sso.GXISignOnScreenRecognizer](#)

[recognizeScreen](#)

### Constructors

(continued from last page)

## GXAbstractSignOnScreenRecognizer

```
public GXAbstractSignOnScreenRecognizer()
```

## Methods

### getPasswordField

```
public GXIField getPasswordField()
```

### setPasswordField

```
public void setPasswordField(GXIField passwordField)
```

For internal use

### getUserField

```
public GXIField getUserField()
```

### setUserField

```
public void setUserField(GXIField userField)
```

For internal use

### get\_PasswordField

```
public GXIField get_PasswordField()
```

### get\_UserField

```
public GXIField get_UserField()
```

## com.sabratec.applinx.framework.sso Class GXAbstractSSOExecutor

java.lang.Object

└-com.sabratec.applinx.framework.sso.GXAbstractSSOExecutor

All Implemented Interfaces:

[GXISSOExecutor](#)

Direct Known Subclasses:

[GXSSOSendKeysExecutor](#), [GXSSOPathExecutor](#)

public abstract class **GXAbstractSSOExecutor**  
extends Object  
implements [GXISSOExecutor](#)

### Constructor Summary

public	<a href="#">GXAbstractSSOExecutor()</a>
--------	---

### Method Summary

void	<a href="#">performSingleSignOn</a> ( <a href="#">GXSSOConfig</a> ssoConfig, <a href="#">GXIClientBaseObject</a> baseObject)
------	--

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.framework.sso.GXISSOExecutor](#)

[performSingleSignOn](#)

### Constructors

#### GXAbstractSSOExecutor

public **GXAbstractSSOExecutor**()

### Methods

#### performSingleSignOn

public void **performSingleSignOn**([GXSSOConfig](#) ssoConfig,  
[GXIClientBaseObject](#) baseObject)  
throws [GXGeneralException](#)

(continued from last page)

## com.sabratec.applinx.framework.sso Interface GXISignOnScreenRecognizer

All Known Implementing Classes:

[GXAbstractSignOnScreenRecognizer](#)

public interface **GXISignOnScreenRecognizer**  
extends java.io.Serializable

### Method Summary

<a href="#">GXSignOnRecognizerResponse</a>	<a href="#">recognizeScreen</a> ( <a href="#">GXIScreen</a> screen)
--	---

### Methods

#### **recognizeScreen**

public [GXSignOnRecognizerResponse](#) **recognizeScreen**([GXIScreen](#) screen)

# com.sabratec.applinx.framework.sso

## Interface GXISSOExecuter

All Known Implementing Classes:  
[GXAbstractSSOExecuter](#)

public interface **GXISSOExecuter**  
extends java.io.Serializable

Method Summary	
void	<a href="#">performSingleSignOn</a> ( <a href="#">GXSSOConfig</a> ssoConfig, <a href="#">GXIClientBaseObject</a> baseObject)

### Methods

#### performSingleSignOn

```
public void performSingleSignOn(GXSSOConfig ssoConfig,  
    GXIClientBaseObject baseObject)  
    throws GXGeneralException
```

## com.sabratec.applinx.framework.sso Interface GXIUserPasswordRetriever

All Known Implementing Classes:

[GXAbstractHttpUserPasswordRetriever](#)

```
public interface GXIUserPasswordRetriever
extends java.io.Serializable
```

### Method Summary

String	<a href="#">getPassword()</a>
String	<a href="#">getUserName()</a>

### Methods

#### getUserName

```
public String getUsername()
```

#### getPassword

```
public String getPassword()
```

## com.sabratec.applinx.framework.sso Class GXSignOnRecognizerResponse

java.lang.Object

└-com.sabratec.applinx.framework.sso.GXSignOnRecognizerResponse

public class **GXSignOnRecognizerResponse**  
extends Object

### Constructor Summary

public	<a href="#">GXSignOnRecognizerResponse()</a>
--------	--

### Method Summary

<a href="#">GXIField</a>	<a href="#">get_PasswordField()</a> For .NET framework usage.
<a href="#">GXIField</a>	<a href="#">get_UserField()</a> For .NET framework usage.
<a href="#">GXIField</a>	<a href="#">getPasswordField()</a>
<a href="#">GXIField</a>	<a href="#">getUserField()</a>
boolean	<a href="#">isSignOnScreen()</a>
void	<a href="#">setPasswordField(<a href="#">GXIField</a> passwordField)</a> For internal use
void	<a href="#">setSignOnScreen(boolean isSignOnScreen)</a>
void	<a href="#">setUserField(<a href="#">GXIField</a> userField)</a> For internal use

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### GXSignOnRecognizerResponse

public **GXSignOnRecognizerResponse()**

### Methods

(continued from last page)

---

## getPasswordField

```
public GXIField getPasswordField()
```

---

## getUserField

```
public GXIField getUserField()
```

---

## setPasswordField

```
public void setPasswordField(GXIField passwordField)
```

For internal use

---

## setUserField

```
public void setUserField(GXIField userField)
```

For internal use

---

## get\_PasswordField

```
public GXIField get_PasswordField()
```

For .NET framework usage. Expose the method as a property.

---

## get\_UserField

```
public GXIField get_UserField()
```

For .NET framework usage. Expose the method as a property.

---

## isSignInScreen

```
public boolean isSignInScreen()
```

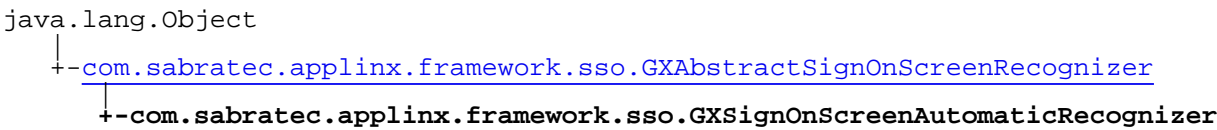
---

## setSignInScreen

```
public void setSignInScreen(boolean isSignInScreen)
```

# com.sabratec.applinx.framework.sso

## Class GXSignOnScreenAutomaticRecognizer



All Implemented Interfaces:  
[GXISignOnScreenRecognizer](#)

Direct Known Subclasses:  
[GXSignOnScreenByNameRecognizer](#)

```
public class GXSignOnScreenAutomaticRecognizer
extends GXAbstractSignOnScreenRecognizer
```

This class recognizes a login screen if the 1st field is an unprotected field, and the 2nd field is a hidden field

Constructor Summary	
public	<a href="#">GXSignOnScreenAutomaticRecognizer()</a>

Method Summary	
<a href="#">GXSignOnRecognizerResponse</a>	<a href="#">recognizeScreen(GXIScreen screen)</a>

Methods inherited from class <a href="#">com.sabratec.applinx.framework.sso.GXAbstractSignOnScreenRecognizer</a>	
<a href="#">get_PasswordField</a> , <a href="#">get_UserField</a> , <a href="#">getPasswordField</a> , <a href="#">getUserField</a> , <a href="#">setPasswordField</a> , <a href="#">setUserField</a>	

Methods inherited from class java.lang.Object	
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>	

Methods inherited from interface <a href="#">com.sabratec.applinx.framework.sso.GXISignOnScreenRecognizer</a>	
<a href="#">recognizeScreen</a>	

## Constructors

```
GXSignOnScreenAutomaticRecognizer
public GXSignOnScreenAutomaticRecognizer()
```

## Methods

(continued from last page)

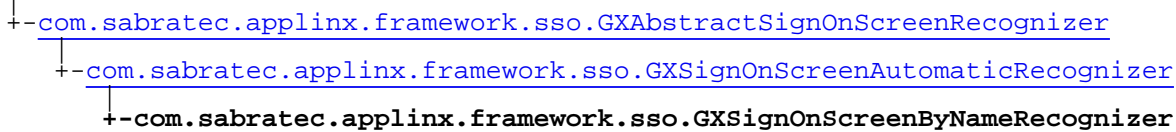
## recognizeScreen

```
public GXSignOnRecognizerResponse recognizeScreen(GXIScreen screen)
```

## com.sabratec.applinx.framework.sso

### Class GXSignOnScreenByNameRecognizer

java.lang.Object



All Implemented Interfaces:

[GXISignOnScreenRecognizer](#)

public class **GXSignOnScreenByNameRecognizer**  
 extends [GXSignOnScreenAutomaticRecognizer](#)

This class determines if a specific screen is a signon screen according to the screen name and/or user/password field name

### Constructor Summary

public	<a href="#">GXSignOnScreenByNameRecognizer()</a>
public	<a href="#">GXSignOnScreenByNameRecognizer</a> (String screenName)
public	<a href="#">GXSignOnScreenByNameRecognizer</a> (String screenName, String userNameFieldName, String passwordFieldName)

### Method Summary

String	<a href="#">get_PasswordFieldName()</a> For .NET framework usage.
String	<a href="#">get_ScreenName()</a> For .NET framework usage.
String	<a href="#">get_UserNameFieldName()</a> For .NET framework usage.
String	<a href="#">getPasswordFieldName()</a> Returns the application field name which determines the password field
String	<a href="#">getScreenName()</a> Returns the screen name which determines that the screen is a signOn screen
String	<a href="#">getUserNameFieldName()</a> Returns the application field name which determines the user name field
<a href="#">GXSignOnRecognizerResponse</a>	<a href="#">recognizeScreen</a> ( <a href="#">GXIScreen</a> screen) Determines if the screen is a signon screen
void	<a href="#">set_PasswordFieldName</a> (String passwordFieldName) For .NET framework usage.

void	<a href="#"><u>set_ScreenName</u></a> (String screenName) For .NET framework usage.
void	<a href="#"><u>set_UserNameFieldName</u></a> (String userNameFieldName) For .NET framework usage.
void	<a href="#"><u>setPasswordFieldName</u></a> (String passwordFieldName) Set the application field name which determines the password field
void	<a href="#"><u>setScreenName</u></a> (String screenName) The name of the SignIn screen, when recognizing the signIn screen according to the screen's name.
void	<a href="#"><u>setUserNameFieldName</u></a> (String userNameFieldName) Returns the application field name which determines the user name field

**Methods inherited from class**[com.sabratec.applinx.framework.sso.GXSignInScreenAutomaticRecognizer](#)[recognizeScreen](#)**Methods inherited from class**[com.sabratec.applinx.framework.sso.GXAbstractSignInScreenRecognizer](#)[get\\_PasswordField](#), [get\\_UserField](#), [getPasswordField](#), [getUserField](#), [setPasswordField](#), [setUserField](#)**Methods inherited from class** java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface** [com.sabratec.applinx.framework.sso.GXISignInScreenRecognizer](#)[recognizeScreen](#)

## Constructors

### GXSignInScreenByNameRecognizer

```
public GXSignInScreenByNameRecognizer()
```

### GXSignInScreenByNameRecognizer

```
public GXSignInScreenByNameRecognizer(String screenName)
```

### GXSignInScreenByNameRecognizer

```
public GXSignInScreenByNameRecognizer(String screenName,
                                       String userNameFieldName,
                                       String passwordFieldName)
```

(continued from last page)

## Methods

### recognizeScreen

```
public GXSignOnRecognizerResponse recognizeScreen(GXIScreen screen)
```

Determines if the screen is a signon screen

**Parameters:**

screen - the screen to check

**Returns:**

---

### getPasswordFieldName

```
public String getPasswordFieldName()
```

Returns the application field name which determines the password field

**Returns:**

the application field name which determines the password field

---

### setPasswordFieldName

```
public void setPasswordFieldName(String passwordFieldName)
```

Set the application field name which determines the password field

**Parameters:**

passwordFieldName - the application field name which determines the password field

---

### getScreenName

```
public String getScreenName()
```

Returns the screen name which determines that the screen is a signOn screen

**Returns:**

the screen name which determines that the screen is a signOn screen

---

### setScreenName

```
public void setScreenName(String screenName)
```

The name of the SignOn screen, when recognizing the signOn screen according to the screen's name.

**Parameters:**

screenName - the screen name which determines that the screen is a signOn screen

---

### getUserNameFieldName

```
public String getUserNameFieldName()
```

Returns the application field name which determines the user name field

**Returns:**

the application field name which determines the user name field

---

## setUserNameFieldName

```
public void setUserNameFieldName(String userNameFieldName)
```

Returns the application field name which determines the user name field

### Parameters:

userNameFieldName - the application field name which determines the user name field

---

## get\_PasswordFieldName

```
public String get_PasswordFieldName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_PasswordFieldName

```
public void set_PasswordFieldName(String passwordFieldName)
```

For .NET framework usage. Expose the method as a property.

---

## get\_ScreenName

```
public String get_ScreenName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ScreenName

```
public void set_ScreenName(String screenName)
```

For .NET framework usage. Expose the method as a property.

---

## get\_UserNameFieldName

```
public String get_UserNameFieldName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UserNameFieldName

```
public void set_UserNameFieldName(String userNameFieldName)
```

For .NET framework usage. Expose the method as a property.

---

## com.sabratec.applinx.framework.sso Class GXSSOConfig

java.lang.Object

└─com.sabratec.applinx.framework.sso.GXSSOConfig

### All Implemented Interfaces:

java.io.Serializable

```
public class GXSSOConfig
extends Object
implements java.io.Serializable
```

This class manages the configuration of a single signOn process against a given host. It holds a [GXISignOnScreenRecognizer](#) which determines whether the screen is a signOn screen, a [GXUserPasswordRetriever](#) which returns a user password from an external resource, and a [GXISSOExecutor](#) which performs the SignOn.

## Constructor Summary

public	<a href="#">GXSSOConfig()</a>
--------	-------------------------------

## Method Summary

boolean	<a href="#">get_Enabled()</a> For .NET framework usage.
<a href="#">GXISignOnScreenRecognizer</a>	<a href="#">getSignOnScreenRecognizer()</a> Determines the type of SignOn screen recognizer to use
<a href="#">GXISSOExecutor</a>	<a href="#">getSsoExecutor()</a> Determines the type of single SignOn executor to use
<a href="#">GXUserPasswordRetriever</a>	<a href="#">getUserPasswordRetriever()</a> Determines the type of user/password retriever to use
boolean	<a href="#">isEnabled()</a> Determines whether to activate a single sign on to the host
void	<a href="#">set_Enabled(boolean enabled)</a> For .NET framework usage.
void	<a href="#">setEnabled(boolean enabled)</a> Determines whether to activate the single sign option.
void	<a href="#">setSignOnScreenRecognizer(GXISignOnScreenRecognizer signOnScreenRecognizer)</a> Determines the method to use to recognize the sign on screen.
void	<a href="#">setSsoExecutor(GXISSOExecutor ssoExecutor)</a> Determines what happens once the SignOn is recognized
void	<a href="#">setUserPasswordRetriever(GXUserPasswordRetriever userPasswordRetriever)</a> Determines the type of user/password retriever to use

**Methods inherited from class `java.lang.Object`**

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### **GXSSOConfig**

```
public GXSSOConfig()
```

## Methods

### **isEnabled**

```
public boolean isEnabled()
```

Determines whether to activate a single sign on to the host

**Returns:**

whether the single signon mechanism is active

### **setEnabled**

```
public void setEnabled(boolean enabled)
```

Determines whether to activate the single sign option.

**Parameters:**

`enabled` - if the sign signon mechanism is active

### **getUserPasswordRetriever**

```
public GX IUserPasswordRetriever getUserPasswordRetriever()
```

Determines the type of user/password retriever to use

**Returns:**

the type of user/password retriever to use

### **setUserPasswordRetriever**

```
public void setUserPasswordRetriever(GX IUserPasswordRetriever userPasswordRetriever)
```

Determines the type of user/password retriever to use

- Based on information retrieved from an HTTP request
- Based on information retrieved from an HTTP session
- Set in the code

**Parameters:**

`userPasswordRetriever` - the type of user/password retriever to use

## getSignInScreenRecognizer

```
public GXISignOnScreenRecognizer getSignInScreenRecognizer()
```

Determines the type of SignIn screen recognizer to use

**Returns:**

the type of SignIn screen recognizer to use

---

## setSignInScreenRecognizer

```
public void setSignInScreenRecognizer(GXISignOnScreenRecognizer signOnScreenRecognizer)
```

Determines the method to use to recognize the sign on screen.

- According to the screen name.
- Automatically recognize the user name and password fields.
- Set from the code.

**Parameters:**

signOnScreenRecognizer - the type of SignIn screen recognizer to use

---

## getSsoExecutor

```
public GXISSOExecutor getSsoExecutor()
```

Determines the type of single SignIn executor to use

**Returns:**

Determines the type of single SignIn screen executor to use

---

## setSsoExecutor

```
public void setSsoExecutor(GXISSOExecutor ssoExecutor)
```

Determines what happens once the SignIn is recognized

- Using an action key (enter an action key)
- Executing a path (enter a path name)
- Set in the code

**Parameters:**

ssoExecutor - the type of single SignIn executor to use

---

## get\_Enabled

```
public boolean get_Enabled()
```

For .NET framework usage. Expose the method as a property.

---

## set\_Enabled

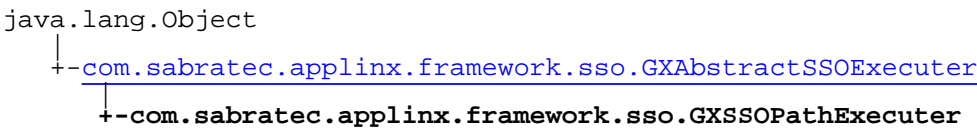
```
public void set_Enabled(boolean enabled)
```

(continued from last page)

For .NET framework usage. Expose the method as a property.

# com.sabratec.applinx.framework.sso

## Class GXSSOPathExecuter



All Implemented Interfaces:  
[GXISSOExecuter](#)

public class **GXSSOPathExecuter**  
extends [GXAbstractSSOExecuter](#)

This path performs a single signOn according to the speficied path name

### Constructor Summary

public	<a href="#">GXSSOPathExecuter</a> ()
public	<a href="#">GXSSOPathExecuter</a> (String pathName)

### Method Summary

String	<a href="#">get_PathName</a> () For .NET framework usage.
String	<a href="#">getPathName</a> ()
void	<a href="#">set_PathName</a> (String pathName) For .NET framework usage.
void	<a href="#">setPathName</a> (String pathName) The name of the path that is to be executed

#### Methods inherited from class [com.sabratec.applinx.framework.sso.GXAbstractSSOExecuter](#)

<a href="#">performSingleSignOn</a>
-------------------------------------

#### Methods inherited from class java.lang.Object

<code>equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait</code>
--

#### Methods inherited from interface [com.sabratec.applinx.framework.sso.GXISSOExecuter](#)

<a href="#">performSingleSignOn</a>
-------------------------------------

### Constructors

(continued from last page)

---

## **GXSSOPathExecuter**

```
public GXSSOPathExecuter()
```

---

## **GXSSOPathExecuter**

```
public GXSSOPathExecuter(String pathName)
```

## Methods

### **getPathName**

```
public String getPathName()
```

---

### **setPathName**

```
public void setPathName(String pathName)
```

The name of the path that is to be executed

---

### **get\_PathName**

```
public String get_PathName()
```

For .NET framework usage. Expose the method as a property.

---

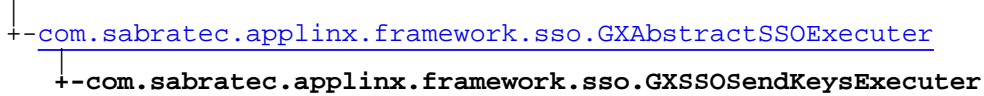
### **set\_PathName**

```
public void set_PathName(String pathName)
```

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.framework.sso Class GXSSOSendKeysExecuter

java.lang.Object



All Implemented Interfaces:

[GXISSOExecuter](#)

public class **GXSSOSendKeysExecuter**  
extends [GXAbstractSSOExecuter](#)

### Constructor Summary

public	<a href="#">GXSSOSendKeysExecuter()</a>
public	<a href="#">GXSSOSendKeysExecuter(String actionKey)</a>

### Method Summary

String	<a href="#">get_ActionKey()</a> For .NET framework usage.
String	<a href="#">getActionKey()</a>
void	<a href="#">set_ActionKey(String actionKey)</a> For .NET framework usage.
void	<a href="#">setActionKey(String actionKey)</a> Enter the action key that is to be activated

Methods inherited from class [com.sabratec.applinx.framework.sso.GXAbstractSSOExecuter](#)

[performSingleSignOn](#)

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.framework.sso.GXISSOExecuter](#)

[performSingleSignOn](#)

### Constructors

(continued from last page)

---

## GXSSOSendKeysExecuter

```
public GXSSOSendKeysExecuter()
```

---

## GXSSOSendKeysExecuter

```
public GXSSOSendKeysExecuter(String actionKey)
```

## Methods

### getActionKey

```
public String getActionKey()
```

---

### setActionKey

```
public void setActionKey(String actionKey)
```

Enter the action key that is to be activated

---

### get\_ActionKey

```
public String get_ActionKey()
```

For .NET framework usage. Expose the method as a property.

---

### set\_ActionKey

```
public void set_ActionKey(String actionKey)
```

For .NET framework usage. Expose the method as a property.

---

**Package**

**com.sabratec.applinx.framework.tables**

## com.sabratec.applinx.framework.tables

### Class GXTableBuildConfig

java.lang.Object

└─com.sabratec.applinx.framework.tables.GXTableBuildConfig

#### All Implemented Interfaces:

java.io.Serializable

public class **GXTableBuildConfig**  
 extends Object  
 implements java.io.Serializable

Summary description for GXITableBuildConfig.

### Constructor Summary

public	<a href="#">GXTableBuildConfig()</a>
--------	--------------------------------------

### Method Summary

boolean	<a href="#">get_AlternateColors()</a> For .NET framework usage.
String	<a href="#">get_AlternateCssName()</a> For .NET framework usage.
boolean	<a href="#">get_CloneComboxOnClientSide()</a> For .NET framework usage.
String	<a href="#">get_DynamicDisableCssClass()</a> For .NET framework usage.
boolean	<a href="#">get_DynamicFieldCheck()</a> <b>Deprecated.</b> Use <i>ReflectHostProtected</i> instead
String	<a href="#">get_Encoding()</a> For .NET framework usage.
int	<a href="#">get_FromRow()</a> For .NET framework usage.
boolean	<a href="#">get_GenerateTrOnClickEvent()</a> <b>Deprecated.</b>
boolean	<a href="#">get_ReflectBGColor()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectFGColor()</a> For .NET framework usage.
boolean	<a href="#">get_ReflectHostProtected()</a> For .NET framework usage.

int	<a href="#"><u>get_ToRow()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_TrimLabels()</u></a> For .NET framework usage.
boolean	<a href="#"><u>getAlternateColors()</u></a> Determines whether to use alternate colors for each second table row.
String	<a href="#"><u>getAlternateCssName()</u></a> The css name used in every alternate table row when AlternateColors is set to true.
boolean	<a href="#"><u>getCloneComboxOnClientSide()</u></a>
String	<a href="#"><u>getDynamicDisableCssClass()</u></a> The name of the css class that is added to an input field, which changed to read only according to the ReflectHostProtected parameter.
boolean	<a href="#"><u>getDynamicFieldCheck()</u></a> <b>Deprecated.</b> Use <i>ReflectHostProtected</i> instead
String	<a href="#"><u>getEncoding()</u></a> <b>Deprecated.</b> - should no longer be used, UTF-8 is now used when needed.
int	<a href="#"><u>getFromRow()</u></a> The starting row of the HTML on which the duplication will be based.
boolean	<a href="#"><u>getGenerateTrOnClickEvent()</u></a> <b>Deprecated.</b>
boolean	<a href="#"><u>getReflectBGColor()</u></a> Set if to dynamically add a relevant back color css to the page fields when calling gx_fillForm.
boolean	<a href="#"><u>getReflectFGColor()</u></a> Determine if to dynamically add a relevant color css to the page fields when calling gx_fillForm.
boolean	<a href="#"><u>getReflectHostProtected()</u></a> Determines whether to dynamically change the page input fields to read only when gx_fillForm is called
int	<a href="#"><u>getToRow()</u></a> The ending row of the HTML on which the duplication will be based.
boolean	<a href="#"><u>getTrimLabels()</u></a>
void	<a href="#"><u>set_AlternateColors()</u></a> (boolean alternateColors) For .NET framework usage.
void	<a href="#"><u>set_AlternateCssName()</u></a> (String alternateCssName) For .NET framework usage.
void	<a href="#"><u>set_CloneComboxOnClientSide()</u></a> (boolean cloneComboxOnClientSide) For .NET framework usage.
void	<a href="#"><u>set_DynamicDisableCssClass()</u></a> (String dynamicDisableCssClass) For .NET framework usage.

void	<a href="#"><u>set_DynamicFieldCheck</u></a> (boolean dynamicFieldCheck) <b>Deprecated.</b> Use <i>ReflectHostProtected</i> instead <i>/** @property</i>
void	<a href="#"><u>set_Encoding</u></a> (String encoding) For .NET framework usage.
void	<a href="#"><u>set_FromRow</u></a> (int fromRow) For .NET framework usage.
void	<a href="#"><u>set_GenerateTrOnClickEvent</u></a> (boolean generateTrOnClickEvent) <b>Deprecated.</b>
void	<a href="#"><u>set_ReflectBGColor</u></a> (boolean reflectBGColor) For .NET framework usage.
void	<a href="#"><u>set_ReflectFGColor</u></a> (boolean reflectFGColor) For .NET framework usage.
void	<a href="#"><u>set_ReflectHostProtected</u></a> (boolean reflectHostProtected) For .NET framework usage.
void	<a href="#"><u>set_ToRow</u></a> (int toRow) For .NET framework usage.
void	<a href="#"><u>set_TrimLabels</u></a> (boolean trimLabels) <b>Deprecated.</b> Use <i>Applinx</i> server configuration, screen content tab
void	<a href="#"><u>setAlternateColors</u></a> (boolean alternateColors) Determines whether to use alternate colors for every other row in a table.
void	<a href="#"><u>setAlternateCssName</u></a> (String alternateCssName) The css name used in every alternate table row when AlternateColors is set to true.
void	<a href="#"><u>setCloneComboxOnClientSide</u></a> (boolean cloneComboxOnClientSide) When this parameter is set to true (by default) an HTML text combo box is rendered once to the browser and all the other combo boxes of the following rows are cloned on the browser side from the first one (this significantly reduces the size of the HTML when there are a large number of options in the combo box).
void	<a href="#"><u>setDynamicDisableCssClass</u></a> (String dynamicDisableCssClass) The name of the css class that is added to an input field, which changed to read only according to the ReflectHostProtected parameter.
void	<a href="#"><u>setDynamicFieldCheck</u></a> (boolean dynamicFieldCheck) <b>Deprecated.</b> Use <i>ReflectHostProtected</i> instead
void	<a href="#"><u>setEncoding</u></a> (String encoding) <b>Deprecated.</b> - should no longer be used, UTF-8 is now used when needed.
void	<a href="#"><u>setFromRow</u></a> (int fromRow) The starting row of the HTML on which the duplication will be based.
void	<a href="#"><u>setGenerateTrOnClickEvent</u></a> (boolean generateTrOnClickEvent) <b>Deprecated.</b>
void	<a href="#"><u>setReflectBGColor</u></a> (boolean reflectBGColor) Set if to dynamically add a relevant back color css to the page fields when calling <i>gx_fillForm</i> .

void	<a href="#"><code>setReflectFGColor</code></a> (boolean reflectFGColor) Determine if to dynamically add a relevant color css to the page fields when calling <code>gx_fillForm</code> .
void	<a href="#"><code>setReflectHostProtected</code></a> (boolean reflectHostProtected) Determines whether to dynamically change the page input fields to read only when <code>gx_fillForm</code> is called
void	<a href="#"><code>setToRow</code></a> (int toRow) The ending row of the HTML on which the duplication will be based.
void	<a href="#"><code>setTrimLabels</code></a> (boolean trimLabels) <b>Deprecated.</b> Use ApplinX server configuration, screen content tab

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXTableBuildConfig

```
public GXTableBuildConfig()
```

## Methods

### getFromRow

```
public int getFromRow()
```

The starting row of the HTML on which the duplication will be based. By default, 1 when there is a header row, and 0 when there is not.

#### Returns:

The starting row of the HTML

### setFromRow

```
public void setFromRow(int fromRow)
```

The starting row of the HTML on which the duplication will be based. By default, 1 when there is a header row, and 0 when there is not.

#### Parameters:

`fromRow` - the starting row of the HTML on which the duplication will be based

### get\_FromRow

```
public int get_FromRow()
```

For .NET framework usage. Expose the method as a property.

(continued from last page)

---

## set\_FromRow

```
public void set_FromRow(int fromRow)
```

For .NET framework usage. Expose the method as a property.

---

## getToRow

```
public int getToRow()
```

The ending row of the HTML on which the duplication will be based. By default, 1 when there is a header row, and 0 when there is not. Both these parameters can be used when you want to duplicate host rows based on more than one HTML row. Note: 0 will be the index of the first HTML row (usually the headers row).

### Returns:

the ending row of the HTML on which the duplication will be based

---

## setToRow

```
public void setToRow(int toRow)
```

The ending row of the HTML on which the duplication will be based. By default, 1 when there is a header row, and 0 when there is not. Both these parameters can be used when you want to duplicate host rows based on more than one HTML row. Note: 0 will be the index of the first HTML row (usually the headers row).

### Parameters:

toRow - the ending row of the HTML on which the duplication will be based

---

## get\_ToRow

```
public int get_ToRow()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ToRow

```
public void set_ToRow(int toRow)
```

For .NET framework usage. Expose the method as a property.

---

## setGenerateTrOnClickEvent

```
public void setGenerateTrOnClickEvent(boolean generateTrOnClickEvent)
```

Deprecated.

---

## getGenerateTrOnClickEvent

```
public boolean getGenerateTrOnClickEvent()
```

Deprecated.

---

## set\_GenerateTrOnClickEvent

```
public void set_GenerateTrOnClickEvent(boolean generateTrOnClickEvent)
```

---

(continued from last page)

**Deprecated.**

---

## get\_GenerateTrOnClickEvent

```
public boolean get_GenerateTrOnClickEvent()
```

**Deprecated.**

---

## setDynamicFieldCheck

```
public void setDynamicFieldCheck(boolean dynamicFieldCheck)
```

**Deprecated.** Use *ReflectHostProtected* instead

---

## getDynamicFieldCheck

```
public boolean getDynamicFieldCheck()
```

**Deprecated.** Use *ReflectHostProtected* instead

---

## set\_DynamicFieldCheck

```
public void set_DynamicFieldCheck(boolean dynamicFieldCheck)
```

**Deprecated.** Use *ReflectHostProtected* instead */\*\* @property*

---

## set\_ReflectHostProtected

```
public void set_ReflectHostProtected(boolean reflectHostProtected)
```

For .NET framework usage. Expose the method as a property.

---

## get\_ReflectHostProtected

```
public boolean get_ReflectHostProtected()
```

For .NET framework usage. Expose the method as a property.

---

## get\_DynamicFieldCheck

```
public boolean get_DynamicFieldCheck()
```

**Deprecated.** Use *ReflectHostProtected* instead

(continued from last page)

## setReflectHostProtected

```
public void setReflectHostProtected(boolean reflectHostProtected)
```

Determines whether to dynamically change the page input fields to read only when gx\_fillForm is called

**Parameters:**

reflectHostProtected - whether to dynamically change the page input fields to read only

---

## getReflectHostProtected

```
public boolean getReflectHostProtected()
```

Determines whether to dynamically change the page input fields to read only when gx\_fillForm is called

**Returns:**

whether to dynamically change the page input fields to read only

---

## setAlternateColors

```
public void setAlternateColors(boolean alternateColors)
```

Determines whether to use alternate colors for every other row in a table. Use AlternateCssName to control the alternate css name.

**Parameters:**

alternateColors - whether to use alternate colors for each second table row

---

## getAlternateColors

```
public boolean getAlternateColors()
```

Determines whether to use alternate colors for each second table row. Use AlternateCssName to control the alternate css name.

**Returns:**

whether to use alternate colors for each second table row

---

## set\_AlternateColors

```
public void set_AlternateColors(boolean alternateColors)
```

For .NET framework usage. Expose the method as a property.

---

## get\_AlternateColors

```
public boolean get_AlternateColors()
```

For .NET framework usage. Expose the method as a property.

---

## setAlternateCssName

```
public void setAlternateCssName(String alternateCssName)
```

The css name used in every alternate table row when AlternateColors is set to true. The default css is: gx\_tbl\_alternate2 , since the default generated table is generated with gx\_tbl\_alternate1.

**Parameters:**

alternateCssName - the css name used in every alternate table row

---

## getAlternateCssName

```
public String getAlternateCssName()
```

The css name used in every alternate table row when AlternateColors is set to true. The default css is: gx\_tbl\_alternate2 , since the default generated table is generated with gx\_tbl\_alternate1.

### Returns:

the css name used in every alternate table row

---

## set\_AlternateCssName

```
public void set_AlternateCssName(String alternateCssName)
```

For .NET framework usage. Expose the method as a property.

---

## get\_AlternateCssName

```
public String get_AlternateCssName()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DynamicDisableCssClass

```
public void set_DynamicDisableCssClass(String dynamicDisableCssClass)
```

For .NET framework usage. Expose the method as a property.

---

## get\_DynamicDisableCssClass

```
public String get_DynamicDisableCssClass()
```

For .NET framework usage. Expose the method as a property.

---

## setDynamicDisableCssClass

```
public void setDynamicDisableCssClass(String dynamicDisableCssClass)
```

The name of the css class that is added to an input field, which changed to read only according to the ReflectHostProtected parameter. Used for giving a different look to this type of input field.

### Parameters:

dynamicDisableCssClass - the name of the css class that is added to an input field

---

## getDynamicDisableCssClass

```
public String getDynamicDisableCssClass()
```

The name of the css class that is added to an input field, which changed to read only according to the ReflectHostProtected parameter. Used for giving a different look to this type of input field.

### Returns:

the name of the css class that is added to an input field

---

## setReflectFGColor

```
public void setReflectFGColor(boolean reflectFGColor)
```

---

(continued from last page)

Determine if to dynamically add a relevant color css to the page fields when calling `gx_fillForm`. The relevant css classes are in `styles_instant.css` in the new application /css\_colors folder in the emulation template.

**Parameters:**

`reflectFGColor` - whether to dynamically add a relevant color css to the page fields

---

## getReflectFGColor

```
public boolean getReflectFGColor()
```

Determine if to dynamically add a relevant color css to the page fields when calling `gx_fillForm`. The relevant css classes are in `styles_instant.css` in the new application /css\_colors folder in the emulation template.

**Returns:**

whether to dynamically add a relevant color css to the page fields

---

## set\_ReflectFGColor

```
public void set_ReflectFGColor(boolean reflectFGColor)
```

For .NET framework usage. Expose the method as a property.

---

## get\_ReflectFGColor

```
public boolean get_ReflectFGColor()
```

For .NET framework usage. Expose the method as a property.

---

## setReflectBGColor

```
public void setReflectBGColor(boolean reflectBGColor)
```

Set if to dynamically add a relevant back color css to the page fields when calling `gx_fillForm`. The relevant css classes are in `styles_instant.css` in the new application /css\_colors folder in the emulation template.

**Parameters:**

`reflectBGColor` - whether to dynamically add a relevant back color css to the page fields

---

## getReflectBGColor

```
public boolean getReflectBGColor()
```

Set if to dynamically add a relevant back color css to the page fields when calling `gx_fillForm`. The relevant css classes are in `styles_instant.css` in the new application /css\_colors folder in the emulation template.

**Returns:**

whether to dynamically add a relevant back color css to the page fields

---

## set\_ReflectBGColor

```
public void set_ReflectBGColor(boolean reflectBGColor)
```

For .NET framework usage. Expose the method as a property.

---

## get\_ReflectBGColor

```
public boolean get_ReflectBGColor()
```

For .NET framework usage. Expose the method as a property.

---

## setCloneComboxOnClientSide

```
public void setCloneComboxOnClientSide(boolean cloneComboxOnClientSide)
```

When this parameter is set to true (by default) an HTML text combo box is rendered once to the browser and all the other combo boxes of the following rows are cloned on the browser side from the first one (this significantly reduces the size of the HTML when there are a large number of options in the combo box). Set this parameter to false when you would like to dynamically manipulate the combo box values using table user exits: gx\_changeTr, gx\_changeTd, gx\_changeControl.

---

## getCloneComboxOnClientSide

```
public boolean getCloneComboxOnClientSide()
```

---

## set\_CloneComboxOnClientSide

```
public void set_CloneComboxOnClientSide(boolean cloneComboxOnClientSide)
```

For .NET framework usage. Expose the method as a property.

---

## get\_CloneComboxOnClientSide

```
public boolean get_CloneComboxOnClientSide()
```

For .NET framework usage. Expose the method as a property.

---

## setTrimLabels

```
public void setTrimLabels(boolean trimLabels)
```

**Deprecated.** Use ApplinX server configuration, screen content tab

---

## getTrimLabels

```
public boolean getTrimLabels()
```

---

## set\_TrimLabels

```
public void set_TrimLabels(boolean trimLabels)
```

**Deprecated.** Use ApplinX server configuration, screen content tab

---

## get\_TrimLabels

```
public boolean get_TrimLabels()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## getEncoding

```
public String getEncoding()
```

**Deprecated.** - *should no longer be used, UTF-8 is now used when needed.*

For JSP usage only. In JSP tables tags (gx:table), XSL serialization is internally used, and the encoding is required in case the table contains non-english character.

**Returns:**

the encoding used for JSP table tags (gx:table)

---

## setEncoding

```
public void setEncoding(String encoding)
```

**Deprecated.** - *should no longer be used, UTF-8 is now used when needed.*

For JSP usage only. In JSP tables tags (gx:table), XSL serialization is internally used, and the encoding is required in case the table contains non-english character

**Parameters:**

`encoding` - the encoding used for JSP table tags (gx:table)

---

## get\_Encoding

```
public String get_Encoding()
```

For .NET framework usage. Expose the method as a property.

---

## set\_Encoding

```
public void set_Encoding(String encoding)
```

For .NET framework usage. Expose the method as a property.

---

**Package**

**com.sabratec.applinx.framework.web.jse  
engine**

# com.sabratec.applinx.framework.web.jsengine

## Class GXClientConfig

java.lang.Object

└-com.sabratec.applinx.framework.web.jsengine.GXClientConfig

public class **GXClientConfig**  
extends Object

### Constructor Summary

public	<a href="#">GXClientConfig()</a>
--------	----------------------------------

### Method Summary

<a href="#">GXEngineConfig</a>	<a href="#">get_EngineConfig()</a> For .NET framework usage.
<a href="#">GXInstantEmulatorConfig</a>	<a href="#">get_InstantEmulatorConfig()</a> For .NET framework usage.
<a href="#">GXWebEmulatorConfig</a>	<a href="#">get_WebEmulatorConfig()</a> For .NET framework usage.
<a href="#">GXWindowConfig</a>	<a href="#">get_WindowConfig()</a> For .NET framework usage.
<a href="#">GXEngineConfig</a>	<a href="#">getEngineConfig()</a>
<a href="#">GXInstantEmulatorConfig</a>	<a href="#">getInstantEmulatorConfig()</a>
<a href="#">GXWebEmulatorConfig</a>	<a href="#">getWebEmulatorConfig()</a>
<a href="#">GXWindowConfig</a>	<a href="#">getWindowConfig()</a>
void	<a href="#">set_EngineConfig(GXEngineConfig engineConfig)</a> For .NET framework usage.
void	<a href="#">set_InstantEmulatorConfig(GXInstantEmulatorConfig instantEmulatorConfig)</a> For .NET framework usage.
void	<a href="#">set_WebEmulatorConfig(GXWebEmulatorConfig webEmulatorConfig)</a> For .NET framework usage.
void	<a href="#">set_WindowConfig(GXWindowConfig windowConfig)</a> For .NET framework usage.
void	<a href="#">setEngineConfig(GXEngineConfig engineConfig)</a>

void	<a href="#">setInstantEmulatorConfig</a> ( <a href="#">GXInstantEmulatorConfig</a> instantEmulatorConfig)
void	<a href="#">setWebEmulatorConfig</a> ( <a href="#">GXWebEmulatorConfig</a> webEmulatorConfig)
void	<a href="#">setWindowConfig</a> ( <a href="#">GXWindowConfig</a> windowConfig)

**Methods inherited from class** `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### **GXClientConfig**

```
public GXClientConfig()
```

## Methods

### **getEngineConfig**

```
public GXEngineConfig getEngineConfig()
```

### **setEngineConfig**

```
public void setEngineConfig(GXEngineConfig engineConfig)
```

### **get\_EngineConfig**

```
public GXEngineConfig get_EngineConfig()
```

For .NET framework usage. Expose the method as a property.

### **set\_EngineConfig**

```
public void set_EngineConfig(GXEngineConfig engineConfig)
```

For .NET framework usage. Expose the method as a property.

### **getInstantEmulatorConfig**

```
public GXInstantEmulatorConfig getInstantEmulatorConfig()
```

### **setInstantEmulatorConfig**

```
public void setInstantEmulatorConfig(GXInstantEmulatorConfig instantEmulatorConfig)
```

---

(continued from last page)

---

## **get\_InstantEmulatorConfig**

```
public GXInstantEmulatorConfig get_InstantEmulatorConfig()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_InstantEmulatorConfig**

```
public void set_InstantEmulatorConfig(GXInstantEmulatorConfig instantEmulatorConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **getWebEmulatorConfig**

```
public GXWebEmulatorConfig getWebEmulatorConfig()
```

## **setWebEmulatorConfig**

```
public void setWebEmulatorConfig(GXWebEmulatorConfig webEmulatorConfig)
```

## **get\_WebEmulatorConfig**

```
public GXWebEmulatorConfig get_WebEmulatorConfig()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_WebEmulatorConfig**

```
public void set_WebEmulatorConfig(GXWebEmulatorConfig webEmulatorConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **getWindowConfig**

```
public GXWindowConfig getWindowConfig()
```

## **setWindowConfig**

```
public void setWindowConfig(GXWindowConfig windowConfig)
```

## **get\_WindowConfig**

```
public GXWindowConfig get_WindowConfig()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## **set\_WindowConfig**

```
public void set_WindowConfig(GXWindowConfig windowConfig)
```

For .NET framework usage. Expose the method as a property.

# com.sabratec.applinx.framework.web.jsengine

## Class GXEngineConfig

java.lang.Object

└-com.sabratec.applinx.framework.web.jsengine.GXEngineConfig

public class **GXEngineConfig**  
extends Object

framework configurations basic class

### Field Summary

public static final	<a href="#">LOG_LEVEL_DEBUG</a> Value: 3
public static final	<a href="#">LOG_LEVEL_ERROR</a> Value: 0
public static final	<a href="#">LOG_LEVEL_MESSAGE</a> Value: 2
public static final	<a href="#">LOG_LEVEL_WARNING</a> Value: 1

### Constructor Summary

public	<a href="#">GXEngineConfig()</a>
--------	----------------------------------

### Method Summary

String	<a href="#">get_DebugModules()</a> For .NET framework usage.
String	<a href="#">get_DefaultFrameColumns()</a> For .NET framework usage.
int	<a href="#">get_LogLevel()</a> For .NET framework usage.
String	<a href="#">get_LogoffMessage()</a> For .NET framework usage.
boolean	<a href="#">get_LogoffOnBrowserClose()</a> For .NET framework usage.
String	<a href="#">get_MozillaLogoffMessage()</a> For .NET framework usage.
boolean	<a href="#">get_PromptOnBrowserClose()</a> For .NET framework usage.

boolean	<a href="#"><u>get_ShowLogConsole()</u></a> For .NET framework usage.
String	<a href="#"><u>getDebugModules()</u></a>
String	<a href="#"><u>getDefaultFrameColumns()</u></a> Used for debug purposes only.
int	<a href="#"><u>getLogLevel()</u></a>
String	<a href="#"><u>getLogoffMessage()</u></a> The popup message that will appear when closing the browser.
String	<a href="#"><u>getMozillaLogoffMessage()</u></a> <b>Deprecated.</b> use <a href="#"><u>getLogoffMessage()</u></a>
boolean	<a href="#"><u>isLogoffOnBrowserClose()</u></a> Determines whether to log off from the host session when the browser is closed.
boolean	<a href="#"><u>isPromptOnBrowserClose()</u></a> Prompt the user whether to exit the browser session.
boolean	<a href="#"><u>isShowLogConsole()</u></a>
void	<a href="#"><u>set_DebugModules()</u></a> (String debugModules) For .NET framework usage.
void	<a href="#"><u>set_DefaultFrameColumns()</u></a> (String defaultFrameColumns) For .NET framework usage.
void	<a href="#"><u>set_LogLevel()</u></a> (int logLevel) For .NET framework usage.
void	<a href="#"><u>set_LogoffMessage()</u></a> (String logoffMessage) For .NET framework usage.
void	<a href="#"><u>set_LogoffOnBrowserClose()</u></a> (boolean logoffOnBrowserClose) For .NET framework usage.
void	<a href="#"><u>set_MozillaLogoffMessage()</u></a> (String mozillaLogoffMessage) For .NET framework usage.
void	<a href="#"><u>set_PromptOnBrowserClose()</u></a> (boolean promptOnBrowserClose) For .NET framework usage.
void	<a href="#"><u>set_ShowLogConsole()</u></a> (boolean showLogConsole) For .NET framework usage.
void	<a href="#"><u>setDebugModules()</u></a> (String debugModules)
void	<a href="#"><u>setDefaultFrameColumns()</u></a> (String defaultFrameColumns) Used for debug purposes only.
void	<a href="#"><u>setLogLevel()</u></a> (int logLevel)
void	<a href="#"><u>setLogoffMessage()</u></a> (String logoffMessage) The popup message that will be displayed when closing the browser.

void	<a href="#">setLogoffOnBrowserClose</a> (boolean logoffOnBrowserClose) Determines whether to log off from the host session when the browser is closed.
void	<a href="#">setMozillaLogoffMessage</a> (String mozillaLogoffMessage) <b>Deprecated.</b> use <a href="#">setLogoffMessage(String)</a>
void	<a href="#">setPromptOnBrowserClose</a> (boolean promptOnBrowserClose) Determines whether to disconnect the session once the browser is closed.
void	<a href="#">setShowLogConsole</a> (boolean showLogConsole)

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### LOG\_LEVEL\_ERROR

```
public static final int LOG_LEVEL_ERROR
```

Constant value: 0

### LOG\_LEVEL\_WARNING

```
public static final int LOG_LEVEL_WARNING
```

Constant value: 1

### LOG\_LEVEL\_MESSAGE

```
public static final int LOG_LEVEL_MESSAGE
```

Constant value: 2

### LOG\_LEVEL\_DEBUG

```
public static final int LOG_LEVEL_DEBUG
```

Constant value: 3

## Constructors

### GXEngineConfig

```
public GXEngineConfig()
```

## Methods

(continued from last page)

---

## getLogLevel

```
public int getLogLevel()
```

---

## setLogLevel

```
public void setLogLevel(int logLevel)
```

---

## get\_LogLevel

```
public int get_LogLevel()
```

For .NET framework usage. Expose the method as a property.

---

## set\_LogLevel

```
public void set_LogLevel(int logLevel)
```

For .NET framework usage. Expose the method as a property.

---

## isLogoffOnBrowserClose

```
public boolean isLogoffOnBrowserClose()
```

Determines whether to log off from the host session when the browser is closed. It is recommended to define a session timeout in the ApplinX Designer (Application Properties dialog box>Host tab, Non-activity timeout) to ensure logging off from a session.

### Returns:

whether to log off from the host session when the browser is closed

---

## setLogoffOnBrowserClose

```
public void setLogoffOnBrowserClose(boolean logoffOnBrowserClose)
```

Determines whether to log off from the host session when the browser is closed. It is recommended to define a session timeout in the ApplinX Designer (Application Properties dialog box>Host tab, Non-activity timeout) to ensure logging off from a session.

### Parameters:

logoffOnBrowserClose -- whether to log off from the host session when the browser is closed

---

## get\_LogoffOnBrowserClose

```
public boolean get_LogoffOnBrowserClose()
```

For .NET framework usage. Expose the method as a property.

---

## set\_LogoffOnBrowserClose

```
public void set_LogoffOnBrowserClose(boolean logoffOnBrowserClose)
```

For .NET framework usage. Expose the method as a property.

---

## getMozillaLogoffMessage

```
public String getMozillaLogoffMessage()
```

**Deprecated.** use [getLogoffMessage\(\)](#)

### Parameters:

mozillaLogoffMessage

---

## set\_MozillaLogoffMessage

```
public void set_MozillaLogoffMessage(String mozillaLogoffMessage)
```

For .NET framework usage. Expose the method as a property.

---

## get\_MozillaLogoffMessage

```
public String get_MozillaLogoffMessage()
```

For .NET framework usage. Expose the method as a property.

---

## setMozillaLogoffMessage

```
public void setMozillaLogoffMessage(String mozillaLogoffMessage)
```

**Deprecated.** use [setLogoffMessage\(String\)](#)

### Parameters:

mozillaLogoffMessage

---

## getLogoffMessage

```
public String getLogoffMessage()
```

The popup message that will appear when closing the browser. Relevant either when closing the configuring `PromptOnBrowserClose` as on or when closing a Mozilla based browsers (FireFox, Netscape). The reason is that it's not possible to determine whether Browser Refresh button/Close button was pressed. Note: The message will not appear when the menu bar in Mozilla is not available (Full screen mode).

### Returns:

the popup message that will appear when closing the browser in Mozilla based browsers

---

## setLogoffMessage

```
public void setLogoffMessage(String logoffMessage)
```

The popup message that will be displayed when closing the browser. Relevant either when closing the configuring `PromptOnBrowserClose` as on or when closing Mozilla based browsers (FireFox, Netscape). The reason is that it's not possible to determine whether Browser Refresh button/Close button was pressed. Note: The message will not appear when the menu bar in Mozilla is not available (Full screen mode).

### Parameters:

logoffMessage - the popup message that will appear when closing the browser in Mozilla based browsers

---

(continued from last page)

---

## getDefaultFrameColumns

```
public String getDefaultFrameColumns()
```

Used for debug purposes only. Shows the invisible frame of the framework frameset. Set it to "70%" for example to see the frameset hidden frames

### Returns:

the columns width of the active frameset

---

## setDefaultFrameColumns

```
public void setDefaultFrameColumns(String defaultFrameColumns)
```

Used for debug purposes only. Shows the invisible frame of the framework frameset. Set it to "70%" for example to see the frameset hidden frames

### Parameters:

defaultFrameColumns - the columns width of the active frameset

---

## getDebugModules

```
public String getDebugModules()
```

---

## setDebugModules

```
public void setDebugModules(String debugModules)
```

---

## get\_DebugModules

```
public String get_DebugModules()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DebugModules

```
public void set_DebugModules(String debugModules)
```

For .NET framework usage. Expose the method as a property.

---

## isShowLogConsole

```
public boolean isShowLogConsole()
```

---

## setShowLogConsole

```
public void setShowLogConsole(boolean showLogConsole)
```

---

(continued from last page)

---

## get\_ShowLogConsole

```
public boolean get_ShowLogConsole()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ShowLogConsole

```
public void set_ShowLogConsole(boolean showLogConsole)
```

For .NET framework usage. Expose the method as a property.

---

## get\_DefaultFrameColumns

```
public String get_DefaultFrameColumns()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DefaultFrameColumns

```
public void set_DefaultFrameColumns(String defaultFrameColumns)
```

For .NET framework usage. Expose the method as a property.

---

## isPromptOnBrowserClose

```
public boolean isPromptOnBrowserClose()
```

Prompt the user whether to exit the browser session. Also triggered when user hits refresh button

### Returns:

whether to exit the browser session

---

## setPromptOnBrowserClose

```
public void setPromptOnBrowserClose(boolean promptOnBrowserClose)
```

Determines whether to disconnect the session once the browser is closed. This parameter also relates to when the Refresh button is pressed.

### Parameters:

promptOnBrowserClose - whether to exit the browser session

---

## get\_PromptOnBrowserClose

```
public boolean get_PromptOnBrowserClose()
```

For .NET framework usage. Expose the method as a property.

---

## set\_PromptOnBrowserClose

```
public void set_PromptOnBrowserClose(boolean promptOnBrowserClose)
```

For .NET framework usage. Expose the method as a property.

---

## get\_LogoffMessage

```
public String get_LogoffMessage()
```

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **set\_LogoffMessage**

```
public void set_LogoffMessage(String logoffMessage)
```

For .NET framework usage. Expose the method as a property.

# com.sabratec.applinx.framework.web.jsengine

## Class GXInstantEmulatorConfig

java.lang.Object

└─com.sabratec.applinx.framework.web.jsengine.GXInstantEmulatorConfig

Direct Known Subclasses:

[GXWebEmulatorConfig](#)

```
public class GXInstantEmulatorConfig
    extends Object
```

Configuration class for html emulation instant page. Generated to XML file and loaded on the client side

### Field Summary

public static final	<a href="#">ARROWS_ALL_SCREEN</a> Value: 2
public static final	<a href="#">ARROWS_INPUT_FIELDS</a> Value: 1
public static final	<a href="#">ARROWS_NONE</a> Value: 0
public static final	<a href="#">LABEL_FOCUS_BOLD</a> Value: 2
public static final	<a href="#">LABEL_FOCUS_NONE</a> Value: 0
public static final	<a href="#">LABEL_FOCUS_UNDERLINE</a> Value: 1

### Constructor Summary

public	<a href="#">GXInstantEmulatorConfig()</a>
--------	---

### Method Summary

boolean	<a href="#">get_AutoSkipAllFields()</a> For .NET framework usage.
boolean	<a href="#">get_BlinkingCaret()</a> For .NET framework usage.
boolean	<a href="#">get_BlockNumericFields()</a> For .NET framework usage.

boolean	<a href="#"><u>get_CaretAtEndOfField()</u></a> For .NET framework usage.
String	<a href="#"><u>get_ColorSets()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_DoJapaneseDataTypeValidityCheck()</u></a> For .NET framework usage.
String	<a href="#"><u>get_DoubleClickActionKey()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_EmulationPasteBehavior()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_HandleMenuSelection()</u></a> For .NET framework usage.
int	<a href="#"><u>get_LabelFocus()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_SelectFieldOnFocus()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_ShowBlinkingFields()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_TabOnInputsOnly()</u></a> For .NET framework usage.
int	<a href="#"><u>get_UseArrows()</u></a> For .NET framework usage.
boolean	<a href="#"><u>get_UseKeyboardPfKeys()</u></a> For .NET framework usage.
String	<a href="#"><u>getColorSets()</u></a> This functionality enables the end user select one of the pre-defined style sheets for instant screens.
String	<a href="#"><u>getDoubleClickActionKey()</u></a> Sends the specified action key to the host when double-clicking on a label.
int	<a href="#"><u>getLabelFocus()</u></a> This option allows you to determine whether the font style of a protected field will become underline, bold or will not change when a user either clicks on the protected field or moves between the protected fields using the keyboard arrow buttons.
int	<a href="#"><u>getUseArrows()</u></a> Determines the arrow keys' navigation behavior (none, input fields or all screen).
boolean	<a href="#"><u>isAutoSkipAllFields()</u></a> Determines whether or not the cursor will automatically move to the next input field once the current input field is filled with text (applies to all input fields).
boolean	<a href="#"><u>isBlinkingCaret()</u></a>
boolean	<a href="#"><u>isBlockNumericFields()</u></a> Determines whether it will be possible to type non-numeric characters in numeric input fields.

boolean	<a href="#"><u>isCaretAtEndOfField()</u></a> Determines whether the caret's position should be at the end of the input fields.
boolean	<a href="#"><u>isDoJapaneseDataTypeValidityCheck()</u></a>
boolean	<a href="#"><u>isEmulationPasteBehavior()</u></a> Determines whether the paste behavior of text will be as in terminal emulators, that once a field is filled, the remaining text is pasted in the following field or whether the text will be pasted in a single field.
boolean	<a href="#"><u>isHandleMenuSelection()</u></a> Enables basic menu support: when clicking on a menu item, the corresponding number is automatically placed in the option field.
boolean	<a href="#"><u>isSelectFieldOnFocus()</u></a> Determines whether when moving the cursor to the following field (using the arrows or TAB key), the field content will be selected and then overwritten or the text will not be selected and when typing, the text will be inserted in the field.
boolean	<a href="#"><u>isShowBlinkingFields()</u></a> Some hosts contain definitions determining that certain fields should blink.
boolean	<a href="#"><u>isTabOnInputsOnly()</u></a> Determines whether the TAB key navigates between the input (unprotected) fields of the screen only, or also between additional buttons and hyperlinks in the Web page.
boolean	<a href="#"><u>isUseKeyboardPfKeys()</u></a> Use PF keys to send actions to the host.
void	<a href="#"><u>set_AutoSkipAllFields</u></a> (boolean autoSkipAllFields) For .NET framework usage.
void	<a href="#"><u>set_BlinkingCaret</u></a> (boolean blinkingCaret) For .NET framework usage.
void	<a href="#"><u>set_BlockNumericFields</u></a> (boolean blockNumericFields) For .NET framework usage.
void	<a href="#"><u>set_CaretAtEndOfField</u></a> (boolean caretAtEndOfField) For .NET framework usage.
void	<a href="#"><u>set_ColorSets</u></a> (String colorSets) For .NET framework usage.
void	<a href="#"><u>set_DoJapaneseDataTypeValidityCheck</u></a> (boolean doJapaneseDataTypeValidityCheck) For .NET framework usage.
void	<a href="#"><u>set_DoubleClickActionKey</u></a> (String doubleClickActionKey) For .NET framework usage.
void	<a href="#"><u>set_EmulationPasteBehavior</u></a> (boolean emulationPasteBehavior) For .NET framework usage.
void	<a href="#"><u>set_HandleMenuSelection</u></a> (boolean handleMenuSelection) For .NET framework usage.
void	<a href="#"><u>set_LabelFocus</u></a> (int labelFocus) For .NET framework usage.

void	<a href="#"><u>set_SelectFieldOnFocus</u></a> (boolean selectFieldOnFocus) For .NET framework usage.
void	<a href="#"><u>set_ShowBlinkingFields</u></a> (boolean showBlinkingFields)
void	<a href="#"><u>set_TabOnInputsOnly</u></a> (boolean tabOnInputsOnly) For .NET framework usage.
void	<a href="#"><u>set_UseArrows</u></a> (int useArrows) For .NET framework usage.
void	<a href="#"><u>set_UseKeyboardPfKeys</u></a> (boolean useKeyboardPfKeys) For .NET framework usage.
void	<a href="#"><u>setAutoSkipAllFields</u></a> (boolean autoSkipAllFields) Determines whether or not the cursor will automatically move to the next input field once the current input field is filled with text (applies to all input fields).
void	<a href="#"><u>setBlinkingCaret</u></a> (boolean blinkingCaret) Displays a blinking caret in input fields
void	<a href="#"><u>setBlockNumericFields</u></a> (boolean blockNumericFields) Determines whether it will be possible to type non-numeric characters in numeric input fields.
void	<a href="#"><u>setCaretAtEndOfField</u></a> (boolean caretAtEndOfField) Determines whether the caret's position should be at the end of the input fields.
void	<a href="#"><u>setColorSets</u></a> (String colorSets) This functionality enables the end user select one of the pre-defined style sheets for instant screens.
void	<a href="#"><u>setDoJapaneseDataTypeValidityCheck</u></a> (boolean doJapaneseDataTypeValidityCheck)
void	<a href="#"><u>setDoubleClickActionKey</u></a> (String doubleClickActionKey) Sends the specified action key to the host when double-clicking on a label.
void	<a href="#"><u>setEmulationPasteBehavior</u></a> (boolean emulationPasteBehavior) Determines whether the paste behavior of text will be as in terminal emulators, that once a field is filled, the remaining text is pasted in the following field or whether the text will be pasted in a single field.
void	<a href="#"><u>setHandleMenuSelection</u></a> (boolean handleMenuSelection) Enables basic menu support: when clicking on a menu item, the corresponding number is automatically placed in the option field.
void	<a href="#"><u>setLabelFocus</u></a> (int labelFocus) This option allows you to determine whether the font style of a protected field will become underline, bold or will not change when a user either clicks on the protected field or moves between the protected fields using the keyboard arrow buttons.
void	<a href="#"><u>setSelectFieldOnFocus</u></a> (boolean selectFieldOnFocus) Determines whether when moving the cursor to the following field (using the arrows or TAB key), the field content will be selected and then overwritten or the text will not be selected and when typing, the text will be inserted in the field.
void	<a href="#"><u>setShowBlinkingFields</u></a> (boolean showBlinkingFields) Some hosts contain definitions determining that certain fields should blink.

void	<a href="#"><code>setTabOnInputsOnly</code></a> (boolean tabOnInputsOnly) Determines whether the TAB key navigates between the input (unprotected) fields of the screen only, or also between additional buttons and hyperlinks in the Web page.
void	<a href="#"><code>setUseArrows</code></a> (int useArrows) This option allows you to determine whether the font style of a protected field will become underline, bold or will not change when a user either clicks on the protected field or moves between the protected fields using the keyboard arrow buttons.
void	<a href="#"><code>setUseKeyboardPfKeys</code></a> (boolean useKeyboardPfKeys) Use PF keys to send actions to the host.

