

# JavaBean

Login SQLite Database

# Xpages Login

## System Requirements:

Download Domino Designer 8.5.2 Environment (DDE)

<http://www.ibm.com/developerworks/downloads/ls/dominodesigner/>

Download SQLite

<http://www.sqlite.org/download.html>

Download SQLiteJDBC jar file

<http://code.google.com/p/sqlite-jdbc/>

## Introduction:

Build a connection to a SQLite Database located on C Drive using a JavaBean and a database connector class. Once connection is established, load specific user to Xpages; user is sent to logOut page if credentials do not check out.

## Disclaimer:

Information contained in the following is presented as is. This tutorial assumes you have basic programming knowledge. All tutorials are based on an Eclipse/Eclipse-based software. Should you need to familiarize yourself with a certain Eclipse environment, prior to continuing this tutorial, please stop now and see our Tutorials page...

## Write Java classes to login

At this point we assume Domino Designer 8.5.2 is downloaded/installed and two Java classes have been created. You are going to use a database connector class with JavaBean to authenticate against SQLite database. Copy and paste below code to your environment for each code samples.

## DBConnector.java

[CODE]

```
/**
 * Copyright 2012 Dököll Solutions, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
```

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
*
*
* @AppName: Docu.nsf
* Program: DBConnector.java
* Created from Copy: 2009.11.03.1.17.PM
*
* Modified: 2011.08.28.3.55.PM
* Added SQLite connection
*/
```

```
package com.dokoll.solutions.inc.development.Utils;
```

```
/**
 *
 * @author Dököll Solutions, Inc.
 * @version 2009.11.03.1.17.PM
 */
```

```
import java.sql.*;
```

```
//load program
```

```
public class DBConnector {
    //get a connection

    public static Connection getConnection() {
        System.out.println("Connection Started...");
        //declare and initialize
        Connection con = null;
        //entering try catch
        try {
            //load driver class
            Class.forName("org.sqlite.JDBC").newInstance();
            System.out.println("Class obtained...");
            //connect to SQLite
            con =
DriverManager.getConnection("jdbc:sqlite:C:\\temp\\WEB_FILES\\SQLite\\sqlite-
3_7_3\\Test.db");
            //...
            System.out.println("Connection forwarded...");
        } catch (Exception e) {
            e.printStackTrace();
        }
        //...
        return con;
    }
}
```

```

    /*public static void main(String[] args) throws Exception {
        //Use this block to test as Java program
        getConnection();
        System.out.println("Program Successful...");
    }*/
}

```

### **Bean it!**

Your JavaBean has been created, and we assume it compiles okay. You are now ready to plug in the connector class to complete login code. You are going to need an Xpages form to submit request to login. Copy and paste below to your environment, areas of interest have been highlighted for your convenience.

## **LoginBeanSQL.java**

```

/**
 * Copyright 2012 Dököll Solutions, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 *
 * @AppName: Docu.nsf
 * Program: LoginBeanSQL.java
 * Created from Copy: 2009.11.03.1.17.PM
 *
 * Modified: 2012.04.07.5.36.AM
 * Added SQLite connection
 */
package com.dokoll.solutions.inc.development.Utils;

//load imports
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

/**
 * @author Dököll Solutions, Inc.
 * @version 2009.11.03.1.17.PM
 *
 */
public class LoginBeanSQL {
    String UserID;
    String PassWord;

    public LoginBeanSQL() {
        //...
    }
}

```

```

}

/**
 * @return the userID
 */
public String getUserID() {
    return UserID;
}

/**
 * @param userID the userID to set
 */
public void setUserID(String userID) {
    UserID = userID;
}

/**
 * @return the passWord
 */
public String getPassWord() {
    return PassWord;
}

/**
 * @param passWord the passWord to set
 */
public void setPassWord(String passWord) {
    PassWord = passWord;
}

//Xpages button code..
public String doGetUserCreds() {

    System.out.println("Entering Try catch Block...");
    try {

        //Declare and initialize variables
        PreparedStatement preparedStmt=null;
        //bring in connection from DBConnector class

        Connection con = DBConnector.getConnection();
        System.out.println("DataSource Obtained, connection made...");
        //set up query string (as a PreparedStatement)
        String query= "SELECT * FROM USER_LOGINS WHERE UserName=? AND Password
=?";

        preparedStmt = con.prepareStatement(query);
        //Grab values from user, pass to program
        preparedStmt.setString(1,UserID);
        preparedStmt.setString(2,PassWord);
        //run query
        ResultSet resultSt = preparedStmt.executeQuery();

