

Building Dynamic Java Web Applications (Part 2)

Glassfish, JAVA EE, Servlets, JSP, EJB

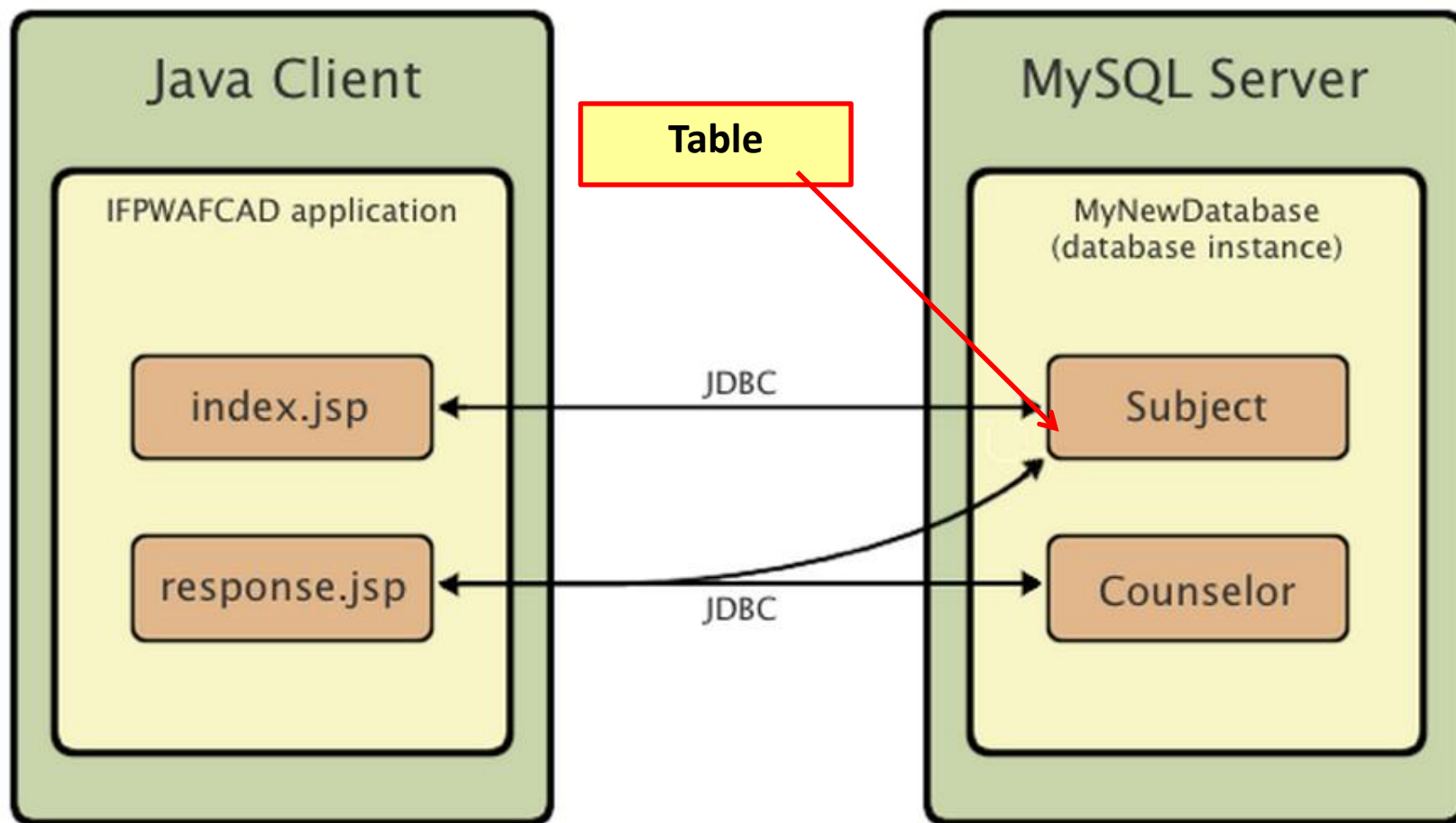


Example: Simple MySQL database

Software or Resource	Version Required
NetBeans IDE, Java bundle	6.8 or 6.9
Java Development Kit (JDK)	6
MySQL database server	5.1
MySQL Connector/J JDBC Driver	version 5.x
GlassFish server	v3 or Open Source Edition 3.0.1

necessary for communication between **Java platforms** and the **MySQL database protocol** (already included inside of NetBeans)

TWO-TIER Architecture: a Java web application communicates directly with a MySQL database using the Java Database



The welcome page (**index.jsp**) presents the user with a simple HTML form. When a browser requests index.jsp, the **JSTL code** within the page initiates a query on **MyNewDatabase**. It retrieves data from the **Subject database table**, and inserts it into the page before it is sent to the browser.

(Sample Run) Final Project

Welcome to IFPWAFCAD, the International Former Professional Wrestlers' Association for Counseling and Development!

IFPWAFCAD offers expert counseling in a wide range of fields.

To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:

Select a subject:

Index.jsp

Submit Subject selected

Response.jsp

The screenshot shows a web browser window with the address bar containing the URL: `http://localhost:8080/IFPWAFCAD/response.jsp?subject_id=3&submit=submit`. The browser tab is titled "Temper Management". The main content area displays a light green box with the following information:

Temper Management	
Description:	<i>Are your angry outbursts affecting your relationships with loved-ones? Do tantrums at work hinder your ability to perform? Temper management helps you to channel your anger into positive, life-changing productivity.</i>
Counselor:	Randy "Macho Man" Savage <i>member since: 2000-11-10</i>
Contact Details:	email: machoman@hotmail.com phone: 555 317-4444

MySQL: Resetting the root password

This example requires that the (username) **root's** password is **not** empty.

Resetting the **root** password

Create a password file:

New password

```
UPDATE mysql.user SET Password=PASSWORD('nbuser') WHERE User='root';  
FLUSH PRIVILEGES;
```

Save the file. For this example, the file will be named C:**mysql-init.txt**.

Start the **MySQL server** with the special **--init-file** option (notice that the backslash in the option value is doubled):

```
C:\Program Files (x86)\EasyPHP-5.3.3\mysql\bin>mysqld --init-file=c:\\mysql-init.txt
```

Displaying existing accounts

You can verify if you succeeded in changing the root password by issuing the following mysql command:

```
mysql> select user, host, password FROM mysql.user;
```

```
mysql> select user, host, password from mysql.user;
+-----+-----+-----+
| user | host      | password |
+-----+-----+-----+
| root | localhost | *9379FCAFCDE4054FE605E2045F76B4F3EBC10C97 |
| dug  | %         | *036DD09491CCE804D71BED5A5B5D33223B4C88F4 |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Encrypted passwords





**Creating a New
Project**

Create a New Project

From the menu, select **File/New Project/Java Web/Web Application**

**International Former Professional Wrestlers'
Association for Counseling and Development**

New Web Application

Steps

1. Choose Project
2. **Name and Location**
3. Server and Settings
4. Frameworks

Name and Location

Project Name: IFPWAFCAD

Project Location: Documents and Settings\nhreyes\My Documents\NetBeansProjects

Project Folder: and Settings\nhreyes\My Documents\NetBeansProjects\IFPWAFCAD

Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

Set as Main Project

Create a New Project

The screenshot shows the 'New Web Application' wizard in an IDE. The 'Steps' panel on the left indicates that the current step is '3. Server and Settings'. The 'Server and Settings' panel contains the following fields:

- Add to Enterprise Application:** <None>
- Server:** GlassFish Server 3 (with an 'Add...' button)
- Use dedicated library folder for server JAR files
- Java EE Version:** Java EE 5 (highlighted with a red arrow)
- Context Path:** /IFPWAFCAD

At the bottom of the wizard, there are navigation buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

We want to see how to create a deployment descriptor (**web.xml**), that's why we want to select **Java EE 5**

Result: Welcome Page (index.jsp)

```
...ava response.jsp x SQL Command 1 x ifpwafcad.sql x SQL Command 2 x SQL Command 3 x index.jsp x
<!--
  Document      : index
  Created on    : 7/10/2010, 00:05:59
  Author       : nhreyes
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=
    <title>JSP Page</title>
  </head>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```

comment

JSP Page directive

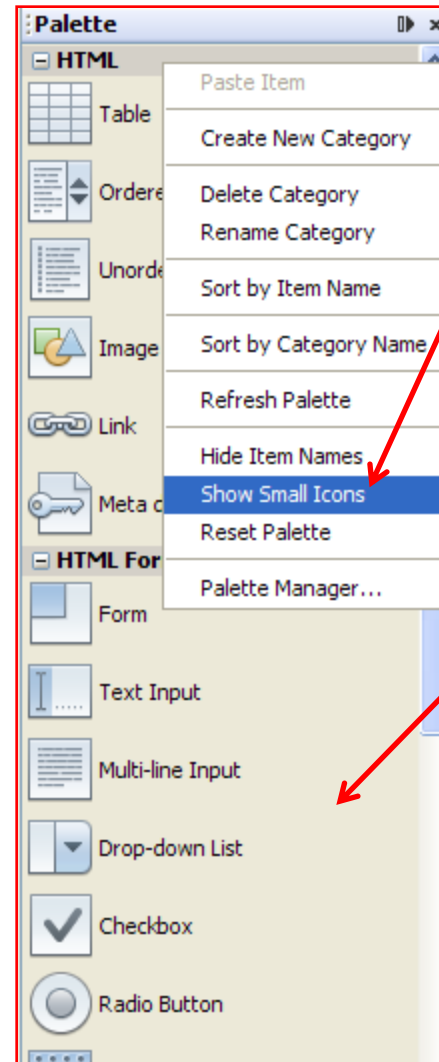
Add Table and Form here

We can easily add a table here using a **Palette**.

**Add some HTML
elements**

Invoke the **Palette Window**

From the menu, select Window/Palette

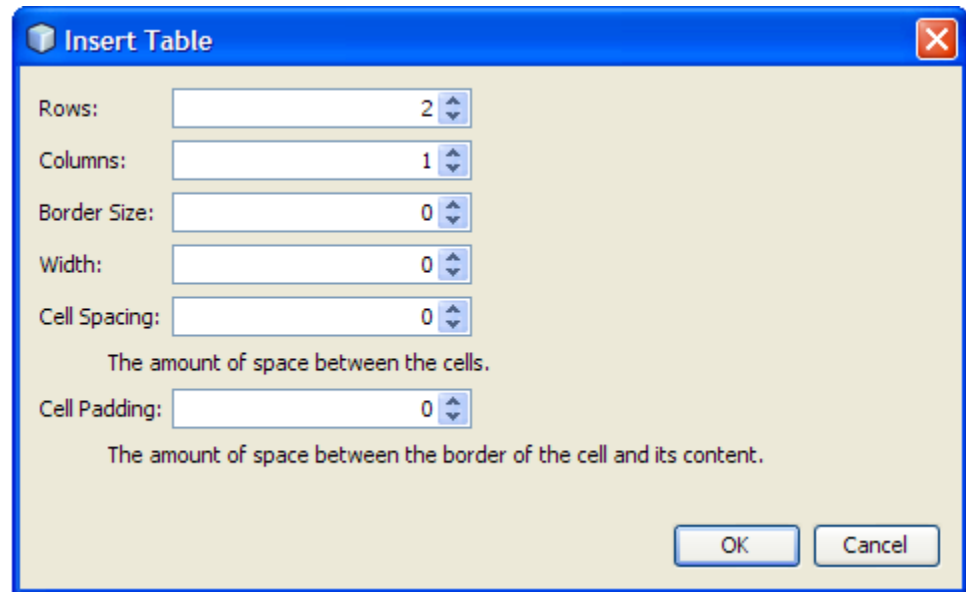
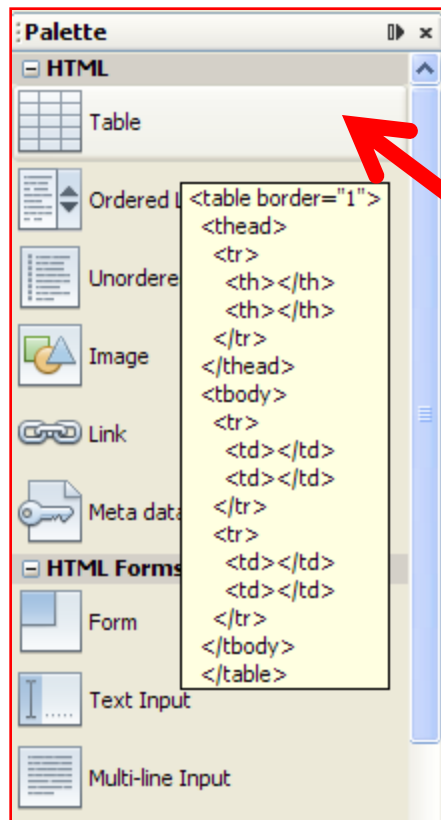


This toggles between Show Big and Small icons

This is the result of changing the display into using Big icons

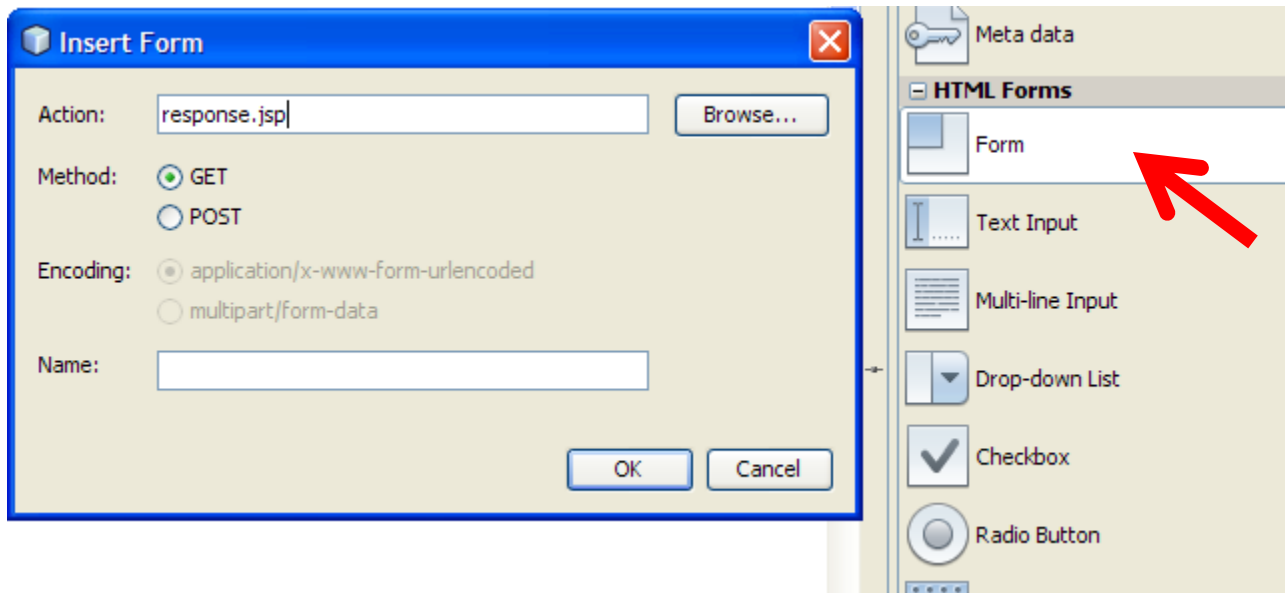
You can also configure the display of the Palette by right-clicking the Palette title bar, and selecting **Show Big Icons**.