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### ARROWS\_NONE

```
public static final int ARROWS_NONE
```

Constant value: 0

### ARROWS\_INPUT\_FIELDS

```
public static final int ARROWS_INPUT_FIELDS
```

Constant value: 1

### ARROWS\_ALL\_SCREEN

```
public static final int ARROWS_ALL_SCREEN
```

Constant value: 2

### LABEL\_FOCUS\_NONE

```
public static final int LABEL_FOCUS_NONE
```

Constant value: 0

### LABEL\_FOCUS\_UNDERLINE

```
public static final int LABEL_FOCUS_UNDERLINE
```

Constant value: 1

### LABEL\_FOCUS\_BOLD

```
public static final int LABEL_FOCUS_BOLD
```

(continued from last page)

Constant value: 2

## Constructors

### GXInstantEmulatorConfig

```
public GXInstantEmulatorConfig()
```

## Methods

### isAutoSkipAllFields

```
public boolean isAutoSkipAllFields()
```

Determines whether or not the cursor will automatically move to the next input field once the current input field is filled with text (applies to all input fields).

**Returns:**

is skip to the next input field when the current input field is filled with content

### setAutoSkipAllFields

```
public void setAutoSkipAllFields(boolean autoSkipAllFields)
```

Determines whether or not the cursor will automatically move to the next input field once the current input field is filled with text (applies to all input fields).

**Parameters:**

autoSkipAllFields - enables the cursor to automatically move to the next field

### get\_AutoSkipAllFields

```
public boolean get_AutoSkipAllFields()
```

For .NET framework usage. Expose the method as a property.

### set\_AutoSkipAllFields

```
public void set_AutoSkipAllFields(boolean autoSkipAllFields)
```

For .NET framework usage. Expose the method as a property.

### getDoubleClickActionKey

```
public String getDoubleClickActionKey()
```

Sends the specified action key to the host when double-clicking on a label.

**Returns:**

the action key to send

### setDoubleClickActionKey

```
public void setDoubleClickActionKey(String doubleClickActionKey)
```

Sends the specified action key to the host when double-clicking on a label.

(continued from last page)

**Parameters:**

doubleClickActionKey - the action key to send

---

**get\_DoubleClickActionKey**

```
public String get_DoubleClickActionKey()
```

For .NET framework usage. Expose the method as a property.

---

**set\_DoubleClickActionKey**

```
public void set_DoubleClickActionKey(String doubleClickActionKey)
```

For .NET framework usage. Expose the method as a property.

---

**getLabelFocus**

```
public int getLabelFocus()
```

This option allows you to determine whether the font style of a protected field will become underline, bold or will not change when a user either clicks on the protected field or moves between the protected fields using the keyboard arrow buttons.

**Returns:**

the look-&-feel of a label when it is clicked

**See Also:**

[LABEL\\_FOCUS\\_NONE](#)

[LABEL\\_FOCUS\\_UNDERLINE](#)

[LABEL\\_FOCUS\\_BOLD](#)

---

**setLabelFocus**

```
public void setLabelFocus(int labelFocus)
```

This option allows you to determine whether the font style of a protected field will become underline, bold or will not change when a user either clicks on the protected field or moves between the protected fields using the keyboard arrow buttons.

**Parameters:**

labelFocus - the look-&-feel of a label when it is clicked

**See Also:**

[LABEL\\_FOCUS\\_NONE](#)

[LABEL\\_FOCUS\\_UNDERLINE](#)

[LABEL\\_FOCUS\\_BOLD](#)

---

**get\_LabelFocus**

```
public int get_LabelFocus()
```

For .NET framework usage. Expose the method as a property.

---

**set\_LabelFocus**

```
public void set_LabelFocus(int labelFocus)
```

For .NET framework usage. Expose the method as a property.

---

## isTabOnInputsOnly

```
public boolean isTabOnInputsOnly()
```

Determines whether the TAB key navigates between the input (unprotected) fields of the screen only, or also between additional buttons and hyperlinks in the Web page.

**Returns:**

the tabbing behavior

---

## setTabOnInputsOnly

```
public void setTabOnInputsOnly(boolean tabOnInputsOnly)
```

Determines whether the TAB key navigates between the input (unprotected) fields of the screen only, or also between additional buttons and hyperlinks in the Web page.

**Parameters:**

tabOnInputsOnly - the tabbing behavior

---

## get\_TabOnInputsOnly

```
public boolean get_TabOnInputsOnly()
```

For .NET framework usage. Expose the method as a property.

---

## set\_TabOnInputsOnly

```
public void set_TabOnInputsOnly(boolean tabOnInputsOnly)
```

For .NET framework usage. Expose the method as a property.

---

## getUseArrows

```
public int getUseArrows()
```

Determines the arrow keys' navigation behavior (none, input fields or all screen).

**Returns:**

the arrow keys navigation behavior

**See Also:**

[ARROWS\\_NONE](#)

[ARROWS\\_INPUT\\_FIELDS](#)

[ARROWS\\_ALL\\_SCREEN](#)

---

## setUseArrows

```
public void setUseArrows(int useArrows)
```

This option allows you to determine whether the font style of a protected field will become underline, bold or will not change when a user either clicks on the protected field or moves between the protected fields using the keyboard arrow buttons.

**Parameters:**

useArrows - the arrow keys navigation behavior

**See Also:**

(continued from last page)

[ARROWS\\_NONE](#)[ARROWS\\_INPUT\\_FIELDS](#)[ARROWS\\_ALL\\_SCREEN](#)

---

## get\_UseArrows

```
public int get_UseArrows()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseArrows

```
public void set_UseArrows(int useArrows)
```

For .NET framework usage. Expose the method as a property.

---

## isUseKeyboardPfKeys

```
public boolean isUseKeyboardPfKeys()
```

Use PF keys to send actions to the host. Default value: true.

### Returns:

if the pf keys on the browser are activated

---

## setUseKeyboardPfKeys

```
public void setUseKeyboardPfKeys(boolean useKeyboardPfKeys)
```

Use PF keys to send actions to the host. Default value: true.

### Parameters:

useKeyboardPfKeys - if the pf keys on the browser should be activated

---

## get\_UseKeyboardPfKeys

```
public boolean get_UseKeyboardPfKeys()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseKeyboardPfKeys

```
public void set_UseKeyboardPfKeys(boolean useKeyboardPfKeys)
```

For .NET framework usage. Expose the method as a property.

---

## getColorSets

```
public String getColorSets()
```

This functionality enables the end user select one of the pre-defined style sheets for instant screens. In the emulation template this functionality is available by default ("change color set" link in the page footer). These relative css file names must be separated by a comma. For example: css/styles\_white.css,css/styles\_black.css

### Returns:

list of relative css file names separate by comma

---

(continued from last page)

---

## setColorSets

```
public void setColorSets(String colorSets)
```

This functionality enables the end user select one of the pre-defined style sheets for instant screens. In the emulation template this functionality is available by default ("change color set" link in the page footer). These relative css file names must be separated by a comma. For example: css/styles\_white.css,css/styles\_black.css

### Parameters:

colorSets - list of relative css file names separate by comma

---

## get\_ColorSets

```
public String get_ColorSets()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ColorSets

```
public void set_ColorSets(String colorSets)
```

For .NET framework usage. Expose the method as a property.

---

## isBlockNumericFields

```
public boolean isBlockNumericFields()
```

Determines whether it will be possible to type non-numeric characters in numeric input fields. The type of the field is a property of the host field which the input field is based on.

### Returns:

if to block numeric fields

---

## setBlockNumericFields

```
public void setBlockNumericFields(boolean blockNumericFields)
```

Determines whether it will be possible to type non-numeric characters in numeric input fields. The type of the field is a property of the host field which the input field is based on.

### Parameters:

blockNumericFields - if to block numeric fields

---

## get\_BlockNumericFields

```
public boolean get_BlockNumericFields()
```

For .NET framework usage. Expose the method as a property.

---

## set\_BlockNumericFields

```
public void set_BlockNumericFields(boolean blockNumericFields)
```

For .NET framework usage. Expose the method as a property.

---

## isCaretAtEndOfField

```
public boolean isCaretAtEndOfField()
```

---

(continued from last page)

Determines whether the caret's position should be at the end of the input fields.

**Returns:**

if to place caret position at the end of input fields

---

## setCaretAtEndOfField

```
public void setCaretAtEndOfField(boolean caretAtEndOfField)
```

Determines whether the caret's position should be at the end of the input fields. Not applicable when "Select content when content on input field" is active.

**Parameters:**

caretAtEndOfField - if to place caret position at the end of input fields

---

## get\_CaretAtEndOfField

```
public boolean get_CaretAtEndOfField()
```

For .NET framework usage. Expose the method as a property.

---

## set\_CaretAtEndOfField

```
public void set_CaretAtEndOfField(boolean caretAtEndOfField)
```

For .NET framework usage. Expose the method as a property.

---

## isHandleMenuSelection

```
public boolean isHandleMenuSelection()
```

Enables basic menu support: when clicking on a menu item, the corresponding number is automatically placed in the option field. The screen should contain only one input field. The clicked label (menu item) must have a number next to it.

**Returns:**

if to support menu selection

---

## setHandleMenuSelection

```
public void setHandleMenuSelection(boolean handleMenuSelection)
```

Enables basic menu support: when clicking on a menu item, the corresponding number is automatically placed in the option field. The screen should contain only one input field. The clicked label (menu item) must have a number next to it.

**Parameters:**

handleMenuSelection - if to support menu selection

---

## get\_HandleMenuSelection

```
public boolean get_HandleMenuSelection()
```

For .NET framework usage. Expose the method as a property.

---

## set\_HandleMenuSelection

```
public void set_HandleMenuSelection(boolean handleMenuSelection)
```

For .NET framework usage. Expose the method as a property.

---

## isSelectFieldOnFocus

```
public boolean isSelectFieldOnFocus( )
```

Determines whether when moving the cursor to the following field (using the arrows or TAB key), the field content will be selected and then overwritten or the text will not be selected and when typing, the text will be inserted in the field.

**Returns:**

if to select the content of input field when it is focus

---

## setSelectFieldOnFocus

```
public void setSelectFieldOnFocus(boolean selectFieldOnFocus)
```

Determines whether when moving the cursor to the following field (using the arrows or TAB key), the field content will be selected and then overwritten or the text will not be selected and when typing, the text will be inserted in the field.

**Parameters:**

selectFieldOnFocus - if to select the content of input field when it is focus

---

## get\_SelectFieldOnFocus

```
public boolean get_SelectFieldOnFocus( )
```

For .NET framework usage. Expose the method as a property.

---

## set\_SelectFieldOnFocus

```
public void set_SelectFieldOnFocus(boolean selectFieldOnFocus)
```

For .NET framework usage. Expose the method as a property.

---

## isShowBlinkingFields

```
public boolean isShowBlinkingFields( )
```

Some hosts contain definitions determining that certain fields should blink. This feature enables these fields to blink in the browser.

**Returns:**

if to show blinking host fields to blink in the browser

---

## setShowBlinkingFields

```
public void setShowBlinkingFields(boolean showBlinkingFields)
```

Some hosts contain definitions determining that certain fields should blink. This feature enables these fields to blink in the browser.

**Parameters:**

showBlinkingFields - if to show blinking host fields to blink in the browser

---

## get\_ShowBlinkingFields

```
public boolean get_ShowBlinkingFields( )
```

For .NET framework usage. Expose the method as a property.

---

---

## set\_ShowBlinkingFields

```
public void set_ShowBlinkingFields(boolean showBlinkingFields)
```

### Parameters:

showBlinkingFields

---

## isDoJapaneseDataTypeValidityCheck

```
public boolean isDoJapaneseDataTypeValidityCheck()
```

---

## setDoJapaneseDataTypeValidityCheck

```
public void setDoJapaneseDataTypeValidityCheck(boolean  
doJapaneseDataTypeValidityCheck)
```

---

## get\_DoJapaneseDataTypeValidityCheck

```
public boolean get_DoJapaneseDataTypeValidityCheck()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DoJapaneseDataTypeValidityCheck

```
public void set_DoJapaneseDataTypeValidityCheck(boolean  
doJapaneseDataTypeValidityCheck)
```

For .NET framework usage. Expose the method as a property.

---

## isBlinkingCaret

```
public boolean isBlinkingCaret()
```

---

## setBlinkingCaret

```
public void setBlinkingCaret(boolean blinkingCaret)
```

Displays a blinking caret in input fields

### Parameters:

blinkingCaret

---

## get\_BlinkingCaret

```
public boolean get_BlinkingCaret()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

---

## set\_BlinkingCaret

```
public void set_BlinkingCaret(boolean blinkingCaret)
```

For .NET framework usage. Expose the method as a property.

---

## isEmulationPasteBehavior

```
public boolean isEmulationPasteBehavior()
```

Determines whether the paste behavior of text will be as in terminal emulators, that once a field is filled, the remaining text is pasted in the following field or whether the text will be pasted in a single field. Only supported in Internet Explorer.

**Returns:**

---

## setEmulationPasteBehavior

```
public void setEmulationPasteBehavior(boolean emulationPasteBehavior)
```

Determines whether the paste behavior of text will be as in terminal emulators, that once a field is filled, the remaining text is pasted in the following field or whether the text will be pasted in a single field. Only supported in Internet Explorer.

**Parameters:**

emulationPasteBehavior

---

## get\_EmulationPasteBehavior

```
public boolean get_EmulationPasteBehavior()
```

For .NET framework usage. Expose the method as a property.

---

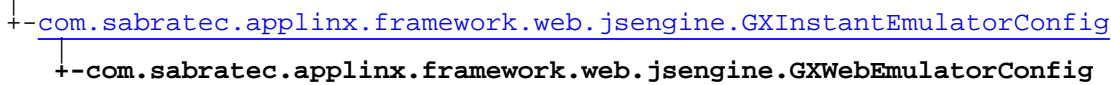
## set\_EmulationPasteBehavior

```
public void set_EmulationPasteBehavior(boolean emulationPasteBehavior)
```

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.framework.web.jsengine Class GXWebEmulatorConfig

java.lang.Object



public class **GXWebEmulatorConfig**  
extends [GXInstantEmulatorConfig](#)

Configuration class for html emulation web pages (generated & instant). Generated to XML file and loaded on the client side

Fields inherited from class [com.sabratec.applinx.framework.web.jsengine.GXInstantEmulatorConfig](#)

[ARROWS\\_ALL\\_SCREEN](#), [ARROWS\\_INPUT\\_FIELDS](#), [ARROWS\\_NONE](#), [LABEL\\_FOCUS\\_BOLD](#), [LABEL\\_FOCUS\\_NONE](#), [LABEL\\_FOCUS\\_UNDERLINE](#)

### Constructor Summary

public	<a href="#">GXWebEmulatorConfig()</a>
--------	---------------------------------------

### Method Summary

boolean	<a href="#">get_AttachInputTagEvents()</a> For .NET framework usage.
boolean	<a href="#">get_CheckHostScreenUpdate()</a> For .NET framework usage.
boolean	<a href="#">get_LoadEmulationComponent()</a> For .NET framework usage.
boolean	<a href="#">get_LoadPrinter()</a> For .NET framework usage.
boolean	<a href="#">get_SupportTypeAhead()</a> For .NET framework usage.
String	<a href="#">get_UserExitsFile()</a> For .NET framework usage.
String	<a href="#">getUserExitsFile()</a> Defines a user exits file for the framework
boolean	<a href="#">isAttachInputTagEvents()</a> Determine whether to capture input tag events : onkey, onkeyup, onkeypress, onselectstart, onpaste, onblur, onclick.
boolean	<a href="#">isCheckHostScreenUpdate()</a> Determines whether to continuously check in the background for changes in the host screen.
boolean	<a href="#">isLoadEmulationComponent()</a> <b>Deprecated.</b> <i>should be configured for Natrual data transfer or type ahead seperatly</i>

boolean	<a href="#"><u>isLoadPrinter()</u></a> Used to determine whether to load the printer applet within internal frame.
boolean	<a href="#"><u>isSupportTypeAhead()</u></a>
void	<a href="#"><u>set_AttachInputTagEvents</u></a> (boolean attachInputTagEvents) For .NET framework usage.
void	<a href="#"><u>set_CheckHostScreenUpdate</u></a> (boolean checkHostScreenUpdate) For .NET framework usage.
void	<a href="#"><u>set_LoadEmulationComponent</u></a> (boolean isLoadEmulationComponent) For .NET framework usage.
void	<a href="#"><u>set_LoadPrinter</u></a> (boolean isLoadPrinter) For .NET framework usage.
void	<a href="#"><u>set_SupportTypeAhead</u></a> (boolean supportTypeAhead) For .NET framework usage.
void	<a href="#"><u>set_UserExitsFile</u></a> (String userExitsFile) For .NET framework usage.
void	<a href="#"><u>setAttachInputTagEvents</u></a> (boolean attachInputTagEvents) Determine whether to capture input tag events : onkey, onkeyup, onkeypress, onselectstart, onpaste, onblur, onclick.
void	<a href="#"><u>setCheckHostScreenUpdate</u></a> (boolean checkHostScreenUpdate) Determines whether to continuously check in the background for changes in the host screen.
void	<a href="#"><u>setLoadEmulationComponent</u></a> (boolean isLoadEmulationComponent) <b>Deprecated.</b> <i>should be configured for Natrual data transfer or type ahead seperatly</i>
void	<a href="#"><u>setLoadPrinter</u></a> (boolean isLoadPrinter) Used to determine whether to load the printer applet within internal frame.
void	<a href="#"><u>setSupportTypeAhead</u></a> (boolean supportTypeAhead) Determines whether to enable the user to type continuously, without waiting for the browser to display the next page.
void	<a href="#"><u>setUserExitsFile</u></a> (String userExitsFile) Defines a user exits file for the framework

**Methods inherited from class**[com.sabratec.applinx.framework.web.jsengine.GXInstantEmulatorConfig](#)

[get\\_AutoSkipAllFields](#), [get\\_BlinkingCaret](#), [get\\_BlockNumericFields](#),  
[get\\_CaretAtEndOfField](#), [get\\_ColorSets](#), [get\\_DoJapaneseDataTypeValidityCheck](#),  
[get\\_DoubleClickActionKey](#), [get\\_EmulationPasteBehavior](#), [get\\_HandleMenuSelection](#),  
[get\\_LabelFocus](#), [get\\_SelectFieldOnFocus](#), [get\\_ShowBlinkingFields](#), [get\\_TabOnInputsOnly](#),  
[get\\_UseArrows](#), [get\\_UseKeyboardPfKeys](#), [getColorSets](#), [getDoubleClickActionKey](#),  
[getLabelFocus](#), [getUseArrows](#), [isAutoSkipAllFields](#), [isBlinkingCaret](#),  
[isBlockNumericFields](#), [isCaretAtEndOfField](#), [isDoJapaneseDataTypeValidityCheck](#),  
[isEmulationPasteBehavior](#), [isHandleMenuSelection](#), [isSelectFieldOnFocus](#),  
[isShowBlinkingFields](#), [isTabOnInputsOnly](#), [isUseKeyboardPfKeys](#), [set\\_AutoSkipAllFields](#),  
[set\\_BlinkingCaret](#), [set\\_BlockNumericFields](#), [set\\_CaretAtEndOfField](#), [set\\_ColorSets](#),  
[set\\_DoJapaneseDataTypeValidityCheck](#), [set\\_DoubleClickActionKey](#),  
[set\\_EmulationPasteBehavior](#), [set\\_HandleMenuSelection](#), [set\\_LabelFocus](#),  
[set\\_SelectFieldOnFocus](#), [set\\_ShowBlinkingFields](#), [set\\_TabOnInputsOnly](#), [set\\_UseArrows](#),  
[set\\_UseKeyboardPfKeys](#), [setAutoSkipAllFields](#), [setBlinkingCaret](#), [setBlockNumericFields](#),  
[setCaretAtEndOfField](#), [setColorSets](#), [setDoJapaneseDataTypeValidityCheck](#),  
[setDoubleClickActionKey](#), [setEmulationPasteBehavior](#), [setHandleMenuSelection](#),  
[setLabelFocus](#), [setSelectFieldOnFocus](#), [setShowBlinkingFields](#), [setTabOnInputsOnly](#),  
[setUseArrows](#), [setUseKeyboardPfKeys](#)

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXWebEmulatorConfig

```
public GXWebEmulatorConfig()
```

## Methods

### isLoadPrinter

```
public boolean isLoadPrinter()
```

Used to determine whether to load the printer applet within internal frame. The printlet is loaded into an invisible frame rather than opened in a new window. Clicking the printer link will display the printlet in the main window, clicking Hide printer in the printlet screen will make it disappear and redisplay the session frame.

#### Returns:

if to load the printer applet within internal frame

### setLoadPrinter

```
public void setLoadPrinter(boolean isLoadPrinter)
```

Used to determine whether to load the printer applet within internal frame. The printlet is loaded into an invisible frame rather than opened in a new window. Clicking the printer link will display the printlet in the main window, clicking Hide printer in the printlet screen will make it disappear and redisplay the session frame.

#### Parameters:

`isLoadPrinter` - determine if to load the printer applet within internal frame

---

## get\_LoadPrinter

```
public boolean get_LoadPrinter()
```

For .NET framework usage. Expose the method as a property.

---

## set\_LoadPrinter

```
public void set_LoadPrinter(boolean isLoadPrinter)
```

For .NET framework usage. Expose the method as a property.

---

## isLoadEmulationComponent

```
public boolean isLoadEmulationComponent()
```

**Deprecated.** *should be configured for Natrual data transfer or type ahead seperatly*

---

## setLoadEmulationComponent

```
public void setLoadEmulationComponent(boolean isLoadEmulationComponent)
```

**Deprecated.** *should be configured for Natrual data transfer or type ahead seperatly*

---

## getUserExitsFile

```
public String getUserExitsFile()
```

Defines a user exits file for the framework

**Returns:**

a user exits file for the framework

---

## setUserExitsFile

```
public void setUserExitsFile(String userExitsFile)
```

Defines a user exits file for the framework

**Parameters:**

userExitsFile - a user exits file for the framework

---

## get\_UserExitsFile

```
public String get_UserExitsFile()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UserExitsFile

```
public void set_UserExitsFile(String userExitsFile)
```

For .NET framework usage. Expose the method as a property.

---

---

## isSupportTypeAhead

```
public boolean isSupportTypeAhead()
```

---

## setSupportTypeAhead

```
public void setSupportTypeAhead(boolean supportTypeAhead)
```

Determines whether to enable the user to type continuously, without waiting for the browser to display the next page. This feature uses an ActiveX component (works in IE only). This feature does not work when a modal window is open, or during the process of opening/closing a pop-up window.

**Parameters:**

supportTypeAhead

---

---

## get\_SupportTypeAhead

```
public boolean get_SupportTypeAhead()
```

For .NET framework usage. Expose the method as a property.

---

---

## set\_SupportTypeAhead

```
public void set_SupportTypeAhead(boolean supportTypeAhead)
```

For .NET framework usage. Expose the method as a property.

---

---

## isCheckHostScreenUpdate

```
public boolean isCheckHostScreenUpdate()
```

Determines whether to continuously check in the background for changes in the host screen. If the host screen has changed, the browser is refreshed and a JavaScript event is triggered. As the server is checked continuously, more server resources are required for this. By default, the server is checked every 3, 6, 12 and then every 24 seconds. These intervals can be customized using user exits (refer to the ApplinX documentation for further details).

**Returns:**

---

## setCheckHostScreenUpdate

```
public void setCheckHostScreenUpdate(boolean checkHostScreenUpdate)
```

Determines whether to continuously check in the background for changes in the host screen. If the host screen has changed, the browser is refreshed and a JavaScript event is triggered. As the server is checked continuously, more server resources are required for this. By default, the server is checked every 3, 6, 12 and then every 24 seconds. These intervals can be customized using user exits (refer to the ApplinX documentation for further details).

**Parameters:**

checkHostScreenUpdate

---

---

## get\_CheckHostScreenUpdate

```
public boolean get_CheckHostScreenUpdate()
```

For .NET framework usage. Expose the method as a property.

---

---

## set\_CheckHostScreenUpdate

```
public void set_CheckHostScreenUpdate(boolean checkHostScreenUpdate)
```

For .NET framework usage. Expose the method as a property.

---

## setAttachInputTagEvents

```
public void setAttachInputTagEvents(boolean attachInputTagEvents)
```

Determine whether to capture input tag events : onkey, onkeyup, onkeypress, onselectstart, onpaste, onblur, onclick. Should be used only when heavy pages contains a huge amount of input tags (over 100) which may slow the web page load time. Disclaimer \*\*\*\*\*: The following features will not work when this configuration is set to false: Blinking caret, Block numeric fields, auto skip, right adjust, auto enter, select field on focus, emulation paste behavior

### Parameters:

attachInputTagEvents

---

## isAttachInputTagEvents

```
public boolean isAttachInputTagEvents()
```

Determine whether to capture input tag events : onkey, onkeyup, onkeypress, onselectstart, onpaste, onblur, onclick. Should be used only when heavy pages contains a huge amount of input tags (over 100) which may slow the web page load time. Disclaimer \*\*\*\*\*: The following features will not work when this configuration is set to false: Blinking caret, Block numeric fields, auto skip, right adjust, auto enter, select field on focus, emulation paste behavior

### Returns:

---

## set\_AttachInputTagEvents

```
public void set_AttachInputTagEvents(boolean attachInputTagEvents)
```

For .NET framework usage. Expose the method as a property.

---

## get\_AttachInputTagEvents

```
public boolean get_AttachInputTagEvents()
```

For .NET framework usage. Expose the method as a property.

---

## get\_LoadEmulationComponent

```
public boolean get_LoadEmulationComponent()
```

For .NET framework usage. Expose the method as a property.

---

## set\_LoadEmulationComponent

```
public void set_LoadEmulationComponent(boolean isLoadEmulationComponent)
```

For .NET framework usage. Expose the method as a property.

---

## com.sabratec.applinx.framework.web.jsengine Class GXWindowConfig

java.lang.Object

└-com.sabratec.applinx.framework.web.jsengine.GXWindowConfig

public class **GXWindowConfig**  
extends Object

### Constructor Summary

public	<a href="#">GXWindowConfig()</a>
--------	----------------------------------

### Method Summary

int	<a href="#">get_DefaultOpenedWindowHeight()</a> For .NET framework usage.
int	<a href="#">get_DefaultOpenedWindowLeft()</a> For .NET framework usage.
int	<a href="#">get_DefaultOpenedWindowTop()</a> For .NET framework usage.
int	<a href="#">get_DefaultOpenedWindowWidth()</a> For .NET framework usage.
String	<a href="#">get_OpenedWindowAttributes()</a> For .NET framework usage.
boolean	<a href="#">get_OpenWindowCenter()</a> For .NET framework usage.
String	<a href="#">get_PopupCloseSentKey()</a> For .NET framework usage.
int	<a href="#">getDefaultOpenedWindowHeight()</a> Determines the default window's height when using the method <code>gx_window.open</code>
int	<a href="#">getDefaultOpenedWindowLeft()</a> Determines the default window's left position when using the method <code>gx_window.open</code>
int	<a href="#">getDefaultOpenedWindowTop()</a> Determines the default window's top position when using the method <code>gx_window.open</code>
int	<a href="#">getDefaultOpenedWindowWidth()</a> Determines the default window's width when using the method <code>gx_window.open</code>
String	<a href="#">getOpenedWindowAttributes()</a> Determines the opened window's attributes when using the method <code>gx_window.open</code>
String	<a href="#">getPopupCloseSentKey()</a> Determines the key to send to the host when the modal window is closed

boolean	<a href="#"><u>isOpenWindowCenter()</u></a> According to the selected radio button, determines whether the popup window is opened in the center of the screen or in a specific position.
void	<a href="#"><u>set_DefaultOpenedWindowHeight(int defaultOpenedWindowHeight)</u></a> For .NET framework usage.
void	<a href="#"><u>set_DefaultOpenedWindowLeft(int defaultOpenedWindowLeft)</u></a> For .NET framework usage.
void	<a href="#"><u>set_DefaultOpenedWindowTop(int defaultOpenedWindowTop)</u></a> For .NET framework usage.
void	<a href="#"><u>set_DefaultOpenedWindowWidth(int defaultOpenedWindowWidth)</u></a> For .NET framework usage.
void	<a href="#"><u>set_OpenedWindowAttributes(String openedWindowAttributes)</u></a> For .NET framework usage.
void	<a href="#"><u>set_OpenWindowCenter(boolean openWindowCenter)</u></a> For .NET framework usage.
void	<a href="#"><u>set_PopupCloseSentKey(String popupCloseSentKey)</u></a> For .NET framework usage.
void	<a href="#"><u>setDefaultOpenedWindowHeight(int defaultOpenedWindowHeight)</u></a> Determines the default window's height when using the method <code>gx_window.open</code>
void	<a href="#"><u>setDefaultOpenedWindowLeft(int defaultOpenedWindowLeft)</u></a> Determines the default window's left position when using the method <code>gx_window.open</code>
void	<a href="#"><u>setDefaultOpenedWindowTop(int defaultOpenedWindowTop)</u></a> Determines the default window's top position when using the method <code>gx_window.open</code>
void	<a href="#"><u>setDefaultOpenedWindowWidth(int defaultOpenedWindowWidth)</u></a> Determines the default window's width when using the method <code>gx_window.open</code>
void	<a href="#"><u>setOpenedWindowAttributes(String openedWindowAttributes)</u></a> Determines the opened window's attributes when using the method <code>gx_window.open</code>
void	<a href="#"><u>setOpenWindowCenter(boolean openWindowCenter)</u></a> According to the selected radio button, determines whether the popup window is opened in the center of the screen or in a specific position.
void	<a href="#"><u>setPopupCloseSentKey(String popupCloseSentKey)</u></a> Determines the key to send to the host when the modal window is closed

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXWindowConfig

```
public GXWindowConfig()
```

(continued from last page)

## Methods

### getDefaultOpenedWindowHeight

```
public int getDefaultOpenedWindowHeight()
```

Determines the default window's height when using the method `gx_window.open`

**Returns:**

the default window height

---

### setDefaultOpenedWindowHeight

```
public void setDefaultOpenedWindowHeight(int defaultOpenedWindowHeight)
```

Determines the default window's height when using the method `gx_window.open`

**Parameters:**

`defaultOpenedWindowHeight` - the default window height

---

### getDefaultOpenedWindowLeft

```
public int getDefaultOpenedWindowLeft()
```

Determines the default window's left position when using the method `gx_window.open`

**Returns:**

the default window left position

---

### setDefaultOpenedWindowLeft

```
public void setDefaultOpenedWindowLeft(int defaultOpenedWindowLeft)
```

Determines the default window's left position when using the method `gx_window.open`

**Parameters:**

`defaultOpenedWindowLeft` - the default window left position

---

### isOpenWindowCenter

```
public boolean isOpenWindowCenter()
```

According to the selected radio button, determines whether the popup window is opened in the center of the screen or in a specific position.

**Returns:**

`openWindowCenter`

---

### setOpenWindowCenter

```
public void setOpenWindowCenter(boolean openWindowCenter)
```

According to the selected radio button, determines whether the popup window is opened in the center of the screen or in a specific position.

**Parameters:**

`openWindowCenter`

---

## getDefaultOpenedWindowTop

```
public int getDefaultOpenedWindowTop()
```

Determines the default window's top position when using the method `gx_window.open`

**Returns:**

the default window top position

---

## setDefaultOpenedWindowTop

```
public void setDefaultOpenedWindowTop(int defaultOpenedWindowTop)
```

Determines the default window's top position when using the method `gx_window.open`

**Parameters:**

`defaultOpenedWindowTop` - the default window top position

---

## getDefaultOpenedWindowWidth

```
public int getDefaultOpenedWindowWidth()
```

Determines the default window's width when using the method `gx_window.open`

**Returns:**

the default window width

---

## setDefaultOpenedWindowWidth

```
public void setDefaultOpenedWindowWidth(int defaultOpenedWindowWidth)
```

Determines the default window's width when using the method `gx_window.open`

**Parameters:**

`defaultOpenedWindowWidth` - the default window width

---

## getOpenedWindowAttributes

```
public String getOpenedWindowAttributes()
```

Determines the opened window's attributes when using the method `gx_window.open`

**Returns:**

the opened window attributes

---

## setOpenedWindowAttributes

```
public void setOpenedWindowAttributes(String openedWindowAttributes)
```

Determines the opened window's attributes when using the method `gx_window.open`

**Parameters:**

`openedWindowAttributes` - the opened window attributes

---

## getPopupCloseSentKey

```
public String getPopupCloseSentKey()
```

---

(continued from last page)

Determines the key to send to the host when the modal window is closed

**Returns:**

---

## setPopupCloseSentKey

```
public void setPopupCloseSentKey(String popupCloseSentKey)
```

Determines the key to send to the host when the modal window is closed

**Parameters:**

popupCloseSentKey

---

## get\_DefaultOpenedWindowHeight

```
public int get_DefaultOpenedWindowHeight()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DefaultOpenedWindowHeight

```
public void set_DefaultOpenedWindowHeight(int defaultOpenedWindowHeight)
```

For .NET framework usage. Expose the method as a property.

---

## get\_DefaultOpenedWindowLeft

```
public int get_DefaultOpenedWindowLeft()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DefaultOpenedWindowLeft

```
public void set_DefaultOpenedWindowLeft(int defaultOpenedWindowLeft)
```

For .NET framework usage. Expose the method as a property.

---

## get\_DefaultOpenedWindowTop

```
public int get_DefaultOpenedWindowTop()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DefaultOpenedWindowTop

```
public void set_DefaultOpenedWindowTop(int defaultOpenedWindowTop)
```

For .NET framework usage. Expose the method as a property.

---

## get\_DefaultOpenedWindowWidth

```
public int get_DefaultOpenedWindowWidth()
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## **set\_DefaultOpenedWindowWidth**

```
public void set_DefaultOpenedWindowWidth(int defaultOpenedWindowWidth)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_OpenedWindowAttributes**

```
public String get_OpenedWindowAttributes()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_OpenedWindowAttributes**

```
public void set_OpenedWindowAttributes(String openedWindowAttributes)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_PopupCloseSentKey**

```
public String get_PopupCloseSentKey()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_PopupCloseSentKey**

```
public void set_PopupCloseSentKey(String popupCloseSentKey)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_OpenWindowCenter**

```
public boolean get_OpenWindowCenter()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_OpenWindowCenter**

```
public void set_OpenWindowCenter(boolean openWindowCenter)
```

For .NET framework usage. Expose the method as a property.

---

---

**Package**

**com.sabratec.applinx.framework.web.nd  
t**

# com.sabratec.applinx.framework.web.ndt

## Class GXNdtConfig

java.lang.Object

└─com.sabratec.applinx.framework.web.ndt.GXNdtConfig

### All Implemented Interfaces:

java.io.Serializable

public class **GXNdtConfig**  
 extends Object  
 implements java.io.Serializable

## Constructor Summary

public	<a href="#">GXNdtConfig()</a>
--------	-------------------------------

## Method Summary

boolean	<a href="#">get_AutomaticDownload()</a> For .NET framework usage.
String	<a href="#">get_DefaultExtension()</a> For .NET framework usage.
boolean	<a href="#">get_DownloadAtOnce()</a> For .NET framework usage.
String	<a href="#">get_HostDownloadForm()</a> For .NET framework usage.
String	<a href="#">get_UploadForm()</a> For .NET framework usage.
boolean	<a href="#">get_UseActiveX()</a> For .NET framework usage.
String	<a href="#">get_WebDownloadForm()</a> For .NET framework usage.
String	<a href="#">getDefaultExtension()</a>
String	<a href="#">getHostDownloadForm()</a> For internal usage
String	<a href="#">getUploadForm()</a> For internal usage
String	<a href="#">getWebDownloadForm()</a> For internal usage

boolean	<a href="#"><u>isAutomaticDownload()</u></a> Determines whether to automatically start downloading a file, when the first download screen is displayed.
boolean	<a href="#"><u>isDownloadAtOnce()</u></a> This parameter determines whether to download the file as a whole, or whether to download the file in blocks, appending each block to the existing file.
boolean	<a href="#"><u>isUseActiveX()</u></a> Determine whether to use ActiveX for natural data transfer
void	<a href="#"><u>set_AutomaticDownload</u></a> (boolean isAutomaticDownload) For .NET framework usage.
void	<a href="#"><u>set_DefaultExtension</u></a> (String defaultExtension) For .NET framework usage.
void	<a href="#"><u>set_DownloadAtOnce</u></a> (boolean downloadAtOnce) For .NET framework usage.
void	<a href="#"><u>set_HostDownloadForm</u></a> (String hostDownloadForm) For .NET framework usage.
void	<a href="#"><u>set_UploadForm</u></a> (String uploadForm) For .NET framework usage.
void	<a href="#"><u>set_UseActiveX</u></a> (boolean useActiveX) For .NET framework usage.
void	<a href="#"><u>set_WebDownloadForm</u></a> (String webDownloadForm) For .NET framework usage.
void	<a href="#"><u>setAutomaticDownload</u></a> (boolean isAutomaticDownload) Determines whether to automatically start downloading a file, when the first download screen is displayed.
void	<a href="#"><u>setDefaultExtension</u></a> (String defaultExtension) Enables defining the default file extension: TXT or CSV.
void	<a href="#"><u>setDownloadAtOnce</u></a> (boolean downloadAtOnce) This parameter determines whether to download the file as a whole, or whether to download the file in blocks, appending each block to the existing file.
void	<a href="#"><u>setHostDownloadForm</u></a> (String hostDownloadForm) For internal usage
void	<a href="#"><u>setUploadForm</u></a> (String uploadForm) For internal usage
void	<a href="#"><u>setUseActiveX</u></a> (boolean isUseActiveX) Defining to use the ActiveX component, enables the work flow's behavior to be identical to that of Entire Connection.
void	<a href="#"><u>setWebDownloadForm</u></a> (String webDownloadForm) For internal usage

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

## Constructors

### **GXNdtConfig**

```
public GXNdtConfig()
```

---

## Methods

### **getHostDownloadForm**

```
public String getHostDownloadForm()
```

For internal usage

---

### **setHostDownloadForm**

```
public void setHostDownloadForm(String hostDownloadForm)
```

For internal usage

---

### **getWebDownloadForm**

```
public String getWebDownloadForm()
```

For internal usage

---

### **setWebDownloadForm**

```
public void setWebDownloadForm(String webDownloadForm)
```

For internal usage

---

### **getUploadForm**

```
public String getUploadForm()
```

For internal usage

---

### **setUploadForm**

```
public void setUploadForm(String uploadForm)
```

For internal usage

---

### **isAutomaticDownload**

```
public boolean isAutomaticDownload()
```

Determines whether to automatically start downloading a file, when the first download screen is displayed. When not selected, a message appears informing you that the host requires you to download a Natural file. If there is a record count from the download, it will be displayed on the page.

**Returns:**

---

## setAutomaticDownload

```
public void setAutomaticDownload(boolean isAutomaticDownload)
```

Determines whether to automatically start downloading a file, when the first download screen is displayed. When not selected, a message appears informing you that the host requires you to download a Natural file. If there is a record count from the download, it will be displayed on the page.

### Parameters:

`isAutomaticDownload` - detrmines if to automatically start download a file, when the first download screen is displayed

---

## get\_AutomaticDownload

```
public boolean get_AutomaticDownload()
```

For .NET framework usage. Expose the method as a property.

---

## set\_AutomaticDownload

```
public void set_AutomaticDownload(boolean isAutomaticDownload)
```

For .NET framework usage. Expose the method as a property.

---

## getDefaultExtension

```
public String getDefaultExtension()
```

---

## setDefaultExtension

```
public void setDefaultExtension(String defaultExtension)
```

Enables defining the default file extension: TXT or CSV.

### Parameters:

`defaultExtension`

---

## get\_DefaultExtension

```
public String get_DefaultExtension()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DefaultExtension

```
public void set_DefaultExtension(String defaultExtension)
```

For .NET framework usage. Expose the method as a property.

---

## isActiveX

```
public boolean isActiveX()
```

Determine whether to use ActiveX for natural data transfer

### Returns:

(continued from last page)

whether to use ActiveX for natural data transfer

---

## setUseActiveX

```
public void setUseActiveX(boolean isUseActiveX)
```

Defining to use the ActiveX component, enables the work flow's behavior to be identical to that of Entire Connection. This is relevant for Internet Explorer only.

### Parameters:

isUseActiveX

---

## get\_UseActiveX

```
public boolean get_UseActiveX()
```

For .NET framework usage. Expose the method as a property.

---

## set\_UseActiveX

```
public void set_UseActiveX(boolean useActiveX)
```

For .NET framework usage. Expose the method as a property.

---

## isDownloadAtOnce

```
public boolean isDownloadAtOnce()
```

This parameter determines whether to download the file as a whole, or whether to download the file in blocks, appending each block to the existing file. This parameter is typically set when expecting to download large files and is only relevant when working with ActiveX and with non Natural UNIX protocols which support Natural data transfer.

### Returns:

---

## setDownloadAtOnce

```
public void setDownloadAtOnce(boolean downloadAtOnce)
```

This parameter determines whether to download the file as a whole, or whether to download the file in blocks, appending each block to the existing file. This parameter is typically set when expecting to download large files and is only relevant when working with ActiveX and with non Natural UNIX protocols which support Natural data transfer.

### Parameters:

downloadAtOnce

---

## get\_DownloadAtOnce

```
public boolean get_DownloadAtOnce()
```

For .NET framework usage. Expose the method as a property.

---

## set\_DownloadAtOnce

```
public void set_DownloadAtOnce(boolean downloadAtOnce)
```

For .NET framework usage. Expose the method as a property.

---

(continued from last page)

## **get\_HostDownloadForm**

```
public String get_HostDownloadForm( )
```

For .NET framework usage. Expose the method as a property.

---

## **set\_HostDownloadForm**

```
public void set_HostDownloadForm(String hostDownloadForm)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_WebDownloadForm**

```
public String get_WebDownloadForm( )
```

For .NET framework usage. Expose the method as a property.

---

## **set\_WebDownloadForm**

```
public void set_WebDownloadForm(String webDownloadForm)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_UploadForm**

```
public String get_UploadForm( )
```

For .NET framework usage. Expose the method as a property.

---

## **set\_UploadForm**

```
public void set_UploadForm(String uploadForm)
```

For .NET framework usage. Expose the method as a property.

---

---

**Package**

**com.sabratec.applinx.framework.web.ss**

**0**

## com.sabratec.applinx.framework.web.sso Class GXAbstractHttpUserPasswordRetriever

java.lang.Object

└─com.sabratec.applinx.framework.web.sso.GXAbstractHttpUserPasswordRetriever

All Implemented Interfaces:

[GXUserPasswordRetriever](#)

Direct Known Subclasses:

[GXHttpSessionUserPasswordRetriever](#), [GXHttpRequestUserPasswordRetriever](#)

public abstract class **GXAbstractHttpUserPasswordRetriever**

extends Object

implements [GXUserPasswordRetriever](#)

### Constructor Summary

public	<a href="#">GXAbstractHttpUserPasswordRetriever</a> ()
public	<a href="#">GXAbstractHttpUserPasswordRetriever</a> (String userParameterName, String passwordParameterName)

### Method Summary

void	<a href="#">adoptHttpWrapper</a> (GXHttpWebWrapper httpWrapper)
String	<a href="#">getPasswordParameterName</a> () For internal use
String	<a href="#">getUserParameterName</a> () For internal use
void	<a href="#">setPasswordParameterName</a> (String passwordParameterName) For internal use
void	<a href="#">setUserParameterName</a> (String userParameterName) For internal use

#### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

#### Methods inherited from interface [com.sabratec.applinx.framework.sso.GXUserPasswordRetriever](#)

[getPassword](#), [getUserName](#)

### Constructors

(continued from last page)

## GXAbstractHttpUserPasswordRetriever

```
public GXAbstractHttpUserPasswordRetriever()
```

---

## GXAbstractHttpUserPasswordRetriever

```
public GXAbstractHttpUserPasswordRetriever(String userParameterName,  
                                             String passwordParameterName)
```

## Methods

### adoptHttpWrapper

```
public void adoptHttpWrapper(GXIHttpWebWrapper httpWrapper)
```

---

### getPasswordParameterName

```
public String getPasswordParameterName()
```

For internal use

---

### setPasswordParameterName

```
public void setPasswordParameterName(String passwordParameterName)
```

For internal use

---

### getUserParameterName

```
public String getUserParameterName()
```

For internal use

---

### setUserParameterName

```
public void setUserParameterName(String userParameterName)
```

For internal use

## com.sabratec.applinx.framework.web.sso Class GXHttpRequestUserPasswordRetriever

java.lang.Object

```

  |
+- com.sabratec.applinx.framework.web.sso.GXAbstractHttpUserPasswordRetriever
  |
+- com.sabratec.applinx.framework.web.sso.GXHttpRequestUserPasswordRetriever

```

All Implemented Interfaces:

[GXUserPasswordRetriever](#)

public class **GXHttpRequestUserPasswordRetriever**  
extends [GXAbstractHttpUserPasswordRetriever](#)

### Constructor Summary

public	<a href="#">GXHttpRequestUserPasswordRetriever()</a>
public	<a href="#">GXHttpRequestUserPasswordRetriever</a> (String userParameterName, String passwordParameterName)

### Method Summary

String	<a href="#">getPassword()</a>
String	<a href="#">getUserName()</a>

#### Methods inherited from class

[com.sabratec.applinx.framework.web.sso.GXAbstractHttpUserPasswordRetriever](#)

[adoptHttpWrapper](#), [getPasswordParameterName](#), [getUserParameterName](#),  
[setPasswordParameterName](#), [setUserParameterName](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.framework.sso.GXUserPasswordRetriever](#)

[getPassword](#), [getUserName](#)

## Constructors

### GXHttpRequestUserPasswordRetriever

public **GXHttpRequestUserPasswordRetriever**()

---

## GXHttpRequestUserPasswordRetriever

```
public GXHttpRequestUserPasswordRetriever(String userParameterName,  
                                           String passwordParameterName)
```

### Methods

#### getUserName

```
public String getUsername()
```

---

#### getPassword

```
public String getPassword()
```

## com.sabratec.applinx.framework.web.sso

### Class GXHttpSessionUserPasswordRetriever

java.lang.Object

```

    |
    +-- com.sabratec.applinx.framework.web.sso.GXAbstractHttpUserPasswordRetriever
         |
         +-- com.sabratec.applinx.framework.web.sso.GXHttpSessionUserPasswordRetriever
  
```

All Implemented Interfaces:

[GXUserPasswordRetriever](#)

public class **GXHttpSessionUserPasswordRetriever**  
 extends [GXAbstractHttpUserPasswordRetriever](#)

#### Constructor Summary

public	<a href="#">GXHttpSessionUserPasswordRetriever()</a>
public	<a href="#">GXHttpSessionUserPasswordRetriever</a> (String userParameterName, String passwordParameterName)

#### Method Summary

String	<a href="#">getPassword()</a>
String	<a href="#">getUserName()</a>

#### Methods inherited from class

[com.sabratec.applinx.framework.web.sso.GXAbstractHttpUserPasswordRetriever](#)

[adoptHttpWrapper](#), [getPasswordParameterName](#), [getUserParameterName](#),  
[setPasswordParameterName](#), [setUserParameterName](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

#### Methods inherited from interface [com.sabratec.applinx.framework.sso.GXUserPasswordRetriever](#)

[getPassword](#), [getUserName](#)

## Constructors

### GXHttpSessionUserPasswordRetriever

public **GXHttpSessionUserPasswordRetriever**()

---

## GXHttpSessionUserPasswordRetriever

```
public GXHttpSessionUserPasswordRetriever(String userParameterName,  
                                           String passwordParameterName)
```

### Methods

#### **getUserName**

```
public String getUserName()
```

---

#### **getPassword**

```
public String getPassword()
```

---

## Package

# com.sabratec.applinx.presentation

This package and its sub packages are responsible for creating a tag representation output from a host screen. The tag representation output is used for creating Instant HTML pages from a host screen.

The following API enables registering as many transformation classes as you require. These transformations can manipulate the output by adding, removing, replacing and changing the screen tag model using a rich library of tags and an easy-to-use API.

The tags are rendered in absolute terms, as oppose to the relative rendering used previously. Rendering the HTML tags relatively caused indentation problems when trying to add or change elements or style sheets such as GUI elements, host windows and different fonts. In the following API, each tag is rendered with corresponding top/left attributes which places the tag in an exact position. The main advantage of this approach is that changing/adding tags/stylesheets doesn't affect the neighboring tags of the modified tag.

## The Main entities used in the API:

### Base Object Entities:

#### Host screen (com.sabratec.applinx.baseobject.GXIScreen)

The host screen entity (which is part of the base object API) is the data model that represents the host screen. The host screen is a collection of fields and additional screen data. The host screen is used in transformations as an event parameter. The host screen event parameter can be used for querying the host screen as to its field contents and properties. (Refer to BaseObject JavaDoc for further documentation).

#### Host field (com.sabratec.applinx.baseobject.GXIField)

The host field entity (which is part of the base object API) contains various ApplinX and/or host properties such as content, protected/unprotected, color, intensified, ApplinX application field name, etc. The host field is also a tag property for tags that are screen based. (Refer to BaseObject JavaDoc for further documentation).

### General Entities:

#### Position (com.sabratec.util.GXPosition)

This class represents a position on the host screen. It is used by the host field and the tag to indicate a specific position on the host screen.

#### Rectangle (com.sabratec.util.GXRectangle)

This class represents a rectangle area of positions on the screen. This class is used to determine the screen area of the tags output in the Renderer configuration, and can perform area queries, on a specific area.

# Instant API Entities:

## Renderer

This entity is represented as an instant tag in the JSP/Instant control in .NET (in instant.jsp/aspx). The Renderer tag renders the HTML output based on the current host screen Renderer configuration settings and registered transformations.

## Renderer Configuration

(com.sabratec.applinx.presentation.GXRenderConfig)

The Renderer Configuration class contains various configurations such as screen area, font size and sub configurations for various built-in transformations. Examples for such sub configurations include host keys, GUI elements, tables, windows, etc. Each sub configuration also allows disabling specific built-in transformations. The Renderer configuration provides transformation registration of the tag listener and completion listener.

## Screen tag model

(com.sabratec.applinx.presentation.tags.GXIScreenTagModel)

The screen tag model is the model which contains a list of tags. The model provides various methods for adding, removing, replacing and searching tags. The final tag list in the screen tag model (after all transformations have been completed) will be rendered to the HTML output by the Renderer entity.

## Tag (com.sabratec.applinx.presentation.tags.GXITag)

The tag entity is the root class of all tag types, such as text fields, labels, links, combo boxes etc. The tag is a host oriented entity.

The tag provides a reference to the host field, and allows locating tags by setting their position in host units (row, column). The initial position is derived from the host field, and can be modified by the developer. The position will be rendered to HTML units as top/left.

Each tag type contains methods relevant to its type. Special HTML tags are also available such as GXHtmlGenericTag for creating a generic tag by specifying the tag name and GXHtmlString for using direct strings and adding them to the screen tag model.

## Tags factory

(com.sabratec.applinx.presentation.tags.html.GXHtmlTagFactory)

The tags factory is used for creating new HTML tags and for improving performance.

**Syntax:** GXHtmlTagFactory.instance().newLink()

This class contains methods for each type of tag.

## Transformations

Transformations are listeners which are registered to the Renderer configuration by the developer, and allow modifying the screen tag model before it is rendered. The transformations can be one of the following types: Tag listeners or Completion listeners.

Note: All ApplinX special entities such as host keys, tables, GUI elements and windows are also represented as built-in transformations, and are performed automatically by the `Renderer` entity.

## Tag listener(`com.sabratec.applinx.presentation.event.GXTagListener`)

When inheriting from this class, you will receive an event on the creation of each type of tag such as: `onNewTextField` , `onNewLabel` , `onNewButton` , etc. It should be used for tag level manipulations, such as changing the content and attributes, adding icons, etc.

## Completion listener(`com.sabratec.applinx.presentation.event.GXICompletionListener`)

When implementing this class you will receive an `onComplete` event, which should be used for manipulating the entire tag model.

## Render Event

### (`com.sabratec.applinx.presentation.event.GXRenderEvent`)

An event is thrown by the `Renderer` to registered transformations when either a new tag is created (tag listener) or the screen tag model has completed its creation process (completion listener). The thrown event passes the `Render` event as a parameter. The `Render` event contains the host screen and the screen tag model. Use the screen tag model for manipulating the HTML output by modifying its contained tags.

## Query (`com.sabratec.applinx.presentation.queries.GXITagQuery`)

Used for performing queries on the screen tag model. Each query retrieves a list of the tags that meet the query criteria. The instant API contains a few built-in queries such as a text query and area query. In addition, you can write your own queries by implementing the `GXITagQuery` interface.

# com.sabratec.applinx.presentation Class GXIllegalPositionException



```
public class GXIllegalPositionException
extends GXGeneralException
```

This exception is thrown when performing a query on a screen tag model that requires a position but the specified position is outside the screen tag model bounds.

## Constructor Summary

public	<a href="#">GXIllegalPositionException</a> (String message)
--------	---

## Methods inherited from class java.lang.Object

<code>equals</code> , <code>getClass</code> , <code>hashCode</code> , <code>notify</code> , <code>notifyAll</code> , <code>toString</code> , <code>wait</code> , <code>wait</code> , <code>wait</code>
--

## Constructors

### GXIllegalPositionException

```
public GXIllegalPositionException(String message)
```

**Parameters:**  
message - The thrown message

## com.sabratec.applinx.presentation Class GXPresentationConstants

java.lang.Object

└─com.sabratec.applinx.presentation.GXPresentationConstants

public class **GXPresentationConstants**  
extends Object

This class contains general constants that are used by the API and its sub packages.

### Field Summary

public static final	<a href="#">ALIGNMENT_HORIZONTAL</a> Represents horizonal alignment Value: 1
public static final	<a href="#">ALIGNMENT_VERTICAL</a> Represents vertical alignment Value: 0

### Constructor Summary

public	<a href="#">GXPresentationConstants</a> ()
--------	--

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### ALIGNMENT\_VERTICAL

public static final int **ALIGNMENT\_VERTICAL**

Represents vertical alignment  
Constant value: 0

#### ALIGNMENT\_HORIZONTAL

public static final int **ALIGNMENT\_HORIZONTAL**

Represents horizonal alignment  
Constant value: 1

### Constructors

#### GXPresentationConstants

public **GXPresentationConstants**()

(continued from last page)

## com.sabratec.applinx.presentation Class GXRenderConfig

java.lang.Object

└--com.sabratec.applinx.presentation.GXRenderConfig

### All Implemented Interfaces:

java.io.Serializable

public class **GXRenderConfig**  
extends Object  
implements java.io.Serializable

This class is used for setting the Renderer configuration. It contains various configurations such as screen area, font size and sub classes for various built-in transformation configurations such as:

Host keys, GUI elements, Tables, Windows, etc.

Each configuration sub class allows disabling a specific built-in transformation.

This class also provides transformations registration of tag listener [GXTagListener](#) and completion listener [GXCompletionListener](#).

## Constructor Summary

public	<a href="#">GXRenderConfig()</a>
--------	----------------------------------

## Method Summary

void	<a href="#">addCompletionListener</a> ( <a href="#">GXCompletionListener</a> listener) Used for registering a listener class that can modify the screen tag model.
void	<a href="#">addTagListener</a> ( <a href="#">GXITagListener</a> listener) Used for registering a listener class which accepts an event for each tag type.
Vector	<a href="#">get_CompleteListenersVector</a> () For .NET framework usage.
<a href="#">GXEmulationTransformConfig</a>	<a href="#">get_EmulationConfig</a> () For .NET framework usage.
String	<a href="#">get_FontFamily</a> () For .NET framework usage.
int	<a href="#">get_FontSize</a> () For .NET framework usage.
<a href="#">GXGuiElementsTransformConfig</a>	<a href="#">get_GuiElementsConfig</a> () For .NET framework usage.
<a href="#">GXHostKeysTransformConfig</a>	<a href="#">get_HostKeysConfig</a> () For .NET framework usage.
GXRectangle	<a href="#">get_Rectangle</a> () For .NET framework usage.

GXRectangle	<a href="#">get_RenderArea()</a> For .NET framework usage.
boolean	<a href="#">get_RenderTagsFromRightToLeft()</a> For .NET framework usage.
int	<a href="#">get_RowSpacingPrecentageFactor()</a> For .NET framework usage.
<a href="#">GXTableTransformConfig</a>	<a href="#">get_TableConfig()</a> For .NET framework usage.
<a href="#">GXWindowFrameTransformConfig</a>	<a href="#">get_WindowFrameConfig()</a> For .NET framework usage.
<a href="#">GXCompletionListener[]</a>	<a href="#">getCompletionListener()</a> Used for retrieving all the registered completion listeners.
<a href="#">GXEmulationTransformConfig</a>	<a href="#">getEmulationConfig()</a> The emulation transformation configuration contains properties which determine the rendered tag look-and-feel and behavior.
String	<a href="#">getFontFamily()</a> Sets the font family to be used for instant rendering.
int	<a href="#">getFontSize()</a> Used for retrieving the font size.
<a href="#">GXGuiElementsTransformConfig</a>	<a href="#">getGuiElementsConfig()</a> The GUI elements transformation configuration.
<a href="#">GXHostKeysTransformConfig</a>	<a href="#">getHostKeysConfig()</a> The Host keys transformation configuration.
GXRectangle	<a href="#">getRenderArea()</a> The rendered area of the host screen which the tags are rendered from.
int	<a href="#">getRowSpacingPrecentageFactor()</a> Returns the row height to be used for instant rendering.
<a href="#">GXTableTransformConfig</a>	<a href="#">getTableConfig()</a> The table transformation configuration.
GXITagListener[]	<a href="#">getTagListeners()</a> Used for retrieving all the registered tag listeners.
<a href="#">GXWindowFrameTransformConfig</a>	<a href="#">getWindowFrameConfig()</a> The window transformation configuration.
boolean	<a href="#">isRenderTagsFromRightToLeft()</a> Determines that the HTML tags will be positioned from right to left.
void	<a href="#">removeAllListeners()</a>
void	<a href="#">set_EmulationConfig(GXEmulationTransformConfig emulationConfig)</a> For .NET framework usage.
void	<a href="#">set_FontFamily(String fontFamily)</a> For .NET framework usage.

void	<a href="#"><u>set_FontSize</u></a> (int fontSize) For .NET framework usage.
void	<a href="#"><u>set_GuiElementsConfig</u></a> ( <a href="#"><u>GXGuiElementsTransformConfig</u></a> guiElementsConfig) For .NET framework usage.
void	<a href="#"><u>set_HostKeysConfig</u></a> ( <a href="#"><u>GXHostKeysTransformConfig</u></a> hostKeysConfig) For .NET framework usage.
void	<a href="#"><u>set_Rectangle</u></a> (GXRectangle rectangle) For .NET framework usage.
void	<a href="#"><u>set_RenderArea</u></a> (GXRectangle rectangle) For .NET framework usage.
void	<a href="#"><u>set_RenderTagsFromRightToLeft</u></a> (boolean renderTagsFromRightToLeft) For .NET framework usage.
void	<a href="#"><u>set_RowSpacingPrecentageFactor</u></a> (int rowSpacingPrecentageFactor) For .NET framework usage.
void	<a href="#"><u>set_TableConfig</u></a> ( <a href="#"><u>GXTableTransformConfig</u></a> tableConfig) For .NET framework usage.
void	<a href="#"><u>set_WindowFrameConfig</u></a> ( <a href="#"><u>GXWindowFrameTransformConfig</u></a> windowConfig) For .NET framework usage.
void	<a href="#"><u>setEmulationConfig</u></a> ( <a href="#"><u>GXEmulationTransformConfig</u></a> emulationConfig)
void	<a href="#"><u>setFontFamily</u></a> (String fontFamily) Sets the font family to be used for instant rendering.
void	<a href="#"><u>setFontSize</u></a> (int fontSize) It is possible to change the default font size to a different size (in pixels).
void	<a href="#"><u>setGuiElementsConfig</u></a> ( <a href="#"><u>GXGuiElementsTransformConfig</u></a> guiElementsConfig)
void	<a href="#"><u>setHostKeysConfig</u></a> ( <a href="#"><u>GXHostKeysTransformConfig</u></a> hostKeysConfig)
void	<a href="#"><u>setRenderArea</u></a> (GXRectangle rectangle) Used to define that a specific area of the host screen will be rendered as HTML in Instant HTML pages by specifying the area's boundaries and in this way hide certain portions of the screen. By default, the entire screen is configured to be rendered to the HTML output.
void	<a href="#"><u>setRenderTagsFromRightToLeft</u></a> (boolean renderTagsFromRightToLeft) Determines that the HTML tags will be positioned from right to left.
void	<a href="#"><u>setRowSpacingPrecentageFactor</u></a> (int rowSpacingPrecentageFactor) Sets the row height to be used for instant rendering.
void	<a href="#"><u>setTableConfig</u></a> ( <a href="#"><u>GXTableTransformConfig</u></a> tableConfig)
void	<a href="#"><u>setWindowFrameConfig</u></a> ( <a href="#"><u>GXWindowFrameTransformConfig</u></a> windowFrameConfig)

**Methods inherited from class** java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

## Constructors

### **GXRenderConfig**

```
public GXRenderConfig()
```

---

## Methods

### **addTagListener**

```
public void addTagListener(GXITagListener listener)
```

Used for registering a listener class which accepts an event for each tag type.

**Parameters:**

listener - The tag listener

---

### **addCompletionListener**

```
public void addCompletionListener(GXICompletionListener listener)
```

Used for registering a listener class that can modify the screen tag model.

**Parameters:**

listener - The completion listener

---

### **setRenderArea**

```
public void setRenderArea(GXRectangle rectangle)
```

Used to define that a specific area of the host screen will be rendered as HTML in Instant HTML pages by specifying the area's boundaries and in this way hide certain portions of the screen.

By default, the entire screen is configured to be rendered to the HTML output.

When setting the area to be rendered, tags positioned outside the rendered area will not be available for query execution on the screen tag model. The rendered area also affects the rendered top/left attributes of the tag. The top/left tag will be calculated relatively to the starting position of the rendered area.

**Parameters:**

rectangle - The positions of the rendered area

---

### **set\_RenderArea**

```
public void set_RenderArea(GXRectangle rectangle)
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[setRenderArea\(GXRectangle\)](#)

---

### **getRenderArea**

```
public GXRectangle getRenderArea()
```

The rendered area of the host screen which the tags are rendered from.