        System.out.println("Query ran...");
        //do stuff based on user access
        while (resultSt.next()) {
            System.out.println("searching resultSet...");
            //Load the results to these variables to begin comparing
            //values entered against values received
            String DBUserName = resultSt.getString("UserName");

```

```

String DBPassword = resultSt.getString("Password");
//2012.04.10.10.29.PM
//TO DO: Log userID, time in, time out for debugging purposes
//compare user submission against values returned from SQLite
    if (UserID.equalsIgnoreCase(DBUserName)
        && Password.equalsIgnoreCase(DBPassword)) {
        //...
        System.out.println("It's a hit!");
        //...
        //load the appropriate Xpage

        return "xsp-success";
    } else {
        System.out.println("Hummm... Let's take a look at that!");
        //...
        //load the appropriate Xpage

        return "xsp-failure";
    }
}
}
//...
catch(Exception ex) {
    ex.printStackTrace();
}
//...
return UserID;
}
}
}

```

### **Build Xpage file**

Full code added below, jump ahead if necessary, areas of interest have been highlighted for your convenience.

## **xploginform.xsp**

```

<?xml version="1.0" encoding="UTF-8"?>
<xp:view xmlns:xp="http://www.ibm.com/xsp/core">

    <html>
        <head>
            <title>Login Form</title>

            <SCRIPT LANGUAGE="JavaScript">
if (window != top) top.location.href = location.href;
</SCRIPT>

            </head>
            <body>
                <font color="maroon" size="4">Welcome to Döcu</font><hr width="100%"
align="left"></hr><br></br><br></br>

```

```

        <xp:form>
            <center><xp:span style="font-size:20pt">Login
Form</xp:span><br></br>

            <table align="center" width="300" height="100">
                <tr>
                    <td style="background-color:rgb(128,128,64)">
                        <xp:span style="font-
weight:bold">UserID</xp:span></td>
                    <td style="background-color:rgb(128,128,64)">
                        <xp:inputText id="userID1"
value="#{LoginBeanSQL.userID}"
defaultValue="#{javascript:session.getCommonUserName()} "></xp:inputText>
                    </td>
                </tr>
                <tr>
                    <td style="background-color:rgb(128,128,64)">
                        <xp:span style="font-
weight:bold;background-color:rgb(128,128,64)"></xp:span><xp:span style="font-
weight:bold">PassWord
                    </xp:span></td>
                    <td style="background-color:rgb(128,128,64)">
                        <xp:inputText id="passWord1"
value="#{LoginBeanSQL.passWord}"
password="true">
                    </xp:inputText>
                </td>
                </tr>
                <tr>
                    <td style="background-
color:rgb(128,128,128)"></td>
                    <td style="background-color:rgb(128,128,128)">
                        <xp:button value="Submit"
id="button1">
                            <xp:eventHandler
event="onclick"
refreshMode="complete" immediate="false"
id="eventHandler1">
                                <xp:this.action><![
CDATA[#{javascript:LoginBeanSQL.doGetUserCreds()}]]></xp:this.
action>
                            </xp:eventHandler>
                        </xp:button>
                        <xp:button value="Reset"
id="button2">
                            <xp:eventHandler
event="onclick"
                                submit="true"
                            </xp:eventHandler>
                        </xp:button>
                    </td>
                </tr>
            </table>

```

```

refreshMode="complete" immediate="false"
save="true"
id="eventHandler2">
<xp:this.action>
<![CDATA[#{javascript:context.redirectToPage\("xploginform"\);}]>
</xp:this.action>
</xp:eventHandler>
</xp:button>
</td>
</tr>
</table>
<table width="300px" align="center"
style="border:0px solid #000000;background-color:#efefef;">
<tr>
<td>
<center><a href="xpissuesform.xsp">Technical
Support</a> &#160;||&#160;<a href="xpchangepassport.xsp">Change
Password</a><br>Copyright © Döcu 2012</br></center>
</td>
</tr>
</table>
</center>
</xp:form>
</body>
</html>
<xp:this.navigationRules><xp:navigationRule outcome="xsp-success"
viewId="/xpmain.xsp"></xp:navigationRule><xp:navigationRule outcome="xsp-failure"
viewId="/xplogoutlink.xsp"></xp:navigationRule></xp:this.navigationRules></xp:view>

```

### Conclusion:

You can now have users login using an Xpages form, send users to specific pages based on access.

**Added info:** You will need to reference [LoginBeansQL.java](#) in your faces-config.xml file.

Questions, comments, please post a brief message on our [Contact](#) form on the main site. Thank you for coming...