Add a **Table** using the **Palette**



Position the cursor at your target coordinates (after the `</h1>` tag), then Insert a **Table** by double-clicking the icon for Table

Add a **Form** using the Palette



Position the cursor at your target coordinates (in between the `<td>` and `</td>` tags), then Insert a **Form** by double-clicking the icon for Forms

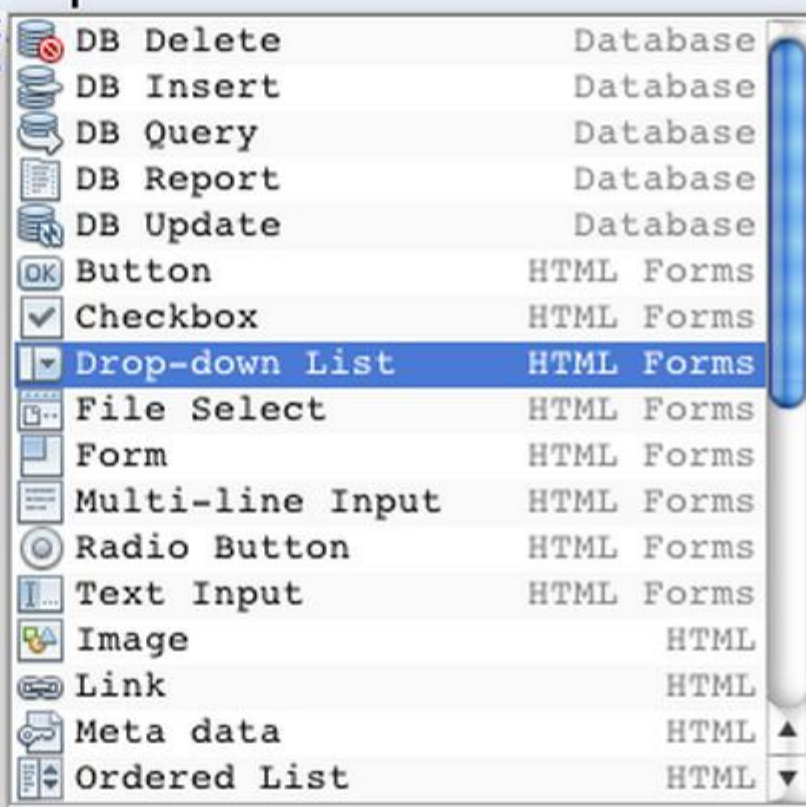
Edit Index.jsp

```
<body>
  <h1>Welcome to IFPWAFCAD, the International Former Professional
Wrestlers' Association for Counseling and Development!</h1>
  <table border="0">
    <thead>
      <tr>
        <th>IFPWAFCAD offers expert counseling in a wide range of fields.</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>To view the contact details of an IFPWAFCAD certified former
professional wrestler in your area, select a subject below:</td>
      </tr>
      <tr>
        <td><form action="response.jsp">
          <strong>Select a subject:</strong>
          </form>
        </td>
      </tr>
    </tbody>
  </table>
</body>
```


Add a Drop-Down List using the Auto-completion support (Ctrl+Space)

```
<tr>
  <td>
    <form action="response.jsp">
      <strong>Select a subject:</strong>

```



The screenshot shows an IDE with an auto-completion menu open. The menu lists various widgets and their categories. The 'Drop-down List' widget is highlighted. The background code shows an HTML form with a text input field.

DB Delete	Database
DB Insert	Database
DB Query	Database
DB Report	Database
DB Update	Database
OK Button	HTML Forms
Checkbox	HTML Forms
Drop-down List	HTML Forms
File Select	HTML Forms
Form	HTML Forms
Multi-line Input	HTML Forms
Radio Button	HTML Forms
Text Input	HTML Forms
Image	HTML
Link	HTML
Meta data	HTML
Ordered List	HTML

```

    </td>
  </tr>
</tbody>
</table>
</body>
</html>

```

Insert Drop-down List

Name:

Options:

Visible Options:

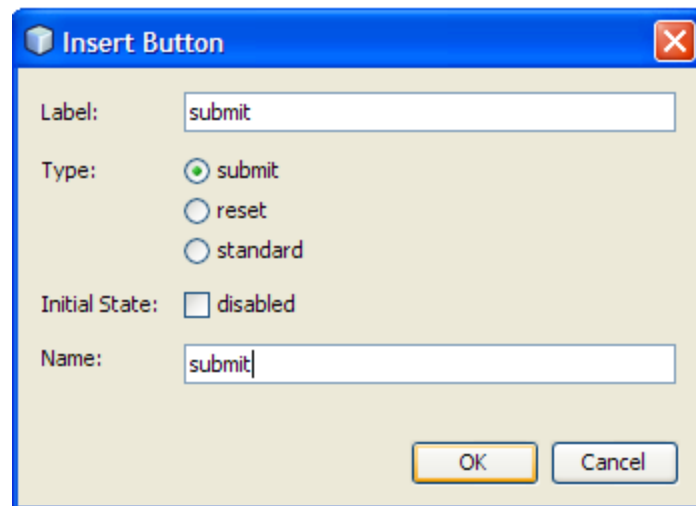
Multiple Selections: allowed

Initial State: disabled

OK Cancel

Press **Ctrl+Space** to invoke the Auto-completion support

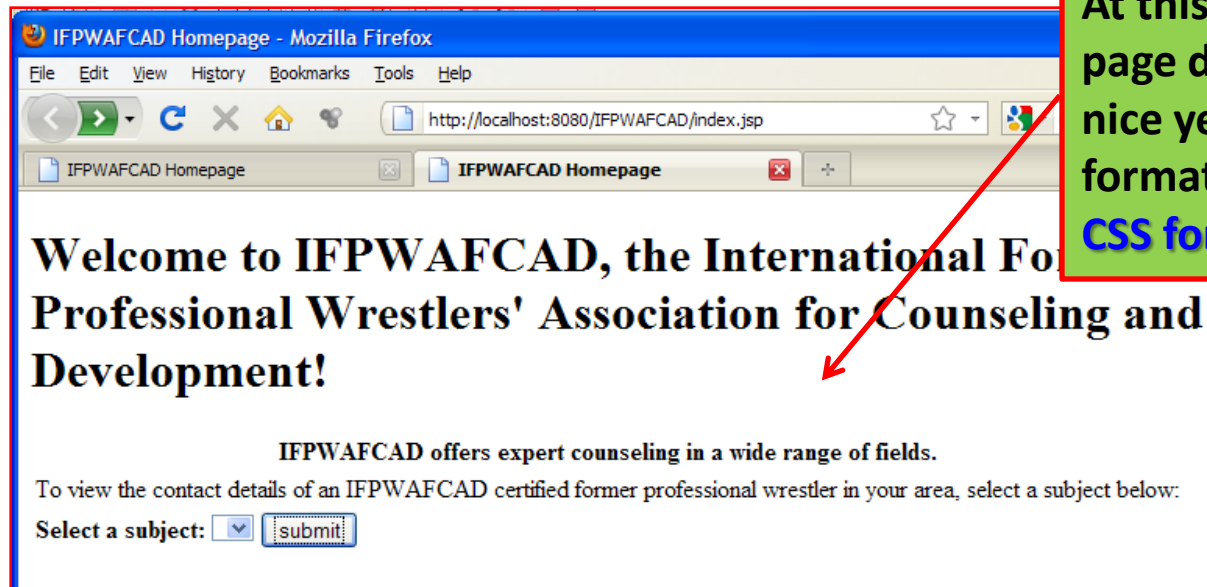
Add a **submit** button



Add a submit button item to a point just after the drop-down list you just added.

Run your index.jsp file

- To view this page in a browser, right-click in the editor and choose **Run File** (Shift-F6).
- When you do this, the JSP page is automatically compiled and deployed to your server.
- The IDE opens your default browser to display the page from its deployed location.



At this stage the web page doesn't look nice yet.. We can format this later using **CSS formatting**.



**Creating the
response page**

Creating the **response** page

New JSP File

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

File Name: response

Project: IFPWAFCAD

Location: Web Pages

Folder: Browse...

Created File: \\Documents and Settings\\nhreyes\\My Documents\\NetBeansProjects\\IFPWAFCAD\\web\\response.jsp

Options:

JSP File (Standard Syntax) Create as a JSP Segment

JSP Document (XML Syntax)

Description:

A JSP file using JSP standard syntax.

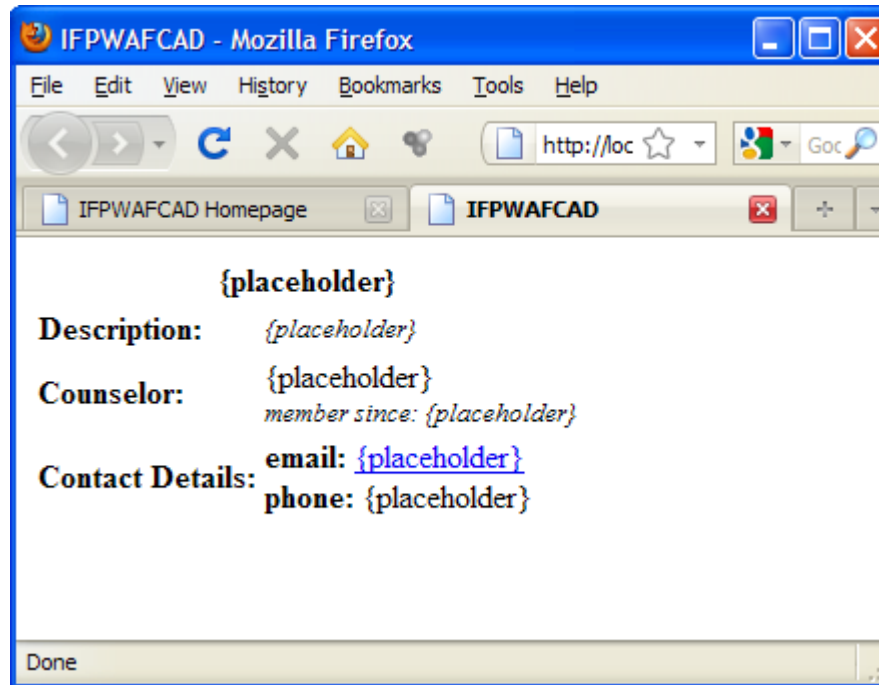
< Back Next > Finish Cancel Help

Right-click the **project**, then select **New / JSP**.

Insert the following **table** into the **response page**

```
<table border="0">
  <thead>
    <tr>
      <th colspan="2">{placeholder}</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td><strong>Description:</strong></td>
      <td><span style="font-size:smaller; font-style:italic;">{placeholder}</span></td>
    </tr>
    <tr>
      <td><strong>Counselor:</strong></td>
      <td>{placeholder}
        <br>
        <span style="font-size:smaller; font-style:italic;">
member since: {placeholder}</span>
      </td>
    </tr>
    <tr>
      <td><strong>Contact Details:</strong></td>
      <td><strong>email:</strong>
        <a href="mailto:{placeholder}">{placeholder}</a>
        <br><strong>phone:</strong> {placeholder}
      </td>
    </tr>
  </tbody>
</table>
```

