EXERCISE

HANDLING OF JAVABEANS
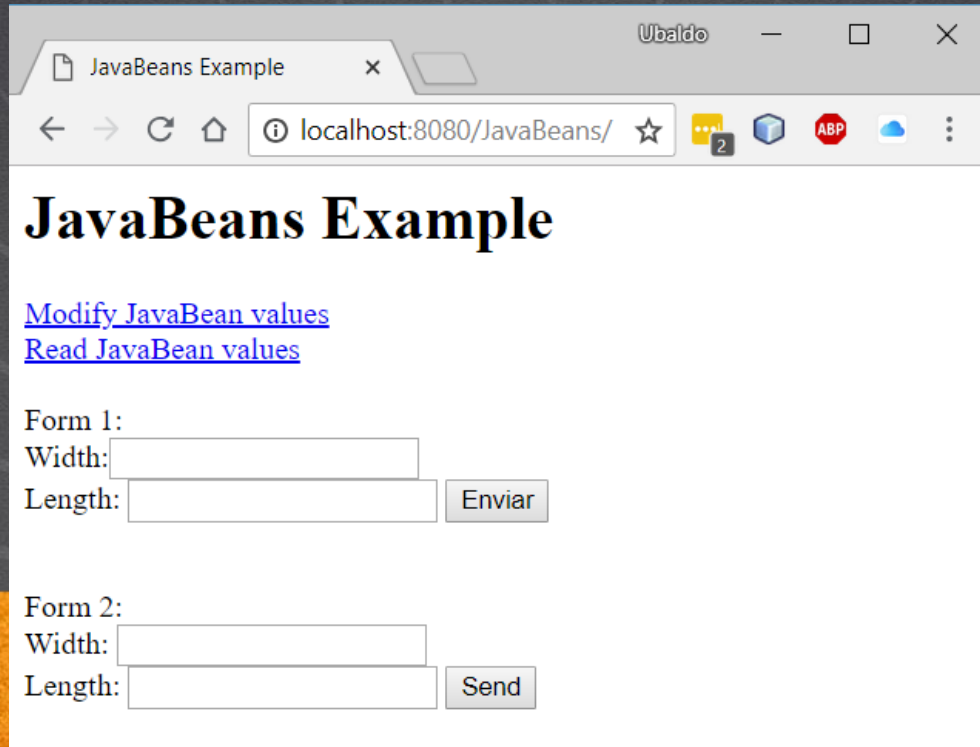

Global Mentoring

EXERCISE OBJECTIVE

Create a project to implement the Handling of JavaBeans with JSPs. At the end we should observe the following:

1. CREATE A NEW PROJECT

Create a new project:

1. CREATE A NEW PROJECT

Create a new project:

