

DBCHECKBOX

The DBCHECKBOX control represents a filter criterion of a database query of type Boolean. Compared to a "normal" CHECKBOX 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 DBCHECKBOX is used for the filter criterion "Female":

The screenshot shows a web application interface with two main sections. The top section, titled "Filter Criteria", contains a label "Female" followed by a checked checkbox and an "Execute" button. The bottom section, titled "Result", displays a table with the following data:

| Title | Last Name | First Name | Cou | State |
|-------|-----------|------------|-----|---------------------|
| MSS | Hahn | Ulla | DE | |
| Dr | Ahlenfeld | Susanne | DE | Nordrhein-Westfalen |
| Dr | Ahlenfeld | Susanne | DE | Nordrhein-Westfalen |

Have a look at the XML layout definition:

```
<rowarea name="Filter Criteria">
  <itr>
    <label name="Female" width="80">
    </label>
    <dbcheckbox valueprop="dbCheckboxFemale" querycolumn="FEMALE">
    </dbcheckbox>
    <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.DBCHECKBOXInfo;
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 DBCHECKBOX_Adapter
    extends Adapter
    implements IDBQUERYConnectionProvider, IDBDemoAdapter
{
    // -----
    // members
    // -----
    ...
    // property >DBCheckboxFemale<
    DBCHECKBOXInfo m_dBCheckboxFemale = new DBCHECKBOXInfo();
    public DBCHECKBOXInfo getDBCheckboxFemale() { return m_dBCheckboxFemale; }

    // property >result<
    DBTEXTGRIDCollection m_result = new DBTEXTGRIDCollection();
    public DBTEXTGRIDCollection getResult() { return m_result; }
    public void setResult(DBTEXTGRIDCollection value) { m_result = value; }
    ...
    // -----
    // public adapter methods
    // -----
    ...
    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_dBCheckboxFemale }, true);
            Class.forName("org.hsqldb.jdbcDriver");
            Connection con = DriverManager.getConnection("jdbc:hsqldb:hsqldb://localhost", "sa", "");
            ResultSet rs = con.createStatement().executeQuery(sb.toString());
            m_result.initWithResultSet(rs);
        }
        catch (Exception exc)
        {
            throw new DelegateError(exc);
        }
    }
}

```

The adapter property `dbCheckboxFemale` is of type `DBCHECKBOXInfo` (from the package `com.softwareag.cis.server.util`). This class has a property value that is manipulated by the `DBCHECKBOX` control. The `DBCHECKBOXInfo` 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 `DBCHECKBOXInfo` 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 <code>DBCHECKBOXInfo</code> -instance. With class <code>DBUtil</code> 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 <code>VALUEHELPTABLE</code> , <code>VALUEHELPCOLUMN</code>). The name is used to build a SQL string in method "toSQLString". | Obligatory | |
| flush | Flush reaction as described in "Common Rules" | Optional | screen server |
| displayonly | If set to "true", the displayed checkbox can not be changed. The default is "false". | Optional | true false |
| width | Width of the control in pixels or as percentage value. This does not change the size of the "box" of the <code>CHECKBOX</code> , but the size of the column in which the <code>CHECKBOX</code> 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 | |