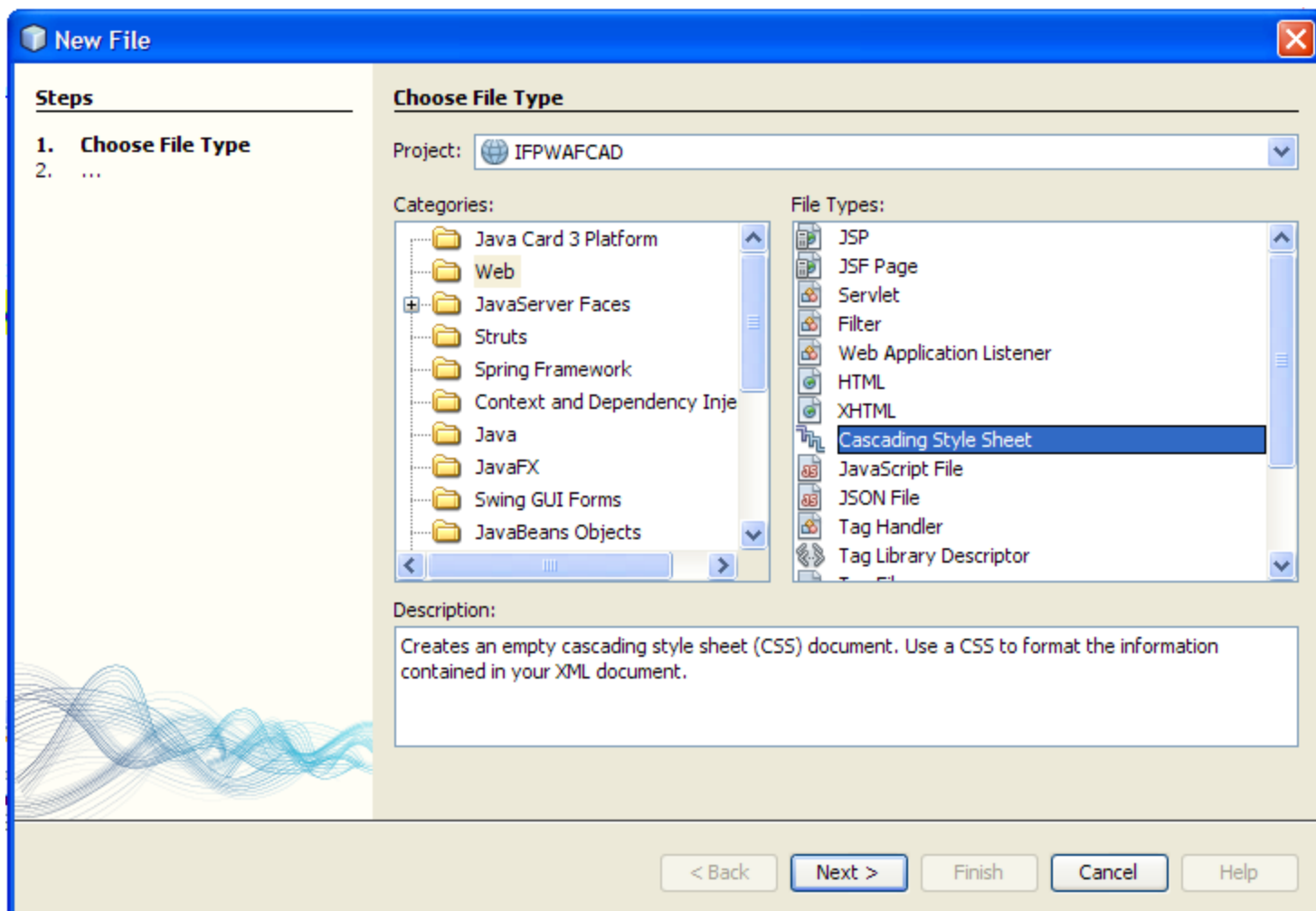
Execute the **response page**



- To view this page in a browser, right-click in the editor and choose **Run File** (Shift-F6).

Adding CSS to your JSPs

Add CSS



Add CSS

New CSS File ✕

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

CSS File Name:


Project:

Location: ▾

Folder:

Created File:

< Back Next > **Finish** Cancel Help



Add CSS

```
body {
  font-family: Verdana, Arial, sans-serif;
  font-size: smaller;
  padding: 50px;
  color: #555;
}

h1 {
  text-align: left;
  letter-spacing: 6px;
  font-size: 1.4em;
  color: #be7429;
  font-weight: normal;
  width: 450px;
}

table {
  width: 580px;
  padding: 10px;
  background-color: #c5e7e0;
}
```

```
th {
  text-align: left;
  border-bottom: 1px solid;
}

td {
  padding: 10px;
}

a:link {
  color: #be7429;
  font-weight: normal;
  text-decoration: none;
}

a:link:hover {
  color: #be7429;
  font-weight: normal;
  text-decoration: underline;
}
```

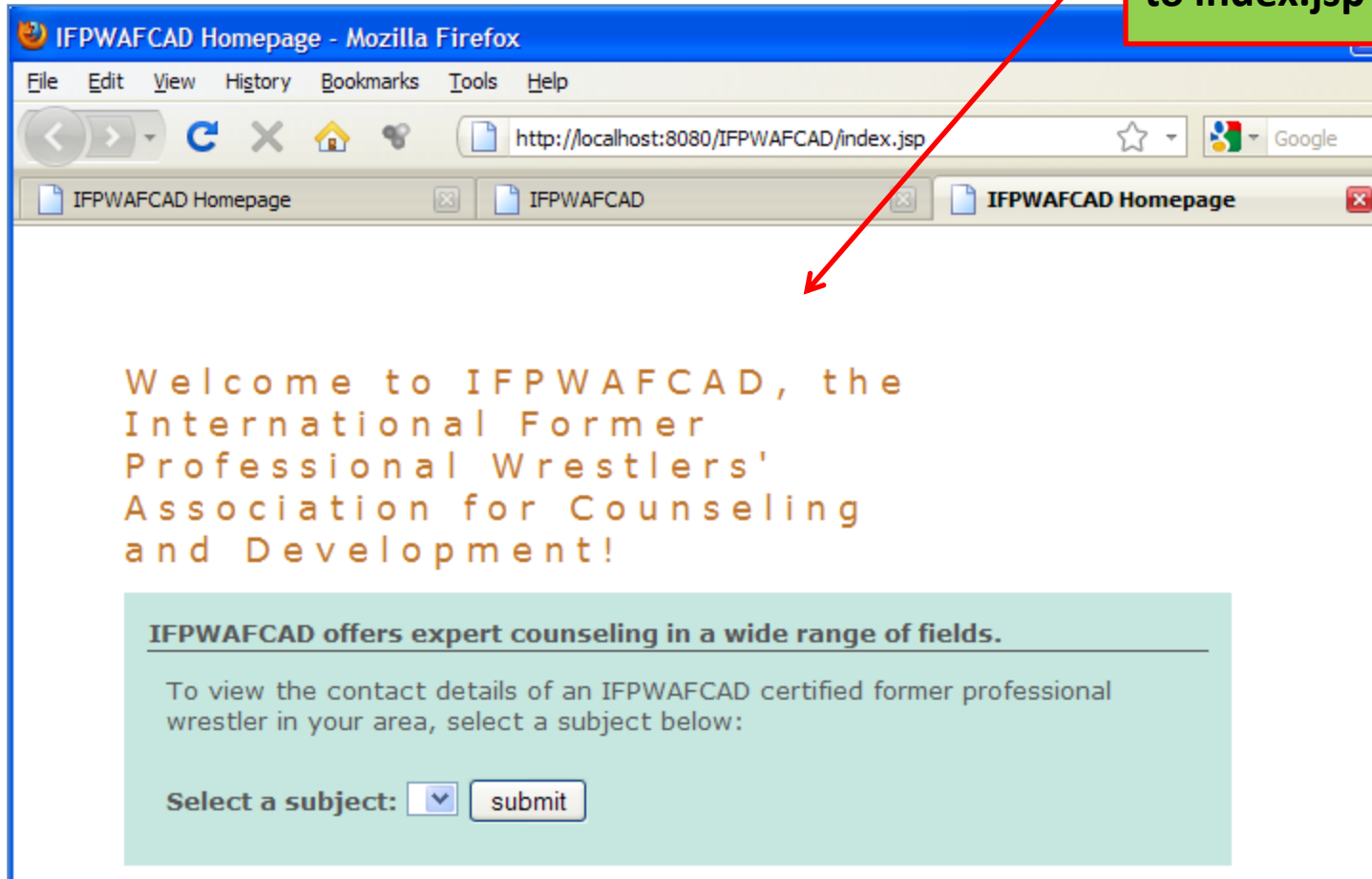
Add CSS

```
<head>  
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
  <link rel="stylesheet" type="text/css" href="style.css">  
  ...  
</head>
```

- Add a link to your **style.css** file in both **index.jsp** and **response.jsp**.
- put the link in the **<head>** section

Display Result: **index.jsp** (with CSS)

Look at the effect of applying the **CSS-style** to index.jsp



IFPWAFCAD Homepage - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:8080/IFPWAFCAD/index.jsp

IFPWAFCAD Homepage IFPWAFCAD IFPWAFCAD Homepage

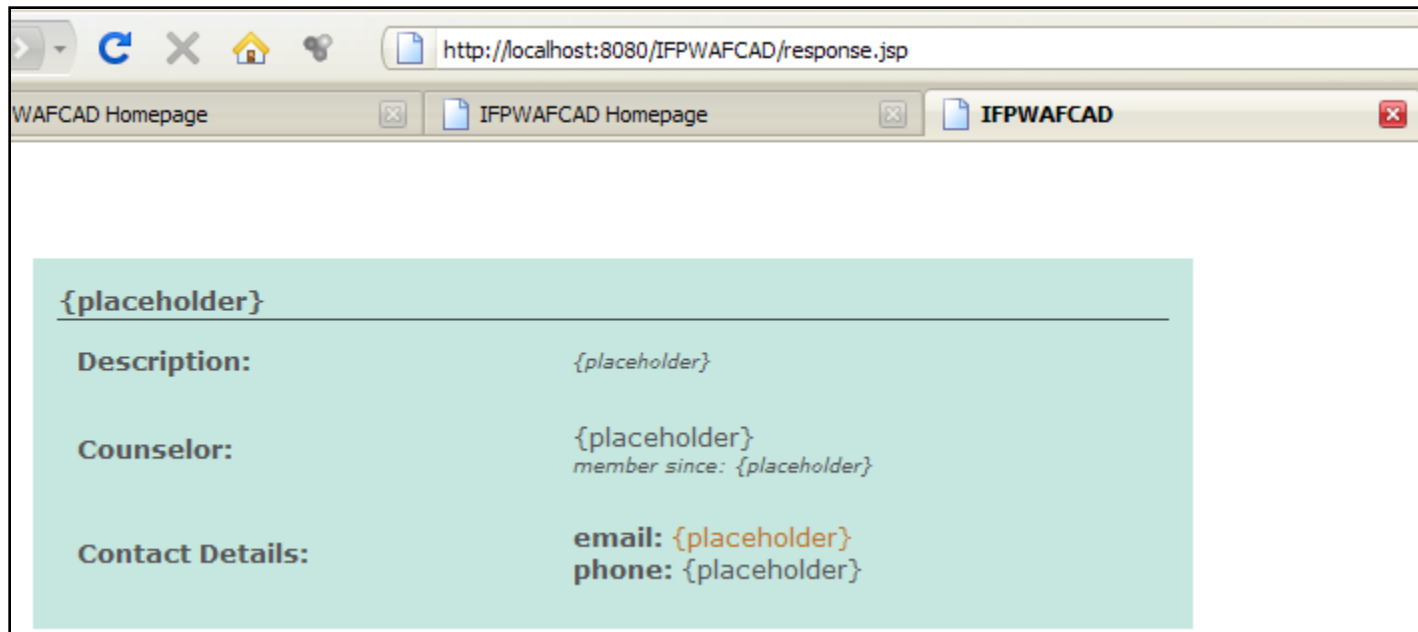
Welcome to IFPWAFCAD, the
International Former
Professional Wrestlers'
Association for Counseling
and Development!

IFPWAFCAD offers expert counseling in a wide range of fields.

To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:

Select a subject:

Display Result: **response.jsp** (with CSS)





**Connecting to a
MySQL database**

Changing the root password

This tutorial uses **nbuser** as an example password. To set your password to nbuser, navigate to your MySQL installation's bin directory in the command-line prompt and enter the following:

```
shell> mysql -u root
```

```
mysql> UPDATE mysql.user SET Password = PASSWORD('nbuser')  
-> WHERE User = 'root';
```

```
mysql> FLUSH PRIVILEGES;
```


Connecting to a MySQL Database

In the **Services window**, expand the **Databases node**, right-click the **MySQL Server node** and choose Properties.

New Database Connection

Basic setting | Advanced

Data Input Mode: Field Entry Direct URL Entry

Driver Name: MySQL (Connector/J driver)

Host: localhost

Port: 3306

Database:

User Name: root

Password:

Display Name (Optional):

Remember password
(see help for information on security risks)

Additional Props:

