Using **Beans** in JSP Pages

Using Beans in JSP Pages

- A Java Bean is a specially constructed Java class written in the Java.

- Java Beans are java classes that have properties, which can be read via a get method or changed via a set method.

- A Java Bean should not have any public variables. All the variables should be accessed using the get/set methods.

JavaBeans Properties

- A JavaBean property is a named attribute that can be accessed by the object. The attribute can be of any Java data type, including the classes that you define.

- JavaBean properties are accessed through two methods in the JavaBean's implementation class.

Using Beans in JSP Pages

getPropertyName()

- For example, if property name is *firstName*, your method name would be **getFirstName()** to read that property.

setPropertyName()

- For example, if property name is *firstName*, your method name would be **setFirstName()** to write that property.

Accessing JavaBeans

- Following JSP action elements are required to use Java bean in a JSP file.
  - **<jsp:useBean>**
  - **<jsp:getProperty>**
  - **<jsp:setProperty>**

Load Java bean inside a JSP :

- We need to use **\<jsp:useBean\>** tag to load a bean into JSP.
- The basic syntax is given below:

  \<jsp:useBean id="objname" class="ClassName" /\>

Getting the Properties of the Bean :

- After the bean gets loaded into the page, the properties can be accessed by using the **\<jsp:getProperty\>** action element.
- The basic syntax is given below:

\<jsp:getProperty name="objname" property="propertyname"/\>

Setting the Properties of the Bean :

- To modify (or) assign value to any variable of the bean object the **\<jsp:setProperty\>** action element is used .
- The basic syntax is as below

  \<jsp:setProperty name="objname" property="name" value="anyvalue"/\>

Example: "**Employee.java**"

Step 1: Create Java Bean

```java
package college;
public class Employee {
    private String name = "Madhu";
    private String department = "CSE";
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getDepartment() {
        return department;
    }
    public void setDepartment(String department) {
        this.department = department;
    }
}
```

Example: "**jbeandemo.jsp**"