---

(continued from last page)

**Returns:**

A rectangle that represent the render area

**See Also:**

[setRenderArea\(GXRectangle\)](#)

---

## get\_Rectangle

```
public GXRectangle get_Rectangle()
```

For .NET framework usage. Expose the method as a property.

---

## set\_Rectangle

```
public void set_Rectangle(GXRectangle rectangle)
```

For .NET framework usage. Expose the method as a property.

---

## get\_RenderArea

```
public GXRectangle get_RenderArea()
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[getRenderArea\(\)](#)

---

## getFontSize

```
public int getFontSize()
```

Used for retrieving the font size.

**Returns:**

The font size

**See Also:**

[setFontSize\(int\)](#)

---

## get\_FontSize

```
public int get_FontSize()
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[getFontSize\(\)](#)

---

## setFontSize

```
public void setFontSize(int fontSize)
```

(continued from last page)

It is possible to change the default font size to a different size (in pixels). Since the change affects the rendering process performed by the framework on the Web server, this setting is not part of the CSS style sheet file (unlike other font settings), but instead it is included in the server-side instant configuration parameters.

Note: Changing the font sizes for generated Web pages is done in the CSS style sheet node.

The possible values include the different pixel possibilities as well as "Dynamic by Resolution". The HTML emulation recognizes user resolution and dynamically changes the css name according to the matching resolution. The resolution can be either 640x480, 800x600, 1024x768 or 1280x1024 pixels. The resolution may also depend on the host resolution (80 or 132 characters). The best-fit functionality is configured by default in the emulation template in all frameworks. It is used by the renderer to calculate the top/left according to each tag's position.

#### Parameters:

fontSize - The font size to set.

---

## set\_FontSize

```
public void set_FontSize(int fontSize)
```

For .NET framework usage. Expose the method as a property.

#### See Also:

[setFontSize\(int\)](#)

---

## getEmulationConfig

```
public GXEmulationTransformConfig getEmulationConfig()
```

The emulation transformation configuration contains properties which determine the rendered tag look-and-feel and behavior.

#### Returns:

The emulation transformation configuration.

---

## setEmulationConfig

```
public void setEmulationConfig(GXEmulationTransformConfig emulationConfig)
```

---

## get\_EmulationConfig

```
public GXEmulationTransformConfig get_EmulationConfig()
```

For .NET framework usage. Expose the method as a property.

#### See Also:

[getEmulationConfig\(\)](#)

---

## set\_EmulationConfig

```
public void set_EmulationConfig(GXEmulationTransformConfig emulationConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getGuiElementsConfig

```
public GXGuiElementsTransformConfig getGuiElementsConfig()
```

The GUI elements transformation configuration. The GUI elements transformation represents GUI elements which were configured using the ApplinX composer.

---

(continued from last page)

**Returns:**

The GUI elements' transformation configuration.

---

## setGuiElementsConfig

```
public void setGuiElementsConfig(GXGuiElementsTransformConfig guiElementsConfig)
```

---

## get\_GuiElementsConfig

```
public GXGuiElementsTransformConfig get_GuiElementsConfig()
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[getGuiElementsConfig\(\)](#)

---

## getHostKeysConfig

```
public GXHostKeysTransformConfig getHostKeysConfig()
```

The Host keys transformation configuration. Host keys transformation represents host keys patterns which were configured using the ApplinX composer.

**Returns:**

The host keys transformation configuration

---

## setHostKeysConfig

```
public void setHostKeysConfig(GXHostKeysTransformConfig hostKeysConfig)
```

---

## get\_HostKeysConfig

```
public GXHostKeysTransformConfig get_HostKeysConfig()
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[getHostKeysConfig\(\)](#)

---

## set\_HostKeysConfig

```
public void set_HostKeysConfig(GXHostKeysTransformConfig hostKeysConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getTableConfig

```
public GXTableTransformConfig getTableConfig()
```

The table transformation configuration. The table transformation is based on a host table that was configured using the ApplinX composer.

**Returns:**

The table configuration

---

## setTableConfig

```
public void setTableConfig(GXTableTransformConfig tableConfig)
```

---

## get\_TableConfig

```
public GXTableTransformConfig get_TableConfig()
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[getTableConfig\(\)](#)

---

## set\_TableConfig

```
public void set_TableConfig(GXTableTransformConfig tableConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getWindowFrameConfig

```
public GXWindowFrameTransformConfig getWindowFrameConfig()
```

The window transformation configuration. Window frame transformation draws a rectangle instead of drawing a host window which is represented as characters or reversed video.

**Returns:**

The window frame configuration

---

## setWindowFrameConfig

```
public void setWindowFrameConfig(GXWindowFrameTransformConfig windowFrameConfig)
```

---

## get\_WindowFrameConfig

```
public GXWindowFrameTransformConfig get_WindowFrameConfig()
```

For .NET framework usage. Expose the method as a property.

**See Also:**

[getWindowFrameConfig\(\)](#)

---

## set\_WindowFrameConfig

```
public void set_WindowFrameConfig(GXWindowFrameTransformConfig windowConfig)
```

For .NET framework usage. Expose the method as a property.

---

## getTagListeners

```
public GXITagListener[] getTagListeners()
```

Used for retrieving all the registered tag listeners.

---

(continued from last page)

**Returns:**

An array of the registered tag listeners

---

## getCompletionListener

```
public GXCompletionListener\[\] getCompletionListener()
```

Used for retrieving all the registered completion listeners.

**Returns:**

An array of the completion listeners

---

## removeAllListeners

```
public void removeAllListeners()
```

---

## getFontFamily

```
public String getFontFamily()
```

Sets the font family to be used for instant rendering.

**Returns:**

the font family in use. null if no specific font to use(defined in the CSS file).

---

## setFontFamily

```
public void setFontFamily(String fontFamily)
```

Sets the font family to be used for instant rendering. It is used for proportion calculation to best fit specific fonts for Japanese & Arabic applications.

For Japanese the best fitted fixed fonts are: "MS Mincho" or "MS Gothic"

For Arabic the best fitted fixed font is: "Courier".

For other languages the best fitted font is "Courier New", which is by default declared in the CSS files. Defining the font here will override the CSS definition.

**Parameters:**

fontFamily - the font family in use. null if no specific font to use(defined in the CSS file).

---

## get\_FontFamily

```
public String get_FontFamily()
```

For .NET framework usage. Expose the method as a property.

---

## set\_FontFamily

```
public void set_FontFamily(String fontFamily)
```

For .NET framework usage. Expose the method as a property.

---

## getRowSpacingPrecentageFactor

```
public int getRowSpacingPrecentageFactor()
```

(continued from last page)

Returns the row height to be used for instant rendering. This height is calculated relatively to the rows font size. The default row height is set to 53% (153% of the font size). For example: When the font size is set to 13px, and using the default row height value of 53%, the row height will be:  $13 \times 1.53 = 19.89 \approx 20\text{px}$ .

**Returns:**

the row height to be used

---

## setRowSpacingPrecentageFactor

```
public void setRowSpacingPrecentageFactor(int rowSpacingPrecentageFactor)
```

Sets the row height to be used for instant rendering. This height is calculated relatively to the rows font size. The default row height is set to 53% (153% of the font size). For example: When the font size is set to 13px, and using the default row height value of 53%, the row height will be:  $13 \times 1.53 = 19.89 \approx 20\text{px}$ .

**Parameters:**

rowHeightPrecentageFactor - the row height to be used

---

## get\_RowSpacingPrecentageFactor

```
public int get_RowSpacingPrecentageFactor()
```

For .NET framework usage. Expose the method as a property.

---

## set\_RowSpacingPrecentageFactor

```
public void set_RowSpacingPrecentageFactor(int rowSpacingPrecentageFactor)
```

For .NET framework usage. Expose the method as a property.

---

## isRenderTagsFromRightToLeft

```
public boolean isRenderTagsFromRightToLeft()
```

Determines that the HTML tags will be positioned from right to left. The usage is in Hebrew/Arabic applications when using non-fixed font, to ensure alignment to the right. By default the tags are positioned from left to right.

**Returns:**


---

## setRenderTagsFromRightToLeft

```
public void setRenderTagsFromRightToLeft(boolean renderTagsFromRightToLeft)
```

Determines that the HTML tags will be positioned from right to left. The usage is in Hebrew/Arabic applications when using non-fixed font, to ensure alignment to the right. By default the tags are positioned from left to right.

**Parameters:**

renderTagsFromRightToLeft

---

## get\_RenderTagsFromRightToLeft

```
public boolean get_RenderTagsFromRightToLeft()
```

For .NET framework usage. Expose the method as a property.

---

## set\_RenderTagsFromRightToLeft

```
public void set_RenderTagsFromRightToLeft(boolean renderTagsFromRightToLeft)
```

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **set\_GuiElementsConfig**

```
public void set_GuiElementsConfig(GXGuiElementsTransformConfig guiElementsConfig)
```

For .NET framework usage. Expose the method as a property.

---

## **get\_CompleteListenersVector**

```
public Vector get_CompleteListenersVector( )
```

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.presentation Class GXTagUtil

java.lang.Object

└--com.sabratec.applinx.presentation.GXTagUtil

public class **GXTagUtil**  
extends Object

### Constructor Summary

public	<a href="#">GXTagUtil()</a>
--------	-----------------------------

### Method Summary

static double	<a href="#">getHeightPixels</a> (int length, int fontSize, int precentageFactor) This method is used for determine the proportion between a given length & font size to actual height in pixels
static double	<a href="#">getTagHeight</a> (GXITag tag, int fontSize, int precentageFactor) This methd returns the tag height in pixels, in order to know if the tag is overriding his below neighbour
static double	<a href="#">getTagWidth</a> (GXITag tag, int fontSize) This method returns the tag width in pixels, in order to know if the tag is overriding his right/left neighbor
static double	<a href="#">getWidthPixels</a> (int length, int fontSize) This method is used for determine the proportion between a given length & font size to actual width in pixels

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### GXTagUtil

public **GXTagUtil**()

### Methods

#### getWidthPixels

public static double **getWidthPixels**(int length,  
int fontSize)

This method is used for determine the proportion between a given length & font size to actual width in pixels

(continued from last page)

**Parameters:**

length - The host field length  
fontSize - The display font size in pixels

**Returns:**

The actual width in pixels

---

## getHeightPixels

```
public static double getHeightPixels(int length,  
    int fontSize,  
    int precentageFactor)
```

This method is used for determine the proportion between a given length & font size to actual height in pixels

**Parameters:**

length - The host field length  
fontSize - The display font size in pixels

**Returns:**

The actual height in pixels

---

## getTagWidth

```
public static double getTagWidth(GXITag tag,  
    int fontSize)
```

This method returns the tag width in pixels, in order to know if the tag is overriding his right/left neighbor

**Parameters:**

tag - The tag which it's width is required  
fontSize - The font size

**Returns:**

The tag width in pixels

---

## getTagHeight

```
public static double getTagHeight(GXITag tag,  
    int fontSize,  
    int precentageFactor)
```

This methd returns the tag height in pixels, in order to know if the tag is overriding his below neighbour

**Parameters:**

tag - The tag which it's height is required  
fontSize - The display font size in pixels

**Returns:**

The tag height in pixles

# com.sabratec.applinx.presentation

## Class GXUndefinedPositionException



```
public class GXUndefinedPositionException
extends GXGeneralException
```

This exception is thrown when performing a query on a screen tag model that requires a position but the specified position is empty.

### Constructor Summary

public	<a href="#">GXUndefinedPositionException</a> ( <a href="#">GXITag</a> tag) Throws an exception with the specified tag details
--------	--

### Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

### Constructors

#### GXUndefinedPositionException

```
public GXUndefinedPositionException(GXITag tag)
```

Throws an exception with the specified tag details

**Parameters:**

tag - The tag whose position is empty

---

## Package

# **com.sabratec.applinx.presentation.event**

This package contains the events and listeners. These classes are used by the Renderer for notifying transformations of the creation of the screen tag model and new tags.

## com.sabratec.applinx.presentation.event Interface GXICompletionListener

All Known Implementing Classes:

[GXHebrewTransform](#)

---

public interface **GXICompletionListener**

This listener is notified when the screen tag model building process is completed.

Classes that implement this listener should be registered in the Render configuration.

When the building process is completed, a GXRenderEvent event is thrown, and the onComplete method is called.

**See Also:**

[GXRenderConfig.addCompletionListener\(GXICompletionListener\)](#), [onComplete\(GXRenderEvent\)](#),  
[GXRenderEvent](#)

---

### Method Summary

void	<a href="#">onComplete(GXRenderEvent event)</a> Invoked when screen tag model build process has been completed.
------	--

---

### Methods

#### onComplete

public void **onComplete**([GXRenderEvent](#) event)

Invoked when screen tag model build process has been completed.

**Parameters:**

event - The event arguments

# com.sabratec.applinx.presentation.event

## Class GXRenderEvent



```
public class GXRenderEvent
extends Object
```

An event that indicates that the screen tag model building process has been completed. This event is used both for tag completion events and screen-model completion events.

A GXRenderEvent is passed to every tag listener [GXTagListener](#) and every completion listener [GXCompletionListener](#) that were registered within the Renderer configuration.

**See Also:**  
[GXCompletionListener](#), [GXTagListener](#)

### Constructor Summary

public	<a href="#">GXRenderEvent</a> ( <a href="#">GXIScreenTagModel</a> screenTagModel, <a href="#">GXIScreen</a> screen) For internal usage.
--------	--

### Method Summary

<a href="#">GXIScreen</a>	<a href="#">getHostScreen</a> () Used for retrieving the current host screen.
<a href="#">GXIScreenTagModel</a>	<a href="#">getScreenTagModel</a> () Used for retrieving the screen tag model created by the renderer.

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXRenderEvent

```
public GXRenderEvent(GXIScreenTagModel screenTagModel,
GXIScreen screen)
```

For internal usage. Initialized by the renderer.

- Parameters:**
- screenTagModel - The screen tag model
  - screen - The host screen

## Methods

(continued from last page)

## getScreenTagModel

```
public GXIScreenTagModel getScreenTagModel( )
```

Used for retrieving the screen tag model created by the renderer.

**Returns:**

The screen model created by the renderer

---

## getHostScreen

```
public GXIScreen getHostScreen( )
```

Used for retrieving the current host screen.

**Returns:**

The current host screen

## com.sabratec.applinx.presentation.event Class GXTagListener

java.lang.Object

└-com.sabratec.applinx.presentation.event.GXTagListener

**Direct Known Subclasses:**

[GXSapPortalTransform](#), [GXCisCssTransform](#)

public abstract class **GXTagListener**  
extends Object

This listener is notified when a tag rendering is completed.

Classes that extend this listener should be registered in the Render configuration.

The class that requires processing a render event extends this abstract [GXTagListener](#) class (overriding only the methods of interest).

When a single tag rendering is completed, a GXRenderEvent event is thrown, and the onNew[TagName] method is called. The methods in this class are empty. This class exists to be used to conveniently create listener objects.

**See Also:**

[GXRenderEvent](#), addTagListener(GXTagListener)

### Constructor Summary

public	<a href="#">GXTagListener</a> ()
--------	----------------------------------

### Method Summary

void	<a href="#">onNewButton</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIButtonTag</a> button) Invoked when a link has been created.
void	<a href="#">onNewCheckbox</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXICheckboxTag</a> checkbox) Invoked when a checkbox has been created.
void	<a href="#">onNewCombobox</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIComboboxTag</a> combobox) Invoked when a combo box has been created.
void	<a href="#">onNewLabel</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILabelTag</a> label) Invoked when a label (protected field) has been created.
void	<a href="#">onNewLink</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILinkTag</a> link) Invoked when a link has been created.
void	<a href="#">onNewRadioButtonsGroup</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIRadioButtonsGroupTag</a> radioGroup) Invoked when a radio buttons group has been created.
void	<a href="#">onNewTable</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableTag</a> table) Invoked when a table tag has been created.
void	<a href="#">onNewTableCell</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableCellTag</a> tableCell) Invoked when a table cell has been created.

void	<a href="#">onNewTableRow</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableRowTag</a> tableRow) Invoked when a table row tag has been created.
void	<a href="#">onNewTextField</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITextFieldTag</a> textField) Invoked when a text field has been created.

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXTagListener

```
public GXTagListener()
```

## Methods

### onNewTextField

```
public void onNewTextField(GXRenderEvent event,
GXITextFieldTag textField)
```

Invoked when a text field has been created. The common usage is for unprotected fields that are rendered as text fields.

#### Parameters:

`event` - The event arguments  
`textField` - The text field tag which has been created

### onNewLabel

```
public void onNewLabel(GXRenderEvent event,
GXILabelTag label)
```

Invoked when a label (protected field) has been created. The common usage is for protected fields that are rendered as text fields.

#### Parameters:

`event` - The event arguments  
`label` - The label tag which has been created

### onNewLink

```
public void onNewLink(GXRenderEvent event,
GXILinkTag link)
```

Invoked when a link has been created. The common usage is for host-keys that may be rendered as links.

#### Parameters:

`event` - The event arguments  
`link` - The linked tag which has been created

(continued from last page)

---

## onNewButton

```
public void onNewButton(GXRenderEvent event,  
    GXIButtonTag button)
```

Invoked when a link has been created. The common usage is for host-keys that may be rendered as buttons.

**Parameters:**

event - the event arguments

button - the button tag which has been created

---

## onNewCombobox

```
public void onNewCombobox(GXRenderEvent event,  
    GXIComboboxTag combobox)
```

Invoked when a combo box has been created. The common usage is for a GUI-element that may be rendered as a combo box.

**Parameters:**

event - The event arguments

combobox - The combo box tag which has been created

---

## onNewCheckbox

```
public void onNewCheckbox(GXRenderEvent event,  
    GXICheckboxTag checkbox)
```

Invoked when a checkbox has been created. The common usage is for a GUI-element that may be rendered as a check box.

---

## onNewRadioButtonGroup

```
public void onNewRadioButtonGroup(GXRenderEvent event,  
    GXIRadioButtonGroupTag radioButtonGroup)
```

Invoked when a radio buttons group has been created. The common usage is for a GUI-element that may be rendered as a radio button's group.

**Parameters:**

event - The event arguments

radioGroup - The radio buttons group tag which has been created

---

## onNewTable

```
public void onNewTable(GXRenderEvent event,  
    GXITableTag table)
```

Invoked when a table tag has been created.

In this case, `table.getHostField()` returns null.

**Parameters:**

event - The event arguments

table - The table tag which has been created

**See Also:**

[GXITable](#)

[GXIScreen](#)

---

(continued from last page)

## onNewTableRow

```
public void onNewTableRow(GXRenderEvent event,  
    GXITableRowTag tableRow)
```

Invoked when a table row tag has been created.

In this case, `table.getHostField()` returns null.

### Parameters:

event - The event arguments

tableRow - The table row tag which has been created

### See Also:

[GXITable](#)

[GXIScreen](#)

---

## onNewTableCell

```
public void onNewTableCell(GXRenderEvent event,  
    GXITableCellTag tableCell)
```

Invoked when a table cell has been created.

Unlike the other table-related event objects, `tableCell.getHostField()` returns the host field the cell is based on.

### Parameters:

event - The event arguments

tableCell - The table tag which has been created

### See Also:

[GXIField](#)

---

---

## Package

# com.sabratec.applinx.presentation.queries

This package contains the query interface and built-in queries such as text query, area query, which should be used for executing queries against the screen tag model.

## Package Specification

##### FILL IN ANY SPECS NEEDED BY JAVA COMPATIBILITY KIT #####

- ##### REFER TO ANY FRAMEMAKER SPECIFICATION HERE #####

## Related Documentation

For overviews, tutorials, examples, guides, and tool documentation, please see:

- ##### REFER TO NON-SPEC DOCUMENTATION HERE #####

## com.sabratec.applinx.presentation.queries

### Class GXAreaQuery

java.lang.Object

↳ com.sabratec.applinx.presentation.queries.GXAreaQuery

All Implemented Interfaces:

[GXITagQuery](#)

public class **GXAreaQuery**  
 extends Object  
 implements [GXITagQuery](#)

This query returns tags in the given area. The query can be either row based by using the constructor [GXAreaQuery\(int, int\)](#) or rectangle based [GXAreaQuery\(GXRectangle, int, boolean\)](#)

See Also:

[GXIScreenTagModel.executeQuery\(GXITagQuery\)](#)

### Field Summary

public static	<a href="#">RETRIEVE_INSIDE_AREA</a> Indicates retrieval of tags from inside the area.
public static	<a href="#">RETRIEVE_OUTSIDE_AREA</a> Indicates retrieval of tags from outside the area.

### Constructor Summary

public	<a href="#">GXAreaQuery()</a> Creates an empty instance.
public	<a href="#">GXAreaQuery(GXRectangle area, int retrievalMode, boolean entireTag)</a> Creates an instance of a rectangular area query.
public	<a href="#">GXAreaQuery(int startRow, int endRow)</a> Creates an instance of a rectangular area query using to start and end rows.

### Method Summary

<a href="#">GXTagList</a>	<a href="#">find(GXIScreenTagModel screenModel)</a> Only tags that have host-screen positions will be returned.
void	<a href="#">setArea(GXRectangle area)</a> Sets the rectangular area according to which tags will be retrieved.
void	<a href="#">setEntireTag(boolean entireTag)</a> Defines whether tags that intersect with the area's frame will be considered as meeting the criteria.
void	<a href="#">setRetrievalMode(int retrievalMode)</a> Sets the retrieval mode, either tags inside or outside the area.
void	<a href="#">setRowRange(int startRow, int endRow)</a> Sets the rectangular area using the start and end rows.

**Methods inherited from class** `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

**Methods inherited from interface** [com.sabratec.applinx.presentation.queries.GXITagQuery](#)

[find](#)

## Fields

### RETRIEVE\_INSIDE\_AREA

```
public static int RETRIEVE_INSIDE_AREA
```

Indicates retrieval of tags from inside the area. This is the default value.

### RETRIEVE\_OUTSIDE\_AREA

```
public static int RETRIEVE_OUTSIDE_AREA
```

Indicates retrieval of tags from outside the area.

## Constructors

### GXAreaQuery

```
public GXAreaQuery()
```

Creates an empty instance.

### GXAreaQuery

```
public GXAreaQuery(GXRectangle area,
                   int retrievalMode,
                   boolean entireTag)
```

Creates an instance of a rectangular area query.

**Parameters:**

`area` - The rectangular area according which determine tags will be retrieved  
`retrievalMode` - indicates retrieval of tags from either inside or outside the area  
`entireTag` - indicates whether tags that intersect with the area's frame will be considered as a matching tags

### GXAreaQuery

```
public GXAreaQuery(int startRow,
                   int endRow)
```

Creates an instance of a rectangular area query using to start and end rows. The criteria is inclusive, i.e. tags which are positioned in the rows themselves will also be retrieved.

**Parameters:**

`startRow` - the row where the range starts, inclusive.  
`endRow` - the row where the range ends, inclusive.

## Methods

(continued from last page)

---

## setArea

```
public void setArea(GXRectangle area)
```

Sets the rectangular area according to which tags will be retrieved.

### Parameters:

area - the rectangular area according to which tags will be retrieved

---

## setRowRange

```
public void setRowRange(int startRow,  
    int endRow)
```

Sets the rectangular area using the start and end rows. The criteria is inclusive, i.e. tags which are positioned in the rows themselves will also be retrieved.

### Parameters:

startRow - the row where the range starts (including this row).

endRow - the row where the range ends (including this row).

---

## setRetrievalMode

```
public void setRetrievalMode(int retrievalMode)
```

Sets the retrieval mode, either tags inside or outside the area.

### Parameters:

retrievalMode - indicates retrieval of tags from either inside or outside the area

### See Also:

[RETRIEVE\\_INSIDE\\_AREA](#)

[RETRIEVE\\_OUTSIDE\\_AREA](#)

---

## setEntireTag

```
public void setEntireTag(boolean entireTag)
```

Defines whether tags that intersect with the area's frame will be considered as meeting the criteria.

### Parameters:

entireTag - true if the entire tag has to be contained in the area

---

## find

```
public GXTagList find(GXIScreenTagModel screenModel)
```

Only tags that have host-screen positions will be returned.

# com.sabratec.applinx.presentation.queries

## Interface GXITagQuery

All Known Implementing Classes:  
[GXSimpleTagQuery](#), [GXAreaQuery](#)

public interface **GXITagQuery**

An interface for queries executed on a screen tag model. By implementing this interface, you should be able to write your own query classes. The query is defined using the [find\(GXIScreenTagModel\)](#) method.

See Also:  
[GXIScreenTagModel.executeQuery\(GXITagQuery\)](#)

Method Summary	
<a href="#">GXTagList</a>	<a href="#">find(GXIScreenTagModel screenModel)</a> Performs a query against a given screen model.

## Methods

### find

public [GXTagList](#) **find**([GXIScreenTagModel](#) screenModel)

Performs a query against a given screen model.

**Parameters:**  
screenModel - The screen model that the query is performed against

**Returns:**  
All tags meeting the query. If there are no tags that meet the query, an empty list is returned

# com.sabratec.applinx.presentation.queries

## Class GXSimpleTagQuery

java.lang.Object

└--com.sabratec.applinx.presentation.queries.GXSimpleTagQuery

All Implemented Interfaces:

[GXITagQuery](#)

Direct Known Subclasses:

[GXTextQuery](#)

public abstract class **GXSimpleTagQuery**  
 extends Object  
 implements [GXITagQuery](#)

This abstract class provides a simple implementation for the [GXITagQuery](#) interface. The implementation consists of iterating over the entire screen model tag, and at the end returning all tags which meet a given criteria.

Overriding classes should implement the abstract method `matchesCriteria(GXITag)`.

See Also:

[GXIScreenTagModel.executeQuery\(GXITagQuery\)](#)

## Constructor Summary

public	<a href="#">GXSimpleTagQuery()</a>
--------	------------------------------------

## Method Summary

<a href="#">GXTagList</a>	<a href="#">find</a> ( <a href="#">GXIScreenTagModel</a> screenTagModel) Finds tags meeting the specific query criteria.
---------------------------	---

Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface [com.sabratec.applinx.presentation.queries.GXITagQuery](#)

[find](#)

## Constructors

### GXSimpleTagQuery

public **GXSimpleTagQuery**()

## Methods

(continued from last page)

## find

```
public GXTagList find(GXIScreenTagModel screenTagModel )
```

Finds tags meeting the specific query criteria.

**Parameters:**

`screenTagModel` - the screen tag model.

**Returns:**

the matching tag list

## com.sabratec.applinx.presentation.queries

### Class GXTextQuery

java.lang.Object

```

  |
  +-- com.sabratec.applinx.presentation.queries.GXSimpleTagQuery
      |
      +-- com.sabratec.applinx.presentation.queries.GXTextQuery
  
```

All Implemented Interfaces:

[GXITagQuery](#)

public class **GXTextQuery**  
 extends [GXSimpleTagQuery](#)

This query returns tags that contain the specified texts. This class requires at least one text that is to be searched for, which should be specified in the constructor. For additional texts use the method [addSearchText\(String\)](#). Tags that contain either one of the search texts will be returned.

See Also:

[GXIScreenTagModel.executeQuery\(GXITagQuery\)](#)

### Constructor Summary

public	<a href="#">GXTextQuery</a> (String searchText) The constructor accepts the text to search for.
--------	--

### Method Summary

void	<a href="#">addSearchText</a> (String searchText) Adds the specified search text to searched texts list
------	--

Methods inherited from class [com.sabratec.applinx.presentation.queries.GXSimpleTagQuery](#)

[find](#)

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.presentation.queries.GXITagQuery](#)

[find](#)

## Constructors

### GXTextQuery

public **GXTextQuery**(String searchText)

The constructor accepts the text to search for.

**Parameters:**

searchText - The text to search for.

## Methods

### **addSearchText**

```
public void addSearchText(String seacrhText)
```

Adds the specified search text to searched texts list

**Parameters:**

seacrhText

---

## Package

# **com.sabratec.applinx.presentation.tags**

This package provides interfaces for the screen tag model,common tags and tag lists.

## com.sabratec.applinx.presentation.tags Interface GXIButtonTag

All Superinterfaces:

[GXIClickableTag](#), [GXITag](#)

public interface **GXIButtonTag**

extends [GXIClickableTag](#)

This interface represents a button tag. The common usage is for a host key when setting the Render configuration for buttons.

An `onNewButton` event is thrown when a new button is created by the Renderer. The method `getContent()` returns the caption of the button.

To create a new button use `GXHtmlTagFactory.instance().newButton(...)`

See Also:

[GXTagListener.onNewButton\(GXRenderEvent, GXIButtonTag\)](#),  
[GXHostKeysTransformConfig.setRenderingType\(int\)](#), [GXTransformConstants.HOSTKEYS\\_BUTTONS](#),  
[GXHtmlTagFactory.newButton\(\)](#)

### Method Summary

String	<a href="#">getCaption()</a>
String	<a href="#">getOnClick()</a>
void	<a href="#">setCaption(String caption)</a>
void	<a href="#">setOnClick(String onClick)</a>

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXIClickableTag](#)

[handleEvent](#)

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#),  
[getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#),  
[hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#),  
[setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#),  
[setVisible](#)

### Methods

#### getCaption

public String **getCaption()**

**Returns:**

The caption of the button. **value** attribute in HTML.

---

## setCaption

```
public void setCaption(String caption)
```

### Parameters:

caption - The caption of the button. **value** attribute in HTML.

---

## getOnClick

```
public String getOnClick()
```

### Returns:

The onClick action to execute. JavaScript command in HTML.

---

## setOnClick

```
public void setOnClick(String onClick)
```

### Parameters:

onClick - The onClick action to execute. JavaScript command in HTML.

## com.sabratec.applinx.presentation.tags Interface GXICheckboxTag

All Superinterfaces:

[GXITag](#)

public interface **GXICheckboxTag**

extends [GXITag](#)

This interface represents a check box tag. The common usage is for when creating a GUI element in ApplinX composer.

An `onNewCheckbox` event is thrown when a new check box is created by the renderer. The method `getContent()` returns the checked/unchecked value of the check box depending if it is checked or not.

To create a new check box use `GXHtmlTagFactory.instance().newCheckbox(...)`

In HTML, this tag renders a hidden tag that saves the checked/unchecked value of the tag, so the checked/unchecked value will be passed to the server when the form is submitted.

### Method Summary

String	<a href="#">getCaption()</a>
String	<a href="#">getCheckedValue()</a>
String	<a href="#">getUncheckedValue()</a>
boolean	<a href="#">isChecked()</a>
void	<a href="#">setCaption(String caption)</a>
void	<a href="#">setChecked(boolean isChecked)</a>
void	<a href="#">setCheckedValue(String checkedValue)</a>
void	<a href="#">setUncheckedValue(String uncheckedValue)</a>

### Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

---

(continued from last page)

## isChecked

```
public boolean isChecked()
```

### Returns:

is the check box checked.

---

## setChecked

```
public void setChecked(boolean isChecked)
```

### Parameters:

isChecked - Set the check box to be checked.

---

## getCheckedValue

```
public String getCheckedValue()
```

### Returns:

the checked value

---

## setCheckedValue

```
public void setCheckedValue(String checkedValue)
```

### Parameters:

checkedValue - the checked value

---

## getUncheckedValue

```
public String getUncheckedValue()
```

### Returns:

the unchecked value

---

## setUncheckedValue

```
public void setUncheckedValue(String uncheckedValue)
```

### Parameters:

uncheckedValue - Set the unchecked value

---

## getCaption

```
public String getCaption()
```

(continued from last page)

**Returns:**

the check box caption

---

## **setCaption**

```
public void setCaption(String caption)
```

**Parameters:**

caption - Set the check box caption

## com.sabratec.applinx.presentation.tags Interface GXIClickableTag

All Superinterfaces:

[GXITag](#)

All Subinterfaces:

[GXImageTag](#), [GXILinkTag](#), [GXIButtonTag](#)

public interface **GXIClickableTag**

extends [GXITag](#)

### Method Summary

void	<a href="#">handleEvent</a> (GXIClickEvent event, <a href="#">GXIScreen</a> screen)
------	---

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#),  
[getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#),  
[hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#),  
[setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#),  
[setVisible](#)

### Methods

#### **handleEvent**

```
public void handleEvent(GXIClickEvent event,  
    GXIScreen screen)
```

## com.sabratec.applinx.presentation.tags Interface GXIComboboxTag

All Superinterfaces:

[GXITag](#)

public interface **GXIComboboxTag**

extends [GXITag](#)

This interface represents a combo box tag. The common usage is for when creating a GUI element in ApplinX composer.

An `onNewCombobox` event is thrown when a new combo box is created by the renderer. The method `getContent()` returns the checked value of the combo box.

To create a new combo box use `GXHtmlTagFactory.instance().newCombobox(...)`

### Method Summary

GXTextValueList	<a href="#">getItems()</a>
String	<a href="#">getSelectedValue()</a> Gets the value of the selected option
void	<a href="#">setItems</a> (GXTextValueList items) Sets a predefined text/value list
void	<a href="#">setSelectedValue</a> (String selectedValue) used for setting the option with the specified value

#### Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

#### getItems

```
public GXTextValueList getItems()
```

**Returns:**

The text/value list

#### setItems

```
public void setItems(GXTextValueList items)
```

Sets a predefined text/value list

(continued from last page)

**Parameters:**

items - The combo box items

---

**getSelectedValue**

```
public String getSelectedValue()
```

Gets the value of the selected option

**Returns:**

the value of the selected option

---

**setSelectedValue**

```
public void setSelectedValue(String selectedValue)
```

used for setting the option with the specified value

**Parameters:**

selectedValue - The option value

## com.sabratec.applinx.presentation.tags Interface GXILabelTag

All Superinterfaces:

[GXITag](#)

public interface **GXILabelTag**  
extends [GXITag](#)

This interface represents a label tag. The common usage is for displaying protected fields received from the host screen.

An `onNewLabel` event is thrown when a new label is created by the `Renderer`. The method `getContent()` returns the text of the label.

To create a new label use `GXHtmlTagFactory.instance().newLabel(...)`

### Method Summary

String	<a href="#">getText()</a> Returns the tag's text.
void	<a href="#">setText(String text)</a> Sets the tag's text.

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

#### getText

```
public String getText()
```

Returns the tag's text.

**Returns:**

the tag's text

#### setText

```
public void setText(String text)
```

Sets the tag's text.

**Parameters:**

text - the tag's text

## com.sabratec.applinx.presentation.tags Interface GXILinkTag

All Superinterfaces:

[GXIClickableTag](#), [GXITag](#)

public interface **GXILinkTag**  
extends [GXIClickableTag](#)

This interface represents a link tag. The common usage is for displaying the host keys as links.

An `onNewLink` event is thrown when a new link is created by the `Renderer`. The method `getContent()` returns the text of the link.

To create a new link use `GXHtmlTagFactory.instance().newLink(...)` TODO roi render config links

### Method Summary

String	<a href="#">getOnClick()</a>
String	<a href="#">getTarget()</a>
String	<a href="#">getText()</a> Returns the tag's text.
void	<a href="#">setOnClick(String onClick)</a>
void	<a href="#">setTarget(String targetUrl)</a>
void	<a href="#">setText(String text)</a> Sets the tag's text.

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXIClickableTag](#)

[handleEvent](#)

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

#### getOnClick

public String **getOnClick()**

---

(continued from last page)

**Returns:**

The onClick action to execute. JavaScript command in HTML.

---

**setOnClick**

```
public void setOnClick(String onClick)
```

**Parameters:**

onClick - The onClick action to execute. JavaScript command in HTML

---

**getTarget**

```
public String getTarget()
```

**Returns:**

The target URL. In HTML the default value is "#", which stands for no action.

---

**setTarget**

```
public void setTarget(String targetUrl)
```

**Parameters:**

targetUrl - The target URL. The default value is "#".

---

**getText**

```
public String getText()
```

Returns the tag's text.

**Returns:**

the tag's text

---

**setText**

```
public void setText(String text)
```

Sets the tag's text.

**Parameters:**

text - the tag's text

---

## com.sabratec.applinx.presentation.tags Interface GXImageTag

All Superinterfaces:

[GXIClickableTag](#), [GXITag](#)

public interface **GXImageTag**

extends [GXIClickableTag](#)

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXIClickableTag](#)

[handleEvent](#)

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#),  
[getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#),  
[hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#),  
[setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#),  
[setVisible](#)

## com.sabratec.applinx.presentation.tags Interface GXIRadioButtonGroupTag

All Superinterfaces:

[GXITag](#)

public interface **GXIRadioButtonGroupTag**

extends [GXITag](#)

This interface represents a radio button group tag. The common usage is for when creating a GUI element in ApplinX composer.

An `onNewRadioButtonGroup` event is thrown when a new radio button group is created by the Renderer. The method `getContent()` returns the value of the selected radio button.

To create a radio button group use `GXHtmlTagFactory.instance().newRadioButtonGroup(...)`

HTML Typical Output:

```
<table>
<tr>
<td>CAPTION1</td> <td><input type="radio" name="<TAG ID>" /></td>
<td>CAPTION2</td>
<td><input type="radio" name="<TAG ID>" /></td>
</tr>
</table>
```

### Method Summary

int	<a href="#">getAlignment()</a> Used to determine if the radio button list will be rendered vertically or horizontally
GXTextValueList	<a href="#">getItems()</a> Retrieves the list of text/values which represents the radio button items.
String	<a href="#">getSelectedValue()</a>
void	<a href="#">setAlignment(int alignment)</a> Used to determine if the radio button list will be rendered vertically or horizontally
void	<a href="#">setItems(GXTextValueList items)</a> Sets a predefined text/value list
void	<a href="#">setSelectedValue(String value)</a> Used for setting the relevant radio button according to a single radio button value

#### Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

(continued from last page)

---

## getAlignment

```
public int getAlignment()
```

Used to determine if the radio button list will be rendered vertically or horizontally

**Returns:**

vertical/horizontal alignment constant

**See Also:**

[GXPresentationConstants.ALIGNMENT\\_HORIZONTAL](#)

[GXPresentationConstants.ALIGNMENT\\_VERTICAL](#)

---

## setAlignment

```
public void setAlignment(int alignment)
```

Used to determine if the radio button list will be rendered vertically or horizontally

**Parameters:**

alignment - either vertical or horizontal constant

**See Also:**

[GXPresentationConstants.ALIGNMENT\\_HORIZONTAL](#)

[GXPresentationConstants.ALIGNMENT\\_VERTICAL](#)

---

## getItems

```
public GXTextValueList getItems()
```

Retrieves the list of text/values which represents the radio button items.

**Returns:**

text/value list of the radio buttons

---

## setItems

```
public void setItems(GXTextValueList items)
```

Sets a predefined text/value list

**Parameters:**

items - The combo box items

---

## getSelectedValue

```
public String getSelectedValue()
```

**Returns:**

The selected radio button value

---

## setSelectedValue

```
public void setSelectedValue(String value)
```

Used for setting the relevant radio button according to a single radio button value

---

(continued from last page)

**Parameters:**

`value` - The value of the radio button to be selected

## com.sabratec.applinx.presentation.tags Interface GXIRadioButtonTag

All Superinterfaces:

[GXITag](#)

public interface **GXIRadioButtonTag**

extends [GXITag](#)

This interface represents a single radio button, with a caption.

It should be used for creating single radio buttons and associating them to other single radio buttons. For example, in host tables and action fields, replace each action field in a table row with a radio button.

Use the `setName` method to associate it with other radio buttons.

The method `getContent()` returns the checked value of the radio button, when it is checked, and `null` when it is not.

To create a new radio button use `GXHtmlTagFactory.instance().newRadioButton(...)`

### Method Summary

String	<a href="#">getCaption()</a> Returns the caption of the radio button.
String	<a href="#">getCheckedValue()</a> Returns the checked value of the radio button.
String	<a href="#">getName()</a> Returns the name of the radio button tag.
boolean	<a href="#">isChecked()</a> Determines if the radio button is checked.
void	<a href="#">setCaption(String caption)</a> Sets the caption.
void	<a href="#">setChecked(boolean checked)</a> Sets if the radio button is checked.
void	<a href="#">setCheckedValue(String checkedValue)</a> Sets the checked value of the radio button.
void	<a href="#">setName(String name)</a> Sets a name for the radio button tag.

### Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

(continued from last page)

## Methods

### getName

```
public String getName()
```

Returns the name of the radio button tag.

**Returns:**

the name of the radio button tag

### setName

```
public void setName(String name)
```

Sets a name for the radio button tag.

**Parameters:**

name - a name for the radio button tag

### isChecked

```
public boolean isChecked()
```

Determines if the radio button is checked.

**Returns:**

If the radio button is checked

### setChecked

```
public void setChecked(boolean checked)
```

Sets if the radio button is checked.

**Parameters:**

checked - is the radio button checked

### getCheckedValue

```
public String getCheckedValue()
```

Returns the checked value of the radio button.

**Returns:**

Returns the checked value of the radio button

### setCheckedValue

```
public void setCheckedValue(String checkedValue)
```

Sets the checked value of the radio button.

**Parameters:**

checkedValue - the checked value of the radio button

(continued from last page)

## getCaption

```
public String getCaption()
```

Returns the caption of the radio button. The caption is written near the radio button tag.

**Returns:**

the caption of the radio button

---

## setCaption

```
public void setCaption(String caption)
```

Sets the caption. The caption is written near the radio button tag.

**Parameters:**

`caption` - the caption of the radio button

## com.sabratec.applinx.presentation.tags Interface GXIScreenTagModel

### public interface GXIScreenTagModel

This class is the model which contains a list of tags.

The model provides various methods for adding, removing, replacing and searching tags.

The final tag list in the screen tag model (after all transformations have been completed) will be rendered to the HTML output by the Renderer entity.

The tags are basically organised in the order they were created, but the order is not important for HTML presentation, since all tags are rendered in absolute terms. The way to search tags is either by checking the direction of the neighboring tag, by ID, by text or by specific position (row, column or both).

### Method Summary

void	<a href="#">add(<a href="#">GXITag</a> tag)</a> Add the specified tag to the screen tag model.
void	<a href="#">addNeighbor(<a href="#">GXITag</a> tag, <a href="#">GXITag</a> neighborTag, <a href="#">GXPosition</a> offset)</a> Add the specified tag to the screen tag model next to the neighboring tag.
<a href="#">GXTagList</a>	<a href="#">executeColumnQuery(int column)</a> Used for executing column based queries on the screen tag model.
<a href="#">GXITag</a>	<a href="#">executePositionQuery(<a href="#">GXPosition</a> position)</a> Used for executing position-based queries on the screen model.
<a href="#">GXITag</a>	<a href="#">executePositionQuery(int row, int column)</a> Used for executing position-based queries, on the screen tag model.
<a href="#">GXTagList</a>	<a href="#">executeQuery(<a href="#">GXITagQuery</a> query)</a> Executes a query on the screen model.
<a href="#">GXTagList</a>	<a href="#">executeRowQuery(int row)</a> Used for executing row based queries on the screen tag model.
<a href="#">GXTagList</a>	<a href="#">executeTextQuery(String searchText)</a> Used for executing text based queries on the screen tag model.
int	<a href="#">getFontSize()</a> The font size (in pixels) in which the tags will be displayed.
<a href="#">GXITag</a>	<a href="#">getNeighbor(<a href="#">GXITag</a> tag, int direction)</a> Returns the neighboring tag in the specified direction.
<a href="#">GXITag</a>	<a href="#">getTagById(String tagId)</a> Finds a tag by the specified tag ID
<a href="#">GXITag[]</a>	<a href="#">getTags()</a> Returns a tag's array which contains all the screen tag model root tags.
void	<a href="#">remove(<a href="#">GXITag</a> tag)</a> Removes the specified tag.

void

[replace](#)([GXITag](#) originalTag, [GXITag](#) newTag)

Replaces the specified tag with the original tag in the screen tag model.

## Methods

### getTagById

```
public GXITag getTagById(String tagId)
```

Finds a tag by the specified tag ID

**Parameters:**

tagId - the ID of the searched tag

**Returns:**

The tag whose ID matches the searched tag ID

### executeQuery

```
public GXTagList executeQuery(GXITagQuery query)
```

Executes a query on the screen model.

The query can be either a built-in query such as:

[GXTextQuery](#),

[GXAreaQuery](#)

or a user defined query (class that implements [GXITagQuery](#))

**Parameters:**

query - the query to execute.

**Returns:**

the tags meeting the query. If the query is not met, then an empty list is returned

### executeTextQuery

```
public GXTagList executeTextQuery(String searchText)
```

Used for executing text based queries on the screen tag model.

**Parameters:**

searchText - the text to search for within the tags. Searches for sub string matches.

**Returns:**

Tags meeting the search criteria.

### executeColumnQuery

```
public GXTagList executeColumnQuery(int column)  
throws GXIllegalPositionException
```

Used for executing column based queries on the screen tag model.

**Parameters:**

column - The searched tags column

**Returns:**

(continued from last page)

Tags meeting the search criteria.

**Throws:**

`GXIllegalPositionException` - when the column is outside the screen tag model bounds

---

## executeRowQuery

```
public GXTagList executeRowQuery(int row)
    throws GXIllegalPositionException
```

Used for executing row based queries on the screen tag model.

**Parameters:**

row - The searched tags row

**Returns:**

tags meeting the search criteria

**Throws:**

`GXIllegalPositionException` - when the column is outside the screen tag model bounds

---

## executePositionQuery

```
public GXITag executePositionQuery(int row,
    int column)
    throws GXIllegalPositionException
```

Used for executing position-based queries, on the screen tag model.

**Parameters:**

row - The searched tag row

column - The searched tag column

**Returns:**

the tag in the specified position

**Throws:**

`GXIllegalPositionException` - when the column is outside the screen tag model bounds

---

## executePositionQuery

```
public GXITag executePositionQuery(GXPosition position)
    throws GXIllegalPositionException
```

Used for executing position-based queries on the screen model.

**Parameters:**

position - The searched position

**Returns:**

the tag in the specified position

**Throws:**

`GXIllegalPositionException` - when the column is outside the screen tag model bounds

---

## getFontSize

```
public int getFontSize()
```

(continued from last page)

The font size (in pixels) in which the tags will be displayed. The font size is also used for calculating actual top/left according to the tag's position.

**Returns:**

the font size.

**See Also:**

[GXRenderConfig.setFontSize\(int\)](#)

---

## getNeighbor

```
public GXITag getNeighbor(GXITag tag,  
                           int direction)  
throws GXUndefinedPositionException
```

Returns the neighboring tag in the specified direction.

**Parameters:**

tag - the tag whose neighbor is being searched for  
direction - The direction to search

**Returns:**

the neighboring tag in the specified direction

**Throws:**

[GXUndefinedPositionException](#) - when the specified tag's position is empty

**See Also:**

[GXTagModelConstants.NEIGHBOR\\_ABOVE](#)

[GXTagModelConstants.NEIGHBOR\\_BELOW](#)

[GXTagModelConstants.NEIGHBOR\\_LEFT](#)

[GXTagModelConstants.NEIGHBOR\\_RIGHT](#)

---

## add

```
public void add(GXITag tag)
```

Add the specified tag to the screen tag model.

Note: The Insert method is not supported since the tags order is not important for the rendering.

**Parameters:**

tag - The new tag to add

---

## getTags

```
public GXITag[] getTags()
```

Returns a tag's array which contains all the screen tag model root tags. Child tags of other tags will not be included in the list, but may be accessed by their parent tag. This method should be used for performing iterations on the tags.

**Returns:**

A tags array

---

## addNeighbor

```
public void addNeighbor(GXITag tag,  
                        GXITag neighborTag,  
                        GXPosition offset)
```

(continued from last page)

Add the specified tag to the screen tag model next to the neighboring tag. The added tag will copy it's position taking into account the offset position from the neighboring tag and the length of the neighboring tag. The tag offset top/left will also be copied, in cases where the neighboring tag has an additional offset from its position. This method can be used for adding icons near a tag. For example, adding a calendar icon next to a date text field.

**Parameters:**

tag - The new tag to add  
neighborTag - The neighboring tag of the new tag  
offset - the new tag offset position from the neighboring tag

---

## replace

```
public void replace(GXITag originalTag,  
                   GXITag newTag)
```

Replaces the specified tag with the original tag in the screen tag model.

The following original tag properties will be copied to the new tag: position, length, offsetTop, offsetLeft, hostField

Is useful for changing a tag type. For example, changing a text to a link.

**Parameters:**

originalTag - the original tag which should be replaced  
newTag - the new tag that should replace the original tag

---

## remove

```
public void remove(GXITag tag)
```

Removes the specified tag. The removed tag will not be displayed in the rendering process result.

**Parameters:**

tag - the tag to remove.

## com.sabratec.applinx.presentation.tags Interface GXITableCellTag

All Superinterfaces:

[GXITag](#)

public interface **GXITableCellTag**

extends [GXITag](#)

This interface represents a table cell tag. The common usage is for creating a table cell within a host table in the ApplinX composer. An `onNewTableCell` event is thrown when a new table cell is created by the Renderer. The method `getContent()` returns the text of the cell. If it's empty, it returns the content of the 1st child tag, which is common behavior in host tables. The method `getHostField()` returns the host field the cell is based on. To create a new table cell use `GXHtmlTagFactory.instance().newTableCell(...)`

### Method Summary

String	<a href="#">getText()</a> Returns the cell's text.
void	<a href="#">setIsHeader</a> (boolean isHeader) Determines if the cell is a table header cell or a table body cell.
void	<a href="#">setText</a> (String text) Sets the cell's text.

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

#### setIsHeader

```
public void setIsHeader(boolean isHeader)
```

Determines if the cell is a table header cell or a table body cell.

**Parameters:**

`isHeader` - if the cell is a header cell

#### getText

```
public String getText()
```

Returns the cell's text. If the cell text is empty, returns the content of the 1st child tag if only one child tag exists.