2. DELETE THE INDEX.HTML FILE

Delete the index.html file:
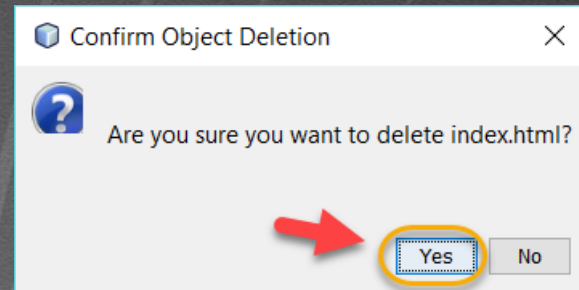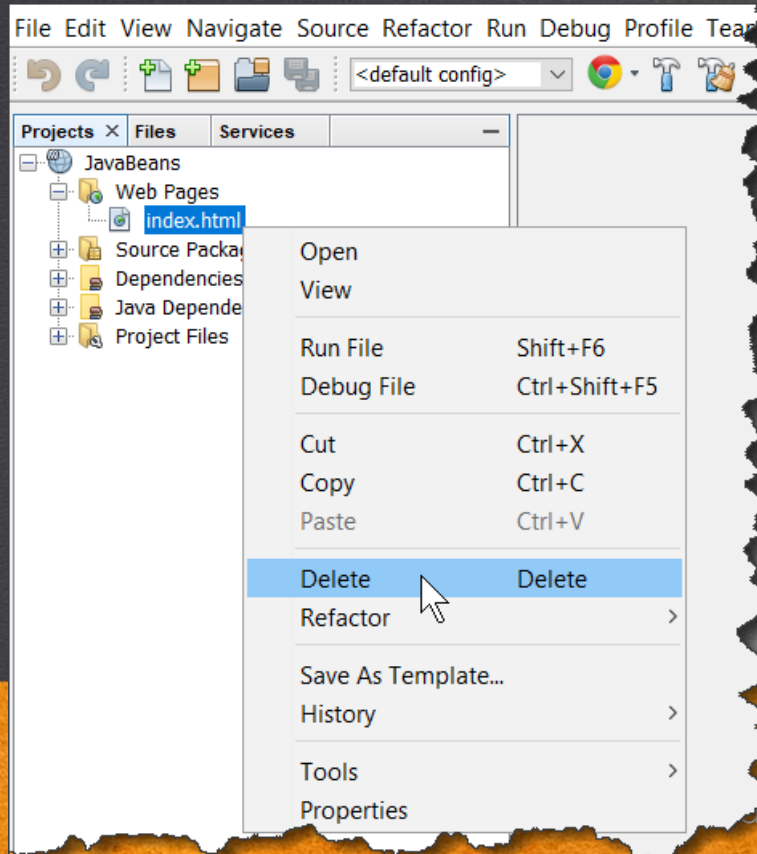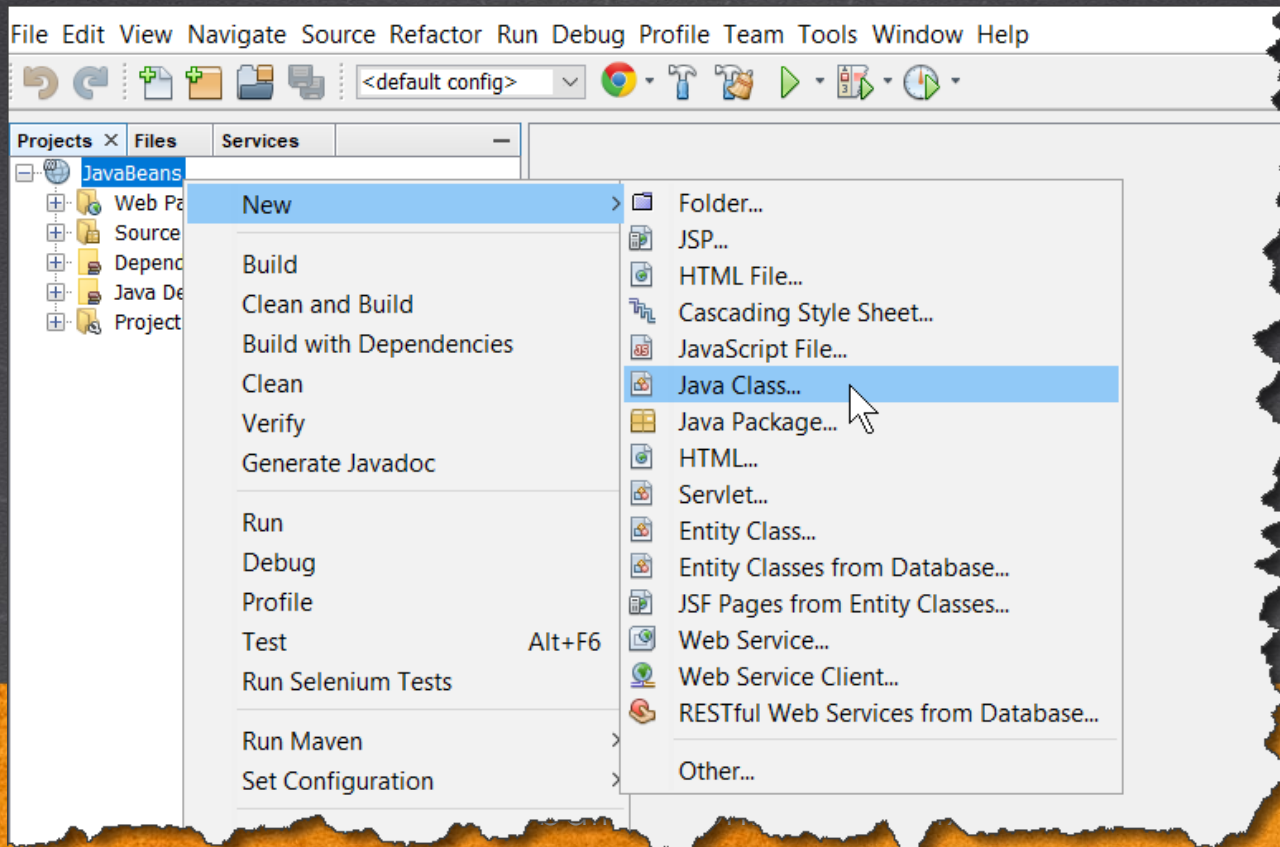
3. CREATE A NEW JAVA CLASS

Create a new Java Class:

3. CREATE A NEW JAVA CLASS

Create a new Java Class:

4. MODIFY THE CODE

Rectangle.java:

```java
package beans;
public class Rectangle {
    private int width;
    private int length;

     public int getWidth() {
        return width;
    }

    public void setWidth(int width) {
        this.width = width;
    }

    public int getLength() {
        return length;
    }
    public void setLength(int length) {
        this.length = length;
    }

    /**
     * We add a method that does not map with any property but is the
     * result of a calculation
     *
     * @return int area value
     */
    public int getArea() {
        return this.width * this.length;
    }
}
```