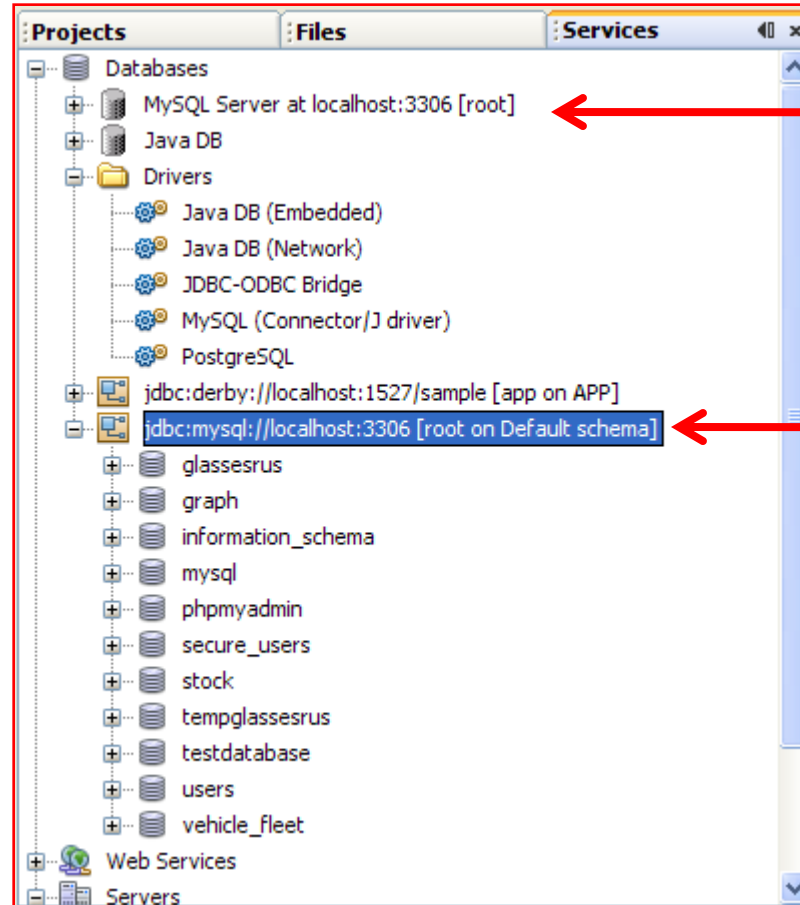
Show JDBC URL

OK Cancel Help

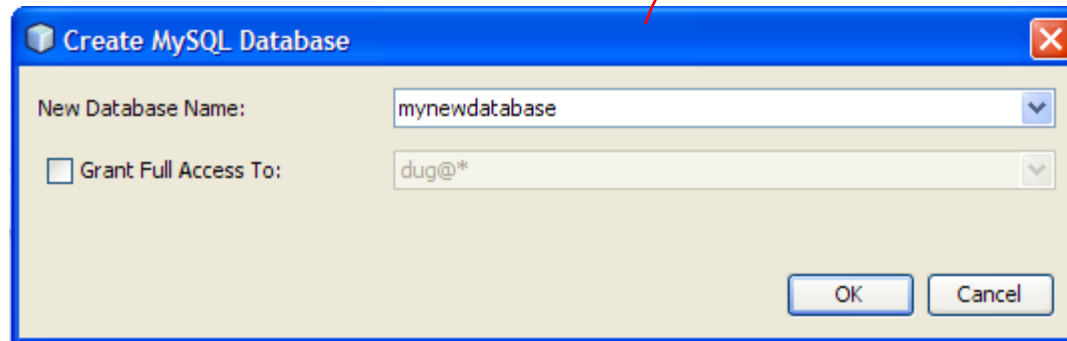
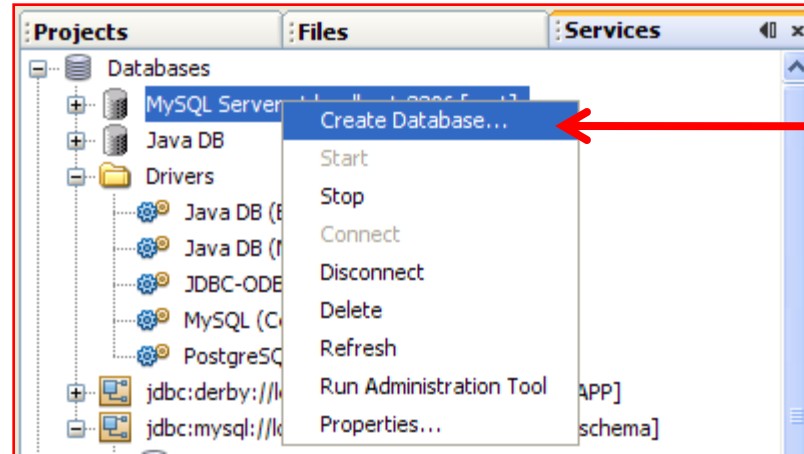
Default port number of MySQL server

Enter password: nbuser

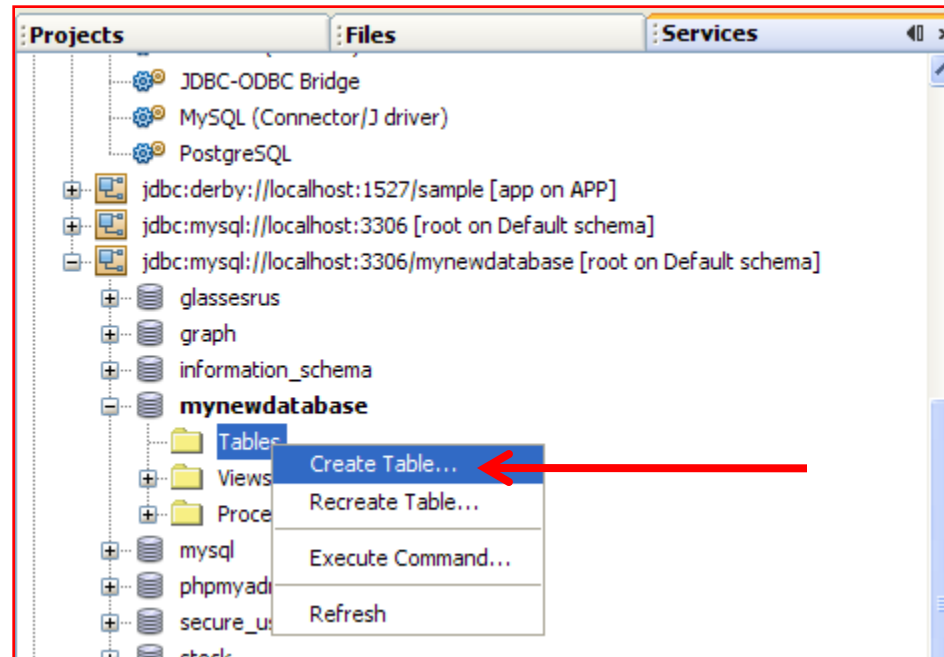
Connecting to a MySQL Database



Creating and Connecting to a MySQL Database Instance

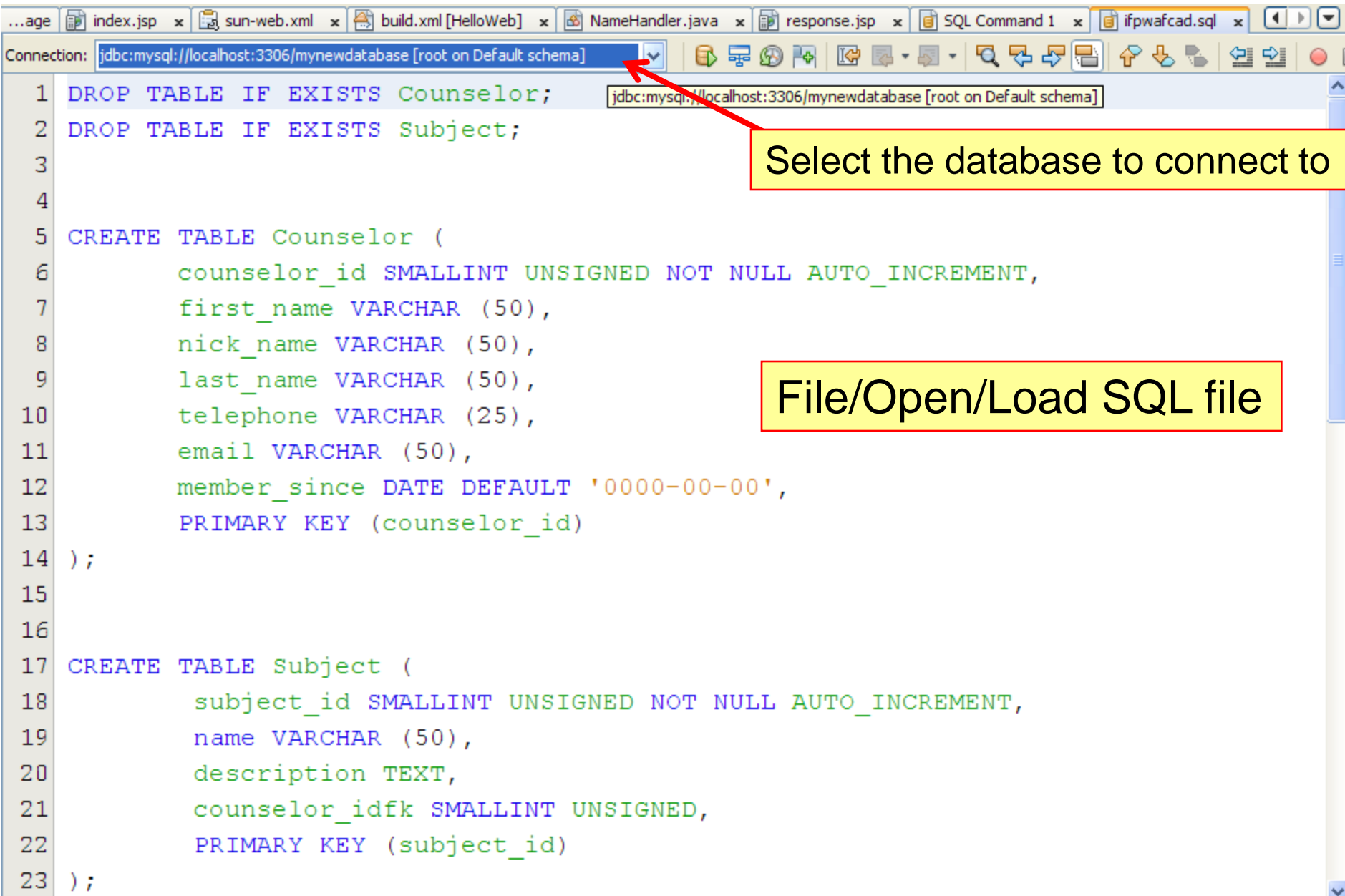


Creating a Table using the Database Explorer



This is optional if you have an **SQL file** containing the entire database structure. You can simply load it later.

Add an Existing SQL Script



The screenshot shows a SQL IDE window with a toolbar and a script editor. The toolbar includes icons for file operations (File, Open, Save, Print, Copy, Paste, Undo, Redo, Find, Replace, Run, Refresh, Stop, Help) and a dropdown menu for database connections. The script editor contains SQL code for dropping and creating tables. A red arrow points from a yellow callout box to the database connection dropdown menu. Another yellow callout box is positioned near the 'File/Open/Load SQL file' icon in the toolbar.

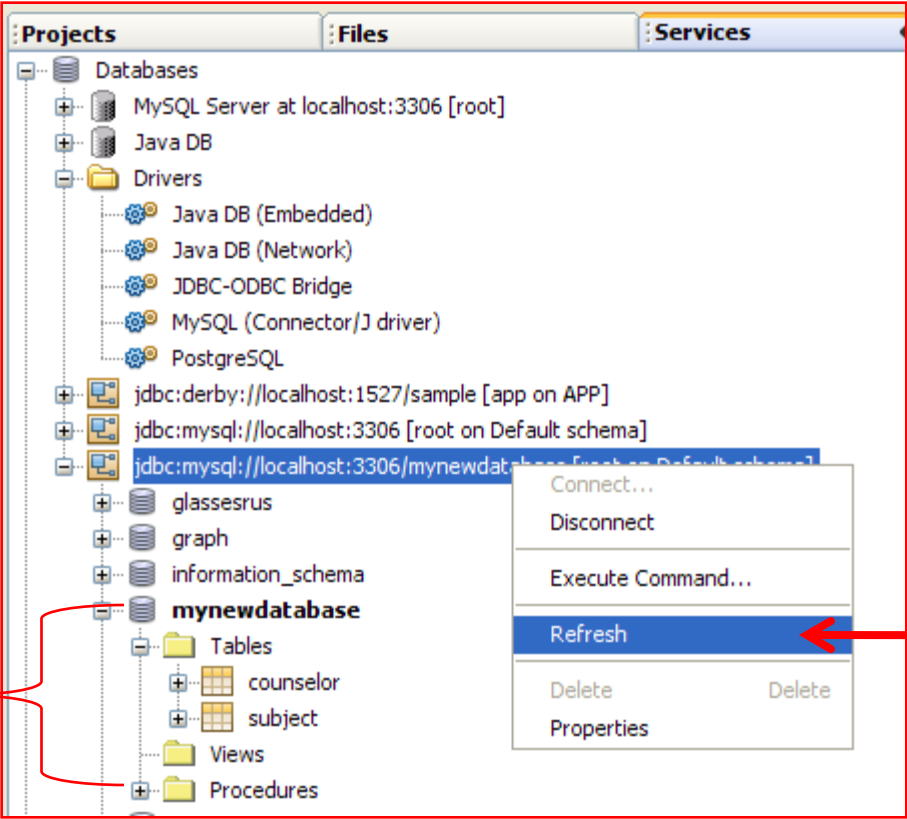
Connection: jdbc:mysql://localhost:3306/mynewdatabase [root on Default schema]

```
1 DROP TABLE IF EXISTS Counselor;
2 DROP TABLE IF EXISTS Subject;
3
4
5 CREATE TABLE Counselor (
6     counselor_id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
7     first_name VARCHAR (50),
8     nick_name VARCHAR (50),
9     last_name VARCHAR (50),
10    telephone VARCHAR (25),
11    email VARCHAR (50),
12    member_since DATE DEFAULT '0000-00-00',
13    PRIMARY KEY (counselor_id)
14 );
15
16
17 CREATE TABLE Subject (
18     subject_id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
19     name VARCHAR (50),
20     description TEXT,
21     counselor_idfk SMALLINT UNSIGNED,
22     PRIMARY KEY (subject_id)
23 );
```