**Returns:**

the cell's text

---

## setText

```
public void setText(String text)
```

Sets the cell's text.

### Parameters:

text - the cell's text

## com.sabratec.applinx.presentation.tags Interface GXITableRowTag

All Superinterfaces:

[GXITag](#)

public interface **GXITableRowTag**

extends [GXITag](#)

This interface represents a table row tag. The common usage is for creating a table row within a host table in the ApplinX composer . An `onNewTableRow` event is thrown when a new table row is created by the Renderer.

The method `getContent()` returns null.

The method `getHostField()` also returns null.

To create a new table row use `GXHtmlTagFactory.instance().newTableRow(...)`

### Method Summary

<a href="#">GXTableCellList</a>	<a href="#">getCells()</a>
---------------------------------	----------------------------

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

#### **getCells**

public [GXTableCellList](#) **getCells()**

**Returns:**

The child cells of the row

## com.sabratec.applinx.presentation.tags Interface GXITableTag

All Superinterfaces:

[GXITag](#)

public interface **GXITableTag**

extends [GXITag](#)

This interface represents a table tag. The common usage for using this tag is when a host table is created in ApplinX composer. An `onNewTable` event is thrown when a new table row is created by the `Renderer`.

The method `getContent()` returns null.

The method `getHostField()` also returns null.

To create a new table use `GXHtmlTagFactory.instance().newTable(...)`

### Method Summary

<a href="#">GXITableRowTag</a>	<a href="#">getHeaders()</a>
<a href="#">GXTableRowList</a>	<a href="#">getRows()</a> Returns the children

Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#), [getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#), [hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#), [setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#), [setVisible](#)

### Methods

#### getHeaders

public [GXITableRowTag](#) **getHeaders()**

**Returns:**

a table row tag which contains the header cell's.

#### getRows

public [GXTableRowList](#) **getRows()**

Returns the children

**Returns:**

a list of the table's row tags

## com.sabratec.applinx.presentation.tags

### Interface GXITag

#### All Subinterfaces:

[GXITextFieldTag](#), [GXITableTag](#), [GXITableRowTag](#), [GXITableCellTag](#), [GXIRadioButtonTag](#), [GXIRadioButtonGroupTag](#), [GXILabelTag](#), [GXIComboboxTag](#), [GXIClickableTag](#), [GXImageTag](#), [GXILinkTag](#), [GXIButtonTag](#), [GXICheckboxTag](#)

#### public interface GXITag

This interface is the root of all tag classes, such as text fields, labels, links, combo boxes, etc.

The tag class is host oriented. It provides a reference to the host field, and allows locating tags by setting their position [setPosition\(int, int\)](#) in host units (row, column).

The initial position is derived from the host field, and can be modified. The position will be rendered to HTML units as top/left. The tag also contains the methods `getAttribute` and `setAttribute` which can be used to read/modify any attribute of the tag.

The tag may have an offset [setOffsetLeft\(double\)](#), [setOffsetTop\(double\)](#) from its position.

The offset is used for aligning the tag when neighboring tags such as a combo box that may overlap the tag.

The offset will be added to the final calculated top/left attributes which are derived from the tag's position. In case the tag doesn't have any position, the offset top/left will be considered as the actual top/left.

A tag may have child tags - [getChildren\(\)](#), and a parent tag [getParent\(\)](#)

HTML specific tags are also available such as:

- `GXHtmlGenericTag` for creating a generic tag by specifying the tag name
- `GXHtmlString` for using direct strings and adding them to the screen tag model.

### Method Summary

void	<a href="#">copyFrom(GXITag tag)</a> Copies tags fields
String	<a href="#">getAttribute(String name)</a> Returns the value of an attribute according to its name.
<a href="#">GXTagList</a>	<a href="#">getChildren()</a> Returns the tag's child tags.
String	<a href="#">getContent()</a> Used for retrieving the content of the tag.
<a href="#">GXIField</a>	<a href="#">getHostField()</a> Returns the host field which the tag is based on.
String	<a href="#">getId()</a> Returns the tag's unique ID.
int	<a href="#">getLength()</a> Returns the length that the tag is considered to occupy on the host screen.
double	<a href="#">getOffsetLeft()</a> Returns the offset left position from the <code>position</code> attribute.

double	<a href="#"><code>getOffsetRight()</code></a> Returns the offset right position.
double	<a href="#"><code>getOffsetTop()</code></a> Returns the offset top position from the <code>position</code> attribute.
<a href="#"><code>GXITag</code></a>	<a href="#"><code>getParent()</code></a> Returns the parent tag.
<a href="#"><code>GXPosition</code></a>	<a href="#"><code>getPosition()</code></a> Returns the position of the tag in host screen units (row, column).
String	<a href="#"><code>getTagName()</code></a> Returns the concrete name of the tag type.
boolean	<a href="#"><code>hasChildren()</code></a> Returns <code>true</code> if the tag has child tags.
boolean	<a href="#"><code>isVisible()</code></a> Determines whether to display the tag or not.
void	<a href="#"><code>removeAttribute(String name)</code></a> Removes an attribute according to its name.
void	<a href="#"><code>setAttribute(String name, String value)</code></a> Sets the value of a given attribute.
void	<a href="#"><code>setId(String id)</code></a> Sets a unique ID for the tag.
void	<a href="#"><code>setLength(int length)</code></a> Sets the actual length of the tag on the screen tag model.
void	<a href="#"><code>setOffsetLeft(double left)</code></a> Sets the offset left position.
void	<a href="#"><code>setOffsetRight(double left)</code></a> Sets the offset right position.
void	<a href="#"><code>setOffsetTop(double top)</code></a> Sets the offset top position.
void	<a href="#"><code>setPosition(GXPosition position)</code></a> Sets the position of the tag in host screen units (row, column).
void	<a href="#"><code>setPosition(int row, int column)</code></a> A convenient method for setting row and column instead of using a position object.
void	<a href="#"><code>setVisible(boolean isVisible)</code></a> Sets the tag to be visible/invisible.
void	<a href="#"><code>setVisible(boolean isVisible, boolean doRender)</code></a> Used for setting the tag to be visible/invisible.

## Methods

(continued from last page)

---

## getParent

```
public GXITag getParent()
```

Returns the parent tag. If this tag has no parent, returns `null`.

**Returns:**

the parent tag.

---

## hasChildren

```
public boolean hasChildren()
```

Returns `true` if the tag has child tags.

**Returns:**

`true` if the tag has child tags. If this tag doesn't support child tags, this method always returns `false`.

---

## getChildren

```
public GXTagList getChildren()
```

Returns the tag's child tags. If there are no child tags defined, returns an empty list.  
If this tag doesn't support child tags, returns `null`.

**Returns:**

the tag's children tags.

---

## getId

```
public String getId()
```

Returns the tag's unique ID.

Naming convension:

- When the tag is based on a host field, it will have the ID: "POS<ABSOLUTE SCREEN POSITION>". For example "POS1232".
- If the tag is based on an application field then the tag will have it's name.
- If the tag is based on a multiple application field then the tag will have the name "<APPFIELD\_NAME>\_00<INDEX>". For example: "ItemNumber\_003"

**Returns:**

the tag's unique ID.

---

## setId

```
public void setId(String id)
```

Sets a unique ID for the tag.

**Parameters:**

`id` - a unique ID for tag

---

(continued from last page)

## getOffsetTop

```
public double getOffsetTop()
```

Returns the offset top position from the `position` attribute.

The default value is 0.

When either a GUI element transformation and/or a Table transformation is activated, this attribute may contain a positive value since GUI elements and tables are "higher" elements than standard labels and text fields, and the transformation may change this attribute to avoid tags overlapping each other.

### Returns:

the offset top position attribute

### See Also:

[setOffsetTop\(double\)](#)  
[getOffsetLeft\(\)](#)  
[setOffsetLeft\(double\)](#)  
[setPosition\(GXPosition\)](#)  
[getPosition\(\)](#)

---

## setOffsetTop

```
public void setOffsetTop(double top)
```

Sets the offset top position. This attribute, along with the `offsetLeft` attribute, adjusts the tag's final top/left positioning which is based on the `position` attribute of the tag. When the tag has no position, the offset top will be the actual top attribute in the tag output.

### Parameters:

`top` - the top position attribute.

### See Also:

[getOffsetTop\(\)](#)  
[getOffsetLeft\(\)](#)  
[setOffsetLeft\(double\)](#)

---

## getOffsetLeft

```
public double getOffsetLeft()
```

Returns the offset left position from the `position` attribute.

The default value is 0.

When a GUI element transformation is activated, this attribute may contain a positive value since GUI elements are "longer" elements than standard text fields, and the transformation may change this attribute to avoid tags overriding each other.

### Returns:

the left position attribute

### See Also:

[setOffsetLeft\(double\)](#)  
[getOffsetTop\(\)](#)  
[setOffsetTop\(double\)](#)  
[setPosition\(GXPosition\)](#)  
[getPosition\(\)](#)

---

(continued from last page)

## getOffsetRight

```
public double getOffsetRight()
```

Returns the offset right position. from the `position` attribute. Rendered only when left is 0.

The default value is 0.

### Returns:

the right position attribute

### See Also:

[setOffsetRight\(double\)](#)  
[getOffsetTop\(\)](#)  
[setOffsetTop\(double\)](#)  
[setPosition\(GXPosition\)](#)  
[getPosition\(\)](#)

---

## setOffsetLeft

```
public void setOffsetLeft(double left)
```

Sets the offset left position. This attribute, along with the `offsetTop` attribute, adjusts the tag's final top/left positioning which is based on the `position` attribute of the tag. When the tag has no position, the offset left will be the actual left attribute in the tag output.

### Parameters:

`left` - the left position attribute

### See Also:

[getOffsetLeft\(\)](#)  
[getOffsetTop\(\)](#)  
[setOffsetTop\(double\)](#)  
[setPosition\(GXPosition\)](#)  
[getPosition\(\)](#)

---

## setOffsetRight

```
public void setOffsetRight(double left)
```

Sets the offset right position. This attribute, along with the `offsetTop` attribute, adjusts the tag's final top/right positioning which is based on the `position` attribute of the tag, and when configuring to render from right to left. When the tag has no position, the offset right will be the actual right attribute in the tag output. For this method to work, you must open the ApplinX Framework Configuration Editor and select the Render tags from right check box in the Instant node.

### Parameters:

`right` - the right position attribute

### See Also:

[getOffsetRight\(\)](#)  
[getOffsetTop\(\)](#)  
[setOffsetTop\(double\)](#)  
[setPosition\(GXPosition\)](#)  
[getPosition\(\)](#)

---

## setPosition

```
public void setPosition(GXPosition position)
```

(continued from last page)

Sets the position of the tag in host screen units (row, column). The position will be translated in the result as top/left attributes, using the configured font size [GXRenderConfig.setFontSize\(int\)](#), and the specified offset top/left attributes of the tag.

Other considerations:

- When a tag has leading spaces, it will increase the tag's left attribute, and the spaces will be trimmed (only in the final result).
- RTL languages - Hebrew/Arabic.

This attribute is not rendered, since it does not contribute any visual information.

**Parameters:**

`position` - the position that the tag is considered to have on the host screen

**See Also:**

[GXIScreenTagModel.getNeighbor\(GXITag, int\)](#)

## setPosition

```
public void setPosition(int row,
                       int column)
```

A convenient method for setting row and column instead of using a position object.

**Parameters:**

`row` - the row of the position

`column` - the column of the position

**See Also:**

[setPosition\(GXPosition\)](#)

## getPosition

```
public GXPosition getPosition()
```

Returns the position of the tag in host screen units (row, column). A tag that was derived from the host screen will have the position of the field it was derived from.

The position may also be used for locating neighboring tags using the [GXIScreenTagModel.getNeighbor\(GXITag, int\)](#) method.

In addition the position may be used for locating tags by position using the [GXIScreenTagModel.executePositionQuery\(GXPosition\)](#) method.

This attribute is not rendered, since it does not contribute any visual information.

**Returns:**

the position of the tag on the screen tag model. If the tag is not related to a host field element, the returned value is null.

## getLength

```
public int getLength()
```

Returns the length that the tag is considered to occupy on the host screen.

This attribute should not be rendered, since it does not contribute any visual information.

**Returns:**

the length of the tag on the host screen. If the tag does not have a length on the host screen, 0 is returned.

## setLength

```
public void setLength(int length)
```

Sets the actual length of the tag on the screen tag model.

The length is primarily used for locating the tag by position using the [GXIScreenTagModel.executePositionQuery\(GXPosition\)](#) method, and for locating neighbors [GXIScreenTagModel.getNeighbor\(GXITag, int\)](#). If a length is not specified the tag will be considered as the size of 1 for search purposes.

For text fields, the length attribute determines the actual size of the text field.

This attribute should not be rendered, since it does not contribute any visual information.

### Parameters:

length - the length of the tag on the screen tag model

### See Also:

[GXIScreenTagModel.executePositionQuery\(GXPosition\)](#)  
[GXIScreenTagModel.getNeighbor\(GXITag, int\)](#)

---

## getTagName

```
public String getTagName()
```

Returns the concrete name of the tag type. Example, 'A' for links in HTML.

### Returns:

the actual tag name (not the Name attribute)

---

## setAttribute

```
public void setAttribute(String name,  
                          String value)
```

Sets the value of a given attribute.

If the attribute doesn't exist - it is added. If the attribute exists - its value is replaced.

The attribute is case sensitive. All the attributes built by the renderer are lower case. When adding an HTML css class, it is appended to the other css classes.

To set a single css class use `removeAttribute("class")` before the above method.

### Parameters:

name - the name of the attribute

value - the value to set for the attribute

---

## getAttribute

```
public String getAttribute(String name)
```

Returns the value of an attribute according to its name.

### Parameters:

name - the name of the attribute

### Returns:

the value of the attribute. If the attribute doesn't exist, null is returned.

---

---

## removeAttribute

```
public void removeAttribute(String name)
```

Removes an attribute according to its name.

**Parameters:**

name - the name of the attribute

---

## isVisible

```
public boolean isVisible()
```

Determines whether to display the tag or not. The tag will still remain part of the tag model.

**Returns:**

if the the tag visible

---

## setVisible

```
public void setVisible(boolean isVisible)
```

Sets the tag to be visible/invisible. The tag will not be included at all in the result.

**Parameters:**

isVisible - if the tag should be visible

**See Also:**

[setVisible\(boolean, boolean\)](#)

---

## setVisible

```
public void setVisible(boolean isVisible,  
    boolean doRender)
```

Used for setting the tag to be visible/invisible. For HTML tags if setVisible is false and doRender is true, then the tag will be rendered with style="display:none" attribute.

**Parameters:**

isVisible - if the tag should be visible

doRender - determines if the tag should be rendered invisibly

---

## getContent

```
public String getContent()
```

Used for retrieving the content of the tag.

A unified method for all types of tags. All tag types implementation returns the content from a different property, in order to be able to query any tag for it's content in a generic way.

For example:

- A combo box returns the selected value as the content
- An HTML label returns it's inner text

**Returns:**

(continued from last page)

the content of the tag

---

## getHostField

```
public GXIField getHostField()
```

Returns the host field which the tag is based on. May be used for querying the host field content and properties.

**Returns:**

the host field that the tag is based on

---

## copyFrom

```
public void copyFrom(GXITag tag)
```

Copies tags fields

**Parameters:**

tag - the tag to copy from

## com.sabratec.applinx.presentation.tags Interface GXITextFieldTag

All Superinterfaces:

[GXITag](#)

public interface **GXITextFieldTag**

extends [GXITag](#)

This interface represents a text field tag. The common usage for using this tag is when a host field is an unprotected field.

An `onNewTextField` event is thrown when a new text field is created by the Renderer.

The method `{#getContent() }` returns the text of the text field (**value** attribute in HTML).

To create a new text field use `GXHtmlTagFactory.intage().newTextField(...)`

See Also:

[GXTagListener.onNewTextField\(GXRenderEvent, GXITextFieldTag\)](#),  
[GXHtmlTagFactory.newTextField\(String\)](#)

### Method Summary

String	<a href="#">getText()</a> Returns the tag's text, <b>value</b> attribute in HTML.
boolean	<a href="#">isPassword()</a> Returns if the text field is a password field.
void	<a href="#">setPassword(boolean isPassword)</a> Set if the text field is a password field.
void	<a href="#">setText(String text)</a> Sets the tag's text, <b>value</b> attribute in HTML.

### Methods inherited from interface [com.sabratec.applinx.presentation.tags.GXITag](#)

[copyFrom](#), [getAttribute](#), [getChildren](#), [getContent](#), [getHostField](#), [getId](#), [getLength](#),  
[getOffsetLeft](#), [getOffsetRight](#), [getOffsetTop](#), [getParent](#), [getPosition](#), [getTagName](#),  
[hasChildren](#), [isVisible](#), [removeAttribute](#), [setAttribute](#), [setId](#), [setLength](#),  
[setOffsetLeft](#), [setOffsetRight](#), [setOffsetTop](#), [setPosition](#), [setPosition](#), [setVisible](#),  
[setVisible](#)

### Methods

#### **isPassword**

public boolean **isPassword()**

Returns if the text field is a password field.

**Returns:**

if the text field is a password field

---

(continued from last page)

## setPassword

```
public void setPassword(boolean isPassword)
```

Set if the text field is a password field.

### Parameters:

isPassword - If the text field is a password field

---

## getText

```
public String getText()
```

Returns the tag's text, **value** attribute in HTML.

### Returns:

the tag's text

---

## setText

```
public void setText(String text)
```

Sets the tag's text, **value** attribute in HTML.

### Parameters:

text - the tag's text

---

## com.sabratec.applinx.presentation.tags

### Class GXTableCellList

```

java.lang.Object
  |
  +--GXInitializableTagList
        |
        +--com.sabratec.applinx.presentation.tags.GXTableCellList
  
```

public class **GXTableCellList**  
 extends GXInitializableTagList

This interface represents a list of table cell tags. The list is a member of a table row tag. It allows retrieving, adding and removing table cells from the cell list. Each cell also may have child tags such as text fields, labels, combo boxes, etc.

**See Also:**

[GXITableRowTag.getCells\(\)](#)

### Constructor Summary

public	<a href="#">GXTableCellList</a> ( <a href="#">GXITableRowTag</a> rowTag) Initializes the list and associates it to the specified row tag.
--------	--

### Method Summary

void	<a href="#">add</a> ( <a href="#">GXITableCellTag</a> cell) Adds a new cell to the table cell list.
<a href="#">GXITableCellTag</a>	<a href="#">get</a> (int index) Returns the table cell according to the index.
void	<a href="#">insert</a> ( <a href="#">GXITableCellTag</a> cell, int index) Inserts the cell at the specified index.

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXTableCellList

public **GXTableCellList**([GXITableRowTag](#) rowTag)

Initializes the list and associates it to the specified row tag. All added cell tags [getParent\(\)](#) method will return the row tag

#### Parameters:

rowTag - the row's parent tag

## Methods

(continued from last page)

## get

```
public GXTableCellTag get(int index)
    throws ArrayIndexOutOfBoundsException
```

Returns the table cell according to the index.

**Parameters:**

index - the index of the requested cell

**Returns:**

the table cell according to the index

**Throws:**

[ArrayIndexOutOfBoundsException](#) - if the index was invalid.

---

## insert

```
public void insert(GXTableCellTag cell,
    int index)
    throws ArrayIndexOutOfBoundsException
```

Inserts the cell at the specified index.

**Parameters:**

cell - the new table cell to insert

index - the cell-list index to insert the cell at

**Throws:**

[ArrayIndexOutOfBoundsException](#) - if the index was invalid.

---

## add

```
public void add(GXTableCellTag cell)
```

Adds a new cell to the table cell list.

**Parameters:**

cell - The new cell to add

## com.sabratec.applinx.presentation.tags

### Class GXTableRowList

```

java.lang.Object
  |
  +--GXInitializableTagList
        |
        +--com.sabratec.applinx.presentation.tags.GXTableRowList
  
```

```

public class GXTableRowList
extends GXInitializableTagList
  
```

This interface represents a list of table row tags. The list is a member of a table tag. It allows retrieving, adding and removing table rows from the row list.

**See Also:**

[GXITableTag.getRows\(\)](#)

### Constructor Summary

public	<a href="#">GXTableRowList</a> ( <a href="#">GXITableTag</a> tableTag) Initializes the list and associates it to the specified table tag.
--------	--

### Method Summary

void	<a href="#">add</a> ( <a href="#">GXITableRowTag</a> row) Adds a new row to the table row list
<a href="#">GXITableRowTag</a>	<a href="#">get</a> (int index) Returns the table row according to the index.
void	<a href="#">insert</a> ( <a href="#">GXITableRowTag</a> row, int index) Inserts the row at the specified index.

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXTableRowList

```
public GXTableRowList(GXITableTag tableTag)
```

Initializes the list and associates it to the specified table tag. All added row tags [getParent\(\)](#) method will return the row tag

**Parameters:**

tableTag - the row's parent tag

## Methods

---

(continued from last page)

## get

```
public GXTableRowTag get(int index)  
    throws ArrayIndexOutOfBoundsException
```

Returns the table row according to the index.

**Parameters:**

index - the index of the requested row

**Returns:**

the table row according to the index

**Throws:**

[ArrayIndexOutOfBoundsException](#) - if the index was invalid.

---

## insert

```
public void insert(GXTableRowTag row,  
    int index)
```

Inserts the row at the specified index.

**Parameters:**

row - the new table row to insert

index - the row-list index to insert the row at

**Throws:**

[ArrayIndexOutOfBoundsException](#) - if the index was invalid.

---

## add

```
public void add(GXTableRowTag row)
```

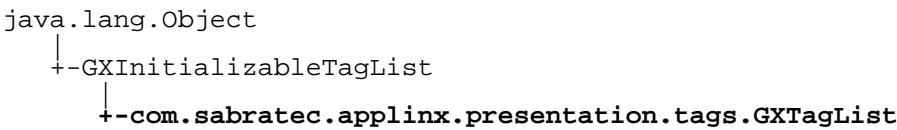
Adds a new row to the table row list

**Parameters:**

row - The new row to add

# com.sabratec.applinx.presentation.tags

## Class GXTagList



```
public class GXTagList
extends GXInitializableTagList
```

This interface represents a list of tags. The list may be retrieved from an executed query on the screen tag model. The list allows retrieving,adding and removing tags from the tags list.

See Also:  
[GXIScreenTagModel.executeQuery\(GXITagQuery\)](#)

### Constructor Summary

public	<a href="#">GXTagList()</a>
public	<a href="#">GXTagList(GXITag parentTag)</a> Initializes the list and associates it to the parent tag.

### Method Summary

void	<a href="#">add(GXITag tag)</a> Add a new tag to the tag list
<a href="#">GXITag</a>	<a href="#">get(int index)</a> Return the cell according to the index.
void	<a href="#">insert(GXITag tag, int index)</a> Inserts the tag at the specified index.
void	<a href="#">remove(GXITag tag)</a> Removes the specified tag from the tag list
void	<a href="#">replace(GXITag originalTag, GXITag newTag)</a> Replaces the specified tag with the original tag in the tag list.

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

### Constructors

#### GXTagList

```
public GXTagList()
```

## GXTagList

```
public GXTagList(GXITag parentTag)
```

Initializes the list and associates it to the parent tag. All added tags `getParent()` method will return the parent tag

### Parameters:

`parentTag` - the parent tag

## Methods

### replace

```
public void replace(GXITag originalTag,  
                   GXITag newTag)
```

Replaces the specified tag with the original tag in the tag list.

The following original tag properties will be copied to the new tag:  
`position, length, offsetTop, offsetLeft, hostField`

### Parameters:

`originalTag` - the original tag which should be replaced

`newTag` - the new tag that should replace the original tag

---

### remove

```
public void remove(GXITag tag)
```

Removes the specified tag from the tag list

### Parameters:

`tag` - The tag to remove

---

### insert

```
public void insert(GXITag tag,  
                  int index)
```

Inserts the tag at the specified index.

### Parameters:

`tag` - the new tag to insert

`index` - the tag-list index to insert the tag at

### Throws:

`ArrayIndexOutOfBoundsException` - if the index was invalid.

---

### get

```
public GXITag get(int index)
```

Return the cell according to the index.

### Parameters:

`index` - the index of the requested tag

(continued from last page)

**Returns:**

the tag according to the index

**Throws:**

`ArrayIndexOutOfBoundsException` - if the index was invalid.

---

## **add**

```
public void add(GXITag tag)
```

Add a new tag to the tag list

**Parameters:**

`tag` - The new tag to add

## com.sabratec.applinx.presentation.tags Class GXTagModelConstants

java.lang.Object

└─com.sabratec.applinx.presentation.tags.GXTagModelConstants

public class **GXTagModelConstants**  
extends Object

This class contains constants used by [GXIScreenTagModel](#)

### Field Summary

public static final	<a href="#">NEIGHBOR_ABOVE</a> Value: 0
public static final	<a href="#">NEIGHBOR_BELOW</a> Value: 1
public static final	<a href="#">NEIGHBOR_LEFT</a> Value: 2
public static final	<a href="#">NEIGHBOR_RIGHT</a> Value: 3

### Constructor Summary

public	<a href="#">GXTagModelConstants()</a>
--------	---------------------------------------

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

#### NEIGHBOR\_ABOVE

public static final int **NEIGHBOR\_ABOVE**

Constant value: 0

#### See Also:

[GXIScreenTagModel.getNeighbor\(GXITag, int\)](#)

#### NEIGHBOR\_BELOW

public static final int **NEIGHBOR\_BELOW**

---

(continued from last page)

Constant value: **1**

**See Also:**

[GXIScreenTagModel.getNeighbor\(GXITag, int\)](#)

---

## NEIGHBOR\_LEFT

public static final int **NEIGHBOR\_LEFT**

Constant value: **2**

**See Also:**

[GXIScreenTagModel.getNeighbor\(GXITag, int\)](#)

---

## NEIGHBOR\_RIGHT

public static final int **NEIGHBOR\_RIGHT**

Constant value: **3**

**See Also:**

[GXIScreenTagModel.getNeighbor\(GXITag, int\)](#)

## Constructors

### GXTagModelConstants

public **GXTagModelConstants()**

---

## Package

# com.sabratec.applinx.presentation.tags.html

This package provides specific HTML tags such as: HTML hidden tag, Generic tag nad HTML string tag.

## com.sabratec.applinx.presentation.tags.html Class GXHtmlGenericTag

```
java.lang.Object
  |
  +--GXHtmlTag
        |
        +--com.sabratec.applinx.presentation.tags.html.GXHtmlGenericTag
```

```
public class GXHtmlGenericTag
extends GXHtmlTag
```

This class represents a generic HTML tag, which has a display functionality only  
For example, <div>, <font>, <p> etc.

To create a new HTML generic tag, use `GXHtmlTagFactory.istance().newGenericTag(<TAG NAME>)`.

Use the `setInnerText` method to set text to the tag. Use `setAttribute` to add an attribute to the tag. Use `getChildren().add(...)` to create child tags.

**See Also:**

[GXHtmlTagFactory.newGenericTag\(String\)](#)

### Method Summary

String	<a href="#">getContent()</a> Returns the inner text of the tag.
void	<a href="#">recycle()</a> Recycles the tag in the HTML tag factory
void	<a href="#">setInnerText(String innerText)</a> Sets the inner text of the tag
void	<a href="#">setTagName(String tagName)</a> Sets the concrete tag name.

#### Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

### Methods

#### setInnerText

```
public void setInnerText(String innerText)
```

Sets the inner text of the tag

**Parameters:**

`innerText` - The inner text of the tag

#### recycle

```
public void recycle()
```

(continued from last page)

Recycles the tag in the HTML tag factory

---

## **setTagName**

```
public void setTagName(String tagName)
```

Sets the concrete tag name.

### **Parameters:**

tagName - the concrete tag name.

---

## **getContent**

```
public String getContent()
```

Returns the inner text of the tag.

### **Returns:**

The inner text of the tag

## com.sabratec.applinx.presentation.tags.html Class GXHtmlHidden

```
java.lang.Object
```

```
└--GXHtmlTag
```

```
└--com.sabratec.applinx.presentation.tags.html.GXHtmlHidden
```

```
public class GXHtmlHidden
extends GXHtmlTag
```

This class represents an HTML hidden input (<input type="hidden").

The method {#getContent()} returns the **value** attribute of the hidden tag.

To create a new hidden tag use `GXHtmlTagFactory.intance().newHiddenTag(...)`

### Method Summary

String	<a href="#">getContent()</a> Returns the value of the hidden tag
String	<a href="#">getValue()</a>
void	<a href="#">initialize()</a> For internal use
void	<a href="#">recycle()</a> For internal use
void	<a href="#">setId(String tagId)</a> Sets the tag ID, also sets the name attribute
void	<a href="#">setValue(String value)</a> Sets the value of the text field

#### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

### Methods

#### getValue

```
public String getValue()
```

##### Returns:

The value of the hidden tag

(continued from last page)

## setValue

```
public void setValue(String value)
```

Sets the value of the text field

### Parameters:

value - the value of the hidden tag

---

## setId

```
public void setId(String tagId)
```

Sets the tag ID, also sets the name attribute

### Parameters:

tagId - the tag ID

---

## initialize

```
public void initialize()
```

For internal use

---

## recycle

```
public void recycle()
```

For internal use

---

## getContent

```
public String getContent()
```

Returns the value of the hidden tag

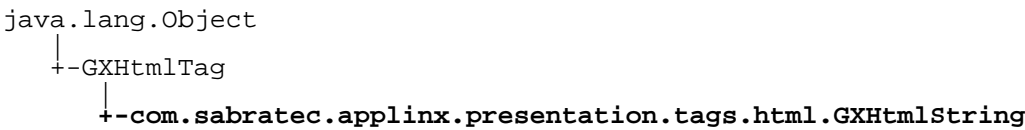
### Returns:

the value of the hidden tag

---

# com.sabratec.applinx.presentation.tags.html

## Class GXHtmlString



```
public class GXHtmlString
extends GXHtmlTag
```

This class is used for adding HTML free text. The text may also contain tags.

To create a new HTML string use `GXHtmlTagFactory.intance().newHtmlString(...)`

### Method Summary

String	<a href="#">getContent()</a>
void	<a href="#">recycle()</a> For internal use
void	<a href="#">renderInto</a> (StringBuffer result) For internal use
void	<a href="#">setTagName</a> (String tagName) Sets the concrete name of the tag
void	<a href="#">setTagText</a> (String htmlText) Sets the tag's text.

Methods inherited from class java.lang.Object
<code>equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait</code>

### Methods

#### setTagText

```
public void setTagText(String htmlText)

    Sets the tag's text. May contain any HTML characters, such as "<" & ">".

Parameters:
    htmlText - HTML text of the tag
```

#### recycle

```
public void recycle()

    For internal use
```

---

## renderInto

```
public void renderInto(StringBuffer result)
```

For internal use

### Parameters:

result

---

## getContent

```
public String getContent()
```

### Returns:

The HTML text of the tag

---

## setTagName

```
public void setTagName(String tagName)
```

Sets the concrete name of the tag

### Parameters:

tagName - The name of the tag

---

# com.sabratec.applinx.presentation.tags.html

## Class GXHtmlTagFactory

java.lang.Object

└─com.sabratec.applinx.presentation.tags.html.GXHtmlTagFactory

public class **GXHtmlTagFactory**  
extends Object

### Method Summary

<a href="#">static GXHtmlTagFactory</a>	<a href="#">instance()</a> Returns an instance of the HTML tag factory.
<a href="#">GXIButtonTag</a>	<a href="#">newButton()</a> Returns a new HTML button tag (<input type="button">)
<a href="#">GXICheckboxTag</a>	<a href="#">newCheckbox(String tagId)</a> Returns a new HTML check box tag.
<a href="#">GXIComboboxTag</a>	<a href="#">newCombobox(String tagId)</a> Returns a new HTML combo box tag (<select>) The tag ID also sets the name of the tag.
<a href="#">GXHtmlGenericTag</a>	<a href="#">newGenericTag(String tagName)</a> Returns a new HTML generic tag, which may be used for creating a tag.
<a href="#">GXHtmlHidden</a>	<a href="#">newHiddenTag(String tagId)</a> Returns a new HTML hidden tag (<input type="hidden">) The tag ID also sets the name of the tag.
<a href="#">GXHtmlString</a>	<a href="#">newHtmlString(String text)</a> Used for creating a free text HTML tag.
<a href="#">GXImageTag</a>	<a href="#">newImage()</a>
<a href="#">GXILabelTag</a>	<a href="#">newLabel(String text)</a> Returns a new HTML label tag (<span>)
<a href="#">GXILinkTag</a>	<a href="#">newLink(String text)</a> Returns a new HTML link tag (<a ...>).
<a href="#">GXIRadioButtonTag</a>	<a href="#">newRadioButton(String tagName, String tagCheckedValue)</a> Returns a new HTML radio button tag (<input type="radio">) The tag name is used for associating single radio buttons by specifying the same name.
<a href="#">GXIRadioButtonGroupTag</a>	<a href="#">newRadioButtonGroup(String tagId)</a> Returns a new HTML radio button group tag.
<a href="#">GXILabelTag</a>	<a href="#">newStaticLabel(String text)</a>
<a href="#">GXITableTag</a>	<a href="#">newTable()</a> Returns a new HTML table tag (<table>)

<a href="#">GXITableCellTag</a>	<a href="#">newTableCell()</a> Returns a new HTML table cell tag (<td>)
<a href="#">GXITableCellTag</a>	<a href="#">newTableHeader()</a>
<a href="#">GXITableRowTag</a>	<a href="#">newTableRow()</a> Returns a new HTML table row tag (<tr>)
<a href="#">GXITextFieldTag</a>	<a href="#">newTextField(String tagId)</a> Returns a new HTML text field tag (<input type="text">,<input type="password">).
void	<a href="#">recycleButton(GXIButtonTag button)</a> Used for recycling the tag.
void	<a href="#">recycleCheckbox(GXICheckboxTag checkbox)</a> Used for recycling the tag.
void	<a href="#">recycleCombobox(GXIComboboxTag combobox)</a> Used for recycling the tag.
void	<a href="#">recycleGenericTag(GXHtmlGenericTag tag)</a> Used for recycling the tag.
void	<a href="#">recycleHidden(GXHtmlHidden hidden)</a> Used for recycling the tag.
void	<a href="#">recycleHtmlString(GXHtmlString htmlStringTag)</a> Used for recycling the tag.
void	<a href="#">recycleImage(GXImageTag image)</a>
void	<a href="#">recycleLabel(GXILabelTag label)</a> Used for recycling the tag.
void	<a href="#">recycleLink(GXILinkTag link)</a> Used for recycling the tag.
void	<a href="#">recycleRadioButton(GXIRadioButtonTag radio)</a> Used for recycling the tag.
void	<a href="#">recycleRadioButtonGroup(GXIRadioButtonGroupTag radioGroup)</a> Used for recycling the tag.
void	<a href="#">recycleTable(GXITableTag table)</a> Used for recycling the tag.
void	<a href="#">recycleTableCell(GXITableCellTag cell)</a> Used for recycling the tag.
void	<a href="#">recycleTableRow(GXITableRowTag row)</a> Used for recycling the tag.
void	<a href="#">recycleTextField(GXITextFieldTag textField)</a> Used for recycling the tag.

**Methods inherited from class** java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Methods

### instance

```
public static GXHtmlTagFactory instance()
```

Returns an instance of the HTML tag factory. This class is a singleton and using this method is the only way to use this class.

**Syntax:**

```
GXHtmlTagFactory.instance().new<LT;TAG NAME>...
```

**Returns:**

an instance of the HTML tag factory.

---

### newLabel

```
public GXILabelTag newLabel(String text)
```

Returns a new HTML label tag (<span>)

**Parameters:**

text - The inner text of the label

**Returns:**

a new HTML label tag(<span>)

---

### newStaticLabel

```
public GXILabelTag newStaticLabel(String text)
```

### newTextField

```
public GXITextFieldTag newTextField(String tagId)
```

Returns a new HTML text field tag (<input type="text">,<input type="password">).

The tag ID also sets the name of the tag. It is required for returning the content to the server when the page is submitted.

To set as a password text field, use the `GXITextFieldTag.setPassword(true)` method.

**Parameters:**

tagId - The text field ID.

**Returns:**

a new HTML text field tag (<input type="text">,<input type="password">)

---

### newTable

```
public GXITableTag newTable()
```

Returns a new HTML table tag (<table>)

**Returns:**

a new HTML table tag (<table>)

---

## newTableRow

```
public GXITableRowTag newTableRow( )
```

Returns a new HTML table row tag (<tr>)

**Returns:**

a new HTML table row tag (<tr>)

---

## newTableCell

```
public GXITableCellTag newTableCell( )
```

Returns a new HTML table cell tag (<td>)

**Returns:**

a new HTML table cell tag (<td>)

---

## newButton

```
public GXIButtonTag newButton( )
```

Returns a new HTML button tag (<input type="button">)

**Returns:**

a new HTML button tag (<input type="button">)

---

## newLink

```
public GXILinkTag newLink(String text)
```

Returns a new HTML link tag (<a ...>).

**Parameters:**

text - The inner text of the link

**Returns:**

a new HTML link tag (<a ...>)

---

## newCombobox

```
public GXIComboboxTag newCombobox(String tagId)
```

Returns a new HTML combo box tag (<select>). The tag ID also sets the name of the tag. It is required for returning the content to the server when the page is submitted.

**Parameters:**

tagId - The tag ID

**Returns:**

a new HTML combo box tag(<select>)

---

## newRadioButtonGroup

```
public GXIRadioButtonGroupTag newRadioButtonGroup(String tagId)
```

(continued from last page)

Returns a new HTML radio button group tag. It is rendered as a table with radio buttons and captions.  
The tag ID sets the name of the radio button tags. It is required for returning the content to the server when the page is submitted.

**Parameters:**

tagId - The radio button group tag ID

**Returns:**

a new HTML label tag (<span>)

---

## newRadioButton

```
public GXIRadioButtonTag newRadioButton(String tagName,  
String tagCheckedValue)
```

Returns a new HTML radio button tag (<input type="radio">) The tag name is used for associating single radio buttons by specifying the same name.

**Parameters:**

tagName - The tag name

tagCheckedValue - The value of the tag when it is selected

**Returns:**

a new HTML radio button tag (<input type="radio">)

---

## newGenericTag

```
public GXHtmlGenericTag newGenericTag(String tagName)
```

Returns a new HTML generic tag, which may be used for creating a tag.

**Parameters:**

tagName - The specified tag name

**Returns:**

a new HTML generic tag

---

## newHiddenTag

```
public GXHtmlHidden newHiddenTag(String tagId)
```

Returns a new HTML hidden tag (<input type="hidden">) The tag ID also sets the name of the tag. It is required for returning the content to the server when the page is submitted.

**Parameters:**

tagId - The tag ID

**Returns:**

a new HTML hidden tag (<input type="hidden">)

---

## newCheckbox

```
public GXICheckboxTag newCheckbox(String tagId)
```

Returns a new HTML check box tag.

**Parameters:**

tagId - The tag ID

**Returns:**

(continued from last page)

a new HTML check box tag

---

## newImage

```
public GXImageTag newImage()
```

---

## newHtmlString

```
public GXHtmlString newHtmlString(String text)
```

Used for creating a free text HTML tag.

**Parameters:**

text - the HTML text

**Returns:**

an HTML string tag

---

## recycleLabel

```
public void recycleLabel(GXILabelTag label)
```

Used for recycling the tag.

**Parameters:**

label - the tag to recycle

---

## recycleTextField

```
public void recycleTextField(GXITextFieldTag textField)
```

Used for recycling the tag.

**Parameters:**

textField - the tag to recycle

---

## recycleTable

```
public void recycleTable(GXITableTag table)
```

Used for recycling the tag.

**Parameters:**

table - the tag to recycle

---

## recycleTableRow

```
public void recycleTableRow(GXITableRowTag row)
```

Used for recycling the tag.

**Parameters:**

row - the tag to recycle

(continued from last page)

---

## recycleTableCell

```
public void recycleTableCell(GXITableCellTag cell)
```

Used for recycling the tag.

**Parameters:**

cell - the tag to recycle

---

## recycleButton

```
public void recycleButton(GXIButtonTag button)
```

Used for recycling the tag.

**Parameters:**

button - the tag to recycle

---

## recycleLink

```
public void recycleLink(GXILinkTag link)
```

Used for recycling the tag.

**Parameters:**

link - the tag to recycle

---

## recycleCombobox

```
public void recycleCombobox(GXIComboboxTag combobox)
```

Used for recycling the tag.

**Parameters:**

combobox - the tag to recycle

---

## recycleRadioButtonGroup

```
public void recycleRadioButtonGroup(GXIRadioButtonGroupTag radioGroup)
```

Used for recycling the tag.

**Parameters:**

radioGroup - the tag to recycle

---

## recycleRadioButton

```
public void recycleRadioButton(GXIRadioButtonTag radio)
```

Used for recycling the tag.

**Parameters:**

radio - the tag to recycle

---

## recycleGenericTag

```
public void recycleGenericTag(GXHtmlGenericTag tag)
```

---

(continued from last page)

Used for recycling the tag.

**Parameters:**

tag - the tag to recycle

---

## recycleHidden

```
public void recycleHidden(GXHtmlHidden hidden)
```

Used for recycling the tag.

**Parameters:**

hidden - the tag to recycle

---

## recycleCheckbox

```
public void recycleCheckbox(GXICheckboxTag checkbox)
```

Used for recycling the tag.

**Parameters:**

checkbox - the tag to recycle

---

## recycleImage

```
public void recycleImage(GXImageTag image)
```

---

## recycleHtmlString

```
public void recycleHtmlString(GXHtmlString htmlStringTag)
```

Used for recycling the tag.

**Parameters:**

htmlStringTag - the tag to recycle

---

## newTableHeader

```
public GXTableCellTag newTableHeader( )
```

---

## Package

# **com.sabratec.applinx.presentation.transf orms**

This package contains the ApplinX built-in transformation configurations. The built-in transformations are: Host keys, Tables, GUI elements, Window frames and emulation behavior.

## com.sabratec.applinx.presentation.transforms

### Class GXCisCssTransform

java.lang.Object

+ [com.sabratec.applinx.presentation.event.GXTagListener](#)  
 + [com.sabratec.applinx.presentation.transforms.GXCisCssTransform](#)

public class **GXCisCssTransform**  
 extends [GXTagListener](#)

This class is registered from sap portal jsp application

#### Constructor Summary

public	<a href="#">GXCisCssTransform()</a>
--------	-------------------------------------

#### Method Summary

void	<a href="#">onNewButton</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIButtonTag</a> button)
void	<a href="#">onNewCheckbox</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXICheckboxTag</a> checkbox)
void	<a href="#">onNewCombobox</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIComboboxTag</a> combobox)
void	<a href="#">onNewLabel</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILabelTag</a> label)
void	<a href="#">onNewLink</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILinkTag</a> link)
void	<a href="#">onNewRadioButtonGroup</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIRadioButtonGroupTag</a> radioButtonGroup)
void	<a href="#">onNewTable</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableTag</a> table)
void	<a href="#">onNewTableCell</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableCellTag</a> tableCell)
void	<a href="#">onNewTableRow</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableRowTag</a> tableRow)
void	<a href="#">onNewTextField</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITextFieldTag</a> textField)

#### Methods inherited from class [com.sabratec.applinx.presentation.event.GXTagListener](#)

[onNewButton](#), [onNewCheckbox](#), [onNewCombobox](#), [onNewLabel](#), [onNewLink](#),  
[onNewRadioButtonGroup](#), [onNewTable](#), [onNewTableCell](#), [onNewTableRow](#), [onNewTextField](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXCisCssTransform

```
public GXCisCssTransform()
```

## Methods

### onNewLabel

```
public void onNewLabel(GXRenderEvent event,  
    GXILabelTag label)
```

Invoked when a label (protected field) has been created. The common usage is for protected fields that are rendered as text fields.

---

### onNewTextField

```
public void onNewTextField(GXRenderEvent event,  
    GXITextFieldTag textField)
```

Invoked when a text field has been created. The common usage is for unprotected fields that are rendered as text fields.

---

### onNewButton

```
public void onNewButton(GXRenderEvent event,  
    GXIButtonTag button)
```

Invoked when a link has been created. The common usage is for host-keys that may be rendered as buttons.

---

### onNewLink

```
public void onNewLink(GXRenderEvent event,  
    GXILinkTag link)
```

Invoked when a link has been created. The common usage is for host-keys that may be rendered as links.

---

### onNewCheckbox

```
public void onNewCheckbox(GXRenderEvent event,  
    GXICheckboxTag checkbox)
```

Invoked when a checkbox has been created. The common usage is for a GUI-element that may be rendered as a check box.

---

### onNewCombobox

```
public void onNewCombobox(GXRenderEvent event,  
    GXIComboboxTag combobox)
```

Invoked when a combo box has been created. The common usage is for a GUI-element that may be rendered as a combo box.

---

(continued from last page)

## onNewRadioButtonGroup

```
public void onNewRadioButtonGroup(GXRenderEvent event,  
    GXIRadioButtonGroupTag radioButton)
```

Invoked when a radio buttons group has been created. The common usage is for a GUI-element that may be rendered as a radio button's group.

---

## onNewTable

```
public void onNewTable(GXRenderEvent event,  
    GXITableTag table)
```

Invoked when a table tag has been created.

In this case, `table.getHostField()` returns null.

---

## onNewTableRow

```
public void onNewTableRow(GXRenderEvent event,  
    GXITableRowTag tableRow)
```

Invoked when a table row tag has been created.

In this case, `table.getHostField()` returns null.

---

## onNewTableCell

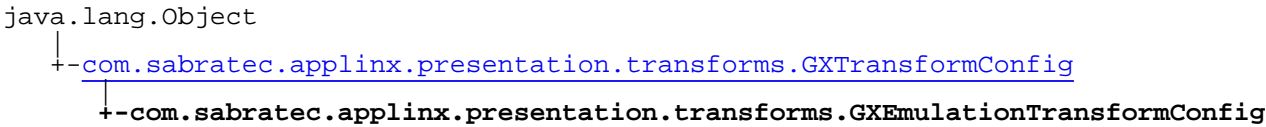
```
public void onNewTableCell(GXRenderEvent event,  
    GXITableCellTag tableCell)
```

Invoked when a table cell has been created.

Unlike the other table-related event objects, `tableCell.getHostField()` returns the host field the cell is based on.

# com.sabratec.applinx.presentation.transforms

## Class GXEmulationTransformConfig



All Implemented Interfaces:  
java.io.Serializable, java.io.Serializable

```
public class GXEmulationTransformConfig
extends GXTransformConfig
implements java.io.Serializable, java.io.Serializable
```

This class is the transformation configuration of various emulation features. It determines the behavior and UI of a tag which may be derived from its host field.  
Among the features:

- Field colors: all, background only, none
- Emulation attributes: data type, automatic skip, right adjust, automatic enter
- Intensified

See Also:  
[GXRenderConfig.getEmulationConfig\(\)](#)

Constructor Summary	
public	<a href="#">GXEmulationTransformConfig()</a>

Method Summary	
boolean	<a href="#">getClientBlinking()</a> For .NET framework usage.
int	<a href="#">getColorMode()</a> For .NET framework usage.
boolean	<a href="#">getRenderAppFieldCss()</a> For .NET framework usage.
boolean	<a href="#">getRenderEmulationAttributes()</a> For .NET framework usage.
boolean	<a href="#">getRenderIntensifiedCss()</a> For .NET framework usage.
int	<a href="#">getColorMode()</a> Determines whether to display all the colors as they are displayed in the host (Background & foreground), display only the background colors or not display any colors at all.
boolean	<a href="#">isClientBlinking()</a> client configuration allows blinking

boolean	<a href="#"><u>isRenderAppFieldCss()</u></a> Determines whether to render a css class for application fields
boolean	<a href="#"><u>isRenderEmulationAttributes()</u></a> Returns if emulation attributes are rendered.
boolean	<a href="#"><u>isRenderIntensifiedCss()</u></a> Determines whether to render an intensified css class for intensified host fields.
void	<a href="#"><u>set_ClientBlinking</u></a> (boolean isClientBlinking) For .NET framework usage.
void	<a href="#"><u>set_ColorMode</u></a> (int colorMode) For .NET framework usage.
void	<a href="#"><u>set_RenderAppFieldCss</u></a> (boolean isRenderAppFieldCss) For .NET framework usage.
void	<a href="#"><u>set_RenderEmulationAttributes</u></a> (boolean isRenderEmulationAttributes) For .NET framework usage.
void	<a href="#"><u>set_RenderIntensifiedCss</u></a> (boolean isRenderIntensifiedCss) For .NET framework usage.
void	<a href="#"><u>setClientBlinking</u></a> (boolean isClientBlinking) client configuration allows blinking
void	<a href="#"><u>setColorMode</u></a> (int colorMode) Determines whether to display all the colors as they are displayed in the host (Background & foreground), display only the background colors or not display any colors at all.
void	<a href="#"><u>setRenderAppFieldCss</u></a> (boolean isRenderAppFieldCss) Sets if to render the application field css class.
void	<a href="#"><u>setRenderEmulationAttributes</u></a> (boolean isRenderEmulationAttributes) Determines whether to render emulation attributes in text fields.
void	<a href="#"><u>setRenderIntensifiedCss</u></a> (boolean isRenderIntensifiedCss) Sets if to render intensified css class for intensified host fields.

Methods inherited from class [com.sabratec.applinx.presentation.transforms.GXTransformConfig](#)

[get\\_Enabled](#), [isEnabled](#), [set\\_Enabled](#), [setEnabled](#)

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXEmulationTransformConfig

```
public GXEmulationTransformConfig()
```

## Methods

(continued from last page)

---

## getColorMode

```
public int getColorMode()
```

Determines whether to display all the colors as they are displayed in the host (Background & foreground), display only the background colors or not display any colors at all.

**Returns:**

Returns the color mode.

**See Also:**

[GXTransformConstants.COLORS\\_ALL](#)

[GXTransformConstants.COLORS\\_NONE](#)

[GXTransformConstants.COLORS\\_REVERSED\\_VIDEO](#)

---

## setColorMode

```
public void setColorMode(int colorMode)
```

Determines whether to display all the colors as they are displayed in the host (Background & foreground), display only the background colors or not display any colors at all.

**Parameters:**

colorMode - The color mode to set.

**See Also:**

[GXTransformConstants.COLORS\\_ALL](#)

[GXTransformConstants.COLORS\\_NONE](#)

[GXTransformConstants.COLORS\\_REVERSED\\_VIDEO](#)

---

## isRenderEmulationAttributes

```
public boolean isRenderEmulationAttributes()
```

Returns if emulation attributes are rendered.

**Returns:**

if emulation attributes are rendered

**See Also:**

[setRenderEmulationAttributes\(boolean\)](#)

---

## setRenderEmulationAttributes

```
public void setRenderEmulationAttributes(boolean isRenderEmulationAttributes)
```

Determines whether to render emulation attributes in text fields. Emulation attributes are special ApplinX attributes that start with "gx\_" and are added to the rendered text field tag. These attributes provide additional information to the ApplinX JavaScript engine, which changes the behavior of the text fields.  
The attributes are:

- Automatic skipping (gx\_autoSkip) - will cause the text field to skip to the neighboring text field, once it is filled with content.
  - Data type (gx\_dt) - The data type of the field. See [GXBaseObjectConstants](#) for possible data types.
  - Right adjustment (gx\_ra) - AS/400 right adjust
  - Automatic ENTER (gx\_au) - AS/400 automatic enter
-

(continued from last page)

**Parameters:**

`isRenderEmulationAttributes` - whether to add emulation attributes in text fields

---

## isRenderAppFieldCss

```
public boolean isRenderAppFieldCss()
```

Determines whether to render a css class for application fields

**Returns:**

if an application field css class is rendered

---

## setRenderAppFieldCss

```
public void setRenderAppFieldCss(boolean isRenderAppFieldCss)
```

Sets if to render the application field css class.

**Parameters:**

`isRenderAppFieldCss` - whether to render an application field's css class

---

## isRenderIntensifiedCss

```
public boolean isRenderIntensifiedCss()
```

Determines whether to render an intensified css class for intensified host fields.

**Returns:**

if an intensified css class is rendered

---

## setRenderIntensifiedCss

```
public void setRenderIntensifiedCss(boolean isRenderIntensifiedCss)
```

Sets if to render intensified css class for intensified host fields.

**Parameters:**

`isRenderIntensifiedCss` - whether to render an intensified css class

---

## isClientBlinking

```
public boolean isClientBlinking()
```

client configuration allows blinking

**Returns:**

---

## setClientBlinking

```
public void setClientBlinking(boolean isClientBlinking)
```

client configuration allows blinking

**Parameters:**

`isClientBlinking`

(continued from last page)

---

## get\_RenderEmulationAttributes

```
public boolean get_RenderEmulationAttributes()
```

For .NET framework usage. Expose the method as a property.

---

## get\_RenderAppFieldCss

```
public boolean get_RenderAppFieldCss()
```

For .NET framework usage. Expose the method as a property.

---

## get\_RenderIntensifiedCss

```
public boolean get_RenderIntensifiedCss()
```

For .NET framework usage. Expose the method as a property.

