Building Dynamic Java Web Applications (Part 2)

Glassfish, JAVA EE, Servlets, JSP, EJB
### Example: Simple MySQL database

<table>
<thead>
<tr>
<th>Software or Resource</th>
<th>Version Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetBeans IDE, Java bundle</td>
<td>6.8 or <strong>6.9</strong></td>
</tr>
<tr>
<td>Java Development Kit (JDK)</td>
<td><strong>6</strong></td>
</tr>
<tr>
<td>MySQL database server</td>
<td><strong>5.1</strong></td>
</tr>
<tr>
<td>MySQL Connector/J JDBC Driver version</td>
<td><strong>5.x</strong></td>
</tr>
<tr>
<td>GlassFish server</td>
<td>v3 or Open Source Edition</td>
</tr>
</tbody>
</table>

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.
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: Temper Management

Submit

Response.jsp

Temper Management

Description:
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.

Counselor: Randy "Macho Man" Savage
member since: 2000-11-10

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:

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

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

We can easily add a table here using a Palette.
Add some HTML elements
Invoke the **Palette Window**

From the menu, select Window/Palette

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

This toggles between Show Big and Small icons.

This is the result of changing the display into using 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
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:

<form action="response.jsp">
  <strong>Select a subject:</strong>
</form>
Add a Drop-Down List using the Auto-completion support (Ctrl+Space)

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

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

New File

Steps
1. Choose File Type
2. ...

Choose File Type
Project: IFPWAFCAD

Categories:
- Java Card 3 Platform
- Web
- JavaServer Faces
- Struts
- Spring Framework
- Context and Dependency Injection
- Java
- JavaFX
- Swing GUI Forms
- JavaBeans Objects

File Types:
- JSP
- JSF Page
- Servlet
- Filter
- Web Application Listener
- HTML
- XHTML
- Cascading Style Sheet
- JavaScript File
- JSON File
- Tag Handler
- Tag Library Descriptor

Description:
Creates an empty cascading style sheet (CSS) document. Use a CSS to format the information contained in your XML document.

< Back  Next >  Finish  Cancel  Help
Add CSS

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

**Name and Location**
- **CSS File Name:** style
- **Project:** IFPWAFCAD
- **Location:** Web Pages
- **Folder:**
- **Created File:** C:\Documents and Settings\nhreyes\My Documents\NetBeansProjects\IFPWAFCAD\web\style.css
Add CSS

• 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

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:  

Submit
<table>
<thead>
<tr>
<th><strong>{placeholder}</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong></td>
</tr>
</tbody>
</table>
| **Counselor:**    | {placeholder}  
|                   | member since: {placeholder} |
| **Contact Details:** | **email:** {placeholder}  
|                   | **phone:** {placeholder} |
Changing the root password

This tutorial uses \texttt{nbuser} as an example password. To set your password to \texttt{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('\texttt{nbuser}')
  -> WHERE User = '\texttt{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.

![Database Connection Window]

**Default port number of MySQL server**

**Enter password:** `nbuser`
Connecting to a MySQL Database

- MySQL Server at localhost:3306 [root]
- jdbc:mysql://localhost:3306 [root on Default schema]
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

```sql
1. DROP TABLE IF EXISTS Counselor;
2. DROP TABLE IF EXISTS Subject;

CREATE TABLE Counselor (
    counselor_id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
    first_name VARCHAR (50),
    nick_name VARCHAR (50),
    last_name VARCHAR (50),
    telephone VARCHAR (25),
    email VARCHAR (50),
    member_since DATE DEFAULT '0000-00-00',
    PRIMARY KEY (counselor_id)
);

CREATE TABLE Subject (
    subject_id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
    name VARCHAR (50),
    description TEXT,
    counselor_id SMALLINT UNSIGNED,
    PRIMARY KEY (subject_id)
);
```
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**
Java Naming and Directory Interface (JNDI) – a naming system is required to manage the lookup between names and resources (e.g. Database).
The database that we want to connect to
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
You need to reference the JDBC resource you just configured from the web application.
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**
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.

Web.xml
Verifying the Results of adding the Resource Reference

Web.xml

```xml
  <session-config>
    <session-timeout>
      30
    </session-timeout>
  </session-config>
  <welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
  </welcome-file-list>
  <resource-ref>
    <description>Database for IFPWAFCAD application</description>
    <res-ref-name>dbc/IFPWAFCAD</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
    <res-sharing-scope>Shareable</res-sharing-scope>
  </resource-ref>
</web-app>
```
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.

You are actually connecting to an instance of the Glassfish application server. Each instance runs in a unique domain, and the driver should be located within the domain (e.g. domain1).
We could verify that our **database driver** has been installed properly by inspecting the actual directory of the domain.
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.

