DBRADIOBUTTON

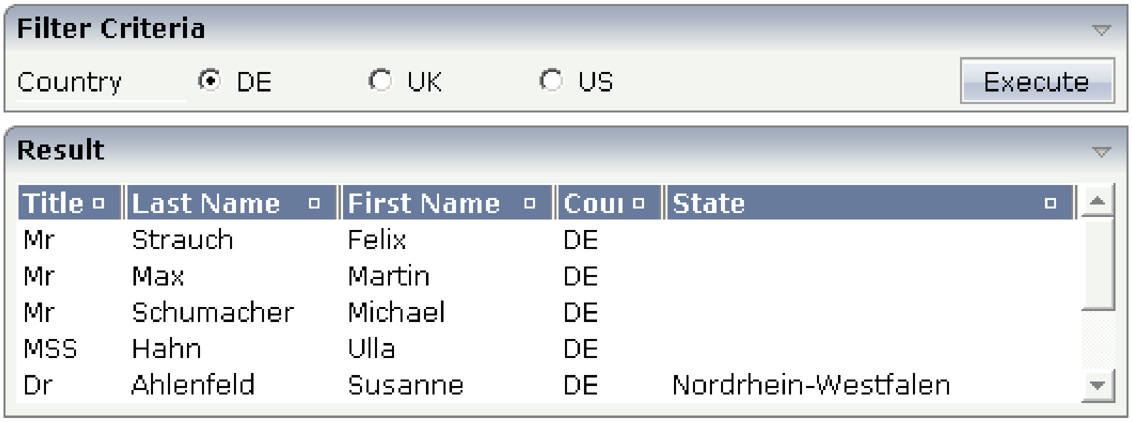
The DBRADIOBUTTON control represents a filter criterion of a database query. Compared to a "normal" RADIOBUTTON control, the only benefit is the convenient way of appending the filter criterion to the SELECT statement.

This chapter covers the following topics:

- Example
- Properties

Example

The following image shows an example in which the DBRADIOBUTTON is used for the filter criterion "Country":



Title	Last Name	First Name	Cou	State
Mr	Strauch	Felix	DE	
Mr	Max	Martin	DE	
Mr	Schumacher	Michael	DE	
MSS	Hahn	Ulla	DE	
Dr	Ahlenfeld	Susanne	DE	Nordrhein-Westfalen

Have a look at the XML layout definition:

```
<rowarea name="Filter Criteria">
    <itr>
        <label name="Country" width="80">
        </label>
        <dbradiobutton valueprop="dBRadioButtonSex" querycolumn="COUNTRY" value="DE">
        </dbradiobutton>
        <hdist>
        </hdist>
        <label name="DE" asplaintext="true">
        </label>
        <hdist width="40">
        </hdist>
        <dbradiobutton valueprop="dBRadioButtonSex" querycolumn="COUNTRY" value="UK">
        </dbradiobutton>
        <hdist>
        </hdist>
        <label name="UK" asplaintext="true">
        </label>
        <hdist width="40">
        </hdist>
        <dbradiobutton valueprop="dBRadioButtonSex" querycolumn="COUNTRY" value="US">
        </dbradiobutton>
        <hdist>
        </hdist>
        <label name="US" asplaintext="true">
        </label>
        <hdist>
        </hdist>
        <hdist width="100%">
        </hdist>
        <button name="Execute" method="onExecute">
        </button>
    </itr>
</rowarea>
<vdist>
</vdist>
<rowarea name="Result" height="140">
    <itr height="100%">
        <textgridsss2 griddataprop="result" rowcount="5" width="100%">
            <column name="Title" property="TITLE" width="50">
            </column>
            <column name="Last Name" property="LASTNAME" width="100">
            </column>
            <column name="First Name" property="FIRSTNAME" width="100">
            </column>
            <column name="Country" property="COUNTRY" width="50">
            </column>
            <column name="State" property="STATE" width="100%">
            </column>
        </textgridsss2>
    </itr>
</rowarea>
```

The corresponding adapter code is:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;

import com.softwareag.cis.server.Adapter;
import com.softwareag.cis.server.util.DBRADIOBUTTONInfo;
import com.softwareag.cis.server.util.DBTEXTGRIDCollection;
import com.softwareag.cis.server.util.DBUtil;
import com.softwareag.cis.server.util.DelegateError;
import com.softwareag.cis.server.util.IDBCondition;
import com.softwareag.cis.server.util.IDBQUERYConnectionProvider;