---

## get\_ColorMode

```
public int get_ColorMode()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ColorMode

```
public void set_ColorMode(int colorMode)
```

For .NET framework usage. Expose the method as a property.

---

## set\_RenderEmulationAttributes

```
public void set_RenderEmulationAttributes(boolean isRenderEmulationAttributes)
```

For .NET framework usage. Expose the method as a property.

---

## set\_RenderAppFieldCss

```
public void set_RenderAppFieldCss(boolean isRenderAppFieldCss)
```

For .NET framework usage. Expose the method as a property.

---

## set\_RenderIntensifiedCss

```
public void set_RenderIntensifiedCss(boolean isRenderIntensifiedCss)
```

For .NET framework usage. Expose the method as a property.

---

## get\_ClientBlinking

```
public boolean get_ClientBlinking()
```

For .NET framework usage. Expose the method as a property.

---

## set\_ClientBlinking

```
public void set_ClientBlinking(boolean isClientBlinking)
```

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.presentation.transforms Class GXGuiElementsTransformConfig

java.lang.Object

```

+--com.sabratec.applinx.presentation.transforms.GXTransformConfig
    |
    +--com.sabratec.applinx.presentation.transforms.GXGuiElementsTransformConfig
  
```

### All Implemented Interfaces:

java.io.Serializable, java.io.Serializable

**Deprecated.** *Gui elements are translated into transformations, and should be controls from ApplinX composer. This class is the transformation configuration for GUI elements as defined in ApplinX composer. It determines the css class for each GUI element and the alignment of radio button groups.*

public class **GXGuiElementsTransformConfig**  
 extends [GXTransformConfig](#)  
 implements java.io.Serializable, java.io.Serializable

### See Also:

[GXRenderConfig.getGuiElementsConfig\(\)](#)

## Constructor Summary

public	<a href="#">GXGuiElementsTransformConfig()</a> <b>Deprecated.</b>
--------	--

## Method Summary

boolean	<a href="#">get_AdjustNeighbours()</a> <b>Deprecated.</b>
int	<a href="#">get_Alignment()</a> <b>Deprecated.</b>
String	<a href="#">get_CheckboxCss()</a> <b>Deprecated.</b>
String	<a href="#">get_ComboboxCss()</a> <b>Deprecated.</b>
String	<a href="#">get_RadioButtonCss()</a> <b>Deprecated.</b>
int	<a href="#">getAlignment()</a> <b>Deprecated.</b>
String	<a href="#">getCheckboxCss()</a> <b>Deprecated.</b>
String	<a href="#">getComboboxCss()</a> <b>Deprecated.</b>

String	<a href="#"><u>getRadioButtonCss()</u></a> <b>Deprecated.</b>
boolean	<a href="#"><u>isAdjustNeighbours()</u></a> <b>Deprecated.</b>
void	<a href="#"><u>set_AdjustNeighbours</u></a> (boolean isAdjustNeighbours) <b>Deprecated.</b>
void	<a href="#"><u>set_Alignment</u></a> (int alignment) <b>Deprecated.</b>
void	<a href="#"><u>set_CheckboxCss</u></a> (String checkboxCss) <b>Deprecated.</b>
void	<a href="#"><u>set_ComboboxCss</u></a> (String comboboxCss) <b>Deprecated.</b>
void	<a href="#"><u>set_RadioButtonCss</u></a> (String radioButtonCss) <b>Deprecated.</b>
void	<a href="#"><u>setAdjustNeighbours</u></a> (boolean isAdjustNeighbours) <b>Deprecated.</b>
void	<a href="#"><u>setAlignment</u></a> (int alignment) <b>Deprecated.</b>
void	<a href="#"><u>setCheckboxCss</u></a> (String checkboxCss) <b>Deprecated.</b>
void	<a href="#"><u>setComboboxCss</u></a> (String comboboxCss) <b>Deprecated.</b>
void	<a href="#"><u>setRadioButtonCss</u></a> (String radioButtonCss) <b>Deprecated.</b>

Methods inherited from class [com.sabratec.applinx.presentation.transforms.GXTransformConfig](#)

[get\\_Enabled](#), [isEnabled](#), [set\\_Enabled](#), [setEnabled](#)

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXGuiElementsTransformConfig

```
public GXGuiElementsTransformConfig()
```

**Deprecated.**

## Methods

(continued from last page)

## getAlignment

```
public int getAlignment()
```

**Deprecated.**

**Returns:**

Returns the alignment.

**See Also:**

[GPresentationConstants.ALIGNMENT\\_HORIZONTAL](#)

[GPresentationConstants.ALIGNMENT\\_VERTICAL](#)

---

## setAlignment

```
public void setAlignment(int alignment)
```

**Deprecated.**

**Parameters:**

alignment - The alignment to set.

**See Also:**

[GPresentationConstants.ALIGNMENT\\_HORIZONTAL](#)

[GPresentationConstants.ALIGNMENT\\_VERTICAL](#)

---

## isAdjustNeighbours

```
public boolean isAdjustNeighbours()
```

**Deprecated.**

**Returns:**

whether the neighboring tags will be adjusted

**See Also:**

[setAdjustNeighbours\(boolean\)](#)

---

## setAdjustNeighbours

```
public void setAdjustNeighbours(boolean isAdjustNeighbours)
```

**Deprecated.**

Determines whether to adjust the neighboring tags when replacing a text field with a GUI element. The replacement is done according to the GUI pixel offset from the original text field. `true` by default.

**Parameters:**

isAdjustNeighbours - whether to adjust the neighboring tags

(continued from last page)

## getCheckboxCss

```
public String getCheckboxCss()
```

### Deprecated.

The css class for a checkbox.

### Returns:

the css class for a check box

---

## setCheckboxCss

```
public void setCheckboxCss(String checkboxCss)
```

### Deprecated.

The css class for a checkbox.

### Parameters:

checkboxCss - the css class for a check box

---

## getComboboxCss

```
public String getComboboxCss()
```

### Deprecated.

The css class for a combo box.

### Returns:

the css class for a combo box

---

## setComboboxCss

```
public void setComboboxCss(String comboboxCss)
```

### Deprecated.

The css class for a combo box.

### Parameters:

comboboxCss - the css class for a combo box

---

## getRadioButtonCss

```
public String getRadioButtonCss()
```

### Deprecated.

The css class for a radio button.

### Returns:

the css class for radio button groups

---

## setRadioButtonCss

```
public void setRadioButtonCss(String radioButtonCss)
```

(continued from last page)

**Deprecated.**

The css class for a radio button.

**Parameters:**

radioButtonCss - the css class for radio button groups

---

**get\_AdjustNeighbours**

```
public boolean get_AdjustNeighbours()
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

**set\_AdjustNeighbours**

```
public void set_AdjustNeighbours(boolean isAdjustNeighbours)
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

**get\_Alignment**

```
public int get_Alignment()
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

**set\_Alignment**

```
public void set_Alignment(int alignment)
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

**get\_CheckboxCss**

```
public String get_CheckboxCss()
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

**set\_CheckboxCss**

```
public void set_CheckboxCss(String checkboxCss)
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

**get\_ComboboxCss**

```
public String get_ComboboxCss()
```

**Deprecated.**

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **set\_ComboboxCss**

```
public void set_ComboboxCss(String comboboxCss)
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

## **get\_RadioButtonCss**

```
public String get_RadioButtonCss()
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

## **set\_RadioButtonCss**

```
public void set_RadioButtonCss(String radioButtonCss)
```

**Deprecated.**

For .NET framework usage. Expose the method as a property.

---

## com.sabratec.applinx.presentation.transforms Class GXHebrewTransform

java.lang.Object

└-com.sabratec.applinx.presentation.transforms.GXHebrewTransform

All Implemented Interfaces:

[GXCompletionListener](#)

public class **GXHebrewTransform**  
extends Object  
implements [GXCompletionListener](#)

### Constructor Summary

public	<a href="#">GXHebrewTransform()</a>
--------	-------------------------------------

### Method Summary

void	<a href="#">onComplete</a> ( <a href="#">GXRenderEvent</a> event)
static void	<a href="#">splitLabel</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILabelTag</a> label)

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface [com.sabratec.applinx.presentation.event.GXCompletionListener](#)

[onComplete](#)

### Constructors

#### GXHebrewTransform

public **GXHebrewTransform()**

### Methods

#### splitLabel

public static void **splitLabel**([GXRenderEvent](#) event,  
[GXILabelTag](#) label)

(continued from last page)

## **onComplete**

```
public void onComplete(GXRenderEvent event)
```

## com.sabratec.applinx.presentation.transforms

### Class GXHostKeysTransformConfig

java.lang.Object

```

+--com.sabratec.applinx.presentation.transforms.GXTransformConfig
    +--com.sabratec.applinx.presentation.transforms.GXHostKeysTransformConfig
  
```

#### All Implemented Interfaces:

java.io.Serializable, java.io.Serializable

public class **GXHostKeysTransformConfig**  
 extends [GXTransformConfig](#)  
 implements java.io.Serializable, java.io.Serializable

This class is the transformation configuration for host keys as defined in the ApplinX application. It determines the rendering type of the host keys (links/buttons), and the css class of the host keys.

#### See Also:

[GXRenderConfig.getHostKeysConfig\(\)](#)

### Constructor Summary

public	<a href="#">GXHostKeysTransformConfig()</a>
--------	---

### Method Summary

String	<a href="#">get_HostKeyCss()</a> For .NET framework usage.
int	<a href="#">get_RenderingType()</a> For .NET framework usage.
String	<a href="#">getHostKeyCss()</a>
int	<a href="#">getRenderingType()</a> The type of the rendered host keys.
void	<a href="#">set_HostKeyCss(String hostKeyCss)</a> For .NET framework usage.
void	<a href="#">set_RenderingType(int renderingType)</a> For .NET framework usage.
void	<a href="#">setHostKeyCss(String hostKeyCss)</a>
void	<a href="#">setRenderingType(int renderingType)</a> The type of the rendered host keys.

Methods inherited from class [com.sabratec.applinx.presentation.transforms.GXTransformConfig](#)

[get\\_Enabled](#), [isEnabled](#), [set\\_Enabled](#), [setEnabled](#)

Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructors

### **GXHostKeysTransformConfig**

```
public GXHostKeysTransformConfig()
```

## Methods

### **setRenderingType**

```
public void setRenderingType(int renderingType)
```

The type of the rendered host keys.

**Parameters:**

renderingType - constant representing the type of the host keys.

**See Also:**

[GXTransformConstants.HOSTKEYS\\_BUTTONS](#)

[GXTransformConstants.HOSTKEYS\\_LINKS](#)

[GXTransformConstants.HOSTKEYS\\_DELETE](#)

### **getRenderingType**

```
public int getRenderingType()
```

The type of the rendered host keys.

**Returns:**

A constant representing the type of the host keys. Listens only to the onComplete event.

**See Also:**

[GXTransformConstants.HOSTKEYS\\_BUTTONS](#)

[GXTransformConstants.HOSTKEYS\\_LINKS](#)

[GXTransformConstants.HOSTKEYS\\_DELETE](#)

### **set\_RenderingType**

```
public void set_RenderingType(int renderingType)
```

For .NET framework usage. Expose the method as a property.

### **get\_RenderingType**

```
public int get_RenderingType()
```

For .NET framework usage. Expose the method as a property.

### **getHostKeyCss**

```
public String getHostKeyCss()
```

---

(continued from last page)

**Returns:**

the CSS class of the host keys

---

**setHostKeyCss**

```
public void setHostKeyCss(String hostKeyCss)
```

**Parameters:**

hostKeyCss - the CSS class of the host keys

---

**get\_HostKeyCss**

```
public String get_HostKeyCss()
```

For .NET framework usage. Expose the method as a property.

---

**set\_HostKeyCss**

```
public void set_HostKeyCss(String hostKeyCss)
```

For .NET framework usage. Expose the method as a property.

---

## com.sabratec.applinx.presentation.transforms

### Class GXSapPortalTransform

java.lang.Object

+ [com.sabratec.applinx.presentation.event.GXTagListener](#)  
 + [com.sabratec.applinx.presentation.transforms.GXSapPortalTransform](#)

public class **GXSapPortalTransform**  
 extends [GXTagListener](#)

This class is registered from sap portal jsp application

#### Constructor Summary

public	<a href="#">GXSapPortalTransform()</a>
--------	--

#### Method Summary

void	<a href="#">onNewButton</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIButtonTag</a> button)
void	<a href="#">onNewCheckbox</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXICheckboxTag</a> checkbox)
void	<a href="#">onNewCombobox</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIComboboxTag</a> combobox)
void	<a href="#">onNewLabel</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILabelTag</a> label)
void	<a href="#">onNewLink</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXILinkTag</a> link)
void	<a href="#">onNewRadioButtonGroup</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXIRadioButtonGroupTag</a> radioButtonGroup)
void	<a href="#">onNewTable</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableTag</a> table)
void	<a href="#">onNewTableCell</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableCellTag</a> tableCell)
void	<a href="#">onNewTableRow</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITableRowTag</a> tableRow)
void	<a href="#">onNewTextField</a> ( <a href="#">GXRenderEvent</a> event, <a href="#">GXITextFieldTag</a> textField)

#### Methods inherited from class [com.sabratec.applinx.presentation.event.GXTagListener](#)

[onNewButton](#), [onNewCheckbox](#), [onNewCombobox](#), [onNewLabel](#), [onNewLink](#),  
[onNewRadioButtonGroup](#), [onNewTable](#), [onNewTableCell](#), [onNewTableRow](#), [onNewTextField](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXSapPortalTransform

```
public GXSapPortalTransform()
```

## Methods

### onNewLabel

```
public void onNewLabel(GXRenderEvent event,  
    GXILabelTag label)
```

Invoked when a label (protected field) has been created. The common usage is for protected fields that are rendered as text fields.

---

### onNewTextField

```
public void onNewTextField(GXRenderEvent event,  
    GXITextFieldTag textField)
```

Invoked when a text field has been created. The common usage is for unprotected fields that are rendered as text fields.

---

### onNewButton

```
public void onNewButton(GXRenderEvent event,  
    GXIButtonTag button)
```

Invoked when a link has been created. The common usage is for host-keys that may be rendered as buttons.

---

### onNewLink

```
public void onNewLink(GXRenderEvent event,  
    GXILinkTag link)
```

Invoked when a link has been created. The common usage is for host-keys that may be rendered as links.

---

### onNewCheckbox

```
public void onNewCheckbox(GXRenderEvent event,  
    GXICheckboxTag checkbox)
```

Invoked when a checkbox has been created. The common usage is for a GUI-element that may be rendered as a check box.

---

### onNewCombobox

```
public void onNewCombobox(GXRenderEvent event,  
    GXIComboboxTag combobox)
```

Invoked when a combo box has been created. The common usage is for a GUI-element that may be rendered as a combo box.

---

(continued from last page)

## onNewRadioButtonGroup

```
public void onNewRadioButtonGroup(GXRenderEvent event,  
    GXIRadioButtonGroupTag radioButton)
```

Invoked when a radio buttons group has been created. The common usage is for a GUI-element that may be rendered as a radio button's group.

---

## onNewTable

```
public void onNewTable(GXRenderEvent event,  
    GXITableTag table)
```

Invoked when a table tag has been created.

In this case, `table.getHostField()` returns null.

---

## onNewTableRow

```
public void onNewTableRow(GXRenderEvent event,  
    GXITableRowTag tableRow)
```

Invoked when a table row tag has been created.

In this case, `table.getHostField()` returns null.

---

## onNewTableCell

```
public void onNewTableCell(GXRenderEvent event,  
    GXITableCellTag tableCell)
```

Invoked when a table cell has been created.

Unlike the other table-related event objects, `tableCell.getHostField()` returns the host field the cell is based on.

## com.sabratec.applinx.presentation.transforms Class GXTableTransformConfig

java.lang.Object

```

  |
  +-- com.sabratec.applinx.presentation.transforms.GXTransformConfig
      |
      +-- com.sabratec.applinx.presentation.transforms.GXTableTransformConfig
  
```

### All Implemented Interfaces:

java.io.Serializable, java.io.Serializable

public class **GXTableTransformConfig**  
 extends [GXTransformConfig](#)  
 implements java.io.Serializable, java.io.Serializable

This class is the transformation configuration for host tables as defined in ApplinX composer. It determines whether to adjust the tags beneath the table tag since a table tag is "taller" then standard label & text field tags. true by default. In additional it determines the various css classes of the tables.

### See Also:

[GXRenderConfig.getTableConfig\(\)](#)

## Constructor Summary

public	<a href="#">GXTableTransformConfig()</a>
--------	--

## Method Summary

boolean	<a href="#">get_AdjustNeighbours()</a> For .NET framework usage.
boolean	<a href="#">get_RenderAlternatingCssClass()</a> For .NET framework usage.
String	<a href="#">get_TableAlternating1CssClass()</a> For .NET framework usage.
String	<a href="#">get_TableAlternating2CssClass()</a> For .NET framework usage.
String	<a href="#">get_TableCssClass()</a> For .NET framework usage.
boolean	<a href="#">get_WindowRelative()</a> For .NET framework usage.
String	<a href="#">getTableAlternating1CssClass()</a>
String	<a href="#">getTableAlternating2CssClass()</a>
String	<a href="#">getTableCssClass()</a>

boolean	<a href="#"><u>isAdjustNeighbours</u></a> ( ) Determines whether to adjust the tags below the table tag, as the table tag is taller than the default tags.
boolean	<a href="#"><u>isRenderAlternatingCssClass</u></a> ( )
boolean	<a href="#"><u>isWindowRelative</u></a> ( ) For internal usage
void	<a href="#"><u>set_AdjustNeighbours</u></a> (boolean isAdjustNeighbours) For .NET framework usage.
void	<a href="#"><u>set_RenderAlternatingCssClass</u></a> (boolean isRenderAlternatingCssClass) For .NET framework usage.
void	<a href="#"><u>set_TableAlternating1CssClass</u></a> (String tableAlternating1CssClass) For .NET framework usage.
void	<a href="#"><u>set_TableAlternating2CssClass</u></a> (String tableAlternating2CssClass) For .NET framework usage.
void	<a href="#"><u>set_TableCssClass</u></a> (String tableCssClass) For .NET framework usage.
void	<a href="#"><u>set_WindowRelative</u></a> (boolean isWindowRelative) For .NET framework usage.
void	<a href="#"><u>setAdjustNeighbours</u></a> (boolean isAdjustNeighbours) Determines whether to adjust the tags below the table tag, as the table tag is taller than the default tags.
void	<a href="#"><u>setRenderAlternatingCssClass</u></a> (boolean isRenderAlternatingCssClass)
void	<a href="#"><u>setTableAlternating1CssClass</u></a> (String tableAlternating1CssClass)
void	<a href="#"><u>setTableAlternating2CssClass</u></a> (String tableAlternating2CssClass)
void	<a href="#"><u>setTableCssClass</u></a> (String tableCssClass)
void	<a href="#"><u>setWindowRelative</u></a> (boolean isWindowRelative) For internal usage

Methods inherited from class [com.sabratec.applinx.presentation.transforms.GXTransformConfig](#)

[get\\_Enabled](#), [isEnabled](#), [set\\_Enabled](#), [setEnabled](#)

Methods inherited from class [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

(continued from last page)

## GXTableTransformConfig

```
public GXTableTransformConfig()
```

## Methods

### getTableAlternating1CssClass

```
public String getTableAlternating1CssClass()
```

**Returns:**

the css class of the odd rows

### setTableAlternating1CssClass

```
public void setTableAlternating1CssClass(String tableAlternating1CssClass)
```

**Parameters:**

tableAlternating1CssClass - the css class of the odd rows

### getTableAlternating2CssClass

```
public String getTableAlternating2CssClass()
```

**Returns:**

the css class of the even rows

### setTableAlternating2CssClass

```
public void setTableAlternating2CssClass(String tableAlternating2CssClass)
```

**Parameters:**

tableAlternating2CssClass - the css class of the even rows

### getTableCssClass

```
public String getTableCssClass()
```

**Returns:**

the css class of the table tag

### setTableCssClass

```
public void setTableCssClass(String tableCssClass)
```

(continued from last page)

**Parameters:**

tableCssClass - the css class of the table tag

---

**isRenderAlternatingCssClass**

```
public boolean isRenderAlternatingCssClass()
```

**Returns:**

whether to render alternating css classes

---

**setRenderAlternatingCssClass**

```
public void setRenderAlternatingCssClass(boolean isRenderAlternatingCssClass)
```

**Parameters:**

isRenderAlternatingCssClass - whether to render alternating css classes

---

**isAdjustNeighbours**

```
public boolean isAdjustNeighbours()
```

Determines whether to adjust the tags below the table tag, as the table tag is taller than the default tags.

**Returns:**

whether to adjust the top/left attributes of the tags below the table tag

---

**setAdjustNeighbours**

```
public void setAdjustNeighbours(boolean isAdjustNeighbours)
```

Determines whether to adjust the tags below the table tag, as the table tag is taller than the default tags.

**Parameters:**

isAdjustNeighbors - whether the top/left attributes of the tags below the table tag will be adjusted

---

**isWindowRelative**

```
public boolean isWindowRelative()
```

For internal usage

**Returns:**

---

**setWindowRelative**

```
public void setWindowRelative(boolean isWindowRelative)
```

For internal usage

**Parameters:**

isWindowRelative

---

## get\_TableAlternating1CssClass

```
public String get_TableAlternating1CssClass()
```

For .NET framework usage. Expose the method as a property.

---

## set\_TableAlternating1CssClass

```
public void set_TableAlternating1CssClass(String tableAlternating1CssClass)
```

For .NET framework usage. Expose the method as a property.

---

## get\_TableAlternating2CssClass

```
public String get_TableAlternating2CssClass()
```

For .NET framework usage. Expose the method as a property.

---

## set\_TableAlternating2CssClass

```
public void set_TableAlternating2CssClass(String tableAlternating2CssClass)
```

For .NET framework usage. Expose the method as a property.

---

## get\_TableCssClass

```
public String get_TableCssClass()
```

For .NET framework usage. Expose the method as a property.

---

## set\_TableCssClass

```
public void set_TableCssClass(String tableCssClass)
```

For .NET framework usage. Expose the method as a property.

---

## get\_RenderAlternatingCssClass

```
public boolean get_RenderAlternatingCssClass()
```

For .NET framework usage. Expose the method as a property.

---

## set\_RenderAlternatingCssClass

```
public void set_RenderAlternatingCssClass(boolean isRenderAlternatingCssClass)
```

For .NET framework usage. Expose the method as a property.

---

## get\_AdjustNeighbours

```
public boolean get_AdjustNeighbours()
```

For .NET framework usage. Expose the method as a property.

---

## set\_AdjustNeighbours

```
public void set_AdjustNeighbours(boolean isAdjustNeighbours)
```

---

(continued from last page)

For .NET framework usage. Expose the method as a property.

---

## **get\_WindowRelative**

```
public boolean get_WindowRelative()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_WindowRelative**

```
public void set_WindowRelative(boolean isWindowRelative)
```

For .NET framework usage. Expose the method as a property.

# com.sabratec.applinx.presentation.transforms

## Class GXTransformConfig

java.lang.Object  
└─com.sabratec.applinx.presentation.transforms.GXTransformConfig

All Implemented Interfaces:  
java.io.Serializable

Direct Known Subclasses:  
[GXWindowFrameTransformConfig](#), [GXTableTransformConfig](#), [GXHostKeysTransformConfig](#),  
[GXGuiElementsTransformConfig](#), [GXEmulationTransformConfig](#)

public abstract class **GXTransformConfig**  
extends Object  
implements java.io.Serializable

Abstract class for all transformation configurations. Contains a method `setEnabled` for enabling/disabling the transformation.

Constructor Summary	
public	<a href="#">GXTransformConfig()</a>

Method Summary	
boolean	<a href="#">get_Enabled()</a> For .NET framework usage.
boolean	<a href="#">isEnabled()</a>
void	<a href="#">set_Enabled(boolean isEnabled)</a> For .NET framework usage.
void	<a href="#">setEnabled(boolean isEnabled)</a>

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

## Constructors

**GXTransformConfig**  
public **GXTransformConfig()**

## Methods

---

(continued from last page)

## **isEnabled**

```
public boolean isEnabled()
```

### **Returns:**

whether the relevant transformation (such as displaying tables, or displaying the graphical window frame) is enabled

---

## **setEnabled**

```
public void setEnabled(boolean isEnabled)
```

### **Parameters:**

`isEnabled` - whether to enable the relevant transformation

---

## **get\_Enabled**

```
public boolean get_Enabled()
```

For .NET framework usage. Expose the method as a property.

---

## **set\_Enabled**

```
public void set_Enabled(boolean isEnabled)
```

For .NET framework usage. Expose the method as a property.

## com.sabratec.applinx.presentation.transforms

### Class GXTransformConstants

```
java.lang.Object
```

```
└--com.sabratec.applinx.presentation.transforms.GXTransformConstants
```

```
public class GXTransformConstants
extends Object
```

This class contains various constants which are used for ApplinX built-in transformations.

#### Field Summary

public static final	<a href="#"><u>COLORS_ALL</u></a> Renders both foreground and background colors for each tag. Value: <b>0</b>
public static final	<a href="#"><u>COLORS_NONE</u></a> Renders no colors Value: <b>1</b>
public static final	<a href="#"><u>COLORS_REVERSED_VIDEO</u></a> Render only back colors Value: <b>2</b>
public static final	<a href="#"><u>HOSTKEYS_BUTTONS</u></a> Renders host keys as buttons. Value: <b>1</b>
public static final	<a href="#"><u>HOSTKEYS_DELETE</u></a> Deletes the button screen text from the screen. Value: <b>2</b>
public static final	<a href="#"><u>HOSTKEYS_LINKS</u></a> Renders host-keys as links. Value: <b>0</b>

#### Constructor Summary

public	<a href="#"><u>GXTransformConstants()</u></a>
--------	---

#### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

#### Fields

##### COLORS\_ALL

```
public static final int COLORS_ALL
```

(continued from last page)

Renders both foreground and background colors for each tag.  
Constant value: **0**

**See Also:**[GXEmulationTransformConfig.setColorMode\(int\)](#)

---

## COLORS\_NONE

```
public static final int COLORS_NONE
```

Renders no colors  
Constant value: **1**

**See Also:**[GXEmulationTransformConfig.setColorMode\(int\)](#)

---

## COLORS\_REVERSED\_VIDEO

```
public static final int COLORS_REVERSED_VIDEO
```

Render only back colors  
Constant value: **2**

**See Also:**[GXEmulationTransformConfig.setColorMode\(int\)](#)

---

## HOSTKEYS\_LINKS

```
public static final int HOSTKEYS_LINKS
```

Renders host-keys as links. The links will be rendered in the same position as the host key's caption, if the host key's action and caption are in the same row. If they are not, the links will be rendered after the last line.  
Constant value: **0**

**See Also:**[GXHostKeysTransformConfig.setRenderingType\(int\)](#)

---

## HOSTKEYS\_BUTTONS

```
public static final int HOSTKEYS_BUTTONS
```

Renders host keys as buttons. The buttons will be rendered in the same position as the host key's caption, if the host key's action and caption are in the same row. If they are not, the buttons will be rendered after the last line.  
Constant value: **1**

**See Also:**[GXHostKeysTransformConfig.setRenderingType\(int\)](#)

---

## HOSTKEYS\_DELETE

```
public static final int HOSTKEYS_DELETE
```

Deletes the button screen text from the screen.  
Constant value: **2**

**See Also:**[GXHostKeysTransformConfig.setRenderingType\(int\)](#)

---

## Constructors

(continued from last page)

## **GXTransformConstants**

```
public GXTransformConstants()
```

## com.sabratec.applinx.presentation.transforms

### Class GXWindowFrameTransformConfig

java.lang.Object

```

+--com.sabratec.applinx.presentation.transforms.GXTransformConfig
    |
    +--com.sabratec.applinx.presentation.transforms.GXWindowFrameTransformConfig

```

#### All Implemented Interfaces:

java.io.Serializable, java.io.Serializable

public class **GXWindowFrameTransformConfig**  
 extends [GXTransformConfig](#)  
 implements java.io.Serializable, java.io.Serializable

This class is the transformation configuration for window frames as defined in the ApplinX application. It replaces the host window characters/reverse video with a solid bordered frame. The configuration determines the CSS class for the window frame transformation.

### Constructor Summary

public	<a href="#">GXWindowFrameTransformConfig()</a>
--------	--

### Method Summary

boolean	<a href="#">get_HideOnly()</a> For .NET framework usage.
String	<a href="#">get_TitleWindowFrameCssClass()</a> For .NET framework usage.
String	<a href="#">get_WindowFrameCssClass()</a> For .NET framework usage.
String	<a href="#">getTitleWindowFrameCssClass()</a> The CSS class of window frame title.
String	<a href="#">getWindowFrameCssClass()</a> The CSS class of window frame.
boolean	<a href="#">isHideOnly()</a>
void	<a href="#">set_HideOnly(boolean hideOnly)</a> For .NET framework usage.
void	<a href="#">set_TitleWindowFrameCssClass(String titleWindowFrameCssClass)</a> For .NET framework usage.
void	<a href="#">set_WindowFrameCssClass(String windowFrameCssClass)</a> For .NET framework usage.
void	<a href="#">setHideOnly(boolean hideOnly)</a>

void	<a href="#"><code>setTitleWindowFrameCssClass</code></a> (String titleWindowFrameCssClass) The CSS class of window frame Title.
void	<a href="#"><code>setWindowFrameCssClass</code></a> (String windowFrameCssClass) The CSS class of window frame.

Methods inherited from class [`com.sabratec.applinx.presentation.transforms.GXTransformConfig`](#)

[`get\_Enabled`](#), [`isEnabled`](#), [`set\_Enabled`](#), [`setEnabled`](#)

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### **GXWindowFrameTransformConfig**

```
public GXWindowFrameTransformConfig()
```

## Methods

### **getWindowFrameCssClass**

```
public String getWindowFrameCssClass()
```

The CSS class of window frame.

**Returns:**

the CSS class of window frame

### **setWindowFrameCssClass**

```
public void setWindowFrameCssClass(String windowFrameCssClass)
```

The CSS class of window frame.

**Parameters:**

`windowFrameCssClass` - the css class of window frame

### **getTitleWindowFrameCssClass**

```
public String getTitleWindowFrameCssClass()
```

The CSS class of window frame title.

**Returns:**

the css class of window frame title

### **setTitleWindowFrameCssClass**

```
public void setTitleWindowFrameCssClass(String titleWindowFrameCssClass)
```

(continued from last page)

The CSS class of window frame Title.

**Parameters:**

windowFrameCssClass - the css class of window frame title

---

**setHideOnly**

```
public void setHideOnly(boolean hideOnly)
```

---

**isHideOnly**

```
public boolean isHideOnly()
```

---

**get\_WindowFrameCssClass**

```
public String get_WindowFrameCssClass()
```

For .NET framework usage. Expose the method as a property.

---

**set\_WindowFrameCssClass**

```
public void set_WindowFrameCssClass(String windowFrameCssClass)
```

For .NET framework usage. Expose the method as a property.

---

**get\_TitleWindowFrameCssClass**

```
public String get_TitleWindowFrameCssClass()
```

For .NET framework usage. Expose the method as a property.

---

**set\_TitleWindowFrameCssClass**

```
public void set_TitleWindowFrameCssClass(String titleWindowFrameCssClass)
```

For .NET framework usage. Expose the method as a property.

---

**set\_HideOnly**

```
public void set_HideOnly(boolean hideOnly)
```

For .NET framework usage. Expose the method as a property.

---

**get\_HideOnly**

```
public boolean get_HideOnly()
```

For .NET framework usage. Expose the method as a property.

---

**Package**  
**com.sabratec.ci**

## com.sabratec.ci

### Class GXLoadBalancer

java.lang.Object

└-com.sabratec.ci.GXLoadBalancer

public class **GXLoadBalancer**  
extends Object

In addition to working with any standard load-balancing tool, ApplinX also has a built-in Load Balancer, which allows a user to define a list of candidate servers, and receive the URL of the server which is least busy. The calculation of the load factor of a server is dependent on variables such as the number of users currently connected to the server, number of users that are idle, and other possible factors. The user can explicitly supply the list of possible servers. Servers, which are currently not running, can be added to the list and the Load Balancer will continue working normally. Alternatively, it can be read from a file containing the URLs of common servers.

#### Constructor Summary

public	<a href="#"><u>GXLoadBalancer()</u></a>
public	<a href="#"><u>GXLoadBalancer</u></a> (String[] serversURLs)
public	<a href="#"><u>GXLoadBalancer</u></a> (String fileName)

#### Method Summary

void	<a href="#"><u>addServerURL</u></a> (String serverURL) Adds a server's URL to the list of possible servers to connect.
String	<a href="#"><u>getAvailableServer</u></a> () A separate thread is instantiated for every server on the list.
String[]	<a href="#"><u>getServers</u></a> () Returns the list of possible servers to connect.
long	<a href="#"><u>getWaitForResponseTimeout</u></a> () Returns a response timeout, which is the time limit for waiting for servers load factor responses.
void	<a href="#"><u>removeAllServerURLs</u></a> () Cleans the list of server urls.
void	<a href="#"><u>setConfigFile</u></a> (String configFile) Sets a configuration file from which server URLs will be taken.
void	<a href="#"><u>setServers</u></a> (String[] serverURLs) Sets the list of possible servers to connect.
void	<a href="#"><u>setWaitForResponseTimeout</u></a> (long timeout) Sets a response timeout, to limit the time to wait for servers load factor responses.

Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

---

## Constructors

### **GXLoadBalancer**

```
public GXLoadBalancer()
```

---

### **GXLoadBalancer**

```
public GXLoadBalancer(String[] serversURLs)
```

---

### **GXLoadBalancer**

```
public GXLoadBalancer(String fileName)
```

---

## Methods

### **addServerURL**

```
public void addServerURL(String serverURL)
```

Adds a server's URL to the list of possible servers to connect.

**Parameters:**

serverURL - The server's URL

---

### **getServers**

```
public String[] getServers()
```

Returns the list of possible servers to connect.

**Returns:**

An array with the servers' URLs

---

### **setServers**

```
public void setServers(String[] serverURLs)
```

Sets the list of possible servers to connect.

**Parameters:**

serverURLs - An array with the servers' URLs

---

### **removeAllServerURLs**

```
public void removeAllServerURLs()
```

---

(continued from last page)

Cleans the list of server urls.

---

## setConfigFile

```
public void setConfigFile(String configFile)
```

Sets a configuration file from which server URLs will be taken. Each URL should be in a separate line

### Parameters:

`configFile` - The full path of the configuration file

---

## setWaitForResponseTimeout

```
public void setWaitForResponseTimeout(long timeout)
```

Sets a response timeout, to limit the time to wait for servers load factor responses. While threads are working, the load balancer waits for a specific amount of time. When the last server has responded, the load balancer stops waiting and continues with the algorithm. This method exists for cases a server does not respond, to refrain from waiting indefinitely. The default timeout is one minute (in milliseconds).

### Parameters:

`timeout` - The timeout in milliseconds

---

## getWaitForResponseTimeout

```
public long getWaitForResponseTimeout()
```

Returns a response timeout, which is the time limit for waiting for servers load factor responses.

### Returns:

The timeout in milliseconds

---

## getAvailableServer

```
public String getAvailableServer()
```

A separate thread is instantiated for every server on the list. Each thread sends a request to the server to return its load factor. The load factor is calculated and the answer is sent back to the load balancer, which compares all load factors and chooses the minimal one.

### Returns:

the server's URL

---

---

## Package

# com.sabratec.ci.print

ApplinX supports printer sessions on AS400 and mainframe hosts. It connects to the host, retrieves the print buffers and analyzes them. The host handles the printers queue and connection of printer sessions to display sessions. ApplinX connects to the host as a printer session to receive the print buffers and allows you to work with them.

The printer session has two levels. The first level is by way of the application programming interface which reflects the data returned into your required environment for example, XML, database, email, web or desktop application. The second level is the default behavior where ApplinX sends all print jobs to the clients machine by means of an applet that runs on the clients browser.

## com.sabratec.ci.print Class GXPage

java.lang.Object

└─com.sabratec.ci.print.GXPage

public class **GXPage**  
extends Object

This class implements one page that is going to be printed. The object can be retrieved from a GXPrintObject by the getPageAt() or getPages() methods once there is a print buffer.

**See Also:**

[GXPrintObject.getPageAt\(int\)](#), [GXPrintObject.getPages\(\)](#)

### Constructor Summary

public	<a href="#">GXPage()</a>
--------	--------------------------

### Method Summary

String	<a href="#">getLineAt(int index)</a> Returns the line at the specified index.
int	<a href="#">getLineCount()</a> Returns the number of lines in the page.
String	<a href="#">getOriginalBuffer()</a> Returns the portion of the buffer that corresponds to the current page as a string.
GXISCSInterface[]	<a href="#">getSCSElements()</a> Returns an array of SCS elements that correspond to the current page.
int	<a href="#">getSeqPageNumber()</a> Returns the sequential number of the page in the current print.
boolean	<a href="#">hasContent()</a> Return whether the page contains at least one char or not.
boolean	<a href="#">isComplete()</a> Returns true if the page is complete.
boolean	<a href="#">isEmphasis(String content)</a>
void	<a href="#">setComplete()</a>
void	<a href="#">setIsWrapTextOnMPP(boolean isWrap)</a>

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

## Constructors

### GXPage

```
public GXPage()
```

## Methods

### setIsWrapTextOnMPP

```
public void setIsWrapTextOnMPP(boolean isWrap)
```

### setComplete

```
public void setComplete()
```

### isComplete

```
public boolean isComplete()
```

Returns true if the page is complete. A page can be not complete if it is the last page that was analyzed from an incomplete buffer.

### isEmphasis

```
public boolean isEmphasis(String content)
```

### getLineCount

```
public int getLineCount()
```

Returns the number of lines in the page.

**Returns:**

number of lines in the page.

### hasContent

```
public boolean hasContent()
```

Return whether the page contains at least one char or not.

**Returns:**

whether the page contains at least one char or not.

### getLineAt

```
public String getLineAt(int index)
```

Returns the line at the specified index.

(continued from last page)

**Parameters:**

index - ranges from 0 to line count?1.

---

**getSeqPageNumber**

```
public int getSeqPageNumber()
```

Returns the sequential number of the page in the current print.

---

**getOriginalBuffer**

```
public String getOriginalBuffer()
```

Returns the portion of the buffer that corresponds to the current page as a string.

---

**getSCSElements**

```
public GXISCSInterface[] getSCSElements()
```

Returns an array of SCS elements that correspond to the current page.

## com.sabratec.ci.print Class GXPrinter

```

java.lang.Object
  |
  +--GXGeneralSession
      |
      +--com.sabratec.ci.print.GXPrinter
  
```

public class **GXPrinter**  
extends GXGeneralSession

This class represents a printer session. The methods to connect to the Applinx Server are similar to the connection methods in the display session.

### Field Summary

public static final	<a href="#">LOG_MODE_DEBUG</a> Value: 1
public static final	<a href="#">LOG_MODE_NORMAL</a> Value: 0

### Constructor Summary

public	<a href="#">GXPrinter()</a>
--------	-----------------------------

### Method Summary

int	<a href="#">attach()</a>
int	<a href="#">getCharactersPerLine()</a>
int	<a href="#">getLinesPerPage()</a>
GXPrintObjectInterface	<a href="#">getPrint()</a> Sends a request to the server to get the latest buffer and returns the GXPrintObject object.
int	<a href="#">getPrinterStatus()</a> Returns the printer status.
void	<a href="#">GXPrinter()</a>
void	<a href="#">setCharactersPerLine(int charactersPerLine)</a>
void	<a href="#">setLinesPerPage(int linesPerPage)</a>
void	<a href="#">setLogLevel(int logLevel)</a>

int	<a href="#">setWaitCondition</a> (int type, String paramString, int intParam1, int intParam2, int caseSensitive, long timeout, long flicker)
-----	--

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### LOG\_MODE\_NORMAL

```
public static final int LOG_MODE_NORMAL
```

Constant value: 0

### LOG\_MODE\_DEBUG

```
public static final int LOG_MODE_DEBUG
```

Constant value: 1

## Constructors

### GXPrinter

```
public GXPrinter()
```

## Methods

### GXPrinter

```
public void GXPrinter()
```

### attach

```
public int attach()
```

### getPrint

```
public GXPrintObjectInterface getPrint()
```

Sends a request to the server to get the latest buffer and returns the GXPrintObject object. If the sequential print number was increased or the current print was not retrieved, it returns a new GXPrintObject object, otherwise it returns null.

### getPrinterStatus

```
public int getPrinterStatus()
```

(continued from last page)

Returns the printer status. Possible printer status parameters and their descriptions: GX\_PRINTER\_STATUS\_BUSY The print job is in process. GX\_PRINTER\_STATUS\_CANCELLING Canceling a print job. GX\_PRINTER\_STATUS\_ERROR The host believes that there is an error in the print session. GX\_PRINTER\_STATUS\_INUSE Trying to connect a print session that is engaged to another device. For example, two print sessions are opened with the same printer. GX\_PRINTER\_STATUS\_INVALID The printer object is not attached to the server (default). GX\_PRINTER\_STATUS\_PAPEROUT ApplinX Server has advised the host that it is out of paper. GX\_PRINTER\_STATUS\_READY The printer session is connected to the host and the printer is ready to accept print jobs from the host. GX\_PRINTER\_STATUS\_TIMEDOUT The host attempts to send a print job, however a problem exists with connecting with the ApplinX Server and the time has elapsed.

---

## setLogLevel

```
public void setLogLevel(int logLevel)
```

### Parameters:

logLevel - possible values: LOG\_MODE\_NORMAL, LOG\_MODE\_DEBUG

---

## setLinesPerPage

```
public void setLinesPerPage(int linesPerPage)
```

---

## getLinesPerPage

```
public int getLinesPerPage()
```

---

## setCharactersPerLine

```
public void setCharactersPerLine(int charactersPerLine)
```

---

## getCharactersPerLine

```
public int getCharactersPerLine()
```

---

## setWaitCondition

```
public int setWaitCondition(int type,  
    String paramString,  
    int intParam1,  
    int intParam2,  
    int caseSensitive,  
    long timeout,  
    long flicker)
```

## com.sabratec.ci.print Class GXPrinterSessionConstants

```
java.lang.Object
```

```
└--com.sabratec.ci.print.GXPrinterSessionConstants
```

```
public class GXPrinterSessionConstants  
extends Object
```

This class holds all the constants of the printer session objects. Used mainly in .NET (where static variables in interfaces are not exposed).

### Field Summary

public static final	<a href="#"><u>GX_APPLICATION_PARM_ASSOCIATE_DEVICE_NAME</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_DEVICE_NAME</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_DEVICE_TYPE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_DISPLAY_DEVICE_NAME</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_HOST_ADDRESS</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_HOST_PORT</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_FIRST_FORM_FEED</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_FORMFEED_IN_TRANSPARENT</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_HOST_FORMATTING</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_HOST_SETTING_FOR_MAX_CHARACTERS_PER_LINE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_HOST_SETTING_FOR_MAX_LINES_PER_PAGE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_TRANSPARENT</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_IGNORE_VERTICAL_CHANNEL_SELECT</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_LANGUAGE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_MESSAGE_LIBRARY</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_MESSAGE_QUEUE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_REPLAY_FILE</u></a>

public static final	<a href="#"><u>GX_APPLICATION_PARM_TRACE_FILE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_TRANSLATE_ESCAPE</u></a>
public static final	<a href="#"><u>GX_APPLICATION_PARM_TRANSPARENT_DATA_IS_ASCII</u></a>
public static final	<a href="#"><u>GX_PRINT_OBJECT_STATUS_COMPLETE</u></a>
public static final	<a href="#"><u>GX_PRINT_OBJECT_STATUS_EMPTY</u></a>
public static final	<a href="#"><u>GX_PRINT_OBJECT_STATUS_NOT_COMPLETE</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_BUSY</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_CANCELLING</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_ERROR</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_INUSE</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_INVALID</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_PAPEROUT</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_READY</u></a>
public static final	<a href="#"><u>GX_PRINTER_STATUS_TIMEDOUT</u></a>
public static final	<a href="#"><u>GX_WAIT_COND_END_OF_PRINT</u></a>
public static final	<a href="#"><u>GX_WAIT_COND_PRINTER_READY</u></a>

## Constructor Summary

public	<a href="#"><u>GXPrinterSessionConstants()</u></a>
--------	--

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### GX\_APPLICATION\_PARM\_LANGUAGE

public static final java.lang.String **GX\_APPLICATION\_PARM\_LANGUAGE**

---

## **GX\_APPLICATION\_PARM\_DEVICE\_TYPE**

```
public static final java.lang.String GX_APPLICATION_PARM_DEVICE_TYPE
```

---

---

## **GX\_APPLICATION\_PARM\_DEVICE\_NAME**

```
public static final java.lang.String GX_APPLICATION_PARM_DEVICE_NAME
```

---

---

## **GX\_APPLICATION\_PARM\_TRACE\_FILE**

```
public static final java.lang.String GX_APPLICATION_PARM_TRACE_FILE
```

---

---

## **GX\_APPLICATION\_PARM\_REPLAY\_FILE**

```
public static final java.lang.String GX_APPLICATION_PARM_REPLAY_FILE
```

---

---

## **GX\_APPLICATION\_PARM\_HOST\_ADDRESS**

```
public static final java.lang.String GX_APPLICATION_PARM_HOST_ADDRESS
```

---

---

## **GX\_APPLICATION\_PARM\_HOST\_PORT**

```
public static final java.lang.String GX_APPLICATION_PARM_HOST_PORT
```

---

---

## **GX\_APPLICATION\_PARM\_ASSOCIATE\_DEVICE\_NAME**

```
public static final java.lang.String GX_APPLICATION_PARM_ASSOCIATE_DEVICE_NAME
```

---

---

## **GX\_APPLICATION\_PARM\_DISPLAY\_DEVICE\_NAME**

```
public static final java.lang.String GX_APPLICATION_PARM_DISPLAY_DEVICE_NAME
```

---

---

## **GX\_APPLICATION\_PARM\_MESSAGE\_QUEUE**

```
public static final java.lang.String GX_APPLICATION_PARM_MESSAGE_QUEUE
```

---

---

## **GX\_APPLICATION\_PARM\_MESSAGE\_LIBRARY**

```
public static final java.lang.String GX_APPLICATION_PARM_MESSAGE_LIBRARY
```

---

(continued from last page)

---

## **GX\_APPLICATION\_PARM\_IGNORE\_HOST\_SETTING\_FOR\_MAX\_CHARACTERS\_PER\_LINE**

```
public static final java.lang.String  
GX_APPLICATION_PARM_IGNORE_HOST_SETTING_FOR_MAX_CHARACTERS_PER_LINE
```

---

## **GX\_APPLICATION\_PARM\_IGNORE\_HOST\_SETTING\_FOR\_MAX\_LINES\_PER\_PAGE**

```
public static final java.lang.String  
GX_APPLICATION_PARM_IGNORE_HOST_SETTING_FOR_MAX_LINES_PER_PAGE
```

---

## **GX\_APPLICATION\_PARM\_IGNORE\_TRANSPARENT**

```
public static final java.lang.String GX_APPLICATION_PARM_IGNORE_TRANSPARENT
```

---

## **GX\_APPLICATION\_PARM\_TRANSPARENT\_DATA\_IS\_ASCII**

```
public static final java.lang.String GX_APPLICATION_PARM_TRANSPARENT_DATA_IS_ASCII
```

---

## **GX\_APPLICATION\_PARM\_IGNORE\_FORMFEED\_IN\_TRANSPARENT**

```
public static final java.lang.String  
GX_APPLICATION_PARM_IGNORE_FORMFEED_IN_TRANSPARENT
```

---

## **GX\_APPLICATION\_PARM\_IGNORE\_VERTICAL\_CHANNEL\_SELECT**

```
public static final java.lang.String  
GX_APPLICATION_PARM_IGNORE_VERTICAL_CHANNEL_SELECT
```

---

## **GX\_APPLICATION\_PARM\_TRANSLATE\_ESCAPE**

```
public static final java.lang.String GX_APPLICATION_PARM_TRANSLATE_ESCAPE
```

---

## **GX\_APPLICATION\_PARM\_IGNORE\_FIRST\_FORM\_FEED**

```
public static final java.lang.String GX_APPLICATION_PARM_IGNORE_FIRST_FORM_FEED
```

---

(continued from last page)

---

## **GX\_APPLICATION\_PARM\_IGNORE\_HOST\_FORMATTING**

```
public static final java.lang.String GX_APPLICATION_PARM_IGNORE_HOST_FORMATTING
```

---

## **GX\_WAIT\_COND\_END\_OF\_PRINT**

```
public static final int GX_WAIT_COND_END_OF_PRINT
```

---

## **GX\_WAIT\_COND\_PRINTER\_READY**

```
public static final int GX_WAIT_COND_PRINTER_READY
```

---

## **GX\_PRINTER\_STATUS\_INVALID**

```
public static final int GX_PRINTER_STATUS_INVALID
```

---

## **GX\_PRINTER\_STATUS\_READY**

```
public static final int GX_PRINTER_STATUS_READY
```

---

## **GX\_PRINTER\_STATUS\_BUSY**

```
public static final int GX_PRINTER_STATUS_BUSY
```

---

## **GX\_PRINTER\_STATUS\_ERROR**

```
public static final int GX_PRINTER_STATUS_ERROR
```

---

## **GX\_PRINTER\_STATUS\_TIMEDOUT**

```
public static final int GX_PRINTER_STATUS_TIMEDOUT
```

---

## **GX\_PRINTER\_STATUS\_PAPEROUT**

```
public static final int GX_PRINTER_STATUS_PAPEROUT
```

---

## **GX\_PRINTER\_STATUS\_INUSE**

```
public static final int GX_PRINTER_STATUS_INUSE
```

---

(continued from last page)

---

## **GX\_PRINTER\_STATUS\_CANCELLING**

```
public static final int GX_PRINTER_STATUS_CANCELLING
```

---

## **GX\_PRINT\_OBJECT\_STATUS\_EMPTY**

```
public static final int GX_PRINT_OBJECT_STATUS_EMPTY
```

---

## **GX\_PRINT\_OBJECT\_STATUS\_NOT\_COMPLETE**

```
public static final int GX_PRINT_OBJECT_STATUS_NOT_COMPLETE
```

---

## **GX\_PRINT\_OBJECT\_STATUS\_COMPLETE**

```
public static final int GX_PRINT_OBJECT_STATUS_COMPLETE
```

## Constructors

### **GXPrinterSessionConstants**

```
public GXPrinterSessionConstants()
```

## com.sabratec.ci.print Class GXPrintObject

java.lang.Object

└─com.sabratec.ci.print.GXPrintObject

public class **GXPrintObject**  
extends Object

This class implements one print job from the host. Every new print job requires a new print object. The object is created by the `getPrint()` method of the `GXPrinter`.

**See Also:**

[GXPrinter.getPrint\(\)](#)

### Constructor Summary

public	<a href="#">GXPrintObject()</a>
--------	---------------------------------

### Method Summary

int	<a href="#">completeBuffer()</a> Completes the buffer using the printer session (if alive).
String	<a href="#">getBuffer()</a> Returns the buffer that currently exists in the object as string.
String	<a href="#">getBuffer(int buffSize)</a> Retrieves a portion of the buffer.
int	<a href="#">getBufferLength()</a> Returns the length of the current buffer.
int	<a href="#">getError()</a>
String	<a href="#">getErrorMessage()</a>
GXIPageInterface	<a href="#">getPageAt(int index)</a> Returns the page at the specified index.
int	<a href="#">getPageCount()</a> Returns the number of pages in the current print buffer, as they were analyzed by ApplinX.
GXIPageInterface[]	<a href="#">getPages()</a> Returns an array of GXPage objects according to the current buffer in the print object.
<a href="#">GXSCS[]</a>	<a href="#">getSCSElements()</a> Returns an array of GXSCS objects that hold all the printer controls that were retrieved along with the current buffer.
int	<a href="#">getSeqPrintNumber()</a> Returns the sequential print number ? the number of prints received from the host since the beginning of the current session.

boolean	<a href="#"><code>isComplete()</code></a> Returns true if the print object holds the complete buffer.
boolean	<a href="#"><code>isCurrentPrint()</code></a> Returns true if this print object is the printer session's current print object.
boolean	<a href="#"><code>isEmpty()</code></a> Returns true if the current print object is empty.
boolean	<a href="#"><code>isPassThroughData()</code></a>
void	<a href="#"><code>setCharactersPerLine(int maxCharactersPerLine)</code></a> Sets the number of characters per page.
void	<a href="#"><code>setLinesPerPage(int maxLinesPerPage)</code></a> Sets the number of lines per page.

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXPrintObject

```
public GXPrintObject()
```

## Methods

### isPassThroughData

```
public boolean isPassThroughData()
```

### isCurrentPrint

```
public boolean isCurrentPrint()
```

Returns true if this print object is the printer session's current print object.

#### Returns:

whether this print object is the current print in the printer session.

### isComplete

```
public boolean isComplete()
```

Returns true if the print object holds the complete buffer. If it returns false further requests must be sent to the server to retrieve the rest of the buffer.

#### Returns:

whether the print buffer is complete.

## isEmpty

```
public boolean isEmpty()
```

Returns true if the current print object is empty.

---

## getSeqPrintNumber

```
public int getSeqPrintNumber()
```

Returns the sequential print number ? the number of prints received from the host since the beginning of the current session.

**Returns:**

the sequential print number since the current print session was started.

---

## getBuffer

```
public String getBuffer()
```

Returns the buffer that currently exists in the object as string.

---

## getBuffer

```
public String getBuffer(int buffSize)
```

Retrieves a portion of the buffer.

**Parameters:**

buffSize - - optional - retrieves a portion of the buffer.

---

## getPageAt

```
public GXIPageInterface getPageAt(int index)
```

Returns the page at the specified index. Page starts from zero.

**Parameters:**

index - - starts from 0.

**Returns:**

analyzed page at the specified index.

---

## getPages

```
public GXIPageInterface[] getPages()
```

Returns an array of GXPage objects according to the current buffer in the print object.

---

## getPageCount

```
public int getPageCount()
```

Returns the number of pages in the current print buffer, as they were analyzed by ApplinX.

**Returns:**

number of pages according to the current buffer in the object.

---

---

## getBufferLength

```
public int getBufferLength()
```

Returns the length of the current buffer.

**Returns:**

length of the current buffer.

---

## getSCSElements

```
public GXSCS\[\] getSCSElements()
```

Returns an array of GXSCS objects that hold all the printer controls that were retrieved along with the current buffer.

**Returns:**

an array of GXSCS objects for the current print buffer.

---

## completeBuffer

```
public int completeBuffer()
```

Completes the buffer using the printer session (if alive). Sends a request to the server to complete the current buffer. This method can be used only if the print object is the current print object and if its status is not complete already.

---

## getError

```
public int getError()
```

---

## getErrorMessage

```
public String getErrorMessage()
```

---

## setLinesPerPage

```
public void setLinesPerPage(int maxLinesPerPage)
```

Sets the number of lines per page. Only applicable if you want to change the current print job settings. The default is the setting of the printer session's configuration, which you configured at the start of the print job.

**Parameters:**

maxLinesPerPage - the maximum lines per page.

---

## setCharactersPerLine

```
public void setCharactersPerLine(int maxCharactersPerLine)
```

Sets the number of characters per page. Only applicable if you want to change the current print job settings. The default is the setting of the printer session's configuration, which you configured at the start of the print job.

**Parameters:**

maxCharactersPerLine - the maximum characters per line.

---

## com.sabratec.ci.print Class GXSCS

java.lang.Object

└─com.sabratec.ci.print.GXSCS

public class **GXSCS**  
extends Object

This class represents a scs element. SCS are controls sent by the host that inform the printer how the print buffers should be presented.

### Method Summary

int	<a href="#"><u>getBufferPosition()</u></a> Retrieves point in complete buffer print-job.
int	<a href="#"><u>getColumnNumber()</u></a> Retrieves the column where the control string should be implemented.
byte[]	<a href="#"><u>getData()</u></a> Retrieves the complete data of the control string (SCS).
int	<a href="#"><u>getKeyAt(int index)</u></a> Retrieves a value key according to the key?s index.
int	<a href="#"><u>getKeysCount()</u></a> Number of dynamic values in the control string data.
int	<a href="#"><u>getLineNumber()</u></a> Retrieves the line where the control string should be implemented.
int	<a href="#"><u>getPageBufferPosition()</u></a> Retrieves a point in the portion of the original buffer that is in the current page.
int	<a href="#"><u>getType()</u></a> Retrieves a type of control string (SCS).
String	<a href="#"><u>toString()</u></a>

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Methods

### getKeysCount

public int **getKeysCount()**

Number of dynamic values in the control string data.

---

## getKeyAt

```
public int getKeyAt(int index)
```

Retrieves a value key according to the key's index. Starts from zero.

---

## getType

```
public int getType()
```

Retrieves a type of control string (SCS). List of types: GX\_TYPE\_UNKNOWN GX\_TYPE\_LINE\_DENSITY GX\_TYPE\_GRAPHIC\_ESCAPE GX\_TYPE\_CHARACTER\_DENSITY GX\_TYPE\_PAGE\_PRESETATION\_MEDIA GX\_TYPE\_VERTICAL\_CHANNEL\_SELECT GX\_TYPE\_TRANSPARENT GX\_TYPE\_HORIZONTAL\_TAB GX\_TYPE\_FORM\_FEED GX\_TYPE\_CARRIAGE\_RETURN GX\_TYPE\_NEW\_LINE GX\_TYPE\_BACKSPACE GX\_TYPE\_INTERCHANGE\_RECORD\_SEPARATOR GX\_TYPE\_LINE\_FEED GX\_TYPE\_HORIZONTAL\_FORMAT GX\_TYPE\_VERTICAL\_FORMAT GX\_TYPE\_VERTICAL\_TAB. SCS controls for double byte languages (Japanese / Korean) GX\_TYPE\_SHIFT\_IN GX\_TYPE\_SHIFT\_OUT GX\_TYPE\_SET\_ATTRIBUTE  
The following SCS type instructions should be ignored. GX\_TYPE\_NO\_OPERATION GX\_TYPE\_ENABLE\_PRESENTATION GX\_TYPE\_INHIBIT\_PRESENTATION GX\_TYPE\_ALERT\_BELL  
The following are applicable to AS/400 hosts only: GX\_TYPE\_GRAPHIC\_ERROR\_ACTION GX\_TYPE\_TEXT\_ORIENTATION GX\_TYPE\_FONT\_ID GX\_TYPE\_DEVICE\_EXCEPTION\_ACTION GX\_TYPE\_JUSTIFY\_TEXT\_FIELD GX\_TYPE\_JUSTIFY\_MODE GX\_TYPE\_HORIZONTAL\_MARGINS GX\_TYPE\_VERTICAL\_MARGINS GX\_TYPE\_PRESENTATION\_PAGE\_SIZE GX\_TYPE\_SUBSCRIPT GX\_TYPE\_SUPERSCRIPT GX\_TYPE SUBSTITUTE GX\_TYPE\_WORD\_UNDERSCORE GX\_TYPE\_BEGIN\_EMPHASIS GX\_TYPE\_BEGIN\_OVERSTRIKE GX\_TYPE\_BEGIN\_UNDERSCORE GX\_TYPE\_END\_EMPHASIS GX\_TYPE\_END\_OVERSTRIKE GX\_TYPE\_END\_UNDERSCORE GX\_TYPE\_REQUIRED\_NEW\_LINE GX\_TYPE\_EXPANDED\_BACKSPACE GX\_TYPE\_UNIT\_BACKSPACE GX\_TYPE\_INDENT\_TAB GX\_TYPE\_INDEX\_RETURN GX\_TYPE\_PRINT\_POSITION GX\_TYPE\_REQUIRED\_FORM\_FEED GX\_TYPE\_CODEPAGE\_LOCAL\_ID GX\_TYPE\_CODEPAGE\_GCID GX\_TYPE\_INITIAL\_CONDITIONS GX\_TYPE\_HORIZONTAL\_TAB\_STOPS GX\_TYPE\_INDENT\_LEVEL GX\_TYPE\_LINE\_SPACING GX\_TYPE\_SINGLE\_LINE\_DISTANCE GX\_TYPE\_ASCII\_TRANSPARENT  
SCS Types for Double Byte Languages (Japanese / Korean) GX\_TYPE\_FONT\_SIZE\_SCALING GX\_TYPE\_LOAD\_ALTERNATE\_CHARACTER GX\_TYPE\_DEFINE\_GRID\_LINE GX\_TYPE\_SET\_IGC\_TYPE GX\_TYPE\_PRESENTATION\_OF\_CONTROL\_CHARACTER  
The following SCS type instructions should be ignored. GX\_TYPE\_REPEAT GX\_TYPE\_SWITCH GX\_TYPE\_PRINT\_SETUP GX\_TYPE\_RELEASE\_LEFT\_MARGIN 1303

---

## getLineNumber

```
public int getLineNumber()
```

Retrieves the line where the control string should be implemented.

---

## getColumnNumber

```
public int getColumnNumber()
```

Retrieves the column where the control string should be implemented.

---

## getBufferPosition

```
public int getBufferPosition()
```

Retrieves point in complete buffer print-job.

---

## getPageBufferPosition

```
public int getPageBufferPosition()
```

Retrieves a point in the portion of the original buffer that is in the current page.

---

---

## **getData**

```
public byte[] getData()
```

Retrieves the complete data of the control string (SCS).

---

## **toString**

```
public String toString()
```

---

**Package**

**com.sabratec.j2ee.framework.web.taglib**

## com.sabratec.j2ee.framework.web.taglib

### Class GXAnchorTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXAnchorTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

#### Direct Known Subclasses:

[GXAbstractAnchorTag](#)

```

public class GXAnchorTag
extends GXAbstractTag
  
```

This class represents an anchor tag (link) which can be used to activate both client side functionality and server side functionality.