Step 2: Create JSP

```html
<html>
<head> <title> Java Bean Demo</title></head>
<body>
<jsp:useBean id="employee" class="college.Employee" />
<h2>Old Employee details </h2>
Name: <jsp:getProperty name="employee" property="name" /><br/>
Department :<jsp:getProperty name="employee" property="department" />
<jsp:setProperty name="employee" property="name" value="Durga" />
<jsp:setProperty name="employee" property="department" value="CSE" />
<h2> New Employee details </h2>
Name:<jsp:getProperty name="employee" property="name" /><br/>
Department :<jsp:getProperty name="employee" property="department" />
</body>
</html>
```
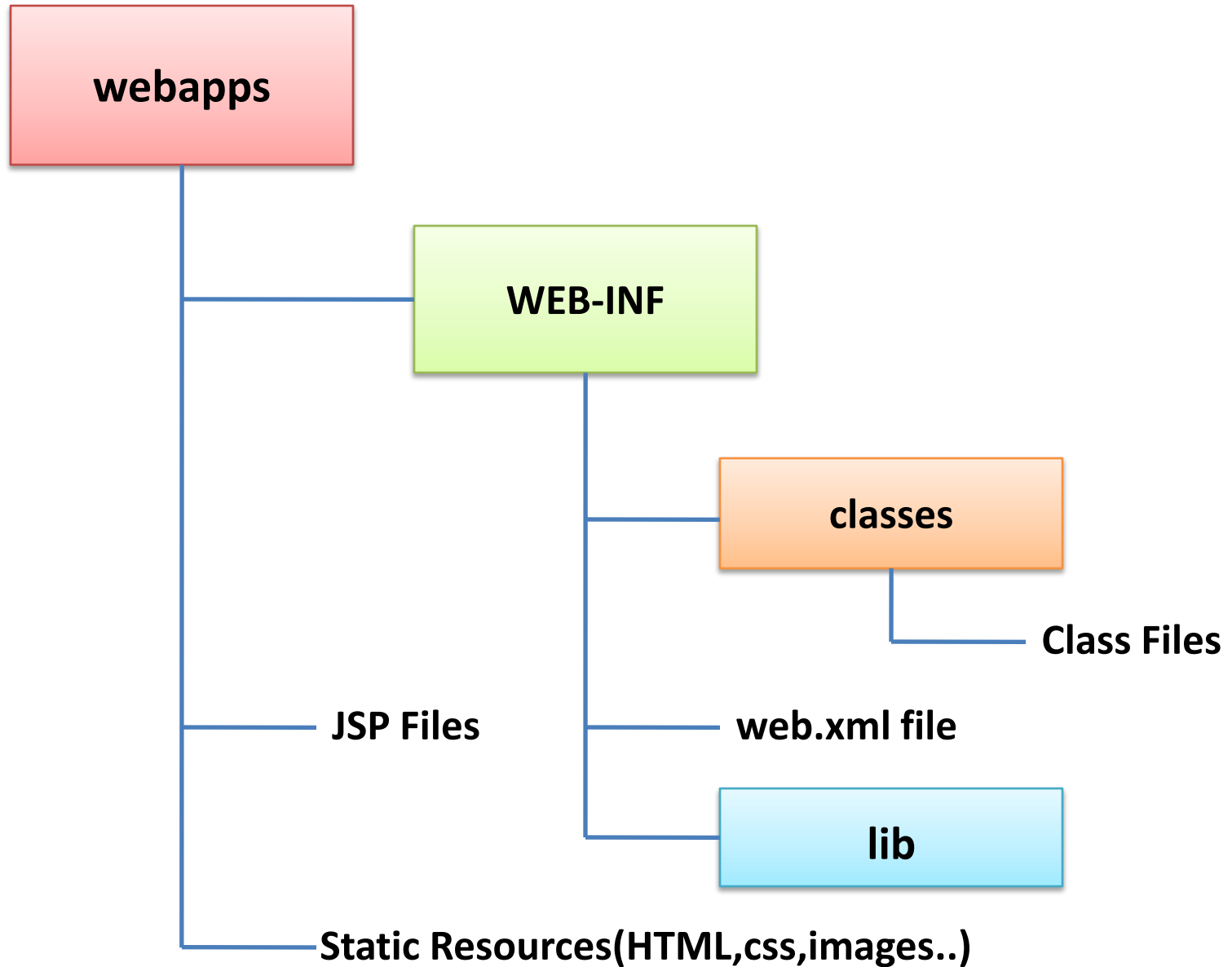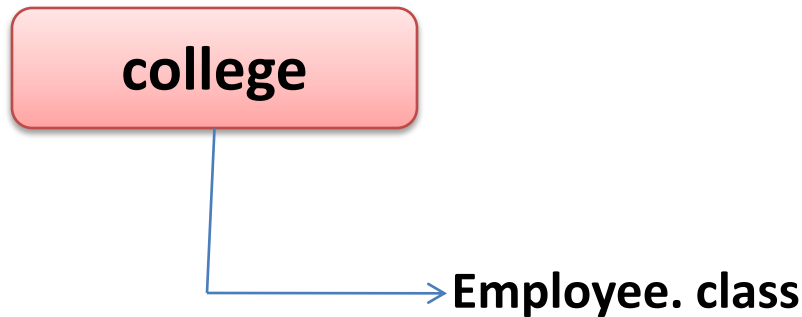
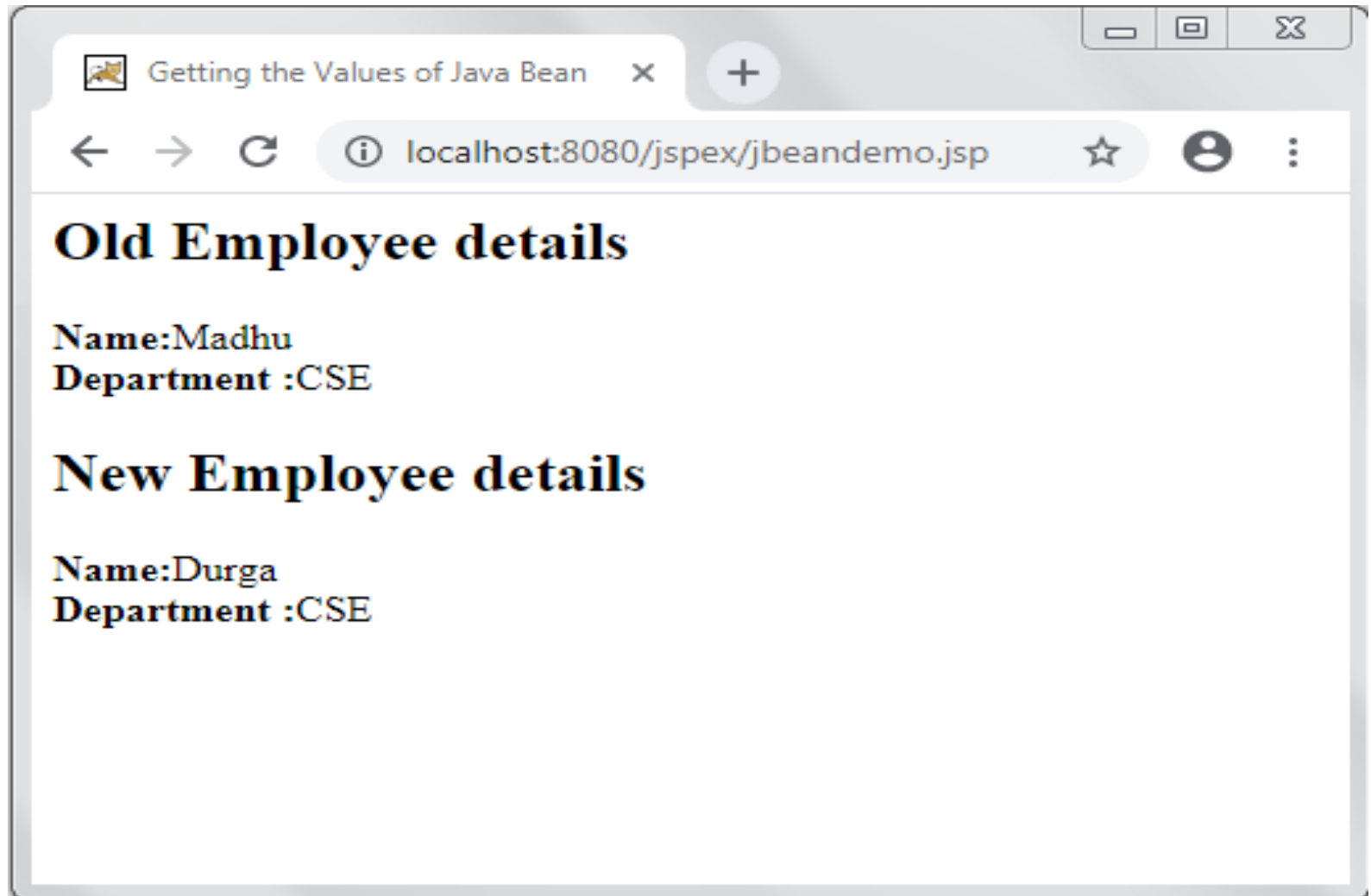# Step 3: Create Folder Structure