// This class is a generated one.

public class DBRadioButton_Adapter
    extends Adapter
    implements IDBQUERYConnectionProvider, IDBDemoAdapter
{
...
    // property >dBRadioButtonSex<
    DBRADIOBUTTONInfo m_dBRadioButtonSex = new DBRADIOBUTTONInfo();
    public DBRADIOBUTTONInfo getDBRadioButtonSex() { return m_dBRadioButtonSex; }

    // property >result<
    DBTEXTGRIDCollection m_result = new DBTEXTGRIDCollection();
    public DBTEXTGRIDCollection getResult() { return m_result; }
    public void setResult(DBTEXTGRIDCollection value) { m_result = value; }
...
    public void onExecute()
    {
        try
        {
            StringBuffer sb = new StringBuffer();
            sb.append("SELECT * FROM BUSINESSPARTNER INNER JOIN ADDRESS ON BUSINESSPARTNER.ID =
                                ADDRESS.BUSINESSPARTNERID");
            DBUtil.addToQuery(sb, new IDBCondition[] { m_dBRadioButtonSex }, true);
            Class.forName("org.hsqldb.jdbcDriver");
            Connection con = DriverManager.getConnection("jdbc:hsqldb:hsql://localhost", "sa", "");
            ResultSet rs = con.createStatement().executeQuery(sb.toString());
            m_result.initWithResultSet(rs);
        }
        catch (Exception exc)
        {
            throw new DelegateError(exc);
        }
    }
...
}
```

The adapter property `dBRadioButtonCountry` is of type `DBRADIOBUTTONInfo` (from the package `com.softwareag.cis.server.util`). This class has a property value that is manipulated by the DBRADIOBUTTON control. The `DBRADIOBUTTONInfo` implements (like all DB controls) the interface `IDBCondition` (`com.softwareag.cis.server.util`). With class `DBUtil` (`com.softwareag.cis.server.util`), there is a convenient way to append the filter value to the SQL query. See the JavaDoc documentation of class `DBRADIOBUTTONInfo` and `DBUtil` for details.

For displaying the result, the class `DBTEXTGRIDCollection` (from the package `com.softwareag.cis.server.util`) is used. This class extends `TEXTGRIDCollection` by the ability to initialise the collection with a result set (`java.sql.ResultSet`). For each line of the result set, it creates an object of class `DBTEXTGRIDLine` (package `com.softwareag.cis.server.util`). Class `DBTEXTGRIDLine` implements the interface `IDynamicAccess`. With this, "normal" text grid controls can be used to visualize the data of the `DBTEXTGRIDCollection`.

Properties

Basic			
valueprop	Property that returns a DBRADIOBUTTONInfo -instance. With class DBUtil this instance provides for a convenient way to append the filter value to query string.	Obligatory	
querycolumn	Name of the column in the query to that the filter criteria belongs to. This column may differ from the value help table/column (properties VALUEHELPTABLE, VALUEHELPCOLUMN). The name is used to build a SQL string in method "toSQLString".	Obligatory	
value	Value that represents this instance of the DBRADIOBUTTON control. The value is set into the adapter property that is defined by the VALUEPROP attribute when the user clicks onto the control. - Vice versa: the control is switched to "marked" when the adapter property holds the value defined.	Obligatory	
displayonly	If set to "true", the displayed checkbox can not be changed. The default is "false".	Optional	true false

flush	Flush reaction as described in "Common Rules"	Optional	screen server
width	Width of the control in pixels or as percentage value. This does not change the size of the "box" of the CHECKBOX, but the size of the column in which the CHECKBOX is located.	Optional	100 120 140 160 180 200 50% 100%
align	Explicit Alignment	Optional	left center right
valign	Explicit Alignment	Optional	top middle bottom
colspan	Number of columns occupied by this control.	Optional	1 2 3 4 5 50 int-value

rowspan	Number of rows occupied by this control.	Optional	1 2 3 4 5 50 int-value
helpid	Identifier that is used for building the URL of the online help page. Please refer to "Online Help Management" for details.	Optional	
comment	Comment without any effect on rendering and behaviour. The comment is shown in the layout editor’s tree view.	Optional	