#### Syntax:

```
<gx:a id="<TAG_ID>" onclick="<SERVER_FUNCTION_NAME>">
```

## Constructor Summary

public	<a href="#">GXAnchorTag</a> ()
--------	--------------------------------

## Method Summary

void	<a href="#">setHref</a> (String value) Called by the JSP page for setting the href attribute to the tag output HTML.
void	<a href="#">setOnClick</a> (String value) Called by the JSP page for setting the onclick attribute to the HTML tag output.
void	<a href="#">setOnmouseout</a> (String value) Called by the JSP page for setting the onmouseout attribute to the tag output HTML.
void	<a href="#">setOnmouseover</a> (String value) Called by the JSP page for setting the onmouseover attribute to the tag output HTML.
void	<a href="#">setOnserverclick</a> (String value) Called by the JSP page for setting a server side function name that will be activated when the anchor (link) is clicked. <b>Syntax:</b> <gx:a id="<TAG_ID>" onclick="<SERVER_FUNCTION_NAME>"> The server side function is in the class which was set in: <gx:html gx_context="CONTEXT CLASS" ...>

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructors

### GXAnchorTag

```
public GXAnchorTag()
```

## Methods

### setOnserverclick

```
public void setOnserverclick(String value)
```

Called by the JSP page for setting a server side function name that will be activated when the anchor (link) is clicked.

**Syntax:**

```
<gx:a id="<TAG_ID>" onserverclick="<SERVER_FUNCTION_NAME>">
```

The server side function is in the class which was set in:

```
<gx:html gx_context="CONTEXT CLASS" ...>
```

**Parameters:**

value - The onserverclick function name (without brackets)

### setOnClick

```
public void setOnClick(String value)
```

Called by the JSP page for setting the onclick attribute to the HTML tag output. If onserverclick is also specified for the tag, the onserverclick will be added before the onclick attribute.

**Parameters:**

value - The onclick attribute value

### setOnmouseover

```
public void setOnmouseover(String value)
```

Called by the JSP page for setting the onmouseover attribute to the tag output HTML.

**Parameters:**

value - The onmouseover attribute value

### setOnmouseout

```
public void setOnmouseout(String value)
```

Called by the JSP page for setting the onmouseout attribute to the tag output HTML.

**Parameters:**

value - The onmouseout attribute value

(continued from last page)

## **setHref**

```
public void setHref(String value)
```

Called by the JSP page for setting the href attribute to the tag output HTML.

### **Parameters:**

value - The href attribute value

## com.sabratec.j2ee.framework.web.taglib

### Class GXBoldTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag
        |
        +-com.sabratec.j2ee.framework.web.taglib.GXBoldTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

public class **GXBoldTag**  
 extends [GXAbstractGenericTag](#)

## Constructor Summary

public	<a href="#">GXBoldTag()</a>
--------	-----------------------------

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag](#)

[setDir](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXBoldTag

public **GXBoldTag()**

## com.sabratec.j2ee.framework.web.taglib Class GXCheckBoxTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXCheckBoxTag
  
```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractCheckBoxTag](#)

```

public class GXCheckBoxTag
extends GXAbstractTag
  
```

Represents a <gx:checkbox> dynamic tag.

#### Syntax:

```
<gx:checkbox id="<TAG ID>" checkedValue="Y" uncheckValue="N" checked="true" />
```

The advantage of this tag over a regular HTML check box is that a regular check box retrieves null when the check box is not selected, whereas this tag returns the value of the cleared check box.

The tag renders a hidden input field along with the check box. To retrieve the check box content (checked/unchecked value) use: `getTagsAccesor().getTagContent("")`

## Constructor Summary

public	<a href="#">GXCheckBoxTag()</a>
--------	---------------------------------

## Method Summary

void	<a href="#">setChecked</a> (String value) Called by the JSP page for setting the checked attribute of the tag.
void	<a href="#">setCheckedValue</a> (String value) Called by the JSP page for setting a checked value when the check box is selected.
void	<a href="#">setOnClick</a> (String value) Called by the JSP page for setting the onclick attribute to the tag output HTML.
void	<a href="#">setUncheckedValue</a> (String value) Called by the JSP page for setting an unchecked value when the check box is not selected.

### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### **GXCheckBoxTag**

```
public GXCheckBoxTag()
```

## Methods

### **setCheckedValue**

```
public void setCheckedValue(String value)
```

Called by the JSP page for setting a checked value when the check box is selected.

**Parameters:**

value - The selected value

---

### **setUncheckedValue**

```
public void setUncheckedValue(String value)
```

Called by the JSP page for setting an unchecked value when the check box is not selected.

**Parameters:**

value - The unchecked value

---

### **setOnClick**

```
public void setOnClick(String value)
```

Called by the JSP page for setting the onclick attribute to the tag output HTML.

**Parameters:**

value - The onclick attribute value

---

### **setChecked**

```
public void setChecked(String value)
```

Called by the JSP page for setting the checked attribute of the tag.

**Parameters:**

value - The onclick attribute value

## com.sabratec.j2ee.framework.web.taglib

### Class GXDivTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag
        |
        +-com.sabratec.j2ee.framework.web.taglib.GXDivTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

public class **GXDivTag**  
 extends [GXAbstractGenericTag](#)

## Constructor Summary

public	<a href="#">GXDivTag()</a>
--------	----------------------------

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag](#)

[setDir](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXDivTag

public **GXDivTag**()

# com.sabratec.j2ee.framework.web.taglib

## Class GXFormTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXFormTag
  
```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractFormTag](#)

```

public class GXFormTag
extends GXAbstractTag
  
```

Represents the main form tag in the JSP page. The tag renders a <form> tag. Along with the form, the JavaScript function `gx_postBack` and a hidden field which saves the target function are rendered.

This function triggers a server function according to the specified function name.

#### Syntax:

```

<gx:form>
...
Field1: <gx:input ...>
Field2: <gx:input ...>
...
</gx:form>
  
```

## Constructor Summary

public	<a href="#">GXFormTag()</a>
--------	-----------------------------

## Method Summary

String	<a href="#">getId()</a> Returns the form ID.
void	<a href="#">setOnsubmit()</a> (String value) Called by the JSP page for setting the <code>onsubmit</code> attribute to the tag output HTML.
void	<a href="#">setSupportFileUpload()</a>

### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### **GXFormTag**

```
public GXFormTag()
```

## Methods

### **setOnsubmit**

```
public void setOnsubmit(String value)
```

Called by the JSP page for setting the onsubmit attribute to the tag output HTML.

**Parameters:**

value - The onsubmit attribute value

---

### **getId**

```
public String getId()
```

Returns the form ID. If an ID is not specified the ID is "GX\_form".

---

### **setSupportFileUpload**

```
public void setSupportFileUpload()
```

# com.sabratec.j2ee.framework.web.taglib

## Class GXHtmlTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXHtmlTag
  
```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractHtmlTag](#), [GXPageTag](#)

```

public class GXHtmlTag
extends GXAbstractTag
  
```

This class is the root tag of the JSP page, and the tag that initializes and activates the context class.

### Syntax:

```

<gx:html gx_context="<CONTEXT CLASS>">
...
<gx:form>
...
</gx:form>
...
</gx:html>
  
```

This class throws the following events in the context class:

- Page Initialization - `gx_onInit()`
- Page Load - `gx_onLoad()`
- Post Back (page is submitted to itself) - activating a target function using reflection from a `serverclick` event.
- Before html rendering - `gx_preRender()`
- Unload (for cleanup) - `gx_onUnload()`
- Error handling - `gx_onError(RuntimeException)()`

## Field Summary

public static final	<a href="#">HTML_TAG_ID</a>
	Value: <code>gx_html</code>

## Constructor Summary

public	<a href="#">GXHtmlTag()</a>
--------	-----------------------------

## Method Summary

int	<a href="#">doEndTag()</a> Method called at end of tag.
int	<a href="#">doStartTag()</a> Method called at beginning of tag.

String	<a href="#">getId()</a>
void	<a href="#">setGx_context</a> (String gx_contextName)
void	<a href="#">setPageContext</a> (PageContext pageContext)

Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Fields

### HTML\_TAG\_ID

```
public static final java.lang.String HTML_TAG_ID
```

Constant value: **gx\_html**

## Constructors

### GXHtmlTag

```
public GXHtmlTag()
```

## Methods

### setPageContext

```
public void setPageContext(PageContext pageContext)
```

Overrides the BodyTagSupport class, and is used for initializing the current tag.

### doStartTag

```
public int doStartTag()
    throws JspException
```

Method called at beginning of tag.

**Returns:**

EVAL\_BODY\_TAG

(continued from last page)

## doEndTag

```
public int doEndTag()  
    throws JspTagException
```

Method called at end of tag.

### Returns:

EVAL\_PAGE

---

## setGx\_context

```
public void setGx_context(String gx_contextName)  
    throws ClassNotFoundException,  
        IllegalAccessException,  
        InstantiationException
```

---

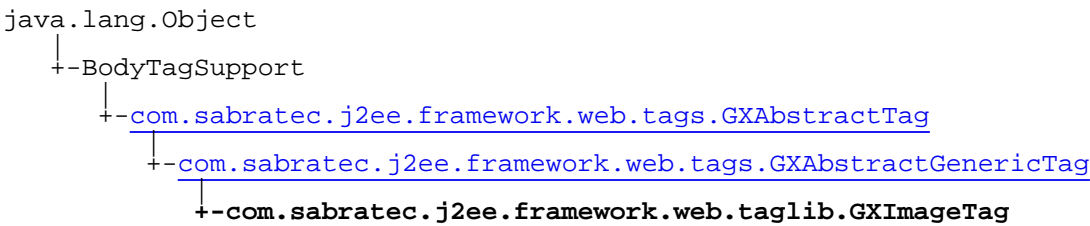
## getId

```
public String getId()
```

Retrieves the ID of the current tag

# com.sabratec.j2ee.framework.web.taglib

## Class GXImageTag



All Implemented Interfaces:  
java.io.Serializable

public class **GXImageTag**  
extends [GXAbstractGenericTag](#)

Constructor Summary	
public	<a href="#">GXImageTag()</a>

Method Summary	
void	<a href="#">setSrc</a> (String value)

Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag</a>
<a href="#">setDir</a>

Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>

Methods inherited from class java.lang.Object
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>

## Constructors

**GXImageTag**  
public **GXImageTag**()

## Methods

(continued from last page)

## **setSrc**

```
public void setSrc(String value)
```

## com.sabratec.j2ee.framework.web.taglib

### Class GXInputTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXInputTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

#### Direct Known Subclasses:

[GXAbstractInputTag](#)

```

public class GXInputTag
extends GXAbstractTag
  
```

This class represents an HTML input tag which can be one of the following types: text, hidden, password, button, image, submit . It should not be used for the following types of inputs:

- Checkbox - use [GXAbstractCheckBoxTag](#)
- Radio - use [GXAbstractRadioButtonList](#)
- 

#### Syntax:

```
<gx:input id="<TAG_ID>" type="text" maxlength="10">
```

```
<gx:input id="<TAG_ID>" type="button" onclick="<SERVER_FUNCTION_NAME>">
```

To dynamically set content to the input field use: `getTagsAccesor().setTagContent(, )`

To dynamically set attribute to the input field use: `getTagsAccesor().setTagAttribute(, , )`

## Constructor Summary

public	<a href="#">GXInputTag()</a>
--------	------------------------------

## Method Summary

void	<a href="#">setChecked</a> (String value)
void	<a href="#">setDir</a> (String value)
void	<a href="#">setMaxlength</a> (String value)
void	<a href="#">setOnchange</a> (String value)
void	<a href="#">setOnClick</a> (String value)
void	<a href="#">setOnkeydown</a> (String value)
void	<a href="#">setOnkeypress</a> (String value)

void	<a href="#"><u>setOnkeyup</u></a> (String value)
void	<a href="#"><u>setOnserverclick</u></a> (String value)
void	<a href="#"><u>setSize</u></a> (String value)
void	<a href="#"><u>setSrc</u></a> (String value)
void	<a href="#"><u>setTabIndex</u></a> (String value)
void	<a href="#"><u>setType</u></a> (String value)
void	<a href="#"><u>setValue</u></a> (String value)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXInputTag

```
public GXInputTag()
```

## Methods

### setOnserverclick

```
public void setOnserverclick(String value)
```

### setValue

```
public void setValue(String value)
```

### setOnClick

```
public void setOnClick(String value)
```

Called by the JSP page for setting the onclick attribute to the tag output html.

---

## setDir

```
public void setDir(String value)
```

---

## setType

```
public void setType(String value)
```

---

## setSize

```
public void setSize(String value)
```

---

## setMaxlength

```
public void setMaxlength(String value)
```

---

## setSrc

```
public void setSrc(String value)
```

---

## setOnkeydown

```
public void setOnkeydown(String value)
```

---

## setOnkeypress

```
public void setOnkeypress(String value)
```

---

## setOnkeyup

```
public void setOnkeyup(String value)
```

---

## setOnChange

```
public void setOnChange(String value)
```

---

## setTabIndex

```
public void setTabIndex(String value)
```

---

(continued from last page)

---

## **setChecked**

```
public void setChecked(String value)
```

## com.sabratec.j2ee.framework.web.taglib

### Class GXPageTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXHtmlTag
        |
        +-com.sabratec.j2ee.framework.web.taglib.GXPageTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

public class **GXPageTag**  
 extends [GXHtmlTag](#)

Fields inherited from class [com.sabratec.j2ee.framework.web.taglib.GXHtmlTag](#)

[HTML\\_TAG\\_ID](#)

## Constructor Summary

public [GXPageTag](#)( )

Methods inherited from class [com.sabratec.j2ee.framework.web.taglib.GXHtmlTag](#)

[doEndTag](#), [doStartTag](#), [getId](#), [setGx\\_context](#), [setPageContext](#)

Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXPageTag

public **GXPageTag**( )

# com.sabratec.j2ee.framework.web.taglib

## Class GXRadioButtonListTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXRadioButtonListTag
  
```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractRadioButtonList](#)

```

public class GXRadioButtonListTag
extends GXAbstractTag
  
```

This class represents a list of radio buttons. This tag is a parent tag of [GXAbstractRadioButton](#),

#### Syntax:

```

<gx:radioButtonList id="<TAG ID>">
  <gx:radioButton value="1" checked="true">View</gx:radioButton>
  <gx:radioButton value="2">Edit</gx:radioButton>
</gx:radioButtonList>
  
```

To dynamically set the selected option use: `getTagsAccesor().setTagContent("TAG ID>", "")`

To dynamically set the radio buttons list use: `getTagsAccesor().setTagList("", GXINameValueCollection)`

To read the the selected option use: `getTagsAccesor().getTagContent(<TAG ID>)`

#### See Also:

[GXAbstractRadioButton](#)

## Constructor Summary

public	<a href="#">GXRadioButtonListTag()</a>
--------	--

## Method Summary

String	<a href="#">getDir()</a>
String	<a href="#">getHorizontal()</a>
void	<a href="#">setBorder</a> (String value)
void	<a href="#">setCellpadding</a> (String value)
void	<a href="#">setCellspacing</a> (String value)
void	<a href="#">setDir</a> (String value)
void	<a href="#">setHorizontal</a> (String value)

Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXRadioButtonListTag

```
public GXRadioButtonListTag()
```

## Methods

### setBorder

```
public void setBorder(String value)
```

### setCellspacing

```
public void setCellspacing(String value)
```

### setCellpadding

```
public void setCellpadding(String value)
```

### setHorizontal

```
public void setHorizontal(String value)
```

### getHorizontal

```
public String getHorizontal()
```

### setDir

```
public void setDir(String value)
```

(continued from last page)

## **getDir**

```
public String getDir()
```

## com.sabratec.j2ee.framework.web.taglib Class GXRadioButtonTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXRadioButtonTag
  
```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractRadioButton](#)

```

public class GXRadioButtonTag
extends GXAbstractTag
  
```

This class represents a single radio button tag within a radio button list. This tag is a child of [GXAbstractRadioButtonList](#), and can't be used as a standalone tag.

#### Syntax:

```

<gx:radioButtonList id="<TAG ID>">
  <gx:radioButton value="1" checked="true">View</gx:radioButton>
  <gx:radioButton value="2">Edit</gx:radioButton>
</gx:radioButtonList>
  
```

To dynamically set the selected option use: `getTagsAccesor().setTagContent("", "")`

To dynamically set the radio buttons list use: `getTagsAccesor().setTagList("", GXINameValueCollection)`

To read the selected option use: `getTagsAccesor().getTagContent(<TAG ID>)`

#### See Also:

[GXAbstractRadioButtonList](#)

## Constructor Summary

public	<a href="#">GXRadioButtonTag()</a>
--------	------------------------------------

## Method Summary

void	<a href="#">setChecked</a> (String value) Called by the JSP page for setting the checked attribute to the tag.
void	<a href="#">setOnClick</a> (String value)
void	<a href="#">setValue</a> (String value) Called by the JSP page for setting the value attribute to the tag.

### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXRadioButtonTag

```
public GXRadioButtonTag()
```

## Methods

### setValue

```
public void setValue(String value)
```

Called by the JSP page for setting the value attribute to the tag.

**Parameters:**

value - The value attribute value

---

### setChecked

```
public void setChecked(String value)
```

Called by the JSP page for setting the checked attribute to the tag.

**Parameters:**

value - The checked attribute value

---

### setOnClick

```
public void setOnClick(String value)
```

Called by the JSP page for setting the onclick attribute to the tag output html.

## com.sabratec.j2ee.framework.web.taglib Class GXSelectTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXSelectTag
  
```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractSelectTag](#)

```

public class GXSelectTag
extends GXAbstractTag
  
```

This class represents a Select tag (HTML combobox).

#### Syntax:

```

<gx:select id="<TAG ID>">
  <option value="1">View</option>
  <option value="2">Edit</option>
</gx:select>
  
```

To dynamically set the selected option use: `getTagsAccesor().setTagContent("", "")`

To dynamically set the combobox option's list use: `getTagsAccesor().setTagList("", GXNameValueCollection)`

To read the the selected option use: `getTagsAccesor().getTagContent("")`

## Constructor Summary

public	<a href="#">GXSelectTag()</a>
--------	-------------------------------

## Method Summary

void	<a href="#">setDir</a> (String value)
void	<a href="#">setMultiple</a> (String value)
void	<a href="#">setOnchange</a> (String value)
void	<a href="#">setOnclick</a> (String value)
void	<a href="#">setSize</a> (String value)

### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnclick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

---

## Constructors

### GXSelectTag

```
public GXSelectTag()
```

## Methods

### setDir

```
public void setDir(String value)
```

---

### setOnChange

```
public void setOnChange(String value)
```

---

### setOnClick

```
public void setOnClick(String value)
```

Called by the JSP page for setting the onclick attribute to the tag output html.

---

### setSize

```
public void setSize(String value)
```

---

### setMultiple

```
public void setMultiple(String value)
```

## com.sabratec.j2ee.framework.web.taglib

### Class GXSpanTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag
        |
        +-com.sabratec.j2ee.framework.web.taglib.GXSpanTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

public class **GXSpanTag**  
 extends [GXAbstractGenericTag](#)

## Constructor Summary

public	<a href="#">GXSpanTag()</a>
--------	-----------------------------

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag](#)

[setDir](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXSpanTag

public **GXSpanTag()**

## com.sabratec.j2ee.framework.web.taglib Class GXTableTag

```

java.lang.Object
|
+-BodyTagSupport
|
+-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
|
+-com.sabratec.j2ee.framework.web.taglib.GXTableTag

```

### All Implemented Interfaces:

java.io.Serializable

### Direct Known Subclasses:

[GXAbstractHtmlTableTag](#)

```

public class GXTableTag
extends GXAbstractTag

```

This class creates an HTML table tag according to the specified inner HTML template (tr, td, etc.), and the specified data table: [GXITable](#). The table tag will duplicate the first HTML row and will insert dynamic values from the data table. The binding of child tags to the data table is done using the ID attributes which match the relevant column name in the data table.

#### Syntax:

```

<gx:table id=i%ItemsTablei%>
<tr>
<th> ItemNumber</th>
<th> Item Description</th>
</tr>
<tr>
<td id=i%ItemNumberi%>dummy data</td>
<td> <input id=i%ItemDesci% size=i%10i% value=i%sample datai%> </td>
</tr>
</gx:table>

```

#### The result will be:

```

<table id=i%ItemsTablei%>
<tr>
<th> ItemNumber</th>
<th> Item Description</th>
</tr>
<tr>
<td id=i%ItemNumber_000i%>1007</td>
<td> <input id=i%ItemDesc_000i% size=i%10i% value=i%Bananai%> </td>
</tr>
<tr>
<td id=i%ItemNumber_001i%>1008</td>
<td> <input id=i%ItemDesc_001i% size=i%10i% value=i%Applei%> </td>
</tr>
...
</table>

```

*Note: The ID attributes of the cloned rows, will be changed to <ORIGINAL\_TAG\_ID>\_00<ROW INDEX>.*

The data table is set to the tag by using the method `getTagsAccesor().setTagTable(String, GXITable, GXITableBuildConfig, GXITableEvents)` which can be accessed using the `getTagsAccesor()` method.

The `setTagTable` method accepts also a table build configuration class which provides common parameters for the table creation and a table-events class, which provides you the ability to change the data and/or HTML of a created row (tr), cell (td), or a control (any other child tag under a cell).

#### See Also:

com.sabratec.applinx.framework.tables.GXITableBuildConfig,  
com.sabratec.j2ee.framework.web.tables.GXITableEvents, [GXIClientBaseObject.getTable\(\)](#),  
org.w3c.dom.Element

## Constructor Summary

public	<a href="#">GXTableTag()</a>
--------	------------------------------

## Method Summary

void	<a href="#">setBorder</a> (String value)
void	<a href="#">setCellpadding</a> (String value)
void	<a href="#">setCellspacing</a> (String value)

### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

### Methods inherited from class [java.lang.Object](#)

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### **GXTableTag**

```
public GXTableTag()
```

## Methods

### **setBorder**

```
public void setBorder(String value)
```

### **setCellspacing**

```
public void setCellspacing(String value)
```

### **setCellpadding**

```
public void setCellpadding(String value)
```

# com.sabratec.j2ee.framework.web.taglib

## Class GXTagsAccesorImpl

java.lang.Object

↳ com.sabratec.j2ee.framework.web.taglib.GXTagsAccesorImpl

### All Implemented Interfaces:

java.io.Serializable, [GXITagsAccesor](#)

public class **GXTagsAccesorImpl**  
 extends Object  
 implements [GXITagsAccesor](#), java.io.Serializable

Impl. class for [GXITagsAccesor](#)

## Constructor Summary

public	<a href="#">GXTagsAccesorImpl</a> (HttpServletRequest request)
--------	--

## Method Summary

void	<a href="#">AddHiddenFieldToForm</a> (String name, String value)
void	<a href="#">AddHtmlTextToForm</a> (String htmlText) Adds an HTML text to the JSP page.
void	<a href="#">AddJSToForm</a> (String jsString)
void	<a href="#">addTableKeyColumn</a> (String refTagId, String keyCol)
String	<a href="#">getBodyAttribute</a> (String attrName)
Hashtable	<a href="#">getBodyAttributes</a> ()
String	<a href="#">getFileName</a> (String refTagId)
String	<a href="#">getFormAttribute</a> (String attrName)
Hashtable	<a href="#">getFormAttributes</a> ()
String	<a href="#">getFormTopString</a> ()
Map	<a href="#">getHiddens</a> ()
java.io.InputStream	<a href="#">getInputStream</a> (String refTagId)
Hashtable	<a href="#">getInputTagAttributes</a> (String refTagid)

String	<a href="#"><u>getPanelToUpdate()</u></a>
String	<a href="#"><u>getPanelToUpdateContent()</u></a>
String	<a href="#"><u>getTableName()</u></a>
String	<a href="#"><u>getTableSelectedKey</u></a> (String tagId)
String[]	<a href="#"><u>getTableSelectedKeys</u></a> (String refTagId)
String	<a href="#"><u>getTagAttribute</u></a> (String refTagId, String attrName)
Hashtable	<a href="#"><u>getTagAttributes</u></a> (String refTagid)
String	<a href="#"><u>getTagContent</u></a> (String tagId)
String[]	<a href="#"><u>getTagContents</u></a> (String refTagName)
boolean	<a href="#"><u>isTagVisible</u></a> (String refTagId)
void	<a href="#"><u>recycleBodyTag</u></a> ()
void	<a href="#"><u>recycleTag</u></a> (String refTagId, Hashtable tagPrivateAttributes)
void	<a href="#"><u>removeTagAttribute</u></a> (String refTagId, String attrName)
void	<a href="#"><u>setAttributeListIfEmpty</u></a> (String refTagId, Hashtable attrList)
void	<a href="#"><u>setBodyAttribute</u></a> (String attrName, String content)
void	<a href="#"><u>setBodyAttributeIfEmpty</u></a> (String attrName, String content)
void	<a href="#"><u>setFormAttribute</u></a> (String attrName, String content)
void	<a href="#"><u>setInputTagAttribute</u></a> (String refTagId, String attrName, String content)
void	<a href="#"><u>setPageVisible</u></a> (boolean visible)
void	<a href="#"><u>setPanelToUpdateContent</u></a> (String content)
void	<a href="#"><u>setTagAttribute</u></a> (String refTagId, String attrName, String content)
void	<a href="#"><u>setTagContent</u></a> (String refTagId, String content)
void	<a href="#"><u>setTagContentIfEmpty</u></a> (String refTagId, String content)
void	<a href="#"><u>setTagList</u></a> (String refTagId, GXINameValueCollection list)

void	<a href="#"><code>setTagList</code></a> (String refTagId, Hashtable list)
void	<a href="#"><code>setTagTable</code></a> (String tagId, <a href="#"><code>GXITable</code></a> table)
void	<a href="#"><code>setTagTable</code></a> (String refTagId, <a href="#"><code>GXITable</code></a> table, com.sabratec.j2ee.framework.web.tables.GXITableBuildConfig config, GXITableEvents events) <b>Deprecated.</b> Use <a href="#"><code>setTagTable(String, GXITable, GXITableBuildConfig, GXITableEvents)</code></a> instead
void	<a href="#"><code>setTagTable</code></a> (String tagId, <a href="#"><code>GXITable</code></a> table, GXITableBuildConfig config)
void	<a href="#"><code>setTagTable</code></a> (String refTagId, <a href="#"><code>GXITable</code></a> table, GXITableBuildConfig config, GXITableEvents events)
void	<a href="#"><code>setTagVisible</code></a> (String refTagId, boolean visible)
void	<a href="#"><code>setTagVisibleIfEmpty</code></a> (String refTagId, boolean visible)

**Methods inherited from class** `java.lang.Object`

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

**Methods inherited from interface** [`com.sabratec.j2ee.framework.web.tags.GXITagsAccesor`](#)
[`AddHiddenFieldToForm`](#), [`AddHtmlTextToForm`](#), [`AddJSToForm`](#), [`addTableKeyColumn`](#), [`getFileName`](#),  
[`getInputStream`](#), [`getTableName`](#), [`getTableSelectedKey`](#), [`getTableSelectedKeys`](#),  
[`getTagContent`](#), [`setBodyAttribute`](#), [`setFormAttribute`](#), [`setPageVisible`](#), [`setTagAttribute`](#),  
[`setTagContent`](#), [`setTagList`](#), [`setTagList`](#), [`setTagTable`](#), [`setTagTable`](#), [`setTagTable`](#),  
[`setTagTable`](#), [`setTagVisible`](#)

## Constructors

### **GXTagsAccesorImpl**

```
public GXTagsAccesorImpl(HttpServletRequest request)
```

## Methods

### **setTagContent**

```
public void setTagContent(String refTagId,  
    String content)
```

### **setTagContentIfEmpty**

```
public void setTagContentIfEmpty(String refTagId,  
    String content)
```

---

## getTagContents

```
public String[] getTagContents(String refTagName)
```

---

---

## getTagContent

```
public String getTagContent(String tagId)
```

---

---

## setTagVisible

```
public void setTagVisible(String refTagId,  
    boolean visible)
```

---

---

## setTagVisibleIfEmpty

```
public void setTagVisibleIfEmpty(String refTagId,  
    boolean visible)
```

---

---

## isTagVisible

```
public boolean isTagVisible(String refTagId)
```

---

---

## setTagList

```
public void setTagList(String refTagId,  
    Hashtable list)
```

---

---

## setTagTable

```
public void setTagTable(String tagId,  
    GXITable table)
```

---

---

## setTagTable

```
public void setTagTable(String tagId,  
    GXITable table,  
    GXITableBuildConfig config)
```

---

(continued from last page)

---

## setTagTable

```
public void setTagTable(String refTagId,  
    GXITable table,  
    com.sabratec.j2ee.framework.web.tables.GXITableBuildConfig config,  
    GXITableEvents events)
```

**Deprecated.** Use [setTagTable\(String, GXITable, GXITableBuildConfig, GXITableEvents\)](#) instead

---

---

## setTagTable

```
public void setTagTable(String refTagId,  
    GXITable table,  
    GXITableBuildConfig config,  
    GXITableEvents events)
```

---

## addTableKeyColumn

```
public void addTableKeyColumn(String refTagId,  
    String keyCol)
```

---

## getTableSelectedKey

```
public String getTableSelectedKey(String tagId)
```

---

## getTableSelectedKeys

```
public String[] getTableSelectedKeys(String refTagId)
```

---

## getTableName

```
public String getTableName()
```

---

## removeTagAttribute

```
public void removeTagAttribute(String refTagId,  
    String attrName)
```

---

## setTagAttribute

```
public void setTagAttribute(String refTagId,  
    String attrName,  
    String content)
```

---

## setInputTagAttribute

```
public void setInputTagAttribute(String refTagId,  
    String attrName,  
    String content)
```

---

## setAttributeListIfEmpty

```
public void setAttributeListIfEmpty(String refTagId,  
    Hashtable attrList)
```

---

## getTagAttribute

```
public String getTagAttribute(String refTagId,  
    String attrName)
```

---

## AddHiddenFieldToForm

```
public void AddHiddenFieldToForm(String name,  
    String value)
```

---

## getHiddens

```
public Map getHiddens()
```

---

## AddJSToForm

```
public void AddJSToForm(String jsString)
```

---

## AddHtmlTextToForm

```
public void AddHtmlTextToForm(String htmlText)
```

Adds an HTML text to the JSP page.

**Parameters:**

htmltext - the html text to add

---

## getFormTopString

```
public String getFormTopString()
```

---

(continued from last page)

---

## setBodyAttributeIfEmpty

```
public void setBodyAttributeIfEmpty(String attrName,  
    String content)
```

---

## setBodyAttribute

```
public void setBodyAttribute(String attrName,  
    String content)
```

---

## getBodyAttribute

```
public String getBodyAttribute(String attrName)
```

---

## setFormAttribute

```
public void setFormAttribute(String attrName,  
    String content)
```

---

## getFormAttribute

```
public String getFormAttribute(String attrName)
```

---

## getTagAttributes

```
public Hashtable getTagAttributes(String refTagid)
```

---

## getInputTagAttributes

```
public Hashtable getInputTagAttributes(String refTagid)
```

---

## getBodyAttributes

```
public Hashtable getBodyAttributes()
```

---

## getFormAttributes

```
public Hashtable getFormAttributes()
```

---

(continued from last page)

---

## setTagList

```
public void setTagList(String refTagId,  
    GXINameValueCollection list)
```

---

## recycleTag

```
public void recycleTag(String refTagId,  
    Hashtable tagPrivateAttributes)
```

---

## recycleBodyTag

```
public void recycleBodyTag()
```

---

## getInputStream

```
public java.io.InputStream getInputStream(String refTagId)  
    throws java.io.IOException
```

---

## getFileName

```
public String getFileName(String refTagId)  
    throws java.io.IOException
```

---

## getPanelToUpdate

```
public String getPanelToUpdate()
```

---

## getPanelToUpdateContent

```
public String getPanelToUpdateContent()
```

---

## setPanelToUpdateContent

```
public void setPanelToUpdateContent(String content)
```

---

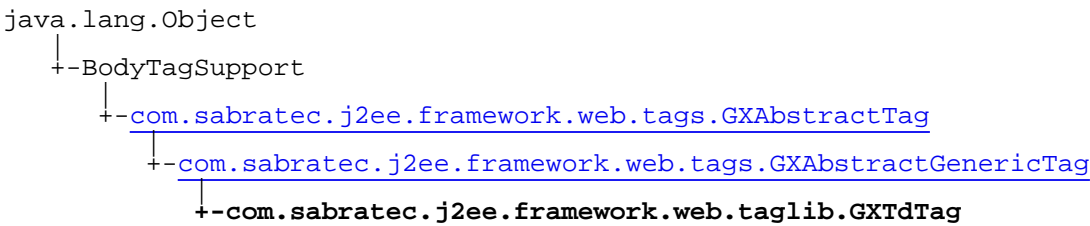
## setPageVisible

```
public void setPageVisible(boolean visible)
```

---

# com.sabratec.j2ee.framework.web.taglib

## Class GXTdTag



All Implemented Interfaces:  
java.io.Serializable

public class **GXTdTag**  
extends [GXAbstractGenericTag](#)

Constructor Summary	
public	<a href="#">GXTdTag()</a>
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag</a>	
<a href="#">setDir</a>	
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	
Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

## Constructors

**GXTdTag**  
public **GXTdTag()**

## com.sabratec.j2ee.framework.web.taglib

### Class GXTextAreaTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag
        |
        +-com.sabratec.j2ee.framework.web.taglib.GXTextAreaTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

public class **GXTextAreaTag**  
 extends [GXAbstractGenericTag](#)

### Constructor Summary

public	<a href="#">GXTextAreaTag()</a>
--------	---------------------------------

### Method Summary

void	<a href="#">setCols</a> (String cols)
void	<a href="#">setRows</a> (String rows)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag](#)

[setDir](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### GXTextAreaTag

public **GXTextAreaTag**()

(continued from last page)

## Methods

### **setCols**

```
public void setCols(String cols)
```

---

### **setRows**

```
public void setRows(String rows)
```

---

## Package

# com.sabratec.j2ee.framework.web.tags

### Overview

The purpose of this package is to provide the developer the ability to add "HTML like" tags to the JSP page. Though this is done without writing any code inside the tags, it is still possible to set dynamic contents to the tags.

This package contains tag library classes and a tag accessor which allows setting dynamic content to the HTML output of the tags.

The final HTML result of each tag is the matching HTML tag with dynamic content/attributes, added dynamically.

Each JSP page must be associated with a Java class that contains an instance of the tags accessor.

The tags accessor sets dynamic content to the tags.

The association is established by setting a root tag: `<gx:html gx_context="" >`

**Tip:** If you don't set a context class the page will work in design mode, and will not be connected to a data source (such as ApplinX) in the context class.

The HTML tag class starts the life cycle of the page:

`gx_onInit -> gx_onLoad -> Post back (submit) -> gx_preRender -> gx_onUnload`

these are all functions where you can use the object instance of the tags accessor `getTagsAccesor()` in order to pass dynamic content to each tag.

The association to each tag is established by setting an ID to each tag, and specifying the ID of the tag in the tags accessor's various methods.

### Using the Tag Library

To set a dynamic content to a tag for example, set the following code:

In `SamplePage.JSP`:

```
<gx:html gx_context="contexts.SamplePage">
```

```
...
```

```
<gx:input id=UserName cssClass=field1/>
```

In `SamplePage.java`:

```
String user = newUser; // can be retrieved from any data source.
```

```
getTagsAccesor().setTagContent(UserName, user)
```

The result will be:

```
<input id=UserName name=UserName class=field1 value=newUser/>
```

### Configuration

*Note: The following configuration is ready for use in any ApplinX application for JSP.*

Check that the tag library contains the following components:

- `gxframework.jar` should be in `WEB-INF/lib` folder.
- `GXTag.tld` should be in `WEB-INF` folder.
- `tags` folder should be in `WEB-INF/classes` including a list of Java classes for each tag type.
- `WEB-INF/web.xml` should contain the following declaration that imports the tag library:

```
<taglib>
  <taglib-uri>com/sabratec/j2ee/framework/tags</taglib-uri>
  <taglib-location>/WEB-INF/GXTags.tld</taglib-location>
</taglib>
```

- Each JSP page should have the following prefix declaration for the tag library:

```
<%@ taglib uri="com/sabratec/j2ee/framework/tags" prefix="gx" %>
```

The classes in this package are abstract in order to allow the developer to extend them easily.

For this purpose, in each new ApplinX project for JSP there is a `tags` folder in `WEB-INF/classes`, which contains Java classes.

Each class in the `tags` folder inherits from an abstract tag class in the tag library.

Associating the tags to the tags' classes is done in `GXTags.tld`.

The `tags` project classes folder is associated to the `GXTags.tld` file (and not directly to the tag library classes) in order to have immediate affect when a developer changes the relevant tag (in the project's `tags` folder).

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractAnchorTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXAnchorTag
        |
        +-com.sabratec.j2ee.framework.web.tags.GXAbstractAnchorTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

**Deprecated.** *If your class contains any customization you should inherit from:*

*com.sabratec.j2ee.framework.web.taglib.GXAnchorTag otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to:*

*com.sabratec.j2ee.framework.web.taglib.GXAnchorTag NOTE: Upgrading the web application using any other method other than the Framework Manager doesn't guaranty complete upgrade*

```

public class GXAbstractAnchorTag
extends GXAnchorTag
  
```

## Constructor Summary

public	<a href="#">GXAbstractAnchorTag()</a> <b>Deprecated.</b>
--------	---

Methods inherited from class [com.sabratec.j2ee.framework.web.taglib.GXAnchorTag](#)

[setHref](#), [setOnClick](#), [setOnmouseout](#), [setOnmouseover](#), [setOnserverclick](#)

Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXAbstractAnchorTag

```
public GXAbstractAnchorTag()
```

**Deprecated.**

(continued from last page)

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractBodyTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXAbstractBodyTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

**Deprecated.** *<gx:body> tag is not required any more is jsp pages*

public class **GXAbstractBodyTag**  
 extends [GXAbstractTag](#)

### Constructor Summary

public	<a href="#">GXAbstractBodyTag()</a> <b>Deprecated.</b>
--------	---

### Method Summary

String	<a href="#">getId()</a> <b>Deprecated.</b>
void	<a href="#">setOnload(String value)</a> <b>Deprecated.</b>

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

### Constructors

#### GXAbstractBodyTag

public **GXAbstractBodyTag()**

**Deprecated.**

(continued from last page)

## Methods

### **getId**

```
public String getId()
```

#### **Deprecated.**

Retrieves the ID of the current tag

---

### **setOnload**

```
public void setOnload(String value)
```

#### **Deprecated.**

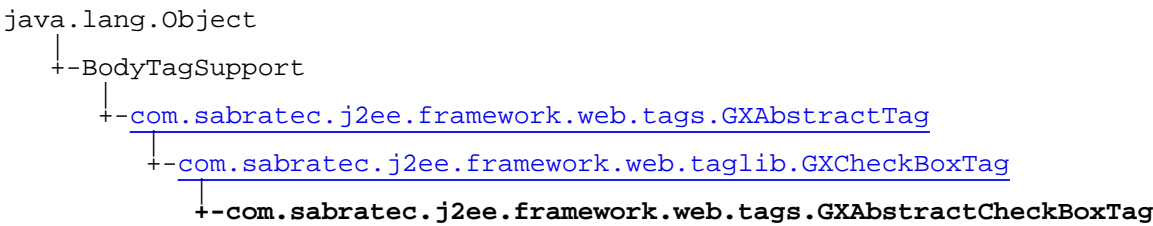
Called by the JSP page for setting the onload attribute to the HTML tag.

#### **Parameters:**

value - The title attribute value

# com.sabratec.j2ee.framework.web.tags

## Class GXAbstractCheckBoxTag



All Implemented Interfaces:  
java.io.Serializable

**Deprecated.** If your class contains any customization you should inherit from: [GXCheckBoxTag](#) otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXCheckBoxTag](#) NOTE: Upgrading the web application using any other method other than the Framework Manager doesn't guaranty complete upgrade

```
public class GXAbstractCheckBoxTag
extends GXCheckBoxTag
```

Constructor Summary	
public	<a href="#">GXAbstractCheckBoxTag()</a> Deprecated.
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.taglib.GXCheckBoxTag</a>	
<a href="#">setChecked</a> , <a href="#">setCheckedValue</a> , <a href="#">setOnClick</a> , <a href="#">setUncheckedValue</a>	
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	
Methods inherited from class java.lang.Object	
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>	

## Constructors

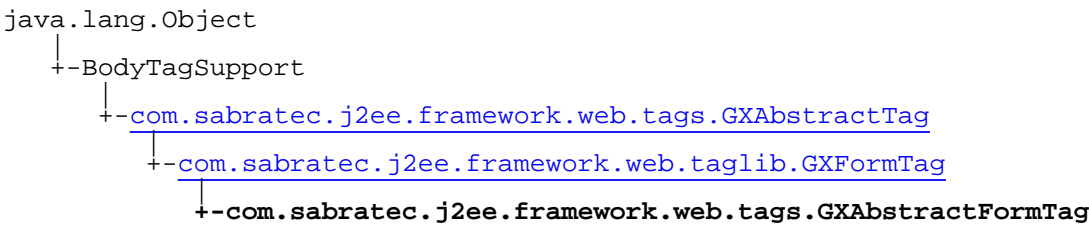
```
GXAbstractCheckBoxTag
```

```
public GXAbstractCheckBoxTag()
```

Deprecated.

# com.sabratec.j2ee.framework.web.tags

## Class GXAbstractFormTag



All Implemented Interfaces:  
java.io.Serializable

**Deprecated.** If your class contains any customization you should inherit from: [GXFormTag](#) otherwise you may delete this class. *GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXFormTag](#) NOTE: Upgrading the web application using any other method other than the Framework Manager doesn't guaranty complete upgrade*

```
public class GXAbstractFormTag
extends GXFormTag
```

Constructor Summary	
public	<a href="#">GXAbstractFormTag()</a> Deprecated.
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.taglib.GXFormTag</a>	
<a href="#">getId</a> , <a href="#">setOnsubmit</a> , <a href="#">setSupportFileUpload</a>	
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	
Methods inherited from class java.lang.Object	
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>	

## Constructors

```
GXAbstractFormTag
```

```
public GXAbstractFormTag()
```

Deprecated.

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractGenericTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXAbstractGenericTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

#### Direct Known Subclasses:

[GXTextAreaTag](#), [GXTdTag](#), [GXSpanTag](#), [GXImageTag](#), [GXDivTag](#), [GXBoldTag](#)

public abstract class **GXAbstractGenericTag**  
 extends [GXAbstractTag](#)

A generic tag that represents all label tags such as: span, div, p, font, b etc. Each label tag should implement `GXAbstractTag.getTagName()` and should be written in `WEB-INF/GXTags.tld` as a tag declaration.

The default behavior of the method `buildHtml` renders the tag name, attributes list, inner HTML, and a closing tag.

To dynamically set content to the tag, use: `getTagsAccesor().setTagContent("", "")`

To dynamically set an attribute to the tag, use: `getTagsAccesor().setTagAttribute("", "", "")`

## Constructor Summary

public	<a href="#">GXAbstractGenericTag()</a>
--------	--

## Method Summary

void	<a href="#">setDir</a> (String value)
------	---------------------------------------

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXAbstractGenericTag

public **GXAbstractGenericTag**()

(continued from last page)

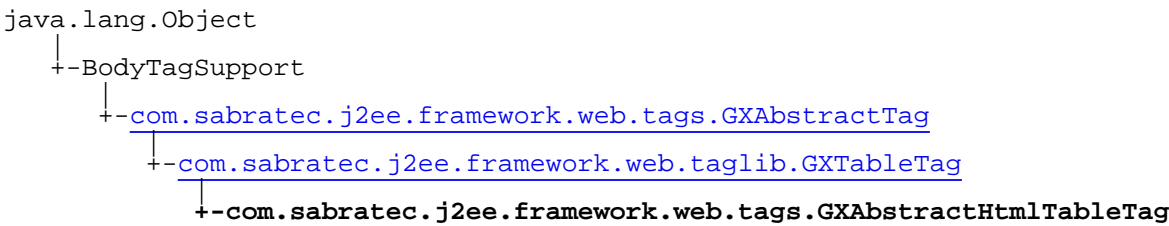
## Methods

### **setDir**

```
public void setDir(String value)
```

# com.sabratec.j2ee.framework.web.tags

## Class GXAbstractHtmlTableTag



All Implemented Interfaces:  
java.io.Serializable

**Deprecated.** If your class contains any customization you should inherit from: [GXTableTag](#) otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXTableTag](#) NOTE: Upgrading the web application using any other method other then the Framework Manager doesn't guaranty complete upgrade

```
public class GXAbstractHtmlTableTag
extends GXTableTag
```

Constructor Summary	
public	<a href="#">GXAbstractHtmlTableTag()</a> Deprecated.
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.taglib.GXTableTag</a>	
<a href="#">setBorder</a> , <a href="#">setCellpadding</a> , <a href="#">setCellspacing</a>	
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	
Methods inherited from class java.lang.Object	
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>	

## Constructors

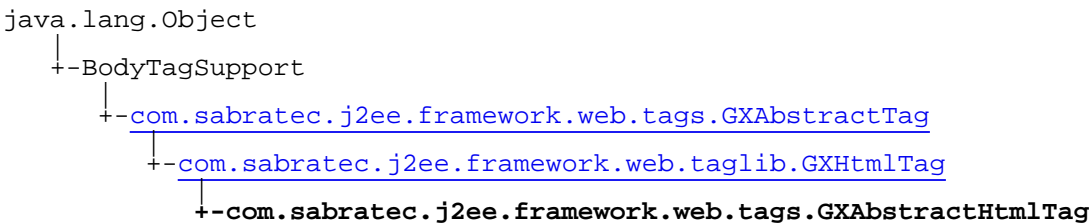
```
GXAbstractHtmlTableTag
```

```
public GXAbstractHtmlTableTag()
```

Deprecated.

# com.sabratec.j2ee.framework.web.tags

## Class GXAbstractHtmlTag



All Implemented Interfaces:  
java.io.Serializable

**Deprecated.** If your class contains any customization you should inherit from: [GXHtmlTag](#) otherwise you may delete this class. *GXTags.tld* was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXHtmlTag](#) NOTE: Upgrading the web application using any other method other than the Framework Manager doesn't guaranty complete upgrade

public class **GXAbstractHtmlTag**  
extends [GXHtmlTag](#)

Fields inherited from class <a href="#">com.sabratec.j2ee.framework.web.taglib.GXHtmlTag</a>	
<a href="#">HTML_TAG_ID</a>	

Constructor Summary	
public	<a href="#">GXAbstractHtmlTag()</a> <b>Deprecated.</b>

Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.taglib.GXHtmlTag</a>	
<a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">setGx_context</a> , <a href="#">setPageContext</a>	

Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	

Methods inherited from class java.lang.Object	
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>	

## Constructors

(continued from last page)

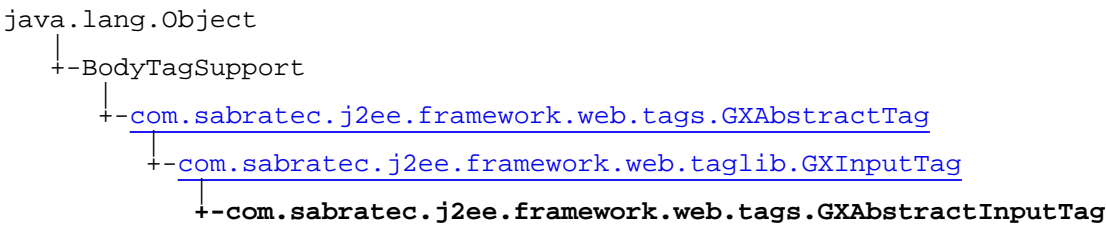
## **GXAbstractHtmlTag**

```
public GXAbstractHtmlTag()
```

**Deprecated.**

# com.sabratec.j2ee.framework.web.tags

## Class GXAbstractInputTag



All Implemented Interfaces:  
java.io.Serializable

**Deprecated.** If your class contains any customization you should inherit from: [GXInputTag](#) otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXInputTag](#) NOTE: Upgrading the web application using any other method other then the Framework Manager doesn't guaranty complete upgrade

public class **GXAbstractInputTag**  
extends [GXInputTag](#)

Constructor Summary	
public	<a href="#">GXAbstractInputTag()</a> <b>Deprecated.</b>
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.taglib.GXInputTag</a>	
<a href="#">setChecked</a> , <a href="#">setDir</a> , <a href="#">setMaxlength</a> , <a href="#">setOnchange</a> , <a href="#">setOnClick</a> , <a href="#">setOnkeydown</a> , <a href="#">setOnkeypress</a> , <a href="#">setOnkeyup</a> , <a href="#">setOnserverclick</a> , <a href="#">setSize</a> , <a href="#">setSrc</a> , <a href="#">setTabIndex</a> , <a href="#">setType</a> , <a href="#">setValue</a>	
Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	
Methods inherited from class java.lang.Object	
<a href="#">equals</a> , <a href="#">getClass</a> , <a href="#">hashCode</a> , <a href="#">notify</a> , <a href="#">notifyAll</a> , <a href="#">toString</a> , <a href="#">wait</a> , <a href="#">wait</a> , <a href="#">wait</a>	

## Constructors

**GXAbstractInputTag**  
public **GXAbstractInputTag()**  
**Deprecated.**

(continued from last page)

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractRadioButton

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXRadioButtonTag
        |
        +-com.sabratec.j2ee.framework.web.tags.GXAbstractRadioButton
  
```

#### All Implemented Interfaces:

java.io.Serializable

**Deprecated.** *If your class contains any customization you should inherit from: [GXRadioButtonTag](#) otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXRadioButtonTag](#) NOTE: Upgrading the web application using any other method other then the Framework Manager doesn't guaranty complete upgrade*

```

public class GXAbstractRadioButton
extends GXRadioButtonTag
  
```

## Constructor Summary

public	<a href="#">GXAbstractRadioButton()</a> <b>Deprecated.</b>
--------	---

#### Methods inherited from class [com.sabratec.j2ee.framework.web.taglib.GXRadioButtonTag](#)

[setChecked](#), [setOnClick](#), [setValue](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXAbstractRadioButton

```
public GXAbstractRadioButton()
```

**Deprecated.**

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractRadioButtonList

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXRadioButtonListTag
        |
        +-com.sabratec.j2ee.framework.web.tags.GXAbstractRadioButtonList
  
```

#### All Implemented Interfaces:

java.io.Serializable

**Deprecated.** *If your class contains any customization you should inherit from: [GXRadioButtonListTag](#) otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXRadioButtonListTag](#) NOTE: Upgrading the web application using any other method other than the Framework Manager doesn't guaranty complete upgrade*

public class **GXAbstractRadioButtonList**  
 extends [GXRadioButtonListTag](#)

## Constructor Summary

public	<a href="#">GXAbstractRadioButtonList()</a> <b>Deprecated.</b>
--------	---

#### Methods inherited from class [com.sabratec.j2ee.framework.web.taglib.GXRadioButtonListTag](#)

[getDir](#), [getHorizontal](#), [setBorder](#), [setCellpadding](#), [setCellspacing](#), [setDir](#), [setHorizontal](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXAbstractRadioButtonList

public **GXAbstractRadioButtonList()**

**Deprecated.**

(continued from last page)

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractSelectTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.taglib.GXSelectTag
        |
        +-com.sabratec.j2ee.framework.web.tags.GXAbstractSelectTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

**Deprecated.** *If your class contains any customization you should inherit from: [GXSelectTag](#) otherwise you may delete this class. GXTags.tld was updated during the web application upgrade process (by the Framework Manager Wizard) to point directly to: [GXSelectTag](#) NOTE: Upgrading the web application using any other method other than the Framework Manager doesn't guaranty complete upgrade*

```

public class GXAbstractSelectTag
extends GXSelectTag
  
```

## Constructor Summary

public	<a href="#">GXAbstractSelectTag()</a> <b>Deprecated.</b>
--------	---

#### Methods inherited from class [com.sabratec.j2ee.framework.web.taglib.GXSelectTag](#)

[setDir](#), [setMultiple](#), [setOnchange](#), [setOnClick](#), [setSize](#)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXAbstractSelectTag

```
public GXAbstractSelectTag()
```

**Deprecated.**

## com.sabratec.j2ee.framework.web.tags

### Class GXAbstractTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

#### Direct Known Subclasses:

[GXJspIncludeTag](#), [GXContextReleaseTag](#), [GXContextInitiatorTag](#), [GXAbstractGenericTag](#), [GXAbstractBodyTag](#), [GXTableTag](#), [GXSelectTag](#), [GXRadioButtonTag](#), [GXRadioButtonListTag](#), [GXInputTag](#), [GXHtmlTag](#), [GXFormTag](#), [GXCheckBoxTag](#), [GXAnchorTag](#)

public abstract class **GXAbstractTag**  
 extends BodyTagSupport  
 implements java.io.Serializable

This class is the abstract class for all the gx tags. It provides common functionality for all the inheriting classes such as:

- [getId\(\)](#)
- [getAttribute\(String\)](#)
- [getAttributes\(\)](#)

Each class that inherits from this class should implement two methods:

[getTagName\(\)](#) - to specify the tag name

[buildHtmlTag\(StringBuffer\)](#) - for building the HTML string of the specific tag

All the inheriting classes are initialized automatically by the JSP page, according to a specified tag in the JSP page.

For example: `<gx:TAG-NAME ...attributes...>...</gx:TAG-NAME>` and should not be initialized by the user.

The attributes are translated into set method in this class or the relevant inherited class.

The JSP tag library specification forces having a specific Set method for each tag. This class also contains a Set method for common HTML attributes: [setCssClass](#), [setStyle](#), etc.

## Constructor Summary

public	<a href="#">GXAbstractTag()</a>
--------	---------------------------------

## Method Summary

int	<a href="#">doAfterBody()</a> Method is invoked after every body evaluation to control whether the body will be reevaluated or not.
int	<a href="#">doEndTag()</a> Triggers the tag build process.
int	<a href="#">doStartTag()</a> Overrides the tag library class BodyTagSupport and performs validation and further initialization.
String	<a href="#">getId()</a> Retrieves the ID of the current tag

String	<a href="#"><u>getInnerHTML()</u></a> Used for retrieving the Inner HTML tag.
GXIWebPageContext	<a href="#"><u>getMainContext()</u></a> Used for getting the main context class for the current tag.
void	<a href="#"><u>setAlign(String value)</u></a> Called by the JSP page for setting the align attribute to the tag output html.
void	<a href="#"><u>setAttributes(String value)</u></a> Used for writing free text of attributes.
void	<a href="#"><u>setBgcolor(String value)</u></a>
void	<a href="#"><u>setCssClass(String cssClass)</u></a> Called by the JSP page for setting the css class of the gx tag.
void	<a href="#"><u>setHeight(String value)</u></a> Called by the JSP page for setting the height attribute to the tag output html.
void	<a href="#"><u>setId(String id)</u></a> Sets an ID identifier to the current tag.
void	<a href="#"><u>setOnClick(String value)</u></a> Called by the JSP page for setting the onclick attribute to the tag output html.
void	<a href="#"><u>setPageContext(PageContext pageContext)</u></a> Overrides the BodyTagSupport class, and is used for initializing the current tag.
void	<a href="#"><u>setStyle(String value)</u></a> Called by the JSP page for setting the style attribute to the tag output html.
void	<a href="#"><u>setTitle(String value)</u></a> Called by the JSP page for setting the title attribute to the tag output html.
void	<a href="#"><u>setVisible(String value)</u></a> Uses for show/hide the tag.
void	<a href="#"><u>setWidth(String value)</u></a> Called by the JSP page for setting the width attribute to the tag output html.

#### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### GXAbstractTag

```
public GXAbstractTag()
```

## Methods

(continued from last page)

---

## getInnerHTML

```
public String getInnerHTML()
```

Used for retrieving the Inner HTML tag.  
For example: <gx:span>some text</gx:span>

**Returns:**

The tag inner HTML

---

## setId

```
public void setId(String id)
```

Sets an ID identifier to the current tag.  
The set method is called automatically by the JSP page when setting the ID attribute to a gx tag.  
The ID can be used to update content for the tag using [GXITagsAccesor.setTagContent\(String, String\)](#).

---

## getId

```
public String getId()
```

Retrieves the ID of the current tag

---

## setCssClass

```
public void setCssClass(String cssClass)
```

Called by the JSP page for setting the css class of the gx tag.  
Translated into the HTML "class" attribute.

**Parameters:**

cssClass - The css class name

---

## setPageContext

```
public void setPageContext(PageContext pageContext)
```

Overrides the BodyTagSupport class, and is used for initializing the current tag.

**Parameters:**

pageContext - The JSP page context

---

## doStartTag

```
public int doStartTag()  
throws JspException
```

Overrides the tag library class BodyTagSupport and performs validation and further initialization.

**See Also:**

doStartTag()

---

## getMainContext

```
public GXIWebPageContext getMainContext()
```

Used for getting the main context class for the current tag.

---

(continued from last page)

**Returns:**

The main context class for the current tag.

---

**doEndTag**

```
public int doEndTag()  
    throws JspTagException
```

Triggers the tag build process. If the tag was set to not visible [GXITagsAccesor.setTagVisible\(String, boolean\)](#), the tag will not be rendered.

This method calls the abstract method `buildHtmlTag(StringBuffer)` in order for each inheriting class to build its HTML content.

---

**doAfterBody**

```
public int doAfterBody()  
    throws JspTagException
```

Method is invoked after every body evaluation to control whether the body will be reevaluated or not.

**Returns:**

SKIP\_BODY

---

**setAttributes**

```
public void setAttributes(String value)
```

Used for writing free text of attributes.

**Syntax:**

```
<gx:tag attributes="onmouseover='showImage();' onmouseout='hideImage();' " ...>
```

**Parameters:**

value - The free text that specifies the attributes

---

**setVisible**

```
public void setVisible(String value)
```

Uses for show/hide the tag. This method is called automatically by the JSP page, when placing a visible attribute to a tag.

Typical Usage: `<gx:tag visible="false">...`

false vaue is causing the tag not to be rendered

**Parameters:**

value - true/false if the tag should be visible.