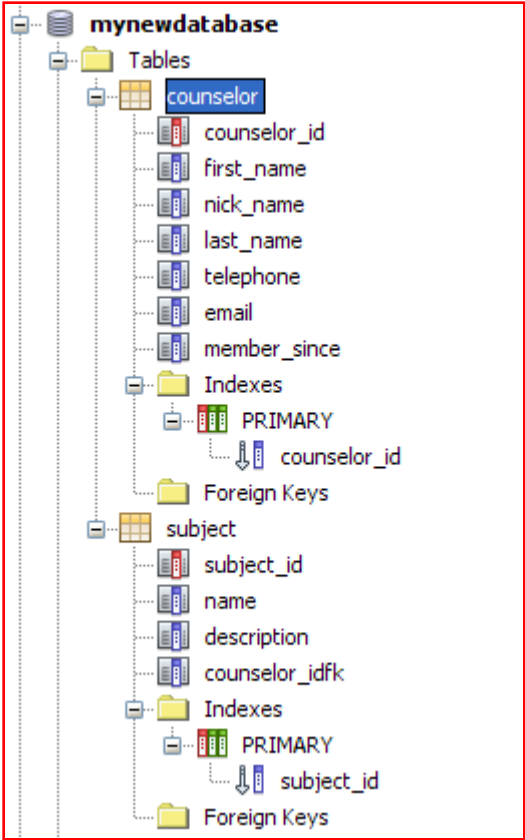
Select the database to connect to

File/Open/Load SQL file

Our newly added database



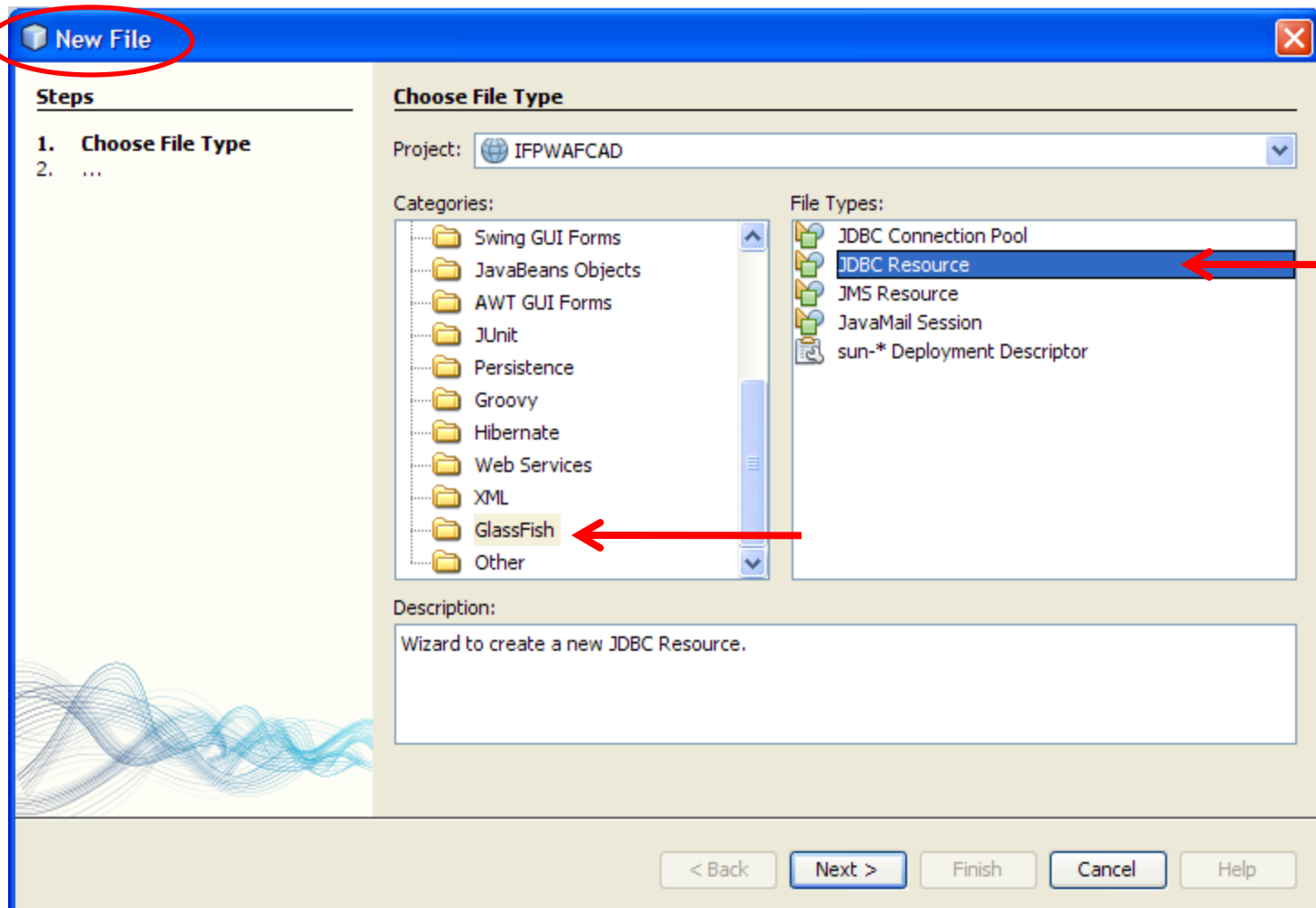
Database Structure



Preparing Communication between the **Application** and **Database**

**Setting up a JDBC data
source and connection
pool**

Setting up a **JDBC** data source and **connection pool**



New JDBC Resource

Steps

1. Choose ...
2. **General Attributes - JDBC Resource**
3. Properties
4. Choose Database Connection
5. Add Connection Pool Properties
6. Add Connection Pool Optional Properties

General Attributes

Provide configuration information for the JDBC Resource.
Either choose an existing JDBC Connection Pool, or create a new JDBC Connection Pool.
Fields with an * mark are required.

Use Existing JDBC Connection Pool

< No JDBC Connection Pool >

Create New JDBC Connection Pool

JNDI Name:*

Object Type:

Enabled:

Description:

< Back Next > Finish Cancel Help

Create a new one

Java Naming and Directory Interface

Java Naming and Directory Interface (JNDI) – a **naming system** is required to manage the lookup between names and resources (e.g. Database).

New JDBC Resource

Steps

1. Choose ...
2. General Attributes - JDBC Resource
3. Properties
- 4. Choose Database Connection**
5. Add Connection Pool Properties
6. Add Connection Pool Optional Properties

Choose Database Connection

Provide configuration information for the JDBC Connection Pool.
Either choose an existing database connection to extract information, or enter the configuration information.
Fields with an * mark are required.

JDBC Connection Pool Name:* IfpwafcadPool

Extract from Existing Connection:

jdbc:mysql://localhost:3306/mynewdatabase [root on Default schema]

New Configuration using Database:

< Select from the list >

XA (Global Transaction)

< Back Next > Finish Cancel Help

The database that we want to connect to

New JDBC Resource

Steps

1. Choose ...
2. General Attributes - JDBC Resource
3. Properties
4. Choose Database Connection
5. **Add Connection Pool Properties**
6. Add Connection Pool Optional Properties

Add Connection Pool Properties

Enter the Datasource Classname, URL, and User to continue.
Hit the Enter key to save values in the Properties table.

Datasource Classname:

Resource Type:

Description:

Properties:

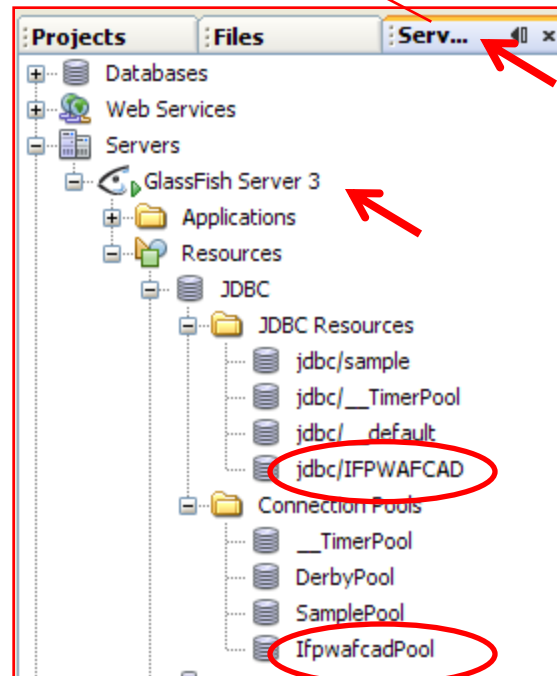
Name	Value
URL	jdbc:mysql://localhost:3306/mynewdatabase
User	root
Password	()

The new password should reflect here

Sun-resources.xml will be created automatically. This will contain entries for the data source and connection pool that you have specified.

Right-click the **IFPWAFCAD project node**, then select **DEPLOY**

Open the **Services window**, and expand the **Servers/Glassfish/Resources/JDBC/JDBC Resources and Connection Pools nodes**



Referencing the data source from the application

You need to **reference** the **JDBC resource** you just configured from the web application

Deployment Descriptor: **Web.xml**

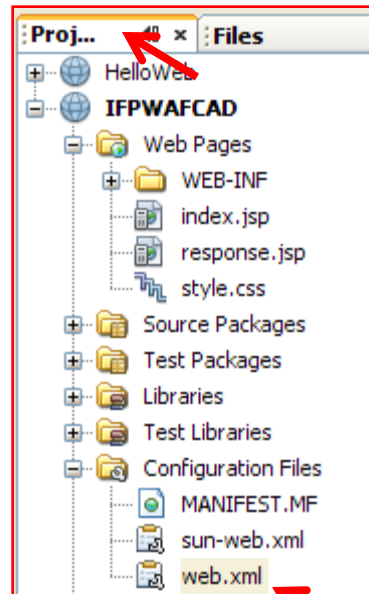
Deployment descriptors are **XML-based** text files that contain information describing **how an application is to be deployed to a specific environment.**

For example, they are normally used to specify:

- application context parameters and behavioural patterns
- security settings
- mappings for servlets, filters and listeners.

Projects Window/ Configuration Files folder/ open web.xml

Web.xml



We need to **reference** the **data source** in the application's deployment descriptor.

Select the **References** tab, click on the **Add button**, then enter details

The screenshot shows the configuration tool interface for Web.xml. The 'References' tab is selected and highlighted with a red dashed box and a red arrow. Below the tabs, the 'Resource References' section is expanded, showing a table with columns for 'Resource Name', 'Resource Type', and 'Authentication'. Below the table are buttons for 'Add...', 'Edit...', and 'Remove'. A red arrow points to the 'Add...' button. An 'Add Resource Reference' dialog box is open in the foreground, with a red arrow pointing to the 'Resource Name' field, which contains the text 'jdbc/IPWAFCAD'. Other fields in the dialog include 'Resource Type' (set to 'javax.sql.DataSource'), 'Authentication' (set to 'Container'), 'Sharing Scope' (set to 'Shareable'), and 'Description' (set to 'Database for IPWAFCAD application'). 'OK' and 'Cancel' buttons are at the bottom of the dialog.

Resource Name	Resource Type	Authentication
---------------	---------------	----------------

Buttons: Add..., Edit..., Remove

Dialog: Add Resource Reference

Resource Name: jdbc/IPWAFCAD

Resource Type: javax.sql.DataSource

Authentication: Container

Sharing Scope: Shareable

Description: Database for IPWAFCAD application

Buttons: OK, Cancel

Web.xml

Verifying the Results of adding the Resource Reference

Web.xml

General Servlets Filters Pages **References** Security XML Resource References

Environment Entries

Resource References

Resource Name	Resource Type	Authentication	Sharing Scope	Description
jdbc/IPWAFCAD	javax.sql.DataSource	Container	Shareable	Database for IPWAFCAD applic

Add... Edit... Remove

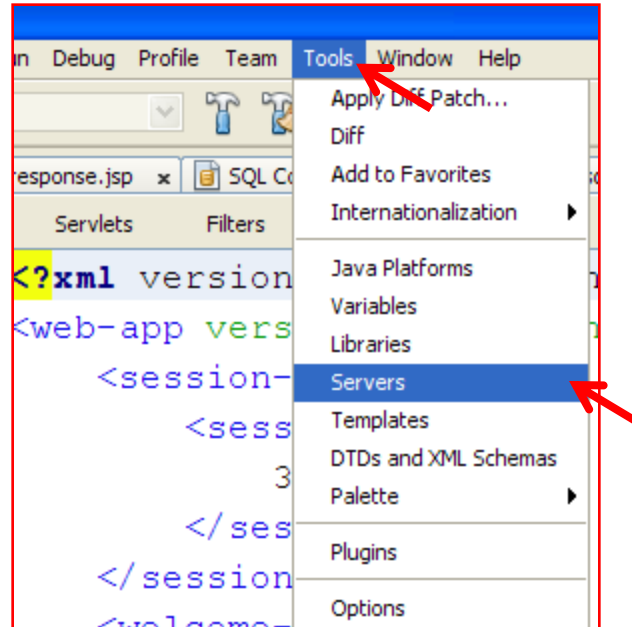
General Servlets Filters Pages References Security **XML**