Execution Procedure:

1. We need to compile **Employee.java** file.

2. After successful compilation we get package along with class file.

3. We need to copy **Employee. class** along with **college** package into **classes** folder which was in **WEB-INF** folder.

4. After copying, We need to execute JSP file(**jbeandemo.jsp)** using Tomcat server.

   i.e., open any browser give following as URL

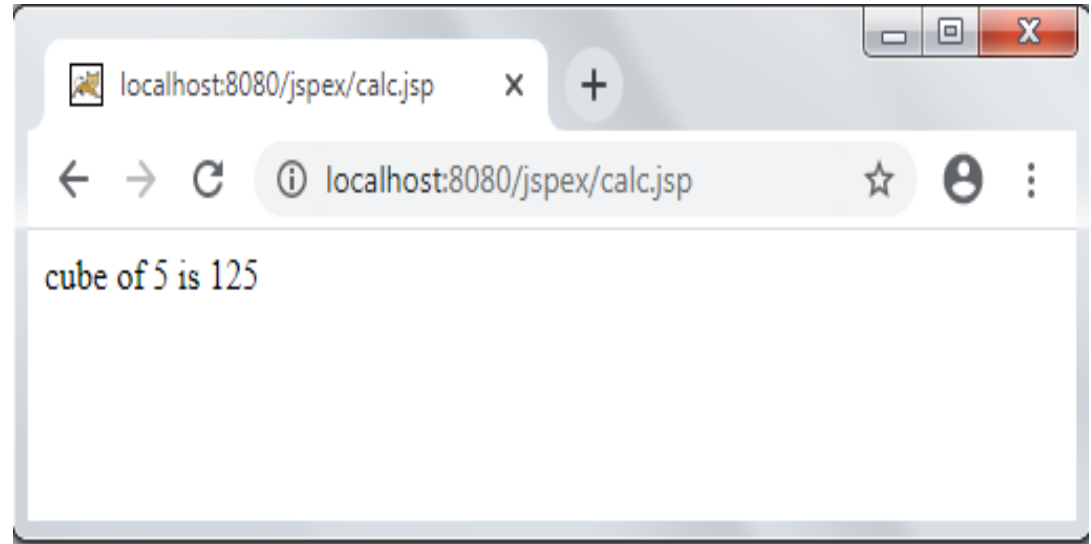   **localhost:8080/foldername/jbeandemo.jsp**

Output:

Example: "**Calculator.java**"

- **Create Java Bean**

package mycalc;

public class Calculator{

public int cube(int n)

   {

        return n*n*n;

   }

}

localhost:8080/jspex/calc.jsp

cube of 5 is 125

- **Create JSP file** "**calc.jsp**"

```
<jsp:useBean id="obj" class="mycalc.Calculator"/>
<%
int m=obj.cube(5);
out.print("cube of 5 is "+m);
%>
```