**See Also:**

[GXITagsAccesor.setTagVisible\(String, boolean\)](#)

---

**setStyle**

```
public void setStyle(String value)
```

Called by the JSP page for setting the style attribute to the tag output html.

**Parameters:**

value - The style attributes value

---

(continued from last page)

## setTitle

```
public void setTitle(String value)
```

Called by the JSP page for setting the title attribute to the tag output html.

**Parameters:**

value - The title attribute value

---

## setOnClick

```
public void setOnClick(String value)
```

Called by the JSP page for setting the onclick attribute to the tag output html.

**Parameters:**

value - The onclick attribute value

---

## setAlign

```
public void setAlign(String value)
```

Called by the JSP page for setting the align attribute to the tag output html.

**Parameters:**

value - The align attribute value

---

## setWidth

```
public void setWidth(String value)
```

Called by the JSP page for setting the width attribute to the tag output html.

**Parameters:**

value - The width attribute value

---

## setHeight

```
public void setHeight(String value)
```

Called by the JSP page for setting the height attribute to the tag output html.

**Parameters:**

value - The height attribute value

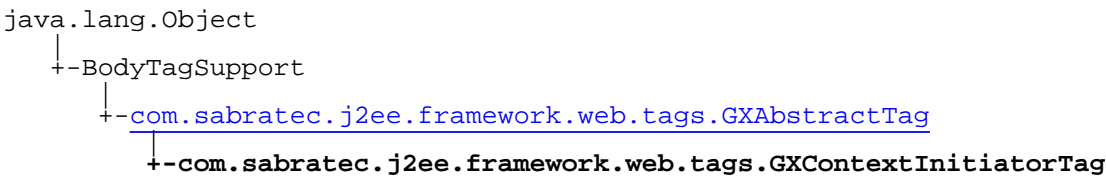
---

## setBgcolor

```
public void setBgcolor(String value)
```

# com.sabratec.j2ee.framework.web.tags

## Class GXContextInitiatorTag



All Implemented Interfaces:  
java.io.Serializable

public class **GXContextInitiatorTag**  
extends [GXAbstractTag](#)

This class is an alternative to the opening tag <gx:html> : [GXAbstractHtmlTag](#). It is required when the Web server doesn't support placing JSP Include files under a server tag.

**Syntax:**  
<gx:init gx\_context="<CONTEXT CLASS"/>  
...  
<gx:form>  
...  
</gx:form>  
...  
<gx:release/>

Constructor Summary		
	public	<a href="#">GXContextInitiatorTag()</a>

Method Summary		
	int	<a href="#">doStartTag()</a>
	void	<a href="#">setGx_context</a> (String gx_contextName)

Methods inherited from class <a href="#">com.sabratec.j2ee.framework.web.tags.GXAbstractTag</a>	
<a href="#">doAfterBody</a> , <a href="#">doEndTag</a> , <a href="#">doStartTag</a> , <a href="#">getId</a> , <a href="#">getInnerHTML</a> , <a href="#">getMainContext</a> , <a href="#">setAlign</a> , <a href="#">setAttributes</a> , <a href="#">setBgcolor</a> , <a href="#">setCssClass</a> , <a href="#">setHeight</a> , <a href="#">setId</a> , <a href="#">setOnClick</a> , <a href="#">setPageContext</a> , <a href="#">setStyle</a> , <a href="#">setTitle</a> , <a href="#">setVisible</a> , <a href="#">setWidth</a>	

Methods inherited from class java.lang.Object	
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

## Constructors

**GXContextInitiatorTag**  
public **GXContextInitiatorTag()**

(continued from last page)

## Methods

### doStartTag

```
public final int doStartTag()  
    throws JspException
```

Overrides the tag library class `BodyTagSupport` and performs validation and further initialization.

---

### setGx\_context

```
public void setGx_context(String gx_contextName)  
    throws ClassNotFoundException,  
        IllegalAccessException,  
        InstantiationException
```

## com.sabratec.j2ee.framework.web.tags

### Class GXContextReleaseTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXContextReleaseTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

public class **GXContextReleaseTag**  
 extends [GXAbstractTag](#)

This class is an alternative to the closing tag `</gx:html>` : [GXAbstractHtmlTag](#). It is required when the web server doesn't support placing JSP Include files under a server tag.

#### Syntax:

```

<gx:init gx_context="<CONTEXT CLASS"/>
...
<gx:form>
...
</gx:form>
...
<gx:release/>
  
```

## Constructor Summary

public	<a href="#">GXContextReleaseTag()</a>
--------	---------------------------------------

## Method Summary

int	<a href="#">doEndTag()</a> Method called at the end of tag.
-----	--

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

[equals](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

## Constructors

### GXContextReleaseTag

public **GXContextReleaseTag()**

## Methods

### **doEndTag**

```
public final int doEndTag()  
    throws JspTagException
```

Method called at the end of tag.

**Returns:**

EVAL\_PAGE

## com.sabratec.j2ee.framework.web.tags Interface GXITagsAccesor

All Known Implementing Classes:

[GXTagsAccesorImpl](#)

public interface **GXITagsAccesor**

This class which can be accessed from the context classes as `getTagsAccesor()`, provides various methods for updating the various gx tags.

It serves both simple tags using `setTagContent`, `setTagAttribute`, or complex tags such as combobox, radio buttons and table.

### Method Summary

void	<a href="#">AddHiddenFieldToForm</a> (String name, String value) Dynamically adds an input hidden field to the JSP page.
void	<a href="#">AddHtmlTextToForm</a> (String htmlText) Adds an HTML text to the JSP page.
void	<a href="#">AddJSToForm</a> (String jsString) Adds a JavaScript code to the JSP page.
void	<a href="#">addTableKeyColumn</a> (String tagId, String keyCol) This method is used for setting the key column(s) of a gx table tag.
String	<a href="#">getFileName</a> (String tagId) Retrieves a file name of an uploaded file according to the tag ID (<gx:input type="file" id="...>).
java.io.InputStream	<a href="#">getInputStream</a> (String tagId) Retrieves an Input Stream for an uploaded file according to the tag ID (<gx:input type="file" id="...>).
String	<a href="#">getTableName</a> () Used for retrieving the table tag ID in the current page
String	<a href="#">getTableSelectedKey</a> (String tagId) Returns the value of the selected key, as specified in <a href="#">addTableKeyColumn(String, String)</a> . If key columns were not added, the row index is returned.
String[]	<a href="#">getTableSelectedKeys</a> (String tagId) Returns the values of the selected keys, as specified in <a href="#">addTableKeyColumn(String, String)</a> .
String	<a href="#">getTagContent</a> (String tagId) Retrieves the tag content from the Request scope.
void	<a href="#">setBodyAttribute</a> (String attrName, String content) <b>Deprecated.</b> Use <a href="#">addJSToForm(String jsString)</a> - <script>document.body.<ATTR_NAME> = <ATTR_VALUE>;</script> instead
void	<a href="#">setFormAttribute</a> (String attrName, String content) Sets a dynamic attribute to <gx:form> tag.
void	<a href="#">setPageVisible</a> (boolean visible) When set to false, the page will not be rendered.

void	<a href="#"><code>setTagAttribute</code></a> (String tagId, String attrName, String content) Sets a dynamic attribute to the specified tag ID.
void	<a href="#"><code>setTagContent</code></a> (String tagId, String content) Sets content to a gx tag with the specified tag ID.
void	<a href="#"><code>setTagList</code></a> (String tagId, GXINameValueCollection list) This method provides a text-value collection to either a radio button list (<gx:radioButtonList>) or to a combo box element (<gx:select>).
void	<a href="#"><code>setTagList</code></a> (String tagId, Hashtable list) <b>Deprecated.</b> Use <code>setTagList(String tagId, com.sabratec.applinx.framework.GXINameValueCollection)</code> instead
void	<a href="#"><code>setTagTable</code></a> (String tagId, <a href="#"><code>GXITable</code></a> table)
void	<a href="#"><code>setTagTable</code></a> (String tagId, <a href="#"><code>GXITable</code></a> table, com.sabratec.j2ee.framework.web.tables.GXITableBuildConfig config, GXITableEvents events)
void	<a href="#"><code>setTagTable</code></a> (String tagId, <a href="#"><code>GXITable</code></a> table, GXITableBuildConfig config)
void	<a href="#"><code>setTagTable</code></a> (String tagId, <a href="#"><code>GXITable</code></a> table, GXITableBuildConfig config, GXITableEvents events) This method is used for filling a <gx:table> tag with the data from the data table <a href="#"><code>GXITable</code></a> .
void	<a href="#"><code>setTagVisible</code></a> (String tagId, boolean visible) When set to false, the specified tag ID will not be rendered.

## Methods

### setTagContent

```
public void setTagContent(String tagId,
    String content)
```

Sets content to a gx tag with the specified tag ID. The content is set to the relevant tag according to the tag type:

- <gx:input> - the content is set in the value attribute.
- <gx:span> (or any other label tag) - the content is set as the inner HTML.
- <gx:select> - the content is set by the option whose value is equal to the specified content.
- <gx:radioButtonList> - the content is set by the radio button whose value is equal to the specified content.
- <gx:checkbox> - the check box is checked if the content equals the checked value.

#### Parameters:

tagId - The ID of the gx tag

content - The dynamic content to set to the gx tag

### getTagContent

```
public String getTagContent(String tagId)
```

Retrieves the tag content from the Request scope. If `setTagContent` for the specified tag ID was already set it will be returned, otherwise the form/queryString value will be returned.

(continued from last page)

**Parameters:**

tagId - the content of the specified tag ID

**Returns:**

the tag's content/value

---

## setTagVisible

```
public void setTagVisible(String tagId,  
                           boolean visible)
```

When set to false, the specified tag ID will not be rendered.

**Parameters:**

tagId - The tag ID to hide

visible - true/false to determine whether to hide/show a tag

---

## setPageVisible

```
public void setPageVisible(boolean visible)
```

When set to false, the page will not be rendered.

**Parameters:**

visible - true/false to determine whether to hide/show the page

---

## setTagAttribute

```
public void setTagAttribute(String tagId,  
                             String attrName,  
                             String content)
```

Sets a dynamic attribute to the specified tag ID. When such a static attribute exists in gx tag in the JSP page, it will be overridden.

**Parameters:**

tagId - the tag ID

attrName - the name of the attribute to set

content - the content for the attribute

---

## setTagList

```
public void setTagList(String tagId,  
                       Hashtable list)
```

**Deprecated.** Use `setTagList(String tagId, com.sabratec.applinx.framework.GXINameValueCollection)` instead

---

## setTagList

```
public void setTagList(String tagId,  
                       GXINameValueCollection list)
```

This method provides a text-value collection to either a radio button list (<gx:radioButtonList>) or to a combo box element (<gx:select>). To use this method, initialize a name-value collection and add text-value pairs to it.

**Parameters:**

(continued from last page)

tagId - the tag ID of the radio button list/combo box which the dynamic list should be provided  
list - the list of text-value items

---

## setTagTable

```
public void setTagTable(String tagId,  
    GXITable table)
```

### See Also:

[setTagTable\(String, GXITable, GXITableBuildConfig, GXITableEvents\)](#)

---

## setTagTable

```
public void setTagTable(String tagId,  
    GXITable table,  
    GXITableBuildConfig config)
```

### See Also:

[setTagTable\(String, GXITable, GXITableBuildConfig, GXITableEvents\)](#)

---

## setTagTable

```
public void setTagTable(String tagId,  
    GXITable table,  
    com.sabratec.j2ee.framework.web.tables.GXITableBuildConfig config,  
    GXITableEvents events)
```

## setTagTable

```
public void setTagTable(String tagId,  
    GXITable table,  
    GXITableBuildConfig config,  
    GXITableEvents events)
```

This method is used for filling a <gx:table> tag with the data from the data table [GXITable](#).

### Parameters:

tagId - The ID of <gx:table>

table - The data table

config - a configuration class with common options of how to build the table HTML

events - a table events class which receives events on every new element creation (row,cell and a tag with a cell).

### See Also:

[GXITableBuildConfig](#)

[GXITableEvents](#)

[GXAbstractHtmlTableTag](#)

[GXIClientBaseObject.getTable\(\)](#)

---

## addTableKeyColumn

```
public void addTableKeyColumn(String tagId,  
    String keyCol)
```

(continued from last page)

This method is used for setting the key column(s) of a gx table tag. Setting a key column is needed when you are interested to know the key column value when a row is selected. For example, if you have collected a multiple screens table with an item number column, and you would like to show details of an item (according to the item number) when clicking on a button in this row.

If you will use `addTableKeyColumn("TABLE TAG ID","ItemNumber")` before the table is displayed, when you will select a row and post back to the context class, you will be able to retrieve the selected item number using [`getTableSelectedKey\(String\)`](#) and use it in order to find and drill down to the relevant row.

**Parameters:**

tagId - The table tag ID

keyCol - The key column name in [`GXITable.getColumns\(\)`](#)

---

## getTableTagName

```
public String getTableTagName()
```

Used for retrieving the table tag ID in the current page

**Returns:**

The last built table name

---

## getTableSelectedKey

```
public String getTableSelectedKey(String tagId)
```

Returns the value of the selected key, as specified in [`addTableKeyColumn\(String, String\)`](#). If key columns were not added, the row index is returned.

**Parameters:**

tagId - The table tag ID

**Returns:**

The key column value

---

## getTableSelectedKeys

```
public String[] getTableSelectedKeys(String tagId)
```

Returns the values of the selected keys, as specified in [`addTableKeyColumn\(String, String\)`](#). Use this method when you specify more then one key column in the method `addTableKeyColumn`. If key columns were not added, the row index is returned.

**Parameters:**

tagId - The table tag ID

**Returns:**

A String array of the key columns values

---

## AddHiddenFieldToForm

```
public void AddHiddenFieldToForm(String name,  
    String value)
```

Dynamically adds an input hidden field to the JSP page.

**Parameters:**

name - The name of the hidden field

value - The value of the hidden field

---

(continued from last page)

---

## AddJSToForm

```
public void AddJSToForm(String jsString)
```

Adds a JavaScript code to the JSP page. The script should be wrapped with <script>...</script> tag.

**Parameters:**

jsString - The JavaScript command(s)

---

## AddHtmlTextToForm

```
public void AddHtmlTextToForm(String htmlText)
```

Adds an HTML text to the JSP page.

**Parameters:**

htmltext - the html text to add

---

## setBodyAttribute

```
public void setBodyAttribute(String attrName,  
    String content)
```

**Deprecated.** Use `addJSToForm(String jsString) - <script>document.body.<ATTR_NAME> = <ATTR_VALUE>;</script>` instead

---

## setFormAttribute

```
public void setFormAttribute(String attrName,  
    String content)
```

Sets a dynamic attribute to <gx:form> tag.

**Parameters:**

attrName - The name of the attribute  
content - The content of the attribute

---

## getInputStream

```
public java.io.InputStream getInputStream(String tagId)  
    throws java.io.IOException
```

Retrieves an Input Stream for an uploaded file according to the tag ID (<gx:input type="file" id="...>).

**Parameters:**

tagId - the input file tag ID

**Returns:**

Input Stream of the uploaded file

---

## getFileName

```
public String getFileName(String tagId)  
    throws java.io.IOException
```

---

(continued from last page)

Retrieves a file name of an uploaded file according to the tag ID (`<gx:input type="file" id="...">`).

**Parameters:**

`tagId` - The input file tag ID

**Returns:**

The file name

## com.sabratec.j2ee.framework.web.tags

### Class GXJspIncludeTag

```

java.lang.Object
  |
  +-BodyTagSupport
    |
    +-com.sabratec.j2ee.framework.web.tags.GXAbstractTag
      |
      +-com.sabratec.j2ee.framework.web.tags.GXJspIncludeTag
  
```

#### All Implemented Interfaces:

java.io.Serializable

```

public class GXJspIncludeTag
extends GXAbstractTag
  
```

This class is the root of a JSP Include file and the tag that initializes and activates the JSP Include context class.

#### Syntax:

```

<gx:include gx_context="<CONTEXT CLASS>">
...
<img src=...
...
</gx:include>
  
```

This class throws the following events in the context class:

- gx\_onLoad - informs about the Load event of the Include file - `gx_onLoad()`
- gx\_preRender - informs before the HTML of the Include file is rendered - `gx_preRender()`
- gx\_onUnload - informs of the Unload event (for cleanup) - `gx_onUnload()`
- gx\_onError - Used for error handling - `gx_onError(RuntimeException())`

## Constructor Summary

public	<a href="#">GXJspIncludeTag()</a>
--------	-----------------------------------

## Method Summary

int	<a href="#">doEndTag()</a> Method called at the end of the tag.
int	<a href="#">doStartTag()</a> Method called at the beginning of the tag.
void	<a href="#">setGx_context</a> (String gx_contextName)
void	<a href="#">setPageContext</a> (PageContext pageContext)

#### Methods inherited from class [com.sabratec.j2ee.framework.web.tags.GXAbstractTag](#)

[doAfterBody](#), [doEndTag](#), [doStartTag](#), [getId](#), [getInnerHTML](#), [getMainContext](#), [setAlign](#), [setAttributes](#), [setBgcolor](#), [setCssClass](#), [setHeight](#), [setId](#), [setOnClick](#), [setPageContext](#), [setStyle](#), [setTitle](#), [setVisible](#), [setWidth](#)

#### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructors

### **GXJspIncludeTag**

```
public GXJspIncludeTag()
```

## Methods

### **setPageContext**

```
public void setPageContext(PageContext pageContext)
```

Overrides the BodyTagSupport class, and is used for initializing the current tag.

### **doStartTag**

```
public final int doStartTag()  
    throws JspException
```

Method called at the beginning of the tag.

**Returns:**

EVAL\_BODY\_TAG

### **doEndTag**

```
public final int doEndTag()  
    throws JspTagException
```

Method called at the end of the tag.

**Returns:**

EVAL\_PAGE

### **setGx\_context**

```
public void setGx_context(String gx_contextName)  
    throws ClassNotFoundException,  
        IllegalAccessException,  
        InstantiationException
```

## com.sabratec.j2ee.framework.web.tags

### Class GXTagFunctions

java.lang.Object

└-com.sabratec.j2ee.framework.web.tags.GXTagFunctions

public class **GXTagFunctions**  
extends Object

Internal Util class for common tasks of the package.

### Constructor Summary

public	<a href="#">GXTagFunctions()</a>
--------	----------------------------------

### Method Summary

static void	<a href="#">addAttributesToTag</a> ( <a href="#">GXITag</a> tag, Hashtable attributes)
static String	<a href="#">hashtableToNameValueString</a> (Hashtable attributes)
static void	<a href="#">renderAndRecycle</a> ( <a href="#">GXITag</a> tag, StringBuffer result)
static void	<a href="#">writeComplexRadio</a> (StringBuffer html, GXIWebContext gx_context, String tagId, String isHorizontal, String dir, String innerHTML, String val, String isChecked, Hashtable attributes)
static void	<a href="#">writeRadio</a> (StringBuffer html, String tagId, GXIWebContext gx_context, String radioVal, String isChecked, Hashtable attributes)

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructors

#### GXTagFunctions

public **GXTagFunctions**()

### Methods

(continued from last page)

---

## writeComplexRadio

```
public static void writeComplexRadio(StringBuffer html,
    GXIWebContext gx_context,
    String tagId,
    String isHorizontal,
    String dir,
    String innerHTML,
    String val,
    String isChecked,
    Hashtable attributes)
```

---

## writeRadio

```
public static void writeRadio(StringBuffer html,
    String tagId,
    GXIWebContext gx_context,
    String radioVal,
    String isChecked,
    Hashtable attributes)
```

---

## hashtableToNameValueString

```
public static String hashtableToNameValueString(Hashtable attributes)
```

---

## addAttributesToTag

```
public static void addAttributesToTag(GXITag tag,
    Hashtable attributes)
```

---

## renderAndRecycle

```
public static void renderAndRecycle(GXITag tag,
    StringBuffer result)
```

---

**Package**  
**com.sabratec.util**

## com.sabratec.util Class GXPosition

java.lang.Object

└─com.sabratec.util.GXPosition

### All Implemented Interfaces:

Cloneable, java.io.Serializable

public class **GXPosition**  
 extends Object  
 implements java.io.Serializable, Cloneable

This class represents a position on the host screen. Each position is described by its coordinates (row and column).

This class is immutable.

### Field Summary

public static	<a href="#">DEFAULT</a>
---------------	-------------------------

### Constructor Summary

public	<a href="#">GXPosition()</a> Ctor for dotNet Xstream.Serializable
public	<a href="#">GXPosition(int row, int column)</a> Creates a new position.

### Method Summary

Object	<a href="#">clone()</a>
int	<a href="#">compareTo(Object otherPosition)</a> Position A is considered to be greater than position B if it is either in a greater row, or in the same row and with a greater column.
boolean	<a href="#">equals(Object otherPosition)</a> Tests if this position is equal to the given position. Two positions are considered equal if they are in the same row and column.
int	<a href="#">getColumn()</a> Returns the column value set for this position.
int	<a href="#">getRow()</a> Returns the row value set for this position
int	<a href="#">hashCode()</a>
String	<a href="#">toString()</a>

### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Fields

### DEFAULT

```
public static com.sabratec.util.GXPosition DEFAULT
```

## Constructors

### GXPosition

```
public GXPosition()
```

Ctor for dotNet Xstream.Serializable

### GXPosition

```
public GXPosition(int row,  
                  int column)
```

Creates a new position.

**Parameters:**

row - the row of the position.

column - the column of the position

## Methods

### getRow

```
public int getRow()
```

Returns the row value set for this position

**Returns:**

the row value set for this position

### getColumn

```
public int getColumn()
```

Returns the column value set for this position.

**Returns:**

the column value set for this position

### equals

```
public boolean equals(Object otherPosition)
```

Tests if this position is equal to the given position.

Two positions are considered equal if they are in the same row and column.

(continued from last page)

**Parameters:**

`otherPosition` - another position object

**Returns:**

Tests if this position is equal to the given position

---

**compareTo**

```
public int compareTo(Object otherPosition)
```

Position A is considered to be greater than position B if it is either in a greater row, or in the same row and with a greater column.

**Returns:**

-1 if this position is smaller than the given position

1 if this position is greater than the given position  
and 0 if the positions are equal

---

**toString**

```
public String toString()
```

---

**hashCode**

```
public int hashCode()
```

---

**clone**

```
public Object clone()
```

---

---

# Package

# com.sabratec.util.ftp

This package supplies an API for the ApplinX FTP mechanism.

It can be used instead of using the FTP dialog screens.

The main class in the package is GXFtpManager.

This is a singleton class and it can be accessed only via the `instance()` method.

Another important class is GXFtpRequest, which is a data structure that contains all the relevant information for the FTP transmission.

**Below are two code examples of uploading and downloading files:**

**Code sample for uploading a file to a Mainframe host:**

```
GXFtpRequest uploadReq = new GXFtpRequest();
uploadReq.setUser(user);
uploadReq.setPassword(pass);
uploadReq.setServer(host ip address);
uploadReq.setRemoteFile(remote file using full path);
uploadReq.setValue(GXMFCCommands.CMD_MF_DATA_TYPE, GXMFCCommands.CMD_VAL_MF_DATA_TYPE_BIN)
try {
uploadReq.readFileToUpload(local file using full path);
GXFtpManager.instance().putFile(uploadReq);
} catch (GXGeneralException e) {
e.printStackTrace();
}
```

**Code sample for downloading a file from a Mainframe host:**

```
GXFtpRequest downloadReq = new GXFtpRequest();
downloadReq.setUser(user);
downloadReq.setPassword(pass);
downloadReq.setServer(host ip address);
downloadReq.setRemoteFile(remote file using full path);
downloadReq.setValue(GXMFCCommands.CMD_MF_DATA_TYPE, GXMFCCommands.CMD_VAL_MF_DATA_TYPE_BIN);
try {
GXFtpResponse resp = GXFtpManager.instance().getFile(downloadReq);
resp.writeFileToDisk(local file using full path);
} catch (GXFtpException e) {
e.getMessage();
}
```

## com.sabratec.util.ftp Class GXFtpConsts

java.lang.Object

└--com.sabratec.util.ftp.GXFtpConsts

public class **GXFtpConsts**  
extends Object

### Field Summary

public static final	<a href="#">ADDITIONAL</a> Value: <b>ADDITIONAL</b>
public static final	<a href="#">DATA</a> Value: <b>DATA</b>
public static final	<a href="#">LOCAL_FILE</a> Value: <b>LOCAL_FILE</b>
public static final	<a href="#">MODE</a> Value: <b>MODE</b>
public static final	<a href="#">PASSWORD</a> Value: <b>PASSWORD</b>
public static final	<a href="#">REMOTE_FILE</a> Value: <b>REMOTE_FILE</b>
public static final	<a href="#">SERVER</a> Value: <b>SERVER</b>
public static final	<a href="#">USER</a> Value: <b>USER</b>

### Constructor Summary

public	<a href="#">GXFtpConsts()</a>
--------	-------------------------------

#### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Fields

(continued from last page)

---

## SERVER

```
public static final java.lang.String SERVER
```

Constant value: **SERVER**

---

## USER

```
public static final java.lang.String USER
```

Constant value: **USER**

---

## PASSWORD

```
public static final java.lang.String PASSWORD
```

Constant value: **PASSWORD**

---

## LOCAL\_FILE

```
public static final java.lang.String LOCAL_FILE
```

Constant value: **LOCAL\_FILE**

---

## REMOTE\_FILE

```
public static final java.lang.String REMOTE_FILE
```

Constant value: **REMOTE\_FILE**

---

## ADDITIONAL

```
public static final java.lang.String ADDITIONAL
```

Constant value: **ADDITIONAL**

---

## MODE

```
public static final java.lang.String MODE
```

Constant value: **MODE**

---

## DATA

```
public static final java.lang.String DATA
```

Constant value: **DATA**

## Constructors

(continued from last page)

## **GXFtpConsts**

```
public GXFtpConsts()
```

## com.sabratec.util.ftp

### Class GXFtpException

```

java.lang.Object
  |
  +-GXGeneralException
    |
    +-com.sabratec.util.ftp.GXFtpException
  
```

```

public class GXFtpException
extends GXGeneralException
  
```

Exception thrown when unexpected or negative reply is received from the server. The `getReply()` method can be used to examine the reply that caused the exception.

### Constructor Summary

public	<a href="#">GXFtpException</a> (GXFtpReply reply, String msg, Throwable e) Constructs new GXFtpException with short explanation of the circumstance.
public	<a href="#">GXFtpException</a> (String msg) Constructs new GXFtpException with short explanation of the circumstance.
public	<a href="#">GXFtpException</a> (GXFtpReply reply) Convenience constructor; Creates new GXFtpException using "Unexpected" as the explanation.

### Method Summary

int	<a href="#">getCode</a> () Convenience method; gets FTP reply code of underlying reply object.
GXFtpReply	<a href="#">getReply</a> () Returns reply that caused this exception.
int	<a href="#">getType</a> () Convenience method; gets FTP reply type of underlying reply object.

### Methods inherited from class java.lang.Object

`equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

## Constructors

### GXFtpException

```

public GXFtpException(GXFtpReply reply,
                      String msg,
                      Throwable e)
  
```

Constructs new GXFtpException with short explanation of the circumstance. The explanation is prepended to the text of the actual reply to form the exception message.

(continued from last page)

## GXFtpException

```
public GXFtpException(String msg)
```

Constructs new GXFtpException with short explanation of the circumstance. Useful when there is still no reply from the server

---

## GXFtpException

```
public GXFtpException(GXFtpReply reply)
```

Convenience constructor; Creates new GXFtpException using "Unexpected" as the explanation.

## Methods

### getReply

```
public GXFtpReply getReply()
```

Returns reply that caused this exception.

**Returns:**

GXFtpReply object that caused exception

---

### getType

```
public int getType()
```

Convenience method; gets FTP reply type of underlying reply object. Equivalent to calling getReply().getType().

**Returns:**

type of GXFtpReply object that caused exception

**See Also:**

getType

---

### getCode

```
public int getCode()
```

Convenience method; gets FTP reply code of underlying reply object. Equivalent to calling getReply().getCode().

**Returns:**

FTP reply code of GXFtpReply object that caused exception

**See Also:**

getCode

## com.sabratec.util.ftp Class GXFtpManager

java.lang.Object

└─com.sabratec.util.ftp.GXFtpManager

public class **GXFtpManager**  
extends Object

This class supplies a batch API for the ApplinX FTP mechanism.

All the data for an ftp transmission is stored in a GXFtpRequest object.

An GXFtpRequest object can be passed as a parameter to two different methods of GXFtpManager: getFile() and putFile()

These methods can be called only via GXFtpManager singleton instance.

Two code examples of uploading and downloading files can be found in the package overview.

### Method Summary

<a href="#">GXFtpResponse</a>	<a href="#">getFile(GXFtpRequest req)</a> getFile is an API method which is used to retrieve a file from a remote host using the FTP protocol.
static <a href="#">GXFtpManager</a>	<a href="#">instance()</a> instance() supplies an access to the class object (singleton). The object is created automatically in the background.
void	<a href="#">putFile(GXFtpRequest req)</a> putFile is an api method which transfers a file to a remote host using the FTP protocol.

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Methods

### instance

public static [GXFtpManager](#) **instance()**

instance() supplies an access to the class object (singleton).  
The object is created automatically in the background.

#### Returns:

The single instance of the class.

### putFile

public void **putFile**([GXFtpRequest](#) req)  
throws [GXFtpException](#)

putFile is an api method which transfers a file to a remote host using the FTP protocol.

#### Throws:

[GXFtpException](#)

## getFile

```
public GXFtpResponse getFile(GXFtpRequest req)  
    throws GXFtpException
```

getFile is an API method which is used to retrieve a file from a remote host using the FTP protocol.

**Returns:**

GXFtpResponse

**Throws:**

[GXFtpException](#)

## com.sabratec.util.ftp Class GXFtpRequest

java.lang.Object

└─com.sabratec.util.ftp.GXFtpRequest

public class **GXFtpRequest**  
extends Object

This class is used as a data structure for all the parameters which are relevant for an FTP transmission. Some of the data is stored in the object by designated methods, e.g.: `setUser()`. The rest of the data is stored using the `setValue()` method and pre defined constants such as:  
`setValue(GXMFCCommands.CMD_MF_DATA_TYPE, GXMFCCommands.CMD_VAL_MF_DATA_TYPE_BIN);`

An initialized object of this class can be passed as a parameter to each of the following GXFtpManager methods:

```
GXFtpManager.instance().putFile(GXFtpRequest);
GXFtpManager.instance().getFile(GXFtpRequest);
```

### Constructor Summary

public	<a href="#">GXFtpRequest()</a> Class constructor
--------	---

### Method Summary

String	<a href="#">getAdditionalCommand()</a> For internal use
byte[]	<a href="#">getLocalFile()</a> For internal use
byte[]	<a href="#">getLocalFileBytes()</a> For internal use
String	<a href="#">getPassword()</a> For internal use
String	<a href="#">getRemoteFile()</a> For internal use
String	<a href="#">getServer()</a> For internal use
String	<a href="#">getUser()</a> For internal use
String	<a href="#">getValue()</a> (String key) For internal use
String	<a href="#">getValues()</a> For internal use
void	<a href="#">readFileToUpload()</a> (String filename) Reads and stores the content of the local file into the request data structure.

void	<a href="#"><code>setAdditional</code></a> (String additionalCommands) Sets an additional user defined command for the FTP request.
void	<a href="#"><code>setLocalFileBytes</code></a> (byte[] fileByBytes) For internal use
void	<a href="#"><code>setPassword</code></a> (String password) Sets the remote host FTP password
void	<a href="#"><code>setRemoteFile</code></a> (String remoteFile) Sets the remote file name using the full path.
void	<a href="#"><code>setServer</code></a> (String server) Sets the FTP remote server IP address.
void	<a href="#"><code>setUser</code></a> (String user) Sets the FTP server user name.
void	<a href="#"><code>setValue</code></a> (String key, String value) Stores command values in the FTP request.

#### Methods inherited from class `java.lang.Object`

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

## Constructors

### GXFtpRequest

```
public GXFtpRequest()
```

Class constructor

## Methods

### getValue

```
public String getValue(String key)
```

For internal use

### getValues

```
public String getValues()
```

For internal use

### setValue

```
public void setValue(String key,  
                    String value)
```

Stores command values in the FTP request.

For example:`setValue(GXMFCCommands.CMD_MF_DATA_TYPE, GXMFCCommands.CMD_VAL_MF_DATA_TYPE_BIN);`

---

(continued from last page)

**Parameters:**

key - command constant which can be either from the [GXMFCommands](#) class or from the [GXAS400Commands](#) class.

value - The cmd value, which can be a constant either from one of these classes ([GXMFCommands](#) or [GXAS400Commands](#)) or a positive number.

---

**getPassword**

```
public String getPassword()
```

For internal use

---

**setPassword**

```
public void setPassword(String password)
```

Sets the remote host FTP password

**Parameters:**

password - host password.

---

**getRemoteFile**

```
public String getRemoteFile()
```

For internal use

---

**setRemoteFile**

```
public void setRemoteFile(String remoteFile)
```

Sets the remote file name using the full path.

**Parameters:**

remoteFile - The full path name of the remote name.

---

**getServer**

```
public String getServer()
```

For internal use

---

**setServer**

```
public void setServer(String server)
```

Sets the FTP remote server IP address.

**Parameters:**

server

---

**getUser**

```
public String getUser()
```

For internal use

---

---

## setUser

```
public void setUser(String user)
```

Sets the FTP server user name.

### Parameters:

user

---

## setLocalFileBytes

```
public void setLocalFileBytes(byte[] fileByBytes)
```

For internal use

---

## getLocalFileBytes

```
public byte[] getLocalFileBytes()
```

For internal use

---

## getLocalFile

```
public byte[] getLocalFile()
```

For internal use

---

## setAdditional

```
public void setAdditional(String additionalCommands)
```

Sets an additional user defined command for the FTP request.

### Parameters:

additionalCommands

---

## getAdditionalCommand

```
public String getAdditionalCommand()
```

For internal use

---

## readFileToUpload

```
public void readFileToUpload(String filename)  
    throws GXGeneralException,  
           java.io.IOException
```

Reads and stores the content of the local file into the request data structure. Should be used before uploading a file.

### Parameters:

filename

### Throws:

GXGeneralException  
IOException

---

## com.sabratec.util.ftp

### Class GXFtpResponse

java.lang.Object

└─com.sabratec.util.ftp.GXFtpResponse

public class **GXFtpResponse**  
extends Object

This class holds the data retrieved from an FTP download request.

### Constructor Summary

public	<a href="#">GXFtpResponse</a> (String fileName, byte[] arr) Constructor - For internal use
--------	---

### Method Summary

void	<a href="#">convertTextIfNeeded</a> (String mode) Converts the downloaded file to Windows/DOS/UNIX file format.
byte[]	<a href="#">getFileAsByteArr</a> ()
java.io.ByteArrayInputStream	<a href="#">getFileAsInputStream</a> () For internal use
String	<a href="#">getFileName</a> () For internal use

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructors

### GXFtpResponse

public **GXFtpResponse**(String fileName,  
byte[] arr)

Constructor - For internal use

## Methods

### getFileAsByteArr

public byte[] **getFileAsByteArr**()

(continued from last page)

## **getFileAsStream**

```
public java.io.ByteArrayInputStream getFileAsStream()
```

For internal use

---

## **getFileName**

```
public String getFileName()
```

For internal use

---

## **convertTextIfNeeded**

```
public void convertTextIfNeeded(String mode)
```

Converts the downloaded file to Windows/DOS/UNIX file format.

### **Parameters:**

mode - possible values: "win", "dos" or "unix"

## com.sabratec.util.ftp Class GXFtpTypes

java.lang.Object

└--com.sabratec.util.ftp.GXFtpTypes

public class **GXFtpTypes**  
extends Object

### Field Summary

public static final	<a href="#">AS400</a> Value: 2
public static final	<a href="#">MAIN_FRAME</a> Value: 1
public static final	<a href="#">SIMPLE</a> Value: 0

### Constructor Summary

public	<a href="#">GXFtpTypes()</a>
--------	------------------------------

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### **SIMPLE**

public static final java.lang.String **SIMPLE**

Constant value: 0

### **MAIN\_FRAME**

public static final java.lang.String **MAIN\_FRAME**

Constant value: 1

### **AS400**

public static final java.lang.String **AS400**

(continued from last page)

Constant value: **2**

## Constructors

### **GXFtpTypes**

```
public GXFtpTypes()
```

---

**Package**

**com.sabratec.util.ftp.as400**

## com.sabratec.util.ftp.as400 Class GXAS400Commands

java.lang.Object

└─com.sabratec.util.ftp.as400.GXAS400Commands

public class **GXAS400Commands**  
extends Object

This class holds constants for the AS400 FTP commands and their possible values:

AS400 designated commands	Possible values	Description
CMD_AS400_DATA_TYPE	CMD_VAL_AS400_DATA_TYPE_ASC	ASCII
	CMD_VAL_AS400_DATA_TYPE_EBC	EBCDIC
	CMD_VAL_AS400_DATA_TYPE_BIN	Binary
	CMD_VAL_AS400_DATA_TYPE_DBCS_ASC	ASCII double byte character set
	CMD_VAL_AS400_DATA_TYPE_DBCS_EBC	EBCDIC double byte character set
	CMD_VAL_AS400_DATA_TYPE_CCSID	Coded character set identifier
CMD_AS400_MODE	CMD_VAL_AS400_MODE_BLOCK	
	CMD_VAL_AS400_MODE_STREAM	
CMD_AS400_DS	CMD_VAL_AS400_DS_FILE	Data structure file
	CMD_VAL_AS400_DS_FILE	Data structure record

### Field Summary

public static final	<a href="#">CMD_AS400_DATA_TYPE</a> Value: <b>type</b>
public static final	<a href="#">CMD_AS400_DS</a> Value: <b>struct</b>
public static final	<a href="#">CMD_AS400_MODE</a> Value: <b>mode</b>
public static final	<a href="#">CMD_VAL_AS400_DATA_TYPE_ASC</a> Value: <b>a</b>
public static final	<a href="#">CMD_VAL_AS400_DATA_TYPE_BIN</a> Value: <b>i</b>
public static final	<a href="#">CMD_VAL_AS400_DATA_TYPE_CCSID</a> Value: <b>c</b>

public static final	<a href="#">CMD_VAL_AS400_DATA_TYPE_DBCS_ASC</a> Value: <b>b</b>
public static final	<a href="#">CMD_VAL_AS400_DATA_TYPE_DBCS_EBC</a> Value: <b>f</b>
public static final	<a href="#">CMD_VAL_AS400_DATA_TYPE_EBC</a> Value: <b>e</b>
public static final	<a href="#">CMD_VAL_AS400_DS_FILE</a> Value: <b>f</b>
public static final	<a href="#">CMD_VAL_AS400_DS_RECORD</a> Value: <b>r</b>
public static final	<a href="#">CMD_VAL_AS400_MODE_BLOCK</a> Value: <b>b</b>
public static final	<a href="#">CMD_VAL_AS400_MODE_STREAM</a> Value: <b>s</b>

## Constructor Summary

public	<a href="#">GXAS400Commands()</a>
--------	-----------------------------------

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### CMD\_AS400\_DATA\_TYPE

public static final java.lang.String **CMD\_AS400\_DATA\_TYPE**

Constant value: **type**

### CMD\_VAL\_AS400\_DATA\_TYPE\_ASC

public static final java.lang.String **CMD\_VAL\_AS400\_DATA\_TYPE\_ASC**

Constant value: **a**

### CMD\_VAL\_AS400\_DATA\_TYPE\_EBC

public static final java.lang.String **CMD\_VAL\_AS400\_DATA\_TYPE\_EBC**

Constant value: **e**

---

## **CMD\_VAL\_AS400\_DATA\_TYPE\_BIN**

```
public static final java.lang.String CMD_VAL_AS400_DATA_TYPE_BIN
```

Constant value: **i**

---

## **CMD\_VAL\_AS400\_DATA\_TYPE\_DBCS\_ASC**

```
public static final java.lang.String CMD_VAL_AS400_DATA_TYPE_DBCS_ASC
```

Constant value: **b**

---

## **CMD\_VAL\_AS400\_DATA\_TYPE\_DBCS\_EBC**

```
public static final java.lang.String CMD_VAL_AS400_DATA_TYPE_DBCS_EBC
```

Constant value: **f**

---

## **CMD\_VAL\_AS400\_DATA\_TYPE\_CCSID**

```
public static final java.lang.String CMD_VAL_AS400_DATA_TYPE_CCSID
```

Constant value: **c**

---

## **CMD\_AS400\_MODE**

```
public static final java.lang.String CMD_AS400_MODE
```

Constant value: **mode**

---

## **CMD\_VAL\_AS400\_MODE\_STREAM**

```
public static final java.lang.String CMD_VAL_AS400_MODE_STREAM
```

Constant value: **s**

---

## **CMD\_VAL\_AS400\_MODE\_BLOCK**

```
public static final java.lang.String CMD_VAL_AS400_MODE_BLOCK
```

Constant value: **b**

---

## **CMD\_AS400\_DS**

```
public static final java.lang.String CMD_AS400_DS
```

Constant value: **struct**

---

(continued from last page)

## **CMD\_VAL\_AS400\_DS\_FILE**

```
public static final java.lang.String CMD_VAL_AS400_DS_FILE
```

Constant value: **f**

---

## **CMD\_VAL\_AS400\_DS\_RECORD**

```
public static final java.lang.String CMD_VAL_AS400_DS_RECORD
```

Constant value: **r**

## Constructors

### **GXAS400Commands**

```
public GXAS400Commands()
```

---

**Package**

**com.sabratec.util.ftp.mainframe**

## com.sabratec.util.ftp.mainframe

### Class GXMFCCommands

```
java.lang.Object
```

```
└--com.sabratec.util.ftp.mainframe.GXMFCCommands
```

```
public class GXMFCCommands
extends Object
```

This class holds constants for the Mainframe FTP commands and their possible values:

MF designated commands	Possible values	Description
CMD_MF_RECORD_FORMAT	CMD_VAL_MF_RECORD_FORMAT_FIXED	Fixed length block
	CMD_VAL_MF_RECORD_FORMAT_VB	Variable length
CMD_MF_DATA_TYPE	CMD_VAL_MF_DATA_TYPE_ASC	ASCII
	CMD_VAL_MF_DATA_TYPE_BIN	Binary
	CMD_VAL_MF_DATA_TYPE_EBCDIC	EBCDIC
	CMD_VAL_MF_DATA_TYPE_DBCS	Double byte character set
	CMD_VAL_MF_DATA_TYPE_UNICODE_2L	Unicode
	CMD_VAL_MF_DATA_TYPE_UNICODE_2B	Unicode
CMD_MF_BLOCKSIZE	#number of bytes	Block size
CMD_MF_PRIMARY	#number of tracks	Primary space
CMD_MF_SECONDARY	#number of tracks	Secondary space
CMD_MF_LRECL	#number	Logical record length

### Field Summary

public static final	<a href="#">CMD_MF_BLOCKSIZE</a> Value: <b>site blocksize=</b>
public static final	<a href="#">CMD_MF_DATA_TYPE</a> Value: <b>type</b>
public static final	<a href="#">CMD_MF_LRECL</a> Value: <b>site lrecl=</b>
public static final	<a href="#">CMD_MF_PRIMARY</a> Value: <b>site primary=</b>
public static final	<a href="#">CMD_MF_RECORD_FORMAT</a> Value: <b>site recfm=</b>
public static final	<a href="#">CMD_MF_SECONDARY</a> Value: <b>site secondary=</b>

public static final	<a href="#">CMD_VAL_MF_DATA_TYPE_ASC</a> Value: <b>a</b>
public static final	<a href="#">CMD_VAL_MF_DATA_TYPE_BIN</a> Value: <b>i</b>
public static final	<a href="#">CMD_VAL_MF_DATA_TYPE_DBCS</a> Value: <b>b</b>
public static final	<a href="#">CMD_VAL_MF_DATA_TYPE_EBCDIC</a> Value: <b>e</b>
public static final	<a href="#">CMD_VAL_MF_DATA_TYPE_UNICODE_2B</a> Value: <b>u 2 b</b>
public static final	<a href="#">CMD_VAL_MF_DATA_TYPE_UNICODE_2L</a> Value: <b>u 2 l</b>
public static final	<a href="#">CMD_VAL_MF_RECORD_FORMAT_FIXED</a> Value: <b>FB</b>
public static final	<a href="#">CMD_VAL_MF_RECORD_FORMAT_VB</a> Value: <b>VB</b>

## Constructor Summary

public	<a href="#">GXMFCCommands</a> ( )
--------	-----------------------------------

### Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Fields

### CMD\_MF\_RECORD\_FORMAT

public static final java.lang.String **CMD\_MF\_RECORD\_FORMAT**

Constant value: **site recfm=**

### CMD\_VAL\_MF\_RECORD\_FORMAT\_FIXED

public static final java.lang.String **CMD\_VAL\_MF\_RECORD\_FORMAT\_FIXED**

Constant value: **FB**

(continued from last page)

---

## CMD\_VAL\_MF\_RECORD\_FORMAT\_VB

```
public static final java.lang.String CMD_VAL_MF_RECORD_FORMAT_VB
```

Constant value: **VB**

---

## CMD\_MF\_BLOCKSIZE

```
public static final java.lang.String CMD_MF_BLOCKSIZE
```

Constant value: **site blocksiz=**

---

## CMD\_MF\_PRIMARY

```
public static final java.lang.String CMD_MF_PRIMARY
```

Constant value: **site primary=**

---

## CMD\_MF\_SECONDARY

```
public static final java.lang.String CMD_MF_SECONDARY
```

Constant value: **site secondary=**

---

## CMD\_MF\_DATA\_TYPE

```
public static final java.lang.String CMD_MF_DATA_TYPE
```

Constant value: **type**

---

## CMD\_VAL\_MF\_DATA\_TYPE\_ASC

```
public static final java.lang.String CMD_VAL_MF_DATA_TYPE_ASC
```

Constant value: **a**

---

## CMD\_VAL\_MF\_DATA\_TYPE\_BIN

```
public static final java.lang.String CMD_VAL_MF_DATA_TYPE_BIN
```

Constant value: **i**

---

## CMD\_VAL\_MF\_DATA\_TYPE\_EBCDIC

```
public static final java.lang.String CMD_VAL_MF_DATA_TYPE_EBCDIC
```

Constant value: **e**

---

## CMD\_VAL\_MF\_DATA\_TYPE\_DBCS

```
public static final java.lang.String CMD_VAL_MF_DATA_TYPE_DBCS
```

---

(continued from last page)

Constant value: **b**

---

## **CMD\_VAL\_MF\_DATA\_TYPE\_UNICODE\_2L**

```
public static final java.lang.String CMD_VAL_MF_DATA_TYPE_UNICODE_2L
```

Constant value: **u 2 l**

---

## **CMD\_VAL\_MF\_DATA\_TYPE\_UNICODE\_2B**

```
public static final java.lang.String CMD_VAL_MF_DATA_TYPE_UNICODE_2B
```

Constant value: **u 2 b**

---

## **CMD\_MF\_LRECL**

```
public static final java.lang.String CMD_MF_LRECL
```

Constant value: **site lrecl=**

## Constructors

### **GXMFCCommands**

```
public GXMFCCommands()
```

# Index

## A

add 521, 539, 541, 544  
addAttributesToTag 702  
addBindingToTemplatePart 349  
addColumn 190, 213, 225, 241, 243, 254  
addColumnns 190, 214  
addCompletionListener 471  
addFilter 233, 262  
AddHiddenFieldToForm 659, 696  
AddHtmlTextToForm 659, 697  
addInputField 15, 16  
addInputFields 16  
ADDITIONAL 709  
AddJSToForm 659, 696  
addNeighbor 521  
addRow 232, 262  
addSearchText 497  
addServerURL 602  
addTableKeyColumn 658, 695  
addTagListener 471  
addVariable 8, 333  
addVariableArray 8  
addWaitCondition 79  
adoptHttpWrapper 457  
ALIGNMENT\_HORIZONTAL 466  
ALIGNMENT\_VERTICAL 466  
ANY\_IDENTIFIED\_SCREEN 45  
AppendFile 359  
APPLICATION\_RELOAD\_FAILED 294  
ARROWS\_ALL\_SCREEN 426  
ARROWS\_INPUT\_FIELDS 426  
ARROWS\_NONE 426  
AS400 365, 721  
attach 609

## B

BY\_APPLINX 365

## C

changeInputField 16

clone 9, 152, 161, 335, 351, 706  
CMD\_AS400\_DATA\_TYPE 725  
CMD\_AS400\_DS 726  
CMD\_AS400\_MODE 726  
CMD\_MF\_BLOCKSIZE 731  
CMD\_MF\_DATA\_TYPE 731  
CMD\_MF\_LRECL 732  
CMD\_MF\_PRIMARY 731  
CMD\_MF\_RECORD\_FORMAT 730  
CMD\_MF\_SECONDARY 731  
CMD\_VAL\_AS400\_DATA\_TYPE\_ASC 725  
CMD\_VAL\_AS400\_DATA\_TYPE\_BIN 726  
CMD\_VAL\_AS400\_DATA\_TYPE\_CCSID 726  
CMD\_VAL\_AS400\_DATA\_TYPE\_DBCS\_ASC 726  
CMD\_VAL\_AS400\_DATA\_TYPE\_DBCS\_EBC 726  
CMD\_VAL\_AS400\_DATA\_TYPE\_EBC 725  
CMD\_VAL\_AS400\_DS\_FILE 726  
CMD\_VAL\_AS400\_DS\_RECORD 727  
CMD\_VAL\_AS400\_MODE\_BLOCK 726  
CMD\_VAL\_AS400\_MODE\_STREAM 726  
CMD\_VAL\_MF\_DATA\_TYPE\_ASC 731  
CMD\_VAL\_MF\_DATA\_TYPE\_BIN 731  
CMD\_VAL\_MF\_DATA\_TYPE\_DBCS 731  
CMD\_VAL\_MF\_DATA\_TYPE\_EBCDIC 731  
CMD\_VAL\_MF\_DATA\_TYPE\_UNICODE\_2B 732  
CMD\_VAL\_MF\_DATA\_TYPE\_UNICODE\_2L 732  
CMD\_VAL\_MF\_RECORD\_FORMAT\_FIXED 730  
CMD\_VAL\_MF\_RECORD\_FORMAT\_VB 730  
COLORS\_ALL 594  
COLORS\_NONE 595  
COLORS\_REVERSED\_VIDEO 595  
compare 195  
compareTo 706  
completeBuffer 620  
confirmCommand 98  
continueDownload 97  
continueUpload 97  
convertTextIfNeeded 720  
copyFrom 535  
createTable 188, 197, 215, 238, 241, 243

## D

DATA 709  
DEFAULT 705

DESTINATION\_FOLDER\_IS\_SAME 294  
DESTINATION\_FOLDER\_IS\_SUBFOLDER 294  
detachSession 54, 103  
doAfterBody 686  
doEndTag 635, 686, 690, 700  
doStartTag 635, 685, 689, 700

## E

endSession 54, 103  
equals 9, 73, 76, 107, 147, 149, 161, 191, 192, 238, 246, 251, 255, 259, 263, 705  
ERROR\_APPLICATION\_NOT\_DEFINED 285  
ERROR\_CANNOT\_CONNECT\_HOST 284  
ERROR\_CANNOT\_CONNECT\_SERVER 284  
ERROR\_CANT\_DELETE\_ASSIGNED\_EXCEPTION\_PATH 288  
ERROR\_CANT\_DELETE\_REFERENCED\_ENTITY 287  
ERROR\_CENTRASITE\_REGISTRATION 297  
ERROR\_CHANGING\_WEB\_SERVICE\_TARGET\_SERVER 296  
ERROR\_CONCURRENT\_NUMBER\_OF\_USERS\_EXCEED  
S\_LICENSE\_AGREEMENT 289  
ERROR\_CONFIGURATION\_SAVE\_FAILED 291  
ERROR\_CONNECTING\_DEVICE\_NAME 289  
ERROR\_CONNECTION\_INFO 285  
ERROR\_CREATING\_ENTITIES\_FROM\_WSDL 296  
ERROR\_CREATING\_TABLE 289  
ERROR\_CURSOR\_POSITION\_NOT\_SET 288  
ERROR\_DB\_GENERAL 287  
ERROR\_DEPLOYING\_WEB\_SERVICE 296  
ERROR\_DEPLOYMENT\_GENERAL 296  
ERROR\_DEPLOYMENT\_HANDLER\_NOT\_AVAILABLE 297  
ERROR\_DISPLAY\_SESSION\_REQUEST\_FOR\_PRINTER\_  
SESSION 289  
ERROR\_EXECUTING\_MAP\_BAD\_DESTINATION 289  
ERROR\_EXECUTING\_MAP\_NO\_MAP\_IN\_APPL 289  
ERROR\_EXECUTING\_MAP\_NO\_PATH\_BETWEEN\_SCRE  
ENS 289  
ERROR\_EXECUTING\_MAP\_SCREEN\_NOT\_IN\_MAP 289  
ERROR\_EXECUTING\_PATH\_FROMSCREENID 286  
ERROR\_EXECUTING\_PATH\_GENERAL 286  
ERROR\_EXECUTING\_PATH\_RESOLVE\_ACTION 288  
ERROR\_EXECUTING\_PATH\_TOSCREENID 286  
ERROR\_EXECUTING\_PATH\_TYPE\_EXCEPTION 288  
ERROR\_EXECUTING\_PROCEDURE\_GENERAL 296  
ERROR\_EXPORTING\_WEB\_SERVICE 296  
ERROR\_FIELD\_CONTENT\_NOT\_SET 288  
ERROR\_GENERAL 284  
ERROR\_HOST\_EXISTS\_IN\_APPLICATION 291  
ERROR\_HOST\_SESSION\_DISCONNECTED 286  
ERROR\_IMPORT\_FILE\_REPOSITORY 288  
ERROR\_INCOMPATIBLE\_VERSIONS 291  
ERROR\_INCOMPATIBLE\_WSS\_VERSION 297  
ERROR\_INCORRECT\_PASSWORD 285  
ERROR\_INDEX\_OUT\_OF\_BOUND 285  
ERROR\_INVALID\_LICENSE\_COMBINATION 292  
ERROR\_INVALID\_LICENSE\_KEY 292  
ERROR\_LICENSE\_FILE\_NOT\_FOUND 292  
ERROR\_LICENSE\_TERMS\_VAIOLATION 290  
ERROR\_LOGIN\_FAILED 286  
ERROR\_MAP\_STEP\_ALREADY\_EXISTS 289  
ERROR\_NATURAL\_DATA\_TRANSFER\_FAILURE 291  
ERROR\_NATURAL\_DATA\_TRANSFER\_INAVLID\_STAT  
E 291  
ERROR\_OVERWRITING 286  
ERROR\_PATH\_NOT\_FOUND 287  
ERROR\_PERMISSION\_DENIED 287  
ERROR\_PREFS\_SAVE\_FAILED 287  
ERROR\_PRINTER\_CONNECT\_ERROR 288  
ERROR\_PRINTER\_DISABLED 286  
ERROR\_REPLACING\_LICENSE 292  
ERROR\_REPLAY\_DEVICE\_INCOMPATIBILITY 288  
ERROR\_REPLAY\_FILE 285  
ERROR\_REPLAY\_FILE\_NOT\_FOUND 285  
ERROR\_RETRIEVING\_WSDL\_SERVICES 295  
ERROR\_RETRIEVING\_WSDL\_SERVICES\_PERMISSION\_  
DENIED 296  
ERROR\_SCREEN\_NOT\_SYNC 287  
ERROR\_SECURITY\_SAVE\_FAILED 291  
ERROR\_SERVICE\_CONNECTION\_NOT\_READY 291  
ERROR\_SERVICE\_CONNECTION\_TIMEOUT\_EXPIRED 290  
ERROR\_SERVICE\_GENERAL 290  
ERROR\_SERVICE\_NO\_AVAILABLE\_CONNECTION 290  
ERROR\_SERVICE\_NOT\_ACTIVE 290  
ERROR\_SERVICE\_NOT\_FOUND 291  
ERROR\_SERVICE\_PATH\_NOT\_FOUND 290  
ERROR\_SESSION\_VARIABLE\_DOESNT\_EXIST 290

ERROR\_SESSION\_VARIABLE\_TYPE\_MISMATCH 290  
ERROR\_SOCKET 284  
ERROR\_SYNCHRONIZING\_REPOSITORY 295  
ERROR\_TIMEOUT\_EXPIRED 286  
ERROR\_UNDEPLOYING\_WEB\_SERVICE 296  
ERROR\_UNKNOWN\_ACTION 287  
ERROR\_UNKNOWN\_SERVER\_ADDRESS 284  
ERROR\_UPDATING\_WEB\_SERVICE 296  
ERROR\_USER\_ALREADY\_ATTACHED 285  
ERROR\_USER\_NOT\_CONNECTED 284  
ERROR\_WAIT\_CONDITION\_NOT\_MET 285  
ERROR\_WAIT\_CONDITION\_NOT\_VALID 287  
executeColumnQuery 519  
executePath 81, 82  
executePositionQuery 520  
executeProcedure 82, 83  
executeQuery 519  
executeRowQuery 520  
executeTextQuery 519  
EXPORT\_APPLICATION\_ENTITIES\_FAILED 294

## F

FAULTY\_SCREEN\_IMAGE 295  
FIELD\_EXISTS\_IN\_ACTIONS 293  
FIELD\_TYPE\_APPFIELD 70  
FIELD\_TYPE\_HOSTFIELD 71  
FILE\_DOES\_NOT\_EXIST 286  
FileName 359, 361  
filter 210, 268, 270, 273  
filterRows 233, 262  
find 493, 494, 495