From the **Palette**, select the **DB Report** icon.
Specify the data source and variable name for the **DB Report**

![Database Report Insertion Window]

- **Variable Name**: `subjects`
- **Data Source**: `jdbc/IFPWAFCAD`
- **Query Statement**:
  ```sql
  SELECT subject_id, name FROM Subject
  ```

**JSTL Tags**

- `<sql:query>` to create query
- `<c:forEach>` to loop through the query’s results set and output the retrieved data
<%@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>

<table border="1">
    <tr>
        <th><c:out value="${columnName}" /></th>
    </tr>
    <c:forEach var="row" items="${subjects.rowsByIndex}"
        var="column">
        <tr>
            <td><c:out value="${column}" /></td>
        </tr>
    </c:forEach>
</table>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@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>

<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>

<!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=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>
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:

<table>
<thead>
<tr>
<th>subject_id</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Financial Consultancy</td>
</tr>
<tr>
<td>2</td>
<td>Existential Psychotherapy</td>
</tr>
<tr>
<td>3</td>
<td>Temper Management</td>
</tr>
<tr>
<td>4</td>
<td>Past-Life Regression</td>
</tr>
<tr>
<td>5</td>
<td>Marriage Guidance</td>
</tr>
<tr>
<td>6</td>
<td>Crisis Management</td>
</tr>
<tr>
<td>7</td>
<td>Dream Analysis</td>
</tr>
<tr>
<td>8</td>
<td>Hypnosis</td>
</tr>
<tr>
<td>9</td>
<td>Reiki</td>
</tr>
</tbody>
</table>
The Project, so far...

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
Next, we would like to put these generated HTML codes into the drop-down list.
Integrating the generated code into the HTML drop-down list
Integrate the `<c:forEach>` tags into the HTML form as follows.

```html
<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>
```

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...

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
- Temper Management
- Past-Life Regression
- Marriage Guidance
- Crisis Management
- Dream Analysis
- Hypnosis
- Reiki

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

Customise the query as indicated below.

```
<%@page
counselorQuery>
page
jdbc/IFPWAFCAD
SELECT * FROM Subject, Counselor WHERE Counselor.counselor_id = Subject.counselor_id AND Subject.subject_id = ? <sql:param value="${param.subject_id}"/>
```
Automatically generated code

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`
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>`.
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>
```
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: Temper Management  [Submit]

Subject selected

Response.jsp

Temper Management

Description: 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.

Counselor: Randy "Macho Man" Savage  
member since: 2000-11-10

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
0. JSP is deployed to the server and compiled into a servlet
We would like to populate the drop-down list with records coming from MyNewDatabase, Subject table.
1. Request for index.jsp

Web browser

MySQL Server - MyNewDatabase

- Subject (Table)
- Counselor (Table)

Project Name: IFPWAFCAD

Name: submit

Glassfish Java EE application server

Index.jsp

Name: subject_id

Jsf-impl.jar

JSTL codes

submit

submit

Server-Client Interaction
1. Request for index.jsp

2. Initiate query on subject table

Web browser

MySQL Server

MyNewDatabase

Subject (Table)

Counselor (Table)

Data resource:

Name: jdbc/IFPWAFCAD

Glassfish Java EE application server

Index.jsp

JSTL codes

Jsf-impl.jar
Server-Client Interaction

Web browser

MySQL Server

MyNewDatabase

- Subject (Table)
- Counselor (Table)

Glassfish Java EE application server

Index.jsp

JSTL codes

Jsfs-impl.jar

Data resource:

Name: jdbc/IWPWAFCAD

3. Send records based on query result

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

2. Initiate query on subject table
5. Send generated page back to browser via HTTP

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

MySQL Server

MyNewDatabase

Subject (Table)

Counselor (Table)

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

DATABASE DRIVER

Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml

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

DATA Resource

Jdbc resource: jdbc/IFPWAFCAD

- New jdbc connection pool: IFpwafcadPool
- Jndi name: jdbc/IFPWAFCAD

Sun-resources.xml

Must reside inside the server’s directory
**Project**

JSP files:

- index.jsp, response.jsp

Configuration files:

- web.xml

**Services**

**Databases**

- Jdbc:mysql://localhost:3306
- mynewdatabase

**Server: Glassfish Server 3**

- Enable JDBC driver deployment

**Domain1**

- 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>`

**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
JSP files:
index.jsp, response.jsp

Configuration files:
web.xml

Services

Databases

Jdbc:mysql://localhost:3306

mynewdatabase

Server: Glassfish Server 3

Domain1

Enable JDBC driver deployment

Must reside inside the server’s directory

Database Driver

Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml

Contains a reference to the data source

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

Jdbc resource: jdbc/IFPWAFCAD

New jdbc connection pool: IFpwafcadPool

Jndi name: jdbc/IFPWAFCAD

Sun-resources.xml

<resource-ref>
      
      
</resource-ref>
Project

JSP files:
index.jsp, response.jsp
Configuration files:
web.xml

Server: Glassfish Server 3

Services

Databases
Jdbc:mysql://localhost:3306
mynewdatabase

Domain1

Enable JDBC driver deployment

Services

Database Driver
Mysql-connector-java.5.1.6-bin.jar

Deployment Descriptor

Web.xml
<resource-ref>
<res-ref>
<res-type>jdbc</res-type>
<res-name>IFPWAFCAD</res-name>
<res-auth>container</res-auth>
<res-root>/IFPWAFCAD</res-root>
</res-ref>
</resource-ref>

Data resource

Jdbc resource: jdbc/IFPWAFCAD
New jdbc connection pool: IFpwafcadPool
Jndi name: jdbc/IFPWAFCAD

Contains a reference to the data source

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

Sun-resources.xml