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="ht
3   <session-config>
4     <session-timeout>
5       30
6     </session-timeout>
7   </session-config>
8   <welcome-file-list>
9     <welcome-file>index.jsp</welcome-file>
10  </welcome-file-list>
11  <resource-ref>
12    <description>Database for IPWAFCAD application</description>
13    <res-ref-name>jdbc/IPWAFCAD</res-ref-name>
14    <res-type>javax.sql.DataSource</res-type>
15    <res-auth>Container</res-auth>
16    <res-sharing-scope>Shareable</res-sharing-scope>
17  </resource-ref>
18 </web-app>
```

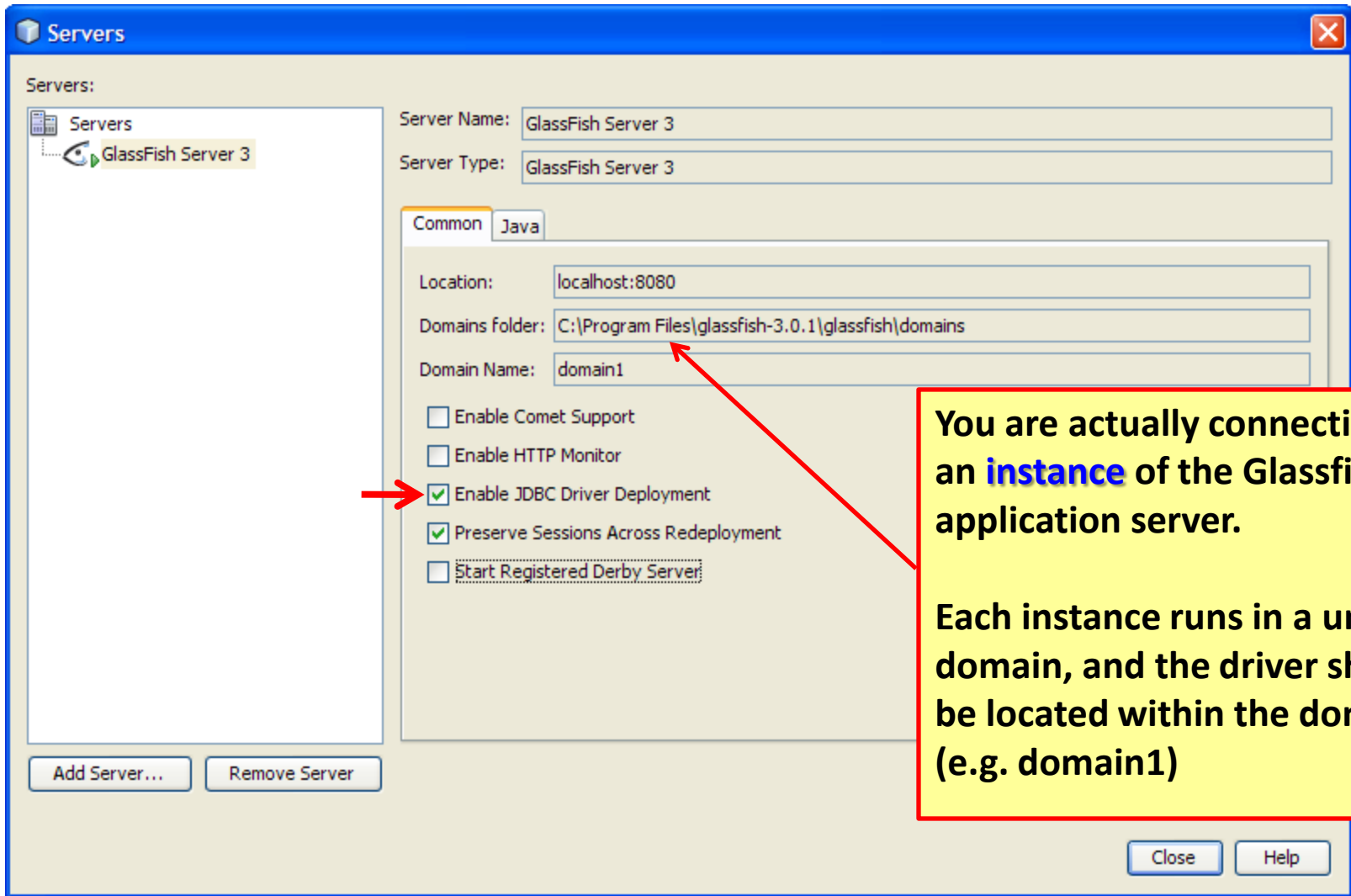
Adding the database driver's JAR file to the server

We need to add the database driver's JAR file **to enable the server to communicate with our database.**

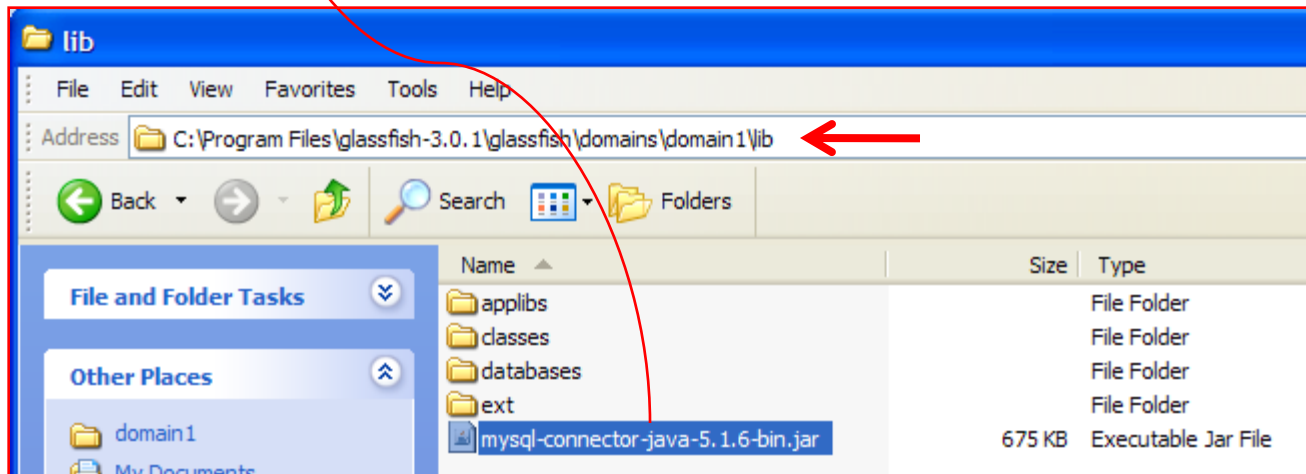
From the menu, select the **Tools, Servers**.



Select Glassfish, then enable the JDBC driver deployment option



We could verify that our **database driver** has been installed properly by inspecting the actual directory of the domain.

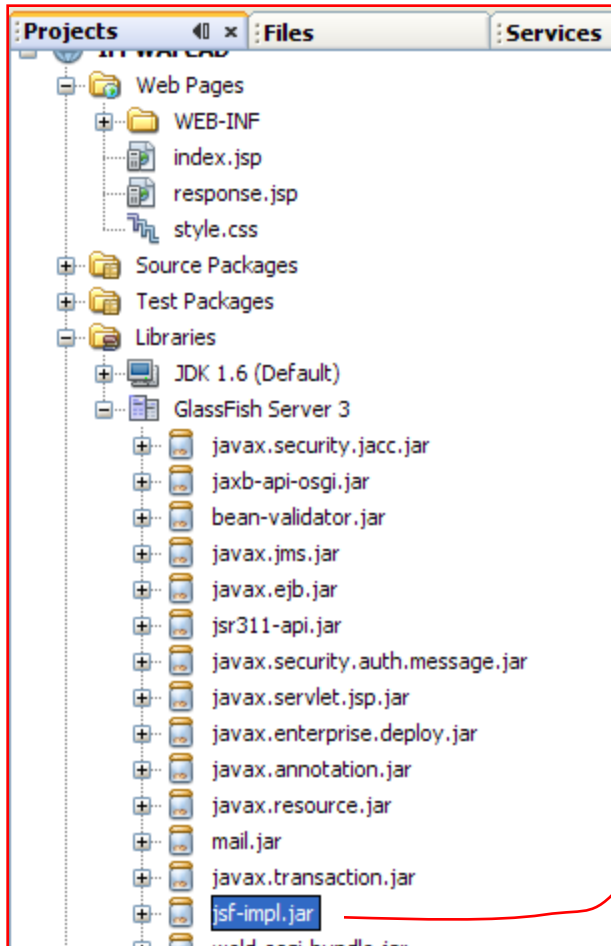


Adding dynamic logic (JSTL)

We can generate content for the **JSP placeholders** that we've created by implementing JSTL codes.

JSTL technology performs the logic that directly queries the database and inserts the retrieved data into the two pages

JavaServer Pages Standard Tag Library (JSTL)



JSTL allows us to **access** and **display** data taken from the database.

Glassfish server includes the **JSTL library** by default.

We need to populate the form inside **index.jsp** with names coming from the **Subject database table**.

The screenshot shows an IDE with a JSP file open. The JSP code is as follows:

```
1 <% --  
2     Document      : index  
3     Created on   : 7/10/2010, 00:05:59  
4     Author      : nhreyes  
5     --%>  
6  
7 <%@page contentType="text/html" pageEncoding="UTF-8"%>  
8 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
9     "http://www.w3.org/TR/html4/loose.dtd">  
10  
11 <html>  
12     <head>  
13         <meta http-equiv="Content-Type" content="text/html; char  
14         <link rel="stylesheet" type="text/css" href="style.css">
```

A red arrow points from a green box labeled "Insert here" to the line number 6. Another red arrow points from the "DB Report" icon in the Database palette to the JSP code area.

The Database palette shows three icons: "DB Query", "DB Report", and "DB Insert". The "DB Report" icon is highlighted with a red arrow.

The Database palette also shows the following code snippet:

```
<sql:query var="result">  
</sql:query>  
<table>  
<!-- column headers -->  
<tr>  
<c:forEach var="columnName" items="${result.columnNames}">  
<th><c:out value="${columnName}"/></th>  
</c:forEach>  
</tr>  
<!-- column data -->
```

From the **Palette**, select the **DB Report** icon.

Specify the data source and variable name for the DB Report

Insert DB Report

Variable Name: subjects

Scope: page

Data Source: jdbc/IFPWAFCAD

Query Statement:

```
SELECT subject_id, name FROM Subject
```

OK Cancel

JSTL Tags

<sql:query> to create query

<c:forEach> to loop through the query's results set and output the retrieved data

Automatically Generated Codes (index.jsp)

```
Start Page x sun-resources.xml x index.jsp x response.jsp x style.css x
table
1  <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
2  <%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
3  <sql:query var="subjects" dataSource="jdbc/IFPWAFCAD">
4      SELECT subject_id, name FROM Subject
5  </sql:query>
6
7  <table border="1">
8      <!-- column headers -->
9      <tr>
10     <c:forEach var="columnName" items="${subjects.columnNames}">
11         <th><c:out value="${columnName}"/></th>
12     </c:forEach>
13 </tr>
14 <!-- column data -->
15 <c:forEach var="row" items="${subjects.rowsByIndex}">
16     <tr>
17     <c:forEach var="column" items="${row}">
18         <td><c:out value="${column}"/></td>
19     </c:forEach>
20     </tr>
21 </c:forEach>
22 </table>
23 <%@page contentType="text/html" pageEncoding="UTF-8"%>
```

Custom tags

<%@taglib directives required for the sql and forEach tags.

(index.jsp)

```
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
<sql:query var="subjects" dataSource="jdbc/IFPWAFCAD">
    SELECT subject_id, name FROM Subject
</sql:query>
```

Custom tags

SQL Query

```
<table border="1">
  <!-- column headers -->
  <tr>
    <c:forEach var="columnName" items="${subjects.columnNames}">
      <th><c:out value="${columnName}" /></th>
    </c:forEach>
  </tr>
  <!-- column data -->
  <c:forEach var="row" items="${subjects.rowsByIndex}">
    <tr>
      <c:forEach var="column" items="{row}">
        <td><c:out value="{column}" /></td>
      </c:forEach>
    </tr>
  </c:forEach>
</table>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional/EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

Table

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <link rel="stylesheet" type="text/css" href="style.css">
  <title>IFPWAFCAD Homepage</title>
</head>
<body>
  <h1>Welcome to IFPWAFCAD, the International Former Professional Wrestlers' Association for Counseling and Development!</h1>
  <table border="0">
    <thead>
      <tr>
        <th>IFPWAFCAD offers expert counseling in a wide range of fields.</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:</td>
      </tr>
      <tr>
        <td><form action="response.jsp">
          <strong>Select a subject:</strong>
          <select name="subject_id">
            <option></option>
            <option></option>
            <option></option>
          </select>
          <input type="submit" value="submit" name="submit" />
        </form></td>
      </tr>
    </tbody>
  </table>
</body>
</html>
```

CSS

Form

Table


html

Run the welcome page (**index.jsp**) to see how it displays in a browser.

The file is:

- **deployed** to the **GlassFish** server,
- **compiled into a servlet**, and then
- **renders in your default browser**.

The code generated from the **DB Report** item creates the following table in the welcome page.



subject_id	name
1	Financial Consultancy
2	Existential Psychotherapy
3	Temper Management
4	Past-Life Regression
5	Marriage Guidance
6	Crisis Management
7	Dream Analysis
8	Hypnosis
9	Reiki

(Sample Run) The Project, so far...

http://localhost:8080/IFPWAFCAD/

How to run your web application?

1. Change **root password** to **nbuser**
2. Start **EasyPHP** to start **phpMyAdmin** and **MySQL server**.
3. Stop **Apache**. Remember, we are using **Glassfish** this time.
4. Run **NetBeans**, run the **web application**

- Netbeans
 - Glassfish Java EE application server
- EasyPHP
 - MySQL server
 - phpMyAdmin
 - Apache web server

subject_id	name
1	Financial Consultancy
2	Existential Psychotherapy
3	Temper Management
4	Past-Life Regression
5	Marriage Guidance
6	Management
7	Dream Analysis
8	
9	Reiki

Welcome to IFPWAFCAD, the International Former Professional Wrestlers' Association for Counseling and Development!

IFPWAFCAD offers expert counseling in a wide range of fields.

To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:

Select a subject:

(Sample Run) The Project, so far...

http://localhost:8080/IFPWAFCAD/

**Next, we would like
to put these
generated HTML
codes into the
drop-down list**

subject_id	name
1	Financial Consultancy
2	Existential Psychotherapy
3	Temper Management
4	Past-Life Regression
5	Marriage Guidance
6	Crisis Management
7	Dream Analysis
8	Hypnosis
9	Reiki

Welcome to IFPWAFCAD, the
International Former
Professional Wrestlers'
Association for Counseling
and Development!

**IFPWAFCAD offers expert counseling in a wide range of
fields.**

To view the contact details of an IFPWAFCAD certified former
professional wrestler in your area, select a subject below:

Select a subject:

**Integrating the
generated code into the
HTML drop-down list**

(index.jsp)

Integrate the `<c:forEach>` tags into the HTML form as follows.

```
<td>
  <form action="response.jsp">
    <strong>Select a subject:</strong>
    <select name="subject_id">
      <c:forEach var="row" items="{subjects.rows}">
        <option value="{row.subject_id}">{row.name}</option>
      </c:forEach>
    </select>
    <input type="submit" value="submit" name="submit" />
  </form>
</td>
```

the `<c:forEach>` tags loop through all `subject_id` and name values from the SQL query, and insert each pair into the **HTML `<option>` tags**. In this manner, the form's drop-down list is populated with data.

(Sample Run) The Project, so far...

6	Crisis Management
7	Dream Analysis
8	Hypnosis
9	Reiki

Welcome to IFPWAFCAD, the
International Former
Professional Wrestlers'
Association for Counseling
and Development!

IFPWAFCAD offers expert counseling in a wide range of fields.

To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:

Select a subject:

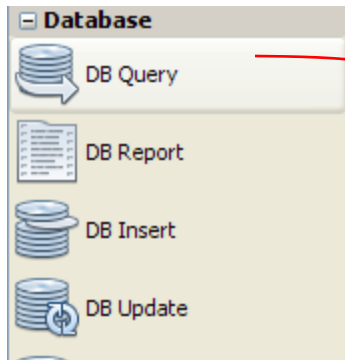
Financial Consultancy ▼

- Financial Consultancy
- Existential Psychotherapy
- Temper Management
- Past-Life Regression
- Marriage Guidance
- Crisis Management
- Dream Analysis
- Hypnosis
- Reiki

submit

Subject chosen

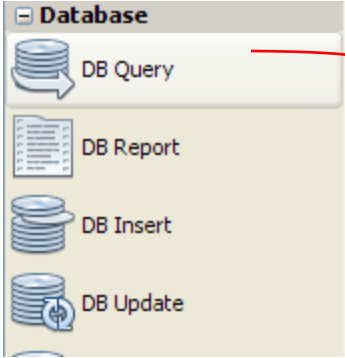
Insert a **DB Query component** on top of `<%@page` tag



```
Start Page x sun-resources.xml x index.jsp x response.jsp x style.css x web.xml x
7
8 <%@page contentType="text/html" pageEncoding="UTF-8"%>
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10   "http://www.w3.org/TR/html4/loose.dtd">
11
12 <html>
13   <head>
14     <meta http-equiv="Content-Type" content="text/html; cha
15     <link rel="stylesheet" type="text/css" href="style.css"
16     <title>IFPWAFCAD </title>
17   </head>
18   <body>
```

Insert a **DB Query component** on top of <%@page tag

Customise the query as indicated below.