## G

get 538, 540, 543  
get\_ActionKey 396  
get\_addDocTypeToFrameset 357  
get\_AddJavascript 348  
get\_AdjustNeighbours 576, 590  
get\_Alignment 576  
get\_AllowTypeAhead 345  
get\_AlternateColors 404  
get\_AlternateCssName 405  
get\_ApplicationName 330

get\_AttachInputTagEvents 441  
get\_AutomaticDownload 452  
get\_AutoSkipAllFields 427  
get\_BlinkingCaret 434  
get\_BlockNumericFields 431  
get\_BoundTemplateParts 355  
get\_Build 308  
get\_CaptureBrowserClose 347  
get\_CaptureTagsEvents 318  
get\_CaretAtEndOfField 432  
get\_CheckboxCss 576  
get\_CheckHostScreenUpdate 440  
get\_ClientBlinking 570  
get\_CloneComboxOnClientSide 407  
get\_ColorMode 570  
get\_ColorSets 431  
get\_ComboboxCss 576  
get\_CompleteListenersVector 478  
get\_DebugModules 419  
get\_DefaultExtension 452  
get\_DefaultFrameColumns 420  
get\_DefaultOpenedWindowHeight 446  
get\_DefaultOpenedWindowLeft 446  
get\_DefaultOpenedWindowTop 446  
get\_DefaultOpenedWindowWidth 446  
get\_Description 330  
get\_DesignMode 308  
get\_DocTypeStr 357  
get\_DoJapaneseDataTypeValidityCheck 434  
get\_DoubleClickActionKey 428  
get\_DownloadAtOnce 453  
get\_DynamicDisableCssClass 347, 405  
get\_DynamicFieldCheck 403  
get\_EmulationConfig 473  
get\_EmulationPasteBehavior 435  
get\_Enabled 391, 593  
get\_Encoding 408  
get\_EncryptData 372  
get\_EngineConfig 411  
get\_ErrorForm 311  
get\_ExternalCssRequestParamName 350  
get\_FailOnWarning 336  
get\_FieldsModification 321  
get\_FieldTypesInUse 309  
get\_Flicker 372

get\_FontFamily 476  
get\_FontSize 472  
get\_FromRow 401  
get\_FtpConfig 319  
get\_GenerateTrOnClickEvent 403  
get\_GuiElementsConfig 474  
get\_HandleMenuSelection 432  
get\_HideOnly 599  
get\_HostAddress 366  
get\_HostDownloadForm 453  
get\_HostKeyCss 582  
get\_HostKeysConfig 474  
get\_HostNewPassword 335  
get\_HostPassword 331  
get\_HostType 366  
get\_HostUserName 329  
get\_IndexForm 323  
get\_Instant 323  
get\_InstantConfig 321  
get\_InstantEmulatorConfig 412  
get\_InstantForm 310  
get\_InstantWinForm 311  
get\_IsInstant 322  
get\_LabelFocus 428  
get\_LoadEmulationComponent 441  
get\_LoadPrinter 439  
get\_LogLevel 417  
get\_LogOffForm 345  
get\_LogoffMessage 420  
get\_LogoffOnBrowserClose 417  
get\_MacroConfig 316  
get\_MacrosFolder 370  
get\_MapPath 315  
get\_MozillaLogoffMessage 418  
get\_NaturalDataTransferConfig 350  
get\_OpenedWindowAttributes 447  
get\_OpenWindowCenter 447  
get\_Password 331  
get\_PasswordField 375, 382  
get\_PasswordFieldName 388  
get\_PathName 394  
get\_PerformSyncCheckInInstant 319  
get\_PopupCloseSentKey 447  
get\_ProceduresConfig 320  
get\_PromptBeforeSessionTimeOut 353  
get\_PromptOnBrowserClose 420  
get\_RadioButtonCss 577  
get\_Rectangle 472  
get\_ReflectBGColor 406  
get\_ReflectBGColors 312  
get\_ReflectEmulationBehaviour 313  
get\_ReflectFGColor 406  
get\_ReflectFGColors 312  
get\_ReflectHostCursor 313  
get\_ReflectHostProtected 315, 403  
get\_ReflectMaxlength 314  
get\_RenderAlternatingCssClass 590  
get\_RenderAppFieldCss 570  
get\_RenderArea 472  
get\_RenderEmulationAttributes 569  
get\_RenderingType 581  
get\_RenderIntensifiedCss 570  
get\_RenderTagsFromRightToLeft 477  
get\_RowSpacingPercentageFactor 477  
get\_SaveUserNameLocally 371  
get\_ScreenLocker 346  
get\_ScreenLockerText 355  
get\_ScreenName 388  
get\_SelectFieldOnFocus 433  
get\_ServerURL 328  
get\_ServiceName 332  
get\_SessionConfig 316  
get\_SessionId 329  
get\_ShowBlinkingFields 433  
get\_ShowIntermediateScreen 334  
get\_ShowLogConsole 419  
get\_SingleSignOnConfig 323  
get\_SsoConfig 322  
get\_SupportDupAndFieldMark 309  
get\_SupportPartialPageRendering 356  
get\_SupportTypeAhead 440  
get\_TableAlternating1CssClass 590  
get\_TableAlternating2CssClass 590  
get\_TableBuildConfig 320  
get\_TableConfig 475  
get\_TableCssClass 590  
get\_TabOnInputsOnly 429  
get\_TerminationPath 355  
get\_TitleWindowFrameCssClass 599  
get\_ToRow 402

get\_TrimLabels 310, 407  
get\_UploadForm 454  
get\_UseActiveX 453  
get\_UseArrows 430  
get\_UseCache 343  
get\_UseCalendar 355  
get\_UseFolders 317  
get\_UseFramesForSubmit 355  
get\_UseJavaScriptsAsResource 351  
get\_UseKeyboardPFKeys 308  
get\_UseKeyboardPfKeys 430  
get\_UseModalWindows 317  
get\_UseOldJSEngine 355  
get\_UserExitsFile 439  
get\_UserField 375, 382  
get\_UserName 371  
get\_UserNameFieldName 388  
get\_UseScreenGroups 318  
get\_UseScriptsInPage 352  
get\_UseStateManager 346  
get\_Variable 335  
get\_Variables 333  
get\_VariablesModified 335  
get\_Version 307  
get\_VirtualDir 344  
get\_WebDownloadForm 454  
get\_WebEmulatorConfig 412  
get\_WebWindowName 356  
get\_WindowConfig 412  
get\_WindowFrameConfig 475  
get\_WindowFrameCssClass 599  
get\_WindowRelative 591  
get\_WritePerformanceLog 349  
getAction 95  
getActionKey 396  
getAdditionalCommand 718  
getAddJavascript 348  
getAlignment 512, 573  
getAllowTypeAhead 345  
getAlternateColors 404  
getAlternateCssName 405  
getAppFieldFullName 221, 250  
getAppFieldName 220, 250  
getAppFields 75, 93  
getApplicationConfig 109  
getApplicationName 12, 51, 330  
getAttribute 533  
getAvailableServer 603  
getBaseObject 53, 54, 102  
getBGColor 72, 90, 201  
getBodyAttribute 660  
getBodyAttributes 660  
getBounds 126  
getBoundTemplateParts 349  
getBuffer 619  
getBufferLength 620  
getBufferPosition 622  
getBuffers 115  
getBuild 308  
getCaption 95, 221, 250, 500, 503, 516  
getCaptureBrowserClose 348  
getCaptureTagsEvents 319  
getCells 525  
getCharactersPerLine 610  
getCheckboxCss 574  
getCheckedValue 503, 516  
getChildren 529  
getClientId 13  
getCloneComboxOnClientSide 407  
getCode 712  
getColorMode 567  
getColorSets 430  
getColumn 225, 254, 705  
getColumnCount 224, 254  
getColumnName 208  
getColumnNumber 622  
getColumns 115, 189, 213, 224, 254  
getComboboxCss 575  
getCompletionListener 476  
getConnectionPool 59  
getContent 72, 89, 192, 207, 217, 245, 534, 549, 551, 553  
getCopyCount 223, 251  
getCornersCoordinates 191  
getCursor 152  
getCursorPosition 115  
getData 48, 49, 129, 133, 623  
getDataTransferCommand 100  
getDataTransferFilename 100  
getDataTransferState 99  
getDataTransferWorkfileCode 100

getDataType 73, 90, 201, 220, 249  
getDebugModules 419  
getDefaultExtension 452  
getDefaultFrameColumns 418  
getDefaultOpenedWindowHeight 444  
getDefaultOpenedWindowLeft 444  
getDefaultOpenedWindowTop 445  
getDefaultOpenedWindowWidth 445  
getDefinedLowerRightCorner 204  
getDefinedUpperLeftCorner 204  
getDescription 330  
getDesignMode 308  
getDestinationScreenName 137  
getDir 645  
getDisplayPosition 95  
getDisplayText 95  
getDocTypeStr 357  
getDoubleClickActionKey 427  
getDynamicDisableCssClass 347, 405  
getDynamicFieldCheck 403  
getEABBuffer 149  
getEABColor 149  
getEmulationConfig 473  
getEncoding 407  
getEncryptData 372  
getEngineConfig 411  
getError 620  
getErrorForm 311  
getErrorMessage 620  
getExternalCssRequestParamName 350  
getFGColor 72, 90, 201  
getFieldAt 75, 92  
getFieldAttributeBuffer 149  
getFieldByCursorPosition 119  
getFieldByPosition 76, 93  
getFieldContent 117, 118  
getFieldCount 75, 92  
getFieldIndex 65  
getFieldInstance 76, 93, 117  
getFieldMatchQuery 177, 180, 181, 182  
getFieldName 65  
getFields 75, 93, 94, 116, 117, 120, 121, 178, 183, 184  
getFieldsModification 321  
getFieldTypesInUse 310  
getFile 714  
getFileAsByteArr 719  
getFileAsInputStream 719  
getFileName 661, 697, 720  
getFilters 233, 262  
getFirstAppfieldColumnIndex 206  
getFirstFieldInstance 75, 93  
getFlicker 18, 372  
getFolderPath 115  
getFontFamily 476  
getFontSize 472, 520  
getFormAttribute 660  
getFormAttributes 660  
getFormatType 4  
getFormTopString 659  
getFromRow 401  
getFtpConfig 320  
getGenerateTrOnClickEvent 402  
getGuiElementsConfig 473  
getHeaderCoordinates 191  
getHeaderLowerRight 204  
getHeaders 526  
getHeaderUpperLeft 204  
getHeightPixels 480  
getHiddens 659  
getHorizontal 645  
getHostAddress 52, 366  
getHostDeviceName 51  
getHostDeviceType 51  
getHostDownloadForm 451  
getHostField 535  
getHostKeyCss 581  
getHostKeys 119  
getHostKeysConfig 474  
getHostKeysTagUserExit 353  
getHostNewPassword 60, 335  
getHostPassword 60, 332  
getHostScreen 485  
getHostService 59  
getHostType 365  
getHostUserName 59, 329  
getId 529, 633, 636, 668, 685  
getIndex 71, 88, 106, 200  
getIndexForm 323  
getInnerHTML 684  
getInputFields 15

getInputMask 73, 91, 201  
getInputStream 661, 697  
getInputTagAttributes 660  
getInstantConfig 321  
getInstantEmulatorConfig 411  
getInstantForm 310  
getInstantWinForm 311  
getItem 228, 234, 257, 262  
getItemContent 229, 234, 258, 263  
getItems 227, 257, 506, 513  
getItemsContents 228, 257  
getKeyAt 622  
getKeyboardMappingXML 52  
getKeys 151  
getKeysCount 621  
getLabelFocus 428  
getLangConvType 52  
getLanguage 52  
getLastAppfieldColumnIndex 206  
getLastFieldInstance 76, 94  
getLength 71, 89, 200, 532  
getLineAt 606  
getLineCount 606  
getLineNumber 622  
getLinesPerPage 610  
getLocalFile 718  
getLocalFileBytes 718  
getLogLevel 416  
getLogOffForm 345  
getLogoffMessage 418  
getLowerRightCorner 205  
getLowerRightPosition 20, 25  
getMacroConfig 316  
getMacrosFolder 370  
getMainContext 685  
getMapName 138  
getMapPath 315  
getMasterScreenName 238, 243  
getMozillaLogoffMessage 418  
getName 71, 88, 106, 113, 123, 163, 189, 200, 213, 220, 249, 515  
getNaturalDataTransferConfig 350  
getNeighbor 521  
getNewRow 190, 214  
getNonActivityTimeOut 52  
getNumProcessedRecords 6  
getOffsetLeft 530  
getOffsetRight 530  
getOffsetTop 529  
getOnClick 501, 509  
getOpenedWindowAttributes 445  
getOriginalBuffer 607  
getPageAt 619  
getPageBufferPosition 622  
getPageCount 619  
getPages 619  
getPanelToUpdate 661  
getPanelToUpdateContent 661  
getParent 528  
getPassword 331, 380, 459, 461, 717  
getPasswordField 375, 381  
getPasswordFieldName 387  
getPasswordParameterName 457  
getPathExecutionLog 146  
getPathName 142, 394  
getPathResponse 140  
getPerformSyncCheckInInstant 319  
getPopupCloseSentKey 445  
getPosition 64, 71, 88, 105, 200, 532  
getPrimaryKey 190, 214, 229, 258  
getPrint 609  
getPrintData 122  
getPrinterStatus 609  
getProceduresConfig 320  
getRadioButtonCss 575  
getReadOnlyBaseObject 54, 102  
getReflectBGColor 406  
getReflectBGColors 312  
getReflectEmulationBehaviour 313  
getReflectFGColor 406  
getReflectFGColors 312  
getReflectHostCursor 314  
getReflectHostProtected 315, 404  
getReflectMaxlength 314  
getRemoteFile 717  
getRenderArea 471  
getRenderingType 581  
getReplayerInfo 111  
getReplayerStatus 111  
getReplayScreenData 111

getReply 712  
getRow 232, 262, 705  
getRowByKey 235, 263  
getRowCount 232, 261  
getRows 115, 190, 214, 232, 261, 526  
getRowSpacingPercentageFactor 476  
getScreen 108, 109  
getScreenAt 120  
getScreenBuffer 149  
getScreenCount 120  
getScreenGroupNames 113  
getScreenLocker 346  
getScreenLockerText 347  
getScreenName 22, 71, 88, 200, 387  
getScreenNames 222, 250  
getScreens 121, 145  
getScreenSequenceNumber 71, 88, 200  
getScreenTagModel 484  
getSCSElements 607, 620  
getSelectedValue 507, 513  
getSeqNumber 114  
getSeqPageNumber 607  
getSeqPrintNumber 619  
getSeqScreenNumber 17, 109, 153  
getServer 717  
getServerContext 12, 334  
getServers 602  
getServerURL 12, 328  
getServiceName 332  
getSessionConfig 316  
getSessionDescription 59  
getSessionID 11, 109  
getSessionId 329  
getSessionPassword 12  
getSignOnScreenRecognizer 391  
getSingleSignOnConfig 322  
getSsoExecutor 391  
getString 24  
getSupportDupAndFieldMark 309  
getTable 83, 84, 118  
getTableAlternating1CssClass 588  
getTableAlternating2CssClass 588  
getTableBuildConfig 320  
getTableConfig 474  
getTableCssClass 588  
getTableName 157, 658, 696  
getTables 118  
getTableSelectedKey 658, 696  
getTableSelectedKeys 658, 696  
getTagAttribute 659  
getTagAttributes 660  
getTagById 519  
getTagContent 657, 693  
getTagContents 657  
getTagHeight 480  
getTagListeners 475  
getTagName 533  
getTags 521  
getTagWidth 480  
getTarget 510  
getTerminationPath 353  
getText 508, 510, 523, 537  
getTimeout 18  
getTitle 126  
getTitleWindowFrameCssClass 598  
getToRow 402  
getTrimLabels 310, 407  
getType 220, 251, 622, 712  
getUncheckedValue 503  
getUploadForm 451  
getUpperLeftCorner 205  
getUpperLeftPosition 20, 25  
getUseArrows 429  
getUseCache 344  
getUseFolders 317  
getUseKeyboardPFKeys 309  
getUseModalWindows 317  
getUser 717  
getUserExitsFile 439  
getUserField 375, 382  
getUserId 62  
getUserName 371, 380, 459, 461  
getUserNameFieldName 387  
getUserParameterName 457  
getUserPasswordRetriever 390  
getUseScreenGroups 318  
getUseStateManager 346  
getValue 106, 550, 716  
getValues 163, 716  
getVariable 145, 333

- getVariableArray 146
- getVariableAt 146
- getVariableCount 146
- getVariableName 221, 250
- getVariableNames 146
- getVariables 8, 333
- getVersion 307
- getVirtualDir 344
- getVisualContent 73, 91, 201
- getWaitCotions 79
- getWaitForResponseTimeout 603
- getWebDownloadForm 451
- getWebEmulatorConfig 412
- getWebWindowName 354
- getWidthPixels 479
- getWindow 118
- getWindowConfig 412
- getWindowFrameConfig 475
- getWindowFrameCssClass 598
- getWindowLowerRight 114
- getWindows 118
- getWindowUpperLeft 114
- getWritePerformanceLog 349
- GX\_APPLICATION\_PARM\_ASSOCIATE\_DEVICE\_NAME 613
- GX\_APPLICATION\_PARM\_DEVICE\_NAME 613
- GX\_APPLICATION\_PARM\_DEVICE\_TYPE 613
- GX\_APPLICATION\_PARM\_DISPLAY\_DEVICE\_NAME 613
- GX\_APPLICATION\_PARM\_HOST\_ADDRESS 613
- GX\_APPLICATION\_PARM\_HOST\_PORT 613
- GX\_APPLICATION\_PARM\_IGNORE\_FIRST\_FORM\_FEE D 614
- GX\_APPLICATION\_PARM\_IGNORE\_FORMFEED\_IN\_TRANSPARENT 614
- GX\_APPLICATION\_PARM\_IGNORE\_HOST\_FORMATTING 614
- GX\_APPLICATION\_PARM\_IGNORE\_HOST\_SETTING\_FOR\_MAX\_CHARACTERS\_PER\_LINE 614
- GX\_APPLICATION\_PARM\_IGNORE\_HOST\_SETTING\_FOR\_MAX\_LINES\_PER\_PAGE 614
- GX\_APPLICATION\_PARM\_IGNORE\_TRANSPARENT 614
- GX\_APPLICATION\_PARM\_IGNORE\_VERTICAL\_CHANNEL\_SELECT 614
- GX\_APPLICATION\_PARM\_LANGUAGE 612
- GX\_APPLICATION\_PARM\_MESSAGE\_LIBRARY 613
- GX\_APPLICATION\_PARM\_MESSAGE\_QUEUE 613
- GX\_APPLICATION\_PARM\_REPLAY\_FILE 613
- GX\_APPLICATION\_PARM\_TRACE\_FILE 613
- GX\_APPLICATION\_PARM\_TRANSLATE\_ESCAPE 614
- GX\_APPLICATION\_PARM\_TRANSPARENT\_DATA\_IS\_ASCII 614
- GX\_FIELD\_ADJUST\_LEFT 45
- GX\_FIELD\_ADJUST\_NONE 45
- GX\_FIELD\_ADJUST\_RIGHT 45
- GX\_FIELD\_ADJUST\_RIGHT\_BLANK\_FILL 45
- GX\_FIELD\_ADJUST\_RIGHT\_ZERO\_FILL 45
- GX\_FIELD\_COLOR\_AQUA 34
- GX\_FIELD\_COLOR\_BLACK 34
- GX\_FIELD\_COLOR\_BLUE 34
- GX\_FIELD\_COLOR\_BROWN 35
- GX\_FIELD\_COLOR\_GRAY 35
- GX\_FIELD\_COLOR\_GREEN 34
- GX\_FIELD\_COLOR\_LIGHT\_AQUA 35
- GX\_FIELD\_COLOR\_LIGHT\_BLUE 35
- GX\_FIELD\_COLOR\_LIGHT\_GREEN 35
- GX\_FIELD\_COLOR\_LIGHT\_PURPLE 35
- GX\_FIELD\_COLOR\_LIGHT\_RED 35
- GX\_FIELD\_COLOR\_LIGHT\_WHITE 36
- GX\_FIELD\_COLOR\_NONE 34
- GX\_FIELD\_COLOR\_PURPLE 34
- GX\_FIELD\_COLOR\_RED 34
- GX\_FIELD\_COLOR\_WHITE 35
- GX\_FIELD\_COLOR\_YELLOW 35
- GX\_FIELD\_DATA\_TYPE\_ALPHA\_ONLY 44
- GX\_FIELD\_DATA\_TYPE\_ALPHANUMERIC 44
- GX\_FIELD\_DATA\_TYPE\_DBCS\_CAN\_CREATE\_SISO 45
- GX\_FIELD\_DATA\_TYPE\_DBCS\_ONLY 44
- GX\_FIELD\_DATA\_TYPE\_DIGITS\_ONLY 44
- GX\_FIELD\_DATA\_TYPE\_KATAKANA\_SHIFT 44
- GX\_FIELD\_DATA\_TYPE\_NUMERIC 44
- GX\_FIELD\_DATA\_TYPE\_REVERSED 45
- GX\_FIELD\_DATA\_TYPE\_SIGNED\_NUMERIC 44
- GX\_PRINT\_OBJECT\_STATUS\_COMPLETE 616
- GX\_PRINT\_OBJECT\_STATUS\_EMPTY 616
- GX\_PRINT\_OBJECT\_STATUS\_NOT\_COMPLETE 616
- GX\_PRINTER\_STATUS\_BUSY 615
- GX\_PRINTER\_STATUS\_CANCELLING 616
- GX\_PRINTER\_STATUS\_ERROR 615

- GX\_PRINTER\_STATUS\_INUSE 615  
GX\_PRINTER\_STATUS\_INVALID 615  
GX\_PRINTER\_STATUS\_PAPEROUT 615  
GX\_PRINTER\_STATUS\_READY 615  
GX\_PRINTER\_STATUS\_TIMEDOUT 615  
GX\_TABLE\_COLUMN\_DATA\_TYPE\_ALPHANUMERIC 219  
GX\_TABLE\_COLUMN\_DATA\_TYPE\_NUMERIC 219  
GX\_TABLE\_COLUMN\_TYPE\_APPFIELD 219  
GX\_TABLE\_COLUMN\_TYPE\_OTHER 220  
GX\_TABLE\_COLUMN\_TYPE\_VARIABLE 219  
GX\_VAR\_APPLICATION\_LANGUAGE 37  
GX\_VAR\_APPLY\_TRANSFORMATIONS 43  
GX\_VAR\_DEVICE\_NAME 36  
GX\_VAR\_DEVICE\_TYPE 36  
GX\_VAR\_DISPLAY\_HIDDEN\_FIELDS 43  
GX\_VAR\_ENCODING\_LOGICAL 41  
GX\_VAR\_FIELDS\_WITH\_SUGGESTED\_NAMES 43  
GX\_VAR\_HOST\_ADDRESS 36  
GX\_VAR\_HOST\_PORT 36  
GX\_VAR\_HOST\_WINDOW\_ENABLED 43  
GX\_VAR\_INCLUDE\_HOST\_FIELDS 43  
GX\_VAR\_NATURAL\_DATA\_TRANSFER\_DELIMITER 43  
GX\_VAR\_NATURAL\_DATA\_TRANSFER\_HEADERS\_USED 44  
GX\_VAR\_NATURAL\_DATA\_TRANSFER\_STREAM 44  
GX\_VAR\_REMOVE\_CHARACTER 40  
GX\_VAR\_REMOVE\_CHARACTER\_BOTH\_SIDES 41  
GX\_VAR\_REPLAY\_DELAY 38  
GX\_VAR\_REPLAY\_FILE 38  
GX\_VAR\_RETURN\_FIELDS 43  
GX\_VAR\_RETURN\_SCREEN\_BUFFERS 39  
GX\_VAR\_SCREEN\_DIRECTION 39  
GX\_VAR\_THROW\_EXCEPTION\_ON\_INVALID\_INPUT 43  
GX\_VAR\_TRACE\_FILE 38  
GX\_VAR\_TRANSLATION\_LANGUAGE 38  
GX\_VAR\_TRIM\_FIELDS 41  
GX\_VAR\_TRIM\_MODE 42  
GX\_VAR\_UPPERCASE\_INPUT 36  
GX\_VAR\_VAL\_BUFFERS\_FIELD\_ATTRIBUTES 40  
GX\_VAR\_VAL\_BUFFERS\_NONE 39  
GX\_VAR\_VAL\_BUFFERS\_SCREEN\_DATA 39  
GX\_VAR\_VAL\_REMOVE\_CHARACTER\_ALL 40  
GX\_VAR\_VAL\_REMOVE\_CHARACTER\_BOTH\_SIDES\_ALL\_FIELDS 41  
GX\_VAR\_VAL\_REMOVE\_CHARACTER\_BOTH\_SIDES\_NUMERIC\_FIELDS\_ONLY 41  
GX\_VAR\_VAL\_REMOVE\_CHARACTER\_INPUT 40  
GX\_VAR\_VAL\_REMOVE\_CHARACTER\_NONE 40  
GX\_VAR\_VAL\_REPLAY\_DELAY\_NO\_DELAY 38  
GX\_VAR\_VAL\_REPLAY\_DELAY\_SIMULATE\_HOST 38  
GX\_VAR\_VAL\_SCREEN\_DIRECTION\_LTR 39  
GX\_VAR\_VAL\_SCREEN\_DIRECTION\_RTL 39  
GX\_VAR\_VAL\_TRIM\_FIELDS\_ALL 42  
GX\_VAR\_VAL\_TRIM\_FIELDS\_INPUT\_AND\_BOTH 42  
GX\_VAR\_VAL\_TRIM\_FIELDS\_INPUT\_ONLY 41  
GX\_VAR\_VAL\_TRIM\_FIELDS\_NONE 42  
GX\_VAR\_VAL\_TRIM\_MODE\_BOTH\_SIDES 42  
GX\_VAR\_VAL\_TRIM\_MODE\_BY\_FIELD\_DIRECTION 42  
GX\_VAR\_VAL\_VT\_SEND\_CURSOR\_ARROWS 37  
GX\_VAR\_VAL\_VT\_SEND\_CURSOR\_NONE 37  
GX\_VAR\_VAL\_VT\_SEND\_CURSOR\_TAB 37  
GX\_VAR\_VT\_AUTO\_SKIP\_WHEN\_FIELD\_IS\_FULL 37  
GX\_VAR\_VT\_DYNAMIC\_INPUT\_CHECK 37  
GX\_VAR\_VT\_KEY\_JUMP\_BETWEEN\_INPUT\_FIELDS 37  
GX\_VAR\_VT\_SEND\_CURSOR 36  
GX\_VAR\_WORKING\_FOLDER 38  
GX\_WAIT\_COND\_END\_OF\_PRINT 615  
GX\_WAIT\_COND\_PRINTER\_READY 615  
GXAbstractAnchorTag 666  
GXAbstractBodyTag 668  
GXAbstractCheckBoxTag 670  
GXAbstractFieldMatchQuery 177  
GXAbstractFormTag 671  
GXAbstractGenericTag 672  
GXAbstractHtmlTableTag 674  
GXAbstractHtmlTag 675  
GXAbstractHttpUserPasswordRetriever 456, 457  
GXAbstractInputTag 677  
GXAbstractNaturalDataTransferRequest 3  
GXAbstractNaturalDataTransferResponse 5  
GXAbstractRadioButton 679  
GXAbstractRadioButtonList 680  
GXAbstractSelectTag 682  
GXAbstractSessionConnectionRequest 11  
GXAbstractSignOnScreenRecognizer 374

- GXAbstractSSOExecuter 376
- GXAbstractTable 188
- GXAbstractTag 684
- GXAddFieldToPreviousRowFilter 268
- GXAnchorTag 626
- GXAppConfig 307
- GXAreaQuery 492
- GXAS400Commands 727
- GXAttachToSessionRequest 27
- GXBaseObjectConstants 45
- GXBinaryRequest 47, 48
- GXBinaryResponse 49
- GXBoldTag 628
- GXCellable 192
- GXCheckBoxTag 630
- GXCisCssTransform 564
- GXClientApplicationConfig 51
- GXClientBaseObjectFactory 53
- GXClientConfig 411
- GXColumnNotFoundException 194
- GXColumnsComparator 195
- GXConfig 328
- GXConnectionException 55
- GXContextInitiatorTag 688
- GXContextReleaseTag 690
- GXCreateSessionRequest 57, 58
- GXCreateSessionResponse 62
- GXCursor 63, 64
- GXDefaultTable 197
- GXDivTag 631
- GXEmulationTransformConfig 567
- GXEndSessionRequest 67, 68
- GXEngineConfig 416
- GXField 71
- GXFieldCollection 75
- GXFieldCollectionQueryAppField 179
- GXFieldCollectionQueryProtected 181
- GXFieldTableCell 199
- GXFormTag 633
- GXFtpConfig 365
- GXFtpConsts 709
- GXFtpException 711, 712
- GXFtpRequest 716
- GXFtpResponse 719
- GXFtpTypes 722
- GXGeneralException 77
- GXGetScreenRequest 78
- GXGuiElementsTransformConfig 573
- GXHebrewTransform 578
- GXHostKeysTransformConfig 581
- GXHostTableUtils 203
- GXHtmlTag 635
- GXHttpRequestUserPasswordRetriever 458, 459
- GXHttpSessionUserPasswordRetriever 460, 461
- GXIllegalPositionException 465
- GXImageTag 637
- GXInnerClientBaseObjectFactory 101
- GXInputField 105
- GXInputTag 640
- GXInstantEmulatorConfig 427
- GXJspIncludeTag 700
- GXLoadBalancer 602
- GXLogNode 360
- GXMacroConfig 370
- GXMasterTable 238
- GXMFCCommands 732
- GXMultipleRecordsTable 241
- GXNaturalDataTransferDownloadRequest 127, 128
- GXNaturalDataTransferDownloadResponse 129
- GXNaturalDataTransferException 131
- GXNaturalDataTransferUploadRequest 132, 133
- GXNaturalDataTransferUploadResponse 134
- GXNavigateException 135
- GXNavigateRequest 137
- GXNdtConfig 451
- GXNonDuplicateRowsFilter 270
- GXNonEmptyRowsFilter 272, 273
- GXPage 605
- GXPageTag 643
- GXPartialRowsTable 243
- GXPathException 140
- GXPathRequest 142
- GXPathResponse 145
- GXPerformanceMonitorNode 362
- GXPosition 705
- GXPresentationConstants 466
- GXPrinter 609
- GXPrinterSessionConstants 616
- GXPrintObject 618
- GXRadioButtonListTag 645

GXRadioButtonTag 648  
GXRenderConfig 471  
GXRenderEvent 484  
GXResponseCodes 297  
GXSapPortalTransform 584  
GXScreenBuffers 148  
GXScreenCollectionFieldNameQuery 184  
GXScreenNameUndefinedException 244  
GXSelectTag 650  
GXSendKeysRequest 151  
GXSessionStatusResponse 153  
GXSetVariablesRequest 154  
GXSignOnRecognizerResponse 381  
GXSignOnScreenAutomaticRecognizer 383  
GXSignOnScreenByNameRecognizer 386  
GXSimpleTagQuery 495  
GXSpanTag 651  
GXSSOConfig 390  
GXSSOPathExecuter 393, 394  
GXSSOSendKeysExecuter 395, 396  
GXTableBuildConfig 401  
GXTableCell 245  
GXTableCellList 538  
GXTableColumn 248, 249  
GXTableColumnCollection 254  
GXTableRequest 157  
GXTableRow 257  
GXTableRowCollection 261  
GXTableRowList 540  
GXTableTag 653  
GXTableTransformConfig 587  
GXTagFunctions 701  
GXTagList 542, 543  
GXTagListener 487  
GXTagModelConstants 546  
GXTagsAccesorImpl 656  
GXTagUtil 479  
GXTdTag 662  
GXTextAreaTag 663  
GXTextQuery 497  
GXTransformConfig 592  
GXTransformConstants 595  
GXUndefinedPositionException 481  
GXVariable 160, 161  
GXVariableArray 162

GXVariableTableCell 265  
GXWaitForCursor 164, 165  
GXWaitForScreen 166  
GXWaitForString 167, 168  
GXWaitHostQuiet 169, 170  
GXWaitWhileCursor 171, 172  
GXWaitWhileScreen 173  
GXWaitWhileString 174, 175  
GXWebAppConfig 343  
GXWebEmulatorConfig 438  
GXWindowConfig 443  
GXWindowFrameTransformConfig 598

## H

handleEvent 505  
hasAlarm 116  
hasChildren 529  
hasContent 606  
hashCode 9, 161, 706  
hashtableToNameValueString 702  
HOSTKEYS\_BUTTONS 595  
HOSTKEYS\_DELETE 595  
HOSTKEYS\_LINKS 595  
HTML\_TAG\_ID 635

## I

IMPORT\_APPLICATION\_CONFIG\_MISSING 295  
IMPORT\_APPLICATION\_COULD\_NOT\_RESTORE\_OVER  
RIDEN\_APP\_FROM\_BACKUP 295  
IMPORT\_APPLICATION\_ENTITIES\_FAILED 294  
IMPORT\_APPLICATION\_FAILED 294  
IMPORT\_APPLICATION\_HOST\_CATEGORY\_UNSUPPO  
RTED 295  
INDEXATION\_PROCESS\_FAILED 294  
initialize 551  
insert 539, 541, 543  
instance 556, 713  
isAddDocTypeToFrameset 356  
isAdjustNeighbours 574, 589  
isAppField 72, 90, 106, 202  
isAttachInputTagEvents 441  
isAutoCursorJump 72, 90, 201  
isAutoEnter 72, 89, 201

- isAutomaticDownload 451
  - isAutoSkipAllFields 427
  - isBlinking 73, 90, 201
  - isBlinkingCaret 434
  - isBlockNumericFields 431
  - isBoundsIncludeBorder 126
  - isCaretAtEndOfField 431
  - isCaseSensitive 25
  - isCell 208
  - isChecked 502, 516
  - isCheckHostScreenUpdate 440
  - isClientBlinking 569
  - isColumnAppfieldColumn 205
  - isComplete 5, 606, 618
  - isCurrentPrint 618
  - isDataTransferCommandSynchronous 100
  - isDebugMode 66, 138, 143, 158
  - isDirectionRTL 116
  - isDisplayChanged 114
  - isDoJapaneseDataTypeValidityCheck 434
  - isDownloadAtOnce 453
  - isEmphasis 606
  - isEmpty 619
  - isEmulationPasteBehavior 435
  - isEnabled 390, 592
  - isFailOnWarning 61, 335
  - isFirstChunk 130
  - isFullDetails 158
  - isHandleMenuSelection 432
  - isHideOnly 599
  - isHighIntensified 72, 89, 201
  - isHostKeyField 73
  - isInstant 322
  - isLoadEmulationComponent 439
  - isLoadPrinter 438
  - isLocked 116
  - isLogoffOnBrowserClose 417
  - isMasterRow 209, 222, 251
  - isMemberOf 118
  - isMultiple 73, 90, 200
  - isNaturalDataTransferSupported 52
  - isOpenWindowCenter 444
  - isPassThroughData 618
  - isPassword 536
  - isPositionInTable 205
  - isPrimaryKeyValid 211, 269, 271, 273
  - isPromptBeforeSessionTimeout 352
  - isPromptOnBrowserClose 420
  - isProtected 72, 89, 200, 266
  - isReadOnly 110
  - isRenderAlternatingCssClass 589
  - isRenderAppFieldCss 569
  - isRenderEmulationAttributes 568
  - isRenderIntensifiedCss 569
  - isRenderTagsFromRightToLeft 477
  - isReversedVideo 73, 91, 202
  - isRowValid 210, 269, 271, 273
  - isSelectFieldOnFocus 433
  - isShowBlinkingFields 433
  - isShowIntermediateScreen 334
  - isShowLogConsole 419
  - isSignOnScreen 382
  - isSupportPartialPageRendering 354
  - isSupportTypeAhead 440
  - isTabbingRTL 116
  - isTabOnInputsOnly 429
  - isTagVisible 657
  - isToCopy 222, 251
  - isTrimFields 52
  - isUnderlined 73, 91, 202
  - isUseActiveX 452
  - isUseCalendar 352
  - isUseFramesForSubmit 352
  - isUseJavaScriptsAsResource 351
  - isUseKeyboardPfKeys 430
  - isUseOldJSEngine 351
  - isUseScriptsInPage 351
  - isVariable 161
  - isVariableArray 145
  - isVariablesModified 334
  - isVisible 72, 89, 200, 534
  - isWaitForSystemUnlock 170
  - isWindow 114
  - isWindowRelative 589
  - isWindowRelativePositioning 16
- L**
- LABEL\_FOCUS\_BOLD 426
  - LABEL\_FOCUS\_NONE 426

LABEL\_FOCUS\_UNDERLINE 426  
LOCAL\_FILE 709  
LOG\_LEVEL\_DEBUG 416  
LOG\_LEVEL\_ERROR 416  
LOG\_LEVEL\_MESSAGE 416  
LOG\_LEVEL\_WARNING 416  
LOG\_MODE\_DEBUG 609  
LOG\_MODE\_NORMAL 609  
LogHistory 360  
LogLevel 360

## M

MAIN\_FRAME 365, 721  
MAPPING\_EXCEEDS\_LINE 293  
MAPPING\_OVERLAP 293  
MAPPING\_UTILITY\_FAILED 294  
MaxFileSize 360  
MESSAGE\_PATH\_BY\_STEP\_COMPLETE 283  
MESSAGE\_PATH\_COMPLETE 284  
MODE 709

## N

navigateTo 84  
NEIGHBOR\_ABOVE 545  
NEIGHBOR\_BELOW 545  
NEIGHBOR\_LEFT 546  
NEIGHBOR\_RIGHT 546  
newButton 557  
newCheckbox 558  
newCombobox 557  
newGenericTag 558  
newHiddenTag 558  
newHtmlString 559  
newImage 559  
newLabel 556  
newLink 557  
newRadioButton 558  
newRadioButtonGroup 557  
newStaticLabel 556  
newTable 556  
newTableCell 557  
newTableHeader 561  
newTableRow 557

newTextField 556  
NO\_ERROR 284  
NO\_WARNING 283  
NULL 123

## O

OBJECT\_ALREADY\_EXISTS 293  
OBJECT\_NOT\_DELETED 293  
OBJECT\_NOT\_FOUND 293  
OBJECT\_NOT\_UPDATED 293  
onComplete 483, 578  
onNewButton 487, 564, 584  
onNewCheckbox 488, 564, 584  
onNewCombobox 488, 564, 584  
onNewLabel 487, 564, 584  
onNewLink 487, 564, 584  
onNewRadioButtonGroup 488, 564, 584  
onNewTable 488, 565, 585  
onNewTableCell 489, 565, 585  
onNewTableRow 488, 565, 585  
onNewTextField 487, 564, 584  
OVERRIDE 123

## P

PASSWORD 709  
performSingleSignOn 376, 379  
PerSessionID 361  
putFile 713

## R

readExternal 4, 8, 13, 16, 27, 48, 61, 68, 79, 128, 133, 138, 143, 147, 152, 154, 158, 191, 238, 241, 243, 255, 259, 264  
readFileToUpload 718  
recognizeScreen 378, 383, 386  
recycle 548, 551, 552  
recycleBodyTag 661  
recycleButton 560  
recycleCheckbox 561  
recycleCombobox 560  
recycleGenericTag 560  
recycleHidden 561  
recycleHtmlString 561

- recycleImage 561
- recycleLabel 559
- recycleLink 560
- recycleRadioButton 560
- recycleRadioButtonsGroup 560
- recycleTable 559
- recycleTableCell 559
- recycleTableRow 559
- recycleTag 661
- recycleTextField 559
- refreshScreen 84
- rejectTransfer 98
- REMOTE\_FILE 709
- remove 522, 543
- removeAllListeners 476
- removeAllServerURLs 602
- removeAttribute 534
- removeColumn 189, 214, 225, 254
- removeFilter 233, 262
- removeInputField 16
- removeItem 230, 258, 259
- removeRow 233, 262
- removeTagAttribute 658
- removeVariable 334
- renderAndRecycle 702
- renderInto 553
- replace 522, 543
- REPOSITORY\_NOT\_AVAILABLE 293
- REPOSITORY\_NOT\_CREATED 293
- REPOSITORY\_NOT\_SYNCHRONIZED 295
- RESTRICTED\_ACCESS\_TO\_URL 297
- RETRIEVE\_INSIDE\_AREA 492
- RETRIEVE\_OUTSIDE\_AREA 492
- S**  
  
saveUserNameLocally 371
- sendKeys 81
- SERVER 708
- SESSION\_CANCELED 297
- SESSION\_CANCELED\_BY\_ADMIN 297
- SESSION\_CANCELED\_BY\_INACTIVE\_TIMEOUT 297
- set\_ActionKey 396
- set\_AddDocTypeToFrameset 356
- set\_AddJavascript 348
- set\_AdjustNeighbours 576, 590
- set\_Alignment 576
- set\_AllowTypeAhead 345
- set\_AlternateColors 404
- set\_AlternateCssName 405
- set\_ApplicationName 330
- set\_AttachInputTagEvents 441
- set\_AutomaticDownload 452
- set\_AutoSkipAllFields 427
- set\_BlinkingCaret 434
- set\_BlockNumericFields 431
- set\_CaptureBrowserClose 348
- set\_CaptureTagsEvents 318
- set\_CaretAtEndOfField 432
- set\_CheckboxCss 576
- set\_CheckHostScreenUpdate 441
- set\_ClientBlinking 570
- set\_CloneComboxOnClientSide 407
- set\_ColorMode 570
- set\_ColorSets 431
- set\_ComboboxCss 577
- set\_DebugModules 419
- set\_DefaultExtension 452
- set\_DefaultFrameColumns 420
- set\_DefaultOpenedWindowHeight 446
- set\_DefaultOpenedWindowLeft 446
- set\_DefaultOpenedWindowTop 446
- set\_DefaultOpenedWindowWidth 446
- set\_Description 330
- set\_DesignMode 308
- set\_DocTypeStr 357
- set\_DoJapaneseDataTypeValidityCheck 434
- set\_DoubleClickActionKey 428
- set\_DownloadAtOnce 453
- set\_DynamicDisableCssClass 347, 405
- set\_DynamicFieldCheck 403
- set\_EmulationConfig 473
- set\_EmulationPasteBehavior 435
- set\_Enabled 391, 593
- set\_Encoding 408
- set\_EncryptData 372
- set\_EngineConfig 411
- set\_ErrorForm 311
- set\_ExternalCssRequestParamName 350
- set\_FailOnWarning 336

set\_FieldsModification 321  
set\_FieldTypesInUse 309  
set\_Flicker 372  
set\_FontFamily 476  
set\_FontSize 473  
set\_FromRow 401  
set\_FtpConfig 323  
set\_GenerateTrOnClickEvent 402  
set\_GuiElementsConfig 478  
set\_HandleMenuSelection 432  
set\_HideOnly 599  
set\_HostAddress 366  
set\_HostDownloadForm 454  
set\_HostKeyCss 582  
set\_HostKeysConfig 474  
set\_HostNewPassword 336  
set\_HostPassword 332  
set\_HostType 366  
set\_HostUserName 329  
set\_IndexForm 323  
set\_Instant 322  
set\_InstantConfig 321  
set\_InstantEmulatorConfig 412  
set\_InstantForm 310  
set\_InstantWinForm 311  
set\_IsInstant 322  
set\_LabelFocus 428  
set\_LoadEmulationComponent 441  
set\_LoadPrinter 439  
set\_LogLevel 417  
set\_LogOffForm 345  
set\_LogoffMessage 421  
set\_LogoffOnBrowserClose 417  
set\_MacroConfig 316  
set\_MacrosFolder 371  
set\_MapPath 315  
set\_MozillaLogoffMessage 418  
set\_NaturalDataTransferConfig 350  
set\_OpenedWindowAttributes 447  
set\_OpenWindowCenter 447  
set\_Password 331  
set\_PasswordFieldName 388  
set\_PathName 394  
set\_PerformSyncCheckInInstant 319  
set\_PopupCloseSentKey 447  
set\_ProceduresConfig 323  
set\_PromptBeforeSessionTimeOut 353  
set\_PromptOnBrowserClose 420  
set\_RadioButtonCss 577  
set\_Rectangle 472  
set\_ReflectBGColor 406  
set\_ReflectBGColors 312  
set\_ReflectEmulationBehaviour 313  
set\_ReflectFGColor 406  
set\_ReflectFGColors 312  
set\_ReflectHostCursor 313  
set\_ReflectHostProtected 315, 403  
set\_ReflectMaxlength 314  
set\_RenderAlternatingCssClass 590  
set\_RenderAppFieldCss 570  
set\_RenderArea 471  
set\_RenderEmulationAttributes 570  
set\_RenderingType 581  
set\_RenderIntensifiedCss 570  
set\_RenderTagsFromRightToLeft 477  
set\_RowSpacingPercentageFactor 477  
set\_SaveUserNameLocally 372  
set\_ScreenLocker 346  
set\_ScreenLockerText 355  
set\_ScreenName 388  
set\_SelectFieldOnFocus 433  
set\_ServerURL 328  
set\_ServiceName 332  
set\_SessionConfig 316  
set\_SessionId 329  
set\_ShowBlinkingFields 434  
set\_ShowIntermediateScreen 335  
set\_ShowLogConsole 420  
set\_SingleSignOnConfig 323  
set\_SsoConfig 323  
set\_SupportDupAndFieldMark 309  
set\_SupportPartialPageRendering 356  
set\_SupportTypeAhead 440  
set\_TableAlternating1CssClass 590  
set\_TableAlternating2CssClass 590  
set\_TableBuildConfig 320  
set\_TableConfig 475  
set\_TableCssClass 590  
set\_TabOnInputsOnly 429  
set\_TerminationPath 356

set\_TitleWindowFrameCssClass 599  
set\_ToRow 402  
set\_TrimLabels 310, 407  
set\_UploadForm 454  
set\_UseActiveX 453  
set\_UseArrows 430  
set\_UseCache 344  
set\_UseCalendar 355  
set\_UseFolders 317  
set\_UseFramesForSubmit 355  
set\_UseJavaScriptsAsResource 351  
set\_UseKeyboardPFKeys 308  
set\_UseKeyboardPfKeys 430  
set\_UseModalWindows 317  
set\_UseOldJSEngine 355  
set\_UserExitsFile 439  
set\_UserName 371  
set\_UserNameFieldName 388  
set\_UseScreenGroups 318  
set\_UseScriptsInPage 352  
set\_UseStateManager 346  
set\_Variables 333  
set\_VirtualDir 344  
set\_WebDownloadForm 454  
set\_WebEmulatorConfig 412  
set\_WebWindowName 356  
set\_WindowConfig 412  
set\_WindowFrameConfig 475  
set\_WindowFrameCssClass 599  
set\_WindowRelative 591  
set\_WritePerformanceLog 349  
setActionKey 396  
setAddDocTypeToFrameset 356  
setAdditional 718  
setAddJavascript 348  
setAdjustNeighbours 574, 589  
setAlign 687  
setAlignment 513, 574  
setAllowTypeAhead 345  
setAlternateColors 404  
setAlternateCssName 404  
setAppFieldFullName 250  
setAppFieldName 221, 250  
setApplicationName 11, 330  
setArea 492  
setAttachInputTagEvents 441  
setAttribute 533  
setAttributeListIfEmpty 659  
setAttributes 686  
setAutomaticDownload 452  
setAutoSkipAllFields 427  
setBgcolor 687  
setBlinkingCaret 434  
setBlockNumericFields 431  
setBodyAttribute 660, 697  
setBodyAttributeIfEmpty 659  
setBorder 645, 653  
setCaption 221, 250, 501, 504, 517  
setCaptureBrowserClose 348  
setCaptureTagsEvents 319  
setCaretAtEndOfField 432  
setCellpadding 645, 653  
setCellspacing 645, 653  
setCharactersPerLine 610, 620  
setCheckboxCss 575  
setChecked 503, 516, 630, 642, 648  
setCheckedValue 503, 516, 630  
setCheckHostScreenUpdate 440  
setClientBlinking 569  
setClientId 13  
setCloneComboxOnClientSide 407  
setColorMode 568  
setColorSets 430  
setCols 663  
setComboboxCss 575  
setComplete 6, 606  
setConfigFile 603  
setConnectionPool 59  
setContent 217, 246  
setCopyCount 222, 251  
setCornersCoordinates 190  
setCssClass 685  
setCursor 152  
setData 48, 49, 130  
setDataTypes 220, 249  
setDebugMode 66, 138, 143, 158  
setDebugModules 419  
setDefaultExtension 452  
setDefaultFrameColumns 419  
setDefaultOpenedWindowHeight 444

setDefaultOpenedWindowLeft 444  
setDefaultOpenedWindowTop 445  
setDefaultOpenedWindowWidth 445  
setDescription 331  
setDesignMode 308  
setDestinationScreenName 138  
setDir 641, 645, 650, 672  
setDocTypeStr 357  
setDoJapaneseDataTypeValidityCheck 434  
setDoubleClickActionKey 427  
setDownloadAtOnce 453  
setDynamicDisableCssClass 347, 405  
setDynamicFieldCheck 403  
setEmulationConfig 473  
setEmulationPasteBehavior 435  
setEnabled 390, 593  
setEncoding 408  
setEncryptData 372  
setEngineConfig 411  
setEntireTag 493  
setErrorForm 311  
setExternalCssRequestParamName 350  
setFailOnWarning 61, 335  
setFieldIndex 65  
setFieldName 65  
setFieldsModification 322  
setFieldTypesInUse 310  
setFirstChunk 130  
setFlicker 19, 372  
setFontFamily 476  
setFontSize 472  
setFormAttribute 660, 697  
setFromRow 401  
setFtpConfig 320  
setFullDetails 158  
setGenerateTrOnClickEvent 402  
setGuiElementsConfig 474  
setGx\_context 636, 689, 700  
setHandleMenuSelection 432  
setHeaderCoordinates 190  
setHeight 687  
setHideOnly 599  
setHorizontal 645  
setHostAddress 366  
setHostDownloadForm 451  
setHostKeyCss 582  
setHostKeysConfig 474  
setHostKeysTagUserExit 354  
setHostNewPassword 60, 335  
setHostPassword 60, 332  
setHostType 366  
setHostUserName 60, 330  
setHref 626  
setId 529, 551, 685  
setIndex 106  
setIndexForm 323  
setInnerText 548  
setInputTagAttribute 659  
setInstantConfig 321  
setInstantEmulatorConfig 411  
setInstantForm 311  
setInstantWinForm 311  
setIsCaseSensitive 25  
setIsHeader 523  
setIsInstant 322  
setIsMasterRow 222, 251  
setIsToCopy 222, 251  
setIsWrapTextOnMPP 606  
setItem 228, 258  
setItemContent 229, 235, 258, 263  
setItems 506, 513  
setKeys 152  
setLabelFocus 428  
setLength 533  
setLinesPerPage 610, 620  
setLoadEmulationComponent 439  
setLoadPrinter 438  
setLocalFileBytes 718  
setLogLevel 417, 610  
setLogOffForm 345  
setLogoffMessage 418  
setLogoffOnBrowserClose 417  
setLowerRightPosition 21, 25  
setMacroConfig 316  
setMacrosFolder 370  
setMapName 138  
setMapPath 315  
setMaxlength 641  
setMozillaLogoffMessage 418  
setMultiple 650

setName 106, 124, 163, 189, 220, 250, 516  
setNaturalDataTransferConfig 350  
setNumProcessedRecords 6  
setOffsetLeft 531  
setOffsetRight 531  
setOffsetTop 530  
setOnChange 641, 650  
setOnClick 501, 510  
setOnclick 626, 630, 640, 648, 650, 687  
setOnkeydown 641  
setOnkeypress 641  
setOnkeyup 641  
setOnload 669  
setOnmouseout 626  
setOnmouseover 626  
setOnserverclick 626, 640  
setOnsubmit 633  
setOpenedWindowAttributes 445  
setOpenWindowCenter 444  
setPageContext 635, 685, 700  
setPageVisible 661, 694  
setPanelToUpdateContent 661  
setPassword 331, 536, 717  
setPasswordField 375, 382  
setPasswordFieldname 387  
setPasswordParameterName 457  
setPathExecutionLog 147  
setPathName 143, 394  
setPerformSyncCheckInInstant 319  
setPopupCloseSentKey 446  
setPosition 64, 105, 531, 532  
setPrimaryKey 190, 215, 258  
setProceduresConfig 321  
setPromptBeforeSessionTimeOut 353  
setPromptOnBrowserClose 420  
setRadioButtonCss 575  
setReflectBGColor 406  
setReflectBGColors 312  
setReflectEmulationBehaviour 313  
setReflectFGColor 405  
setReflectFGColors 312  
setReflectHostCursor 314  
setReflectHostProtected 315, 403  
setReflectMaxlength 314  
setRemoteFile 717  
setRenderAlternatingCssClass 589  
setRenderAppFieldCss 569  
setRenderArea 471  
setRenderEmulationAttributes 568  
setRenderingType 581  
setRenderIntensifiedCss 569  
setRenderTagsFromRightToLeft 477  
setReplayScreen 111  
setRetrievalMode 493  
setRowRange 493  
setRows 664  
setRowSpacingPercentageFactor 477  
setSaveUserNameLocally 371  
setScreenLocker 346  
setScreenLockerText 347  
setScreenName 22, 387  
setScreenNames 222, 251  
setSelectedValue 507, 513  
setSelectFieldOnFocus 433  
setSeqScreenNumber 17  
setServer 717  
setServerContext 12, 334  
setServers 602  
setServerURL 12, 329  
setServiceName 332  
setSessionConfig 316  
setSessionDescription 58  
setSessionID 11  
setSessionId 329  
setSessionPassword 12  
setShowBlinkingFields 433  
setShowFirstRowsWithoutMaster 241  
setShowIntermediateScreen 334  
setShowLogConsole 419  
setSignOnScreen 382  
setSignOnScreenRecognizer 391  
setSingleSignOnConfig 322  
setSize 641, 650  
setSrc 637, 641  
setSsoExecuter 391  
setString 25  
setStyle 686  
setSupportDupAndFieldMark 309  
setSupportFileUpload 633  
setSupportPartialPageRendering 354

setSupportTypeAhead 440  
setTabIndex 641  
setTableAlternating1CssClass 588  
setTableAlternating2CssClass 588  
setTableBuildConfig 320  
setTableConfig 475  
setTableCssClass 588  
setTableName 158  
setTabOnInputsOnly 429  
setTagAttribute 658, 694  
setTagContent 656, 693  
setTagContentIfEmpty 656  
setTagList 657, 660, 694  
setTagName 549, 553  
setTagTable 657, 658, 695  
setTagText 552  
setTagVisible 657, 694  
setTagVisibleIfEmpty 657  
setTarget 510  
setTerminationPath 353  
setText 508, 510, 524, 537  
setTimeout 18  
setTitle 686  
setTitleWindowFrameCssClass 598  
setToRow 402  
setTrimLabels 310, 407  
setType 641  
setUncheckedValue 503, 630  
setUploadForm 451  
setUpperLeftPosition 21, 25  
setUseActiveX 453  
setUseArrows 429  
setUseCache 344  
setUseCalendar 352  
setUseFolders 317  
setUseFramesForSubmit 352  
setUseHostService 59  
setUseJavaScriptsAsResource 351  
setUseKeyboardPFKeys 309  
setUseKeyboardPfKeys 430  
setUseModalWindows 317  
setUseOldJSEngine 351  
setUser 718  
setUserExitsFile 439  
setUserField 375, 382

setUserName 371  
setUserNameFieldName 388  
setUserParameterName 457  
setUserPasswordRetriever 390  
setUseScreenGroups 318  
setUseScriptsInPage 351  
setUseStateManager 346  
setValue 106, 550, 640, 648, 716  
setValues 163  
setVariableName 221, 250  
setVariables 8, 333  
setVirtualDir 344  
setVisible 534, 686  
setWaitCondition 610  
setWaitForResponseTimeout 603  
setWebDownloadForm 451  
setWebEmulatorConfig 412  
setWebWindowName 354  
setWidth 687  
setWindowConfig 412  
setWindowFrameConfig 475  
setWindowFrameCssClass 598  
setWindowRelative 589  
setWindowRelativePositioning 16  
setWritePerformanceLog 349  
ShortLongDescription 362  
SIMPLE 365, 721  
sort 189, 216, 236, 263  
splitLabel 578  
startDownload 96  
startUpload 97

## T

toString 17, 65, 107, 143, 152, 191, 202, 251, 255, 259, 264, 273, 335, 623, 706  
TraceSessions 362  
TraceSessionsIDs 362  
TYPE\_APPFIELD 87  
TYPE\_HOSTFIELD 87  
TYPE\_HOSTKEY\_FIELD 88  
TYPE\_MULTIPLE\_APPFIELD 88

## U

UPDATE\_NONE\_EXISTING\_ENTITY 295

USER 709

## W

WARNING\_30\_DAYS\_EVAL\_VERSION 292

WARNING\_60\_DAYS\_EVAL\_VERSION 292

WARNING\_90\_DAYS\_EVAL\_VERSION 292

WARNING\_DURING\_CONNECT 283

WARNING\_EVALUATION\_EXPIRED 292

WARNING\_FIELD\_CONTENT\_ALREADY\_EXIST 283

WARNING\_FIELD\_IS\_PROTECTED 283

WARNING\_PATH\_SCREEN\_NOT\_ADDED\_TO\_RESPONS

E 283

WARNING\_SCREEN\_IMAGE\_INSERT\_FAILED 283

WARNING\_WAIT\_CONDITION\_FAILED 283

writeComplexRadio 701

writeExternal 4, 9, 13, 17, 27, 48, 61, 68, 79, 128, 133, 139,  
143, 147, 152, 155, 159, 191, 238, 241, 243, 255, 259, 264

writeRadio 702