```
Start Page x sun-resources
7
8 <%@page cc
9 <!DOCTYPE
10 "http:
11
12 <html>
13 <head>
14 <m
15 <l
16 <t
17 </head
18 <body>
```

Insert DB Query

Variable Name:

Scope:

Data Source:

Query Statement:

```
SELECT * FROM Subject, Counselor WHERE Counselor.counselor_id = Subject.counselor_idfk
AND Subject.subject_id = ? <sql:param value="{param.subject_id}"/>
```

OK Cancel

(response.jsp)

Automatically generated code

```
1 <%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
2 <!--
3 Document : response
4 Created on : 7/10/2010, 00:53:2
5 Author : nhreyes
6 --%>
7
8 <sql:query var="counselorQuery" dataSource="jdbc/IFPWAFCAD">
9 SELECT * FROM Subject, Counselor WHERE Counselor.counselor_id = Subject.counselor_idfk
10 AND Subject.subject_id = ? <sql:param value="${param.subject_id}"/>
11 </sql:query>
12 <%@page contentType="text/html" pageEncoding="UTF-8"%>
13 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
14 "http://www.w3.org/TR/html4/loose.dtd">
```

The IDE automatically added the `<%@taglib>` directive needed for the `<sql:query>` tag

Extracts the user's choice from the **drop-down list** of `index.jsp`

(response.jsp)

Add a **<c:set variable tag** to store the first record returned by the query.

Add a **<%@taglib directive** to allow us to use the <c:set tag.

```
1 <%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
2 <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
3 <!--
4     Document      : response
5     Created on    : 7/10/2010, 00:53:26
6     Author       : nhreyes
7 --%>
8
9 <sql:query var="counselorQuery" dataSource="jdbc/IFPWAFCAD">
10     SELECT * FROM Subject, Counselor WHERE Counselor.counselor_id = Subject.counselor_idfk
11     AND Subject.subject_id = ? <sql:param value='${param.subject_id}'/>
12 </sql:query>
13 <c:set var="counselorDetails" value='${counselorQuery.rows[0]}'/>
14 <%@page contentType="text/html" pageEncoding="UTF-8"%>
15 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
16     "http://www.w3.org/TR/html4/loose.dtd">
```

Added lines

Replace the **{placeholders}** with the following **Expression Language (EL) statements (blue highlights)**

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <link rel="stylesheet" type="text/css" href="style.css">
  <title>${counselorDetails.name} </title>
</head>
<body>
  <table border="0">
    <thead>
      <tr>
        <th colspan="2">${counselorDetails.name}</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td><strong>Description: </strong></td>
        <td><span style="font-size:smaller; font-style:italic;">${counselorDetails.description}</span></td>
      </tr>
      <tr>
        <td><strong>Counselor: </strong></td>
        <td><strong>${counselorDetails.first_name} ${counselorDetails.nick_name} ${counselorDetails.last_name}</strong>
          <br>
          <span style="font-size:smaller; font-style:italic;">
            member since: ${counselorDetails.member_since}</span>
        </td>
      </tr>
      <tr>
        <td><strong>Contact Details: </strong></td>
        <td><strong>email: </strong>
          <a href="mailto:${counselorDetails.email}">${counselorDetails.email}</a>
          <br>
          <strong>phone: </strong>${counselorDetails.telephone}
        </td>
      </tr>
    </tbody>
  </table>
</body>
</html>
```

(response.jsp)

Run the application!

(Sample Run) Final Project

Welcome to IFPWAFCAD, the International Former Professional Wrestlers' Association for Counseling and Development!

IFPWAFCAD offers expert counseling in a wide range of fields.

To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:

Select a subject:

Index.jsp

Submit Subject selected

Response.jsp

The screenshot shows a web browser window with the address bar containing the URL: `http://localhost:8080/IFPWAFCAD/response.jsp?subject_id=3&submit=submit`. The browser tab is titled "Temper Management". The main content area displays a light green box with the following information:

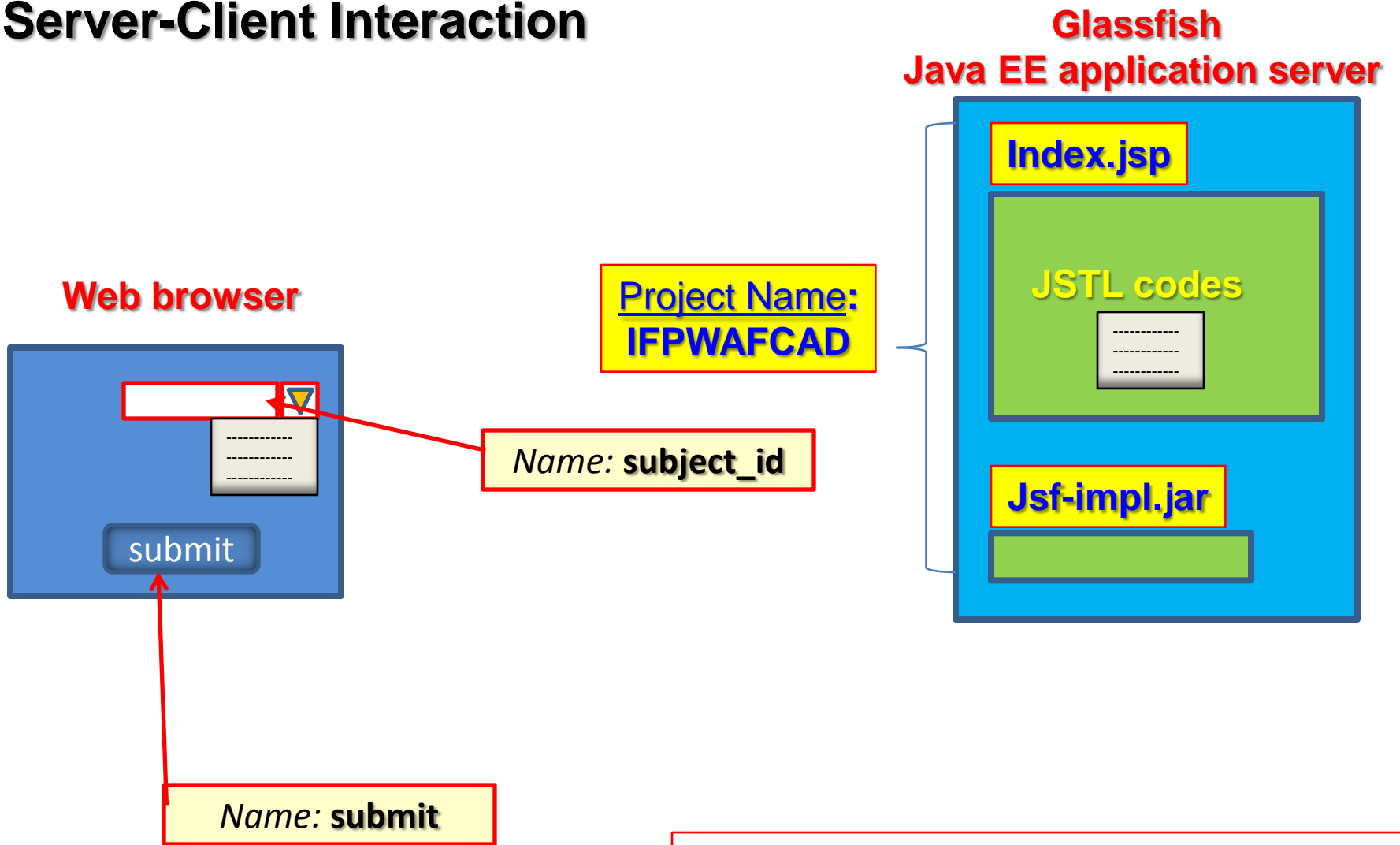
Temper Management	
Description:	<i>Are your angry outbursts affecting your relationships with loved-ones? Do tantrums at work hinder your ability to perform? Temper management helps you to channel your anger into positive, life-changing productivity.</i>
Counselor:	Randy "Macho Man" Savage <i>member since: 2000-11-10</i>
Contact Details:	email: machoman@hotmail.com phone: 555 317-4444

Project: IFPWAFCAD

JSP documents
MySQL server, database with 2 tables
Glassfish Java EE application server

JSTL – for dynamic logic
Jsf-impl.jar – required library to utilise JSTL codes
Jdbc/IFPWAFCAD – name of data resource

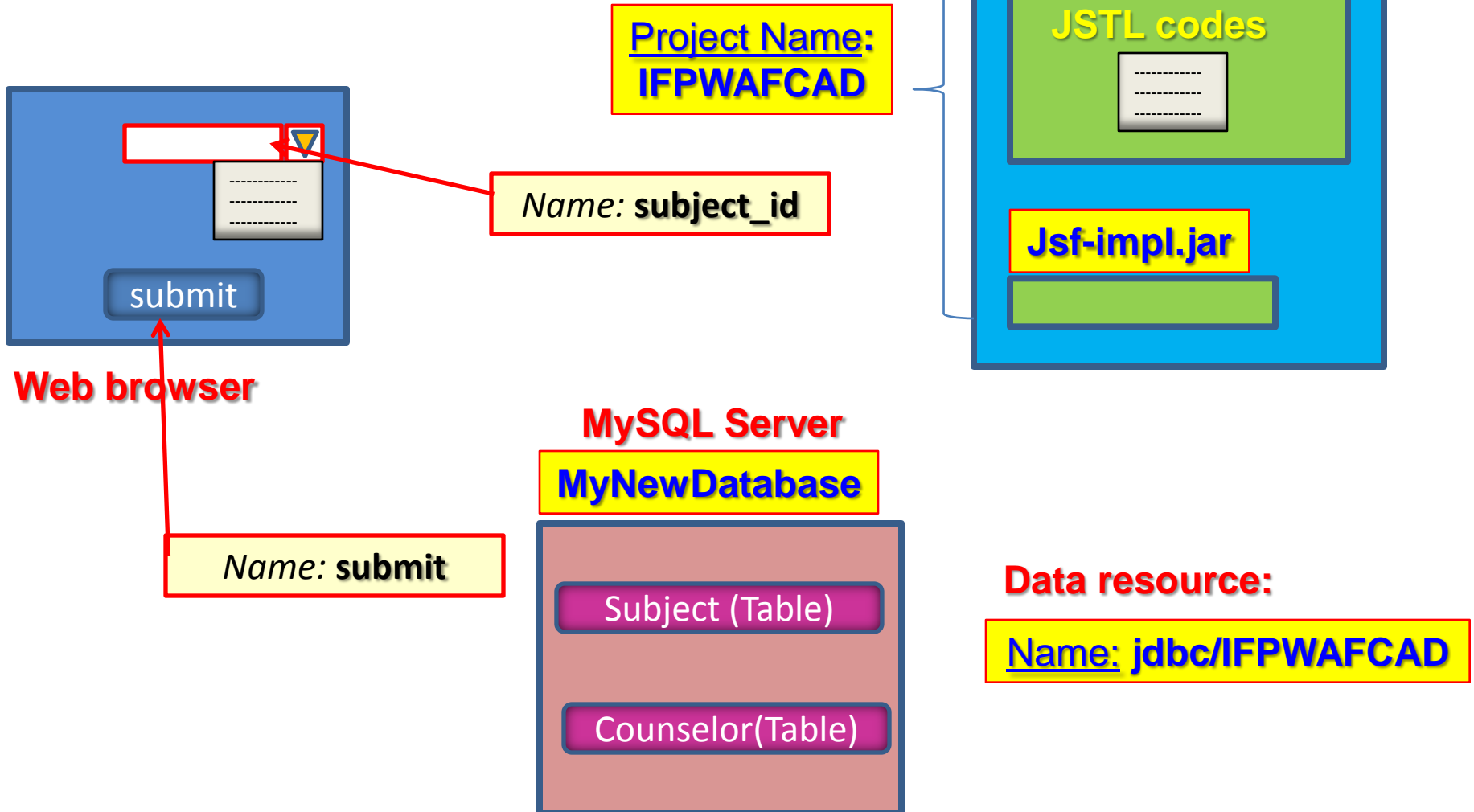
Server-Client Interaction



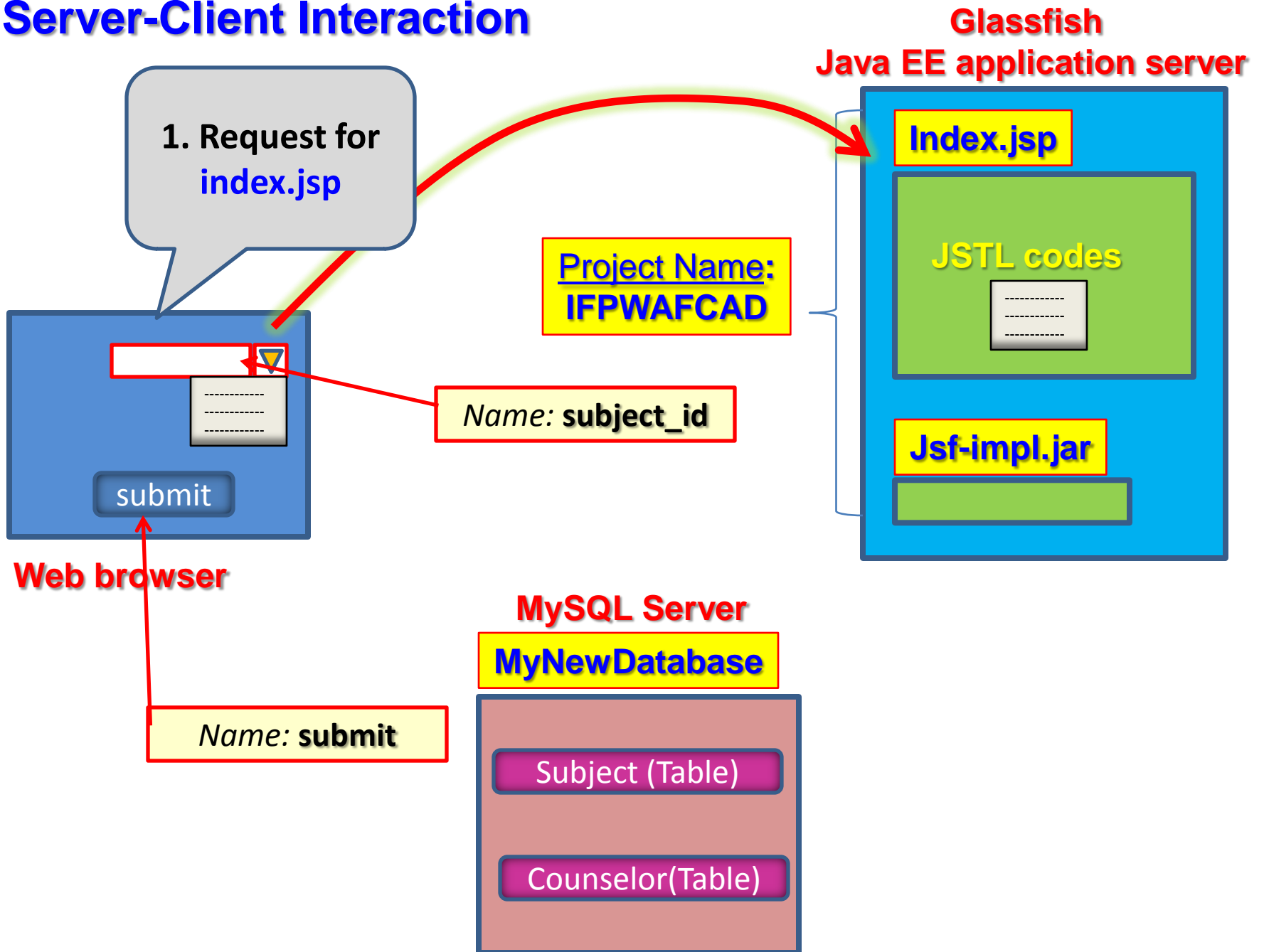
0. JSP is deployed to the server and compiled into a servlet

Server-Client Interaction

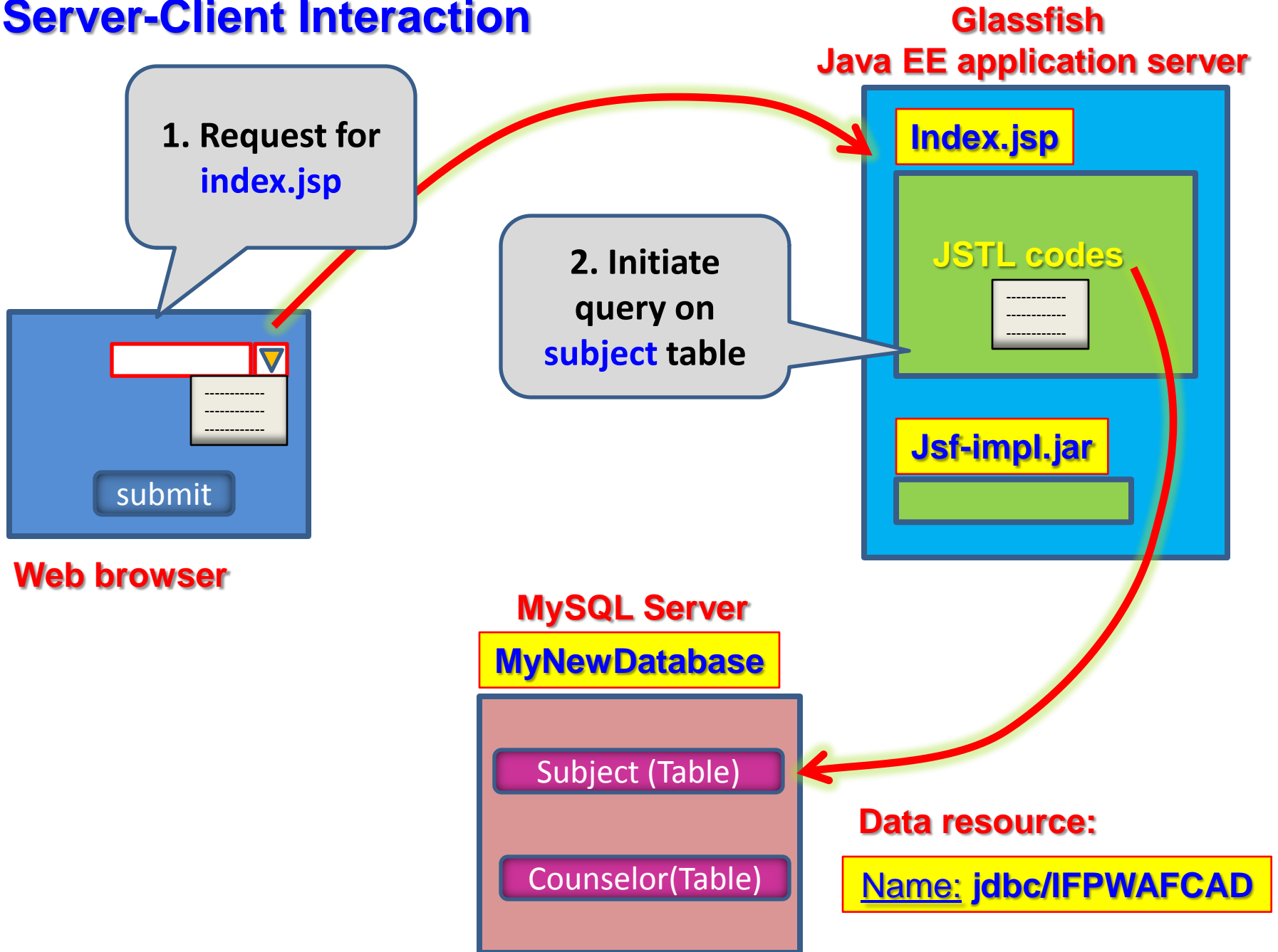
We would like to populate the drop-down list with records coming from MyNewDatabase, Subject table.



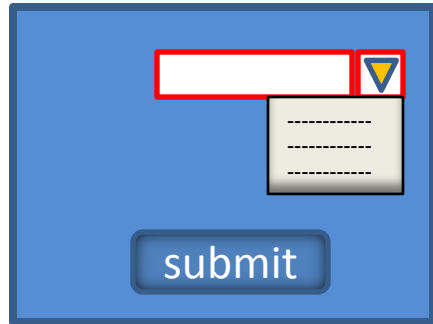
Server-Client Interaction



Server-Client Interaction



Server-Client Interaction



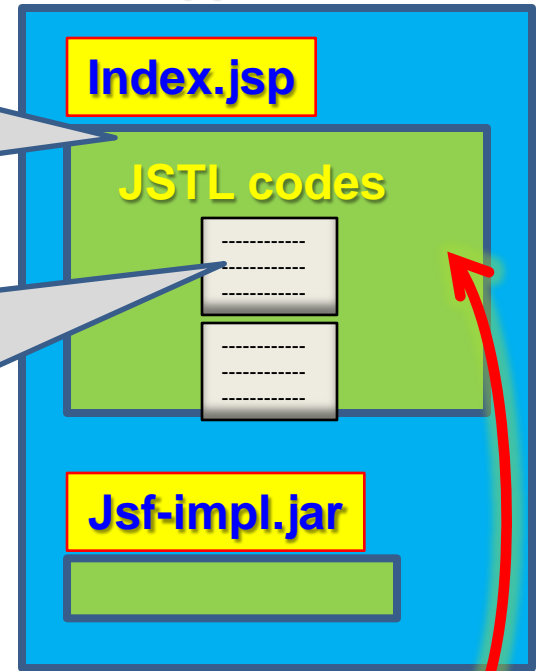
Web browser

2. Initiate query on **subject** table

4. **Insert records** into page by referring to the name of data source

3. Send **records** based on query result

Glassfish
Java EE application server



MySQL Server

MyNewDatabase

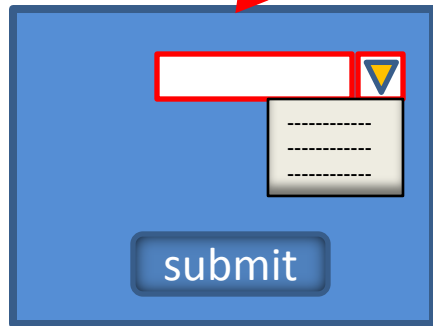
Subject (Table)

Counselor(Table)

Data resource:

Name: jdbc/IFPWAFCAD

Server-Client Interaction

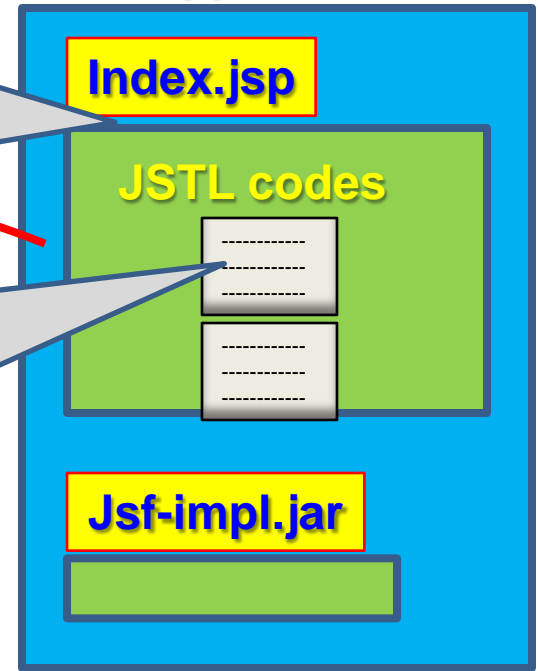


Web browser

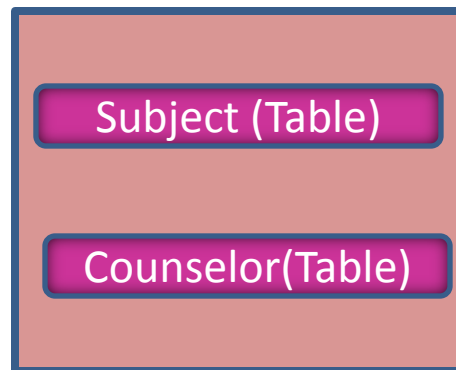
5. Send generated page back to browser via HTTP

4. Insert records into page by referring to the name of data source

Glassfish
Java EE application server



MySQL Server
MyNewDatabase



Data resource:

Name: jdbc/IFPWAFCAD

Project: IFPWAFCAD

Database Connection

Project

JSP files:

index.jsp, response.jsp

Configuration files:

web.xml

Server: Glassfish Server 3

Enable JDBC driver deployment

Domain1

Services

Databases

Jdbc:mysql://localhost:3306

mynewdatabase

DATA resource

Jdbc resource: jdbc/IFPWAFCAD

New jdbc connection pool: **IFpwafcadPool**

Jndi name: jdbc/IFPWAFCAD

Must reside inside the server's directory

Database Driver

Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml

```
<resource-ref>
  -----
  -----
</resource-ref>
```

Sun-resources.xml

Project

JSP files:

index.jsp, response.jsp

Configuration files:

web.xml

Server: Glassfish Server 3



Enable JDBC driver deployment

Domain1

Services

Databases

Jdbc:mysql://localhost:3306

mynewdatabase



DATA resource

Jdbc resource: jdbc/IFPWAFCAD

New jdbc connection pool: **IFpwafcadPool**

Jndi name: jdbc/IFPWAFCAD

Sun-resources.xml

Both data source and connection pool need to be registered with Glassfish server

Must reside inside the server's directory

Database Driver

Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml

```
<resource-ref>
  -----
  -----
</resource-ref>
```

Project

JSP files:

index.jsp, response.jsp

Configuration files:

web.xml

Server: Glassfish Server 3



Enable JDBC driver deployment

Domain1

Services

Databases

Jdbc:mysql://localhost:3306

mynewdatabase



DATA resource

Jdbc resource: jdbc/IFPWAFCAD

New jdbc connection pool: **IFpwafcadPool**

Jndi name: jdbc/IFPWAFCAD

Both data source and connection pool need to be registered with Glassfish server

Sun-resources.xml

Must reside inside the server's directory

Database Driver

Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml

```
<resource-ref>
  -----
  -----
</resource-ref>
```

Contains a reference to the data source

Project

JSP files:

index.jsp, response.jsp

Configuration files:

web.xml

Server: Glassfish Server 3



Enable JDBC driver deployment

Domain1

Services

Databases

Jdbc:mysql://localhost:3306

mynewdatabase



DATA resource

Jdbc resource: jdbc/IFPWAFCAD

New jdbc connection pool: **IFpwafcadPool**

Jndi name: jdbc/IFPWAFCAD

Sun-resources.xml

Both data source and connection pool need to be registered with Glassfish server

Must reside inside the server's directory

Database Driver

Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml

```
<resource-ref>
  -----
  -----
</resource-ref>
```

Contains a reference to the data source