5. CREATE A NEW JSP

Create a new JSP:
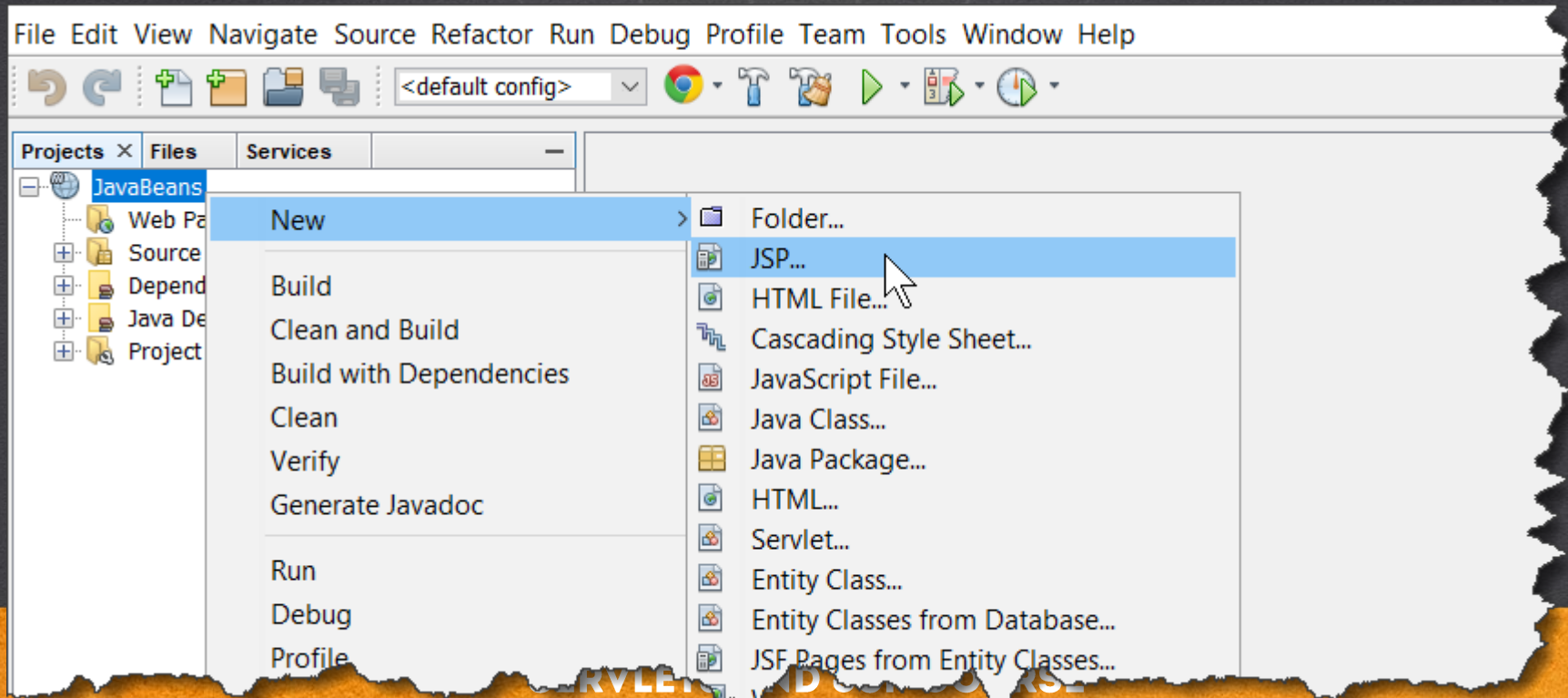
5. CREATE A NEW JSP

Create a new JSP:

6. MODIFY THE CODE

getterJsp.jsp:

```html
<html>
    <head>
        <meta charset="UTF-8">
        <title>JSP that reads JavaBean values</title>
    </head>
    <body>
        <h1>JSP that reads JavaBean values</h1>
        <!-- We declare the JavaBean to use
         if there already exists a bean in the session called "rectangle",
         return the same object and do not create a new one -->
        <jsp:useBean id="rectangle" class="beans.Rectangle" scope="session" />

        <!--We retrieve the values and show them-->
        <br>
        Width value: <jsp:getProperty name="rectangle" property="width" />
        <br>
        Length value: <jsp:getProperty name="rectangle" property="length" />
        <br>
         <!--In this case, it does not matter that the property does not exist
         in the JavaBean, just with having a method called getArea ()
         it is possible to simulate a property -->
        Area value: <jsp:getProperty name="rectangle" property="area" />
        <br>
        <br>
        <a href="index.jsp">Return to index</a>
    </body>
</html>
```

7. CREATE A NEW JSP

Create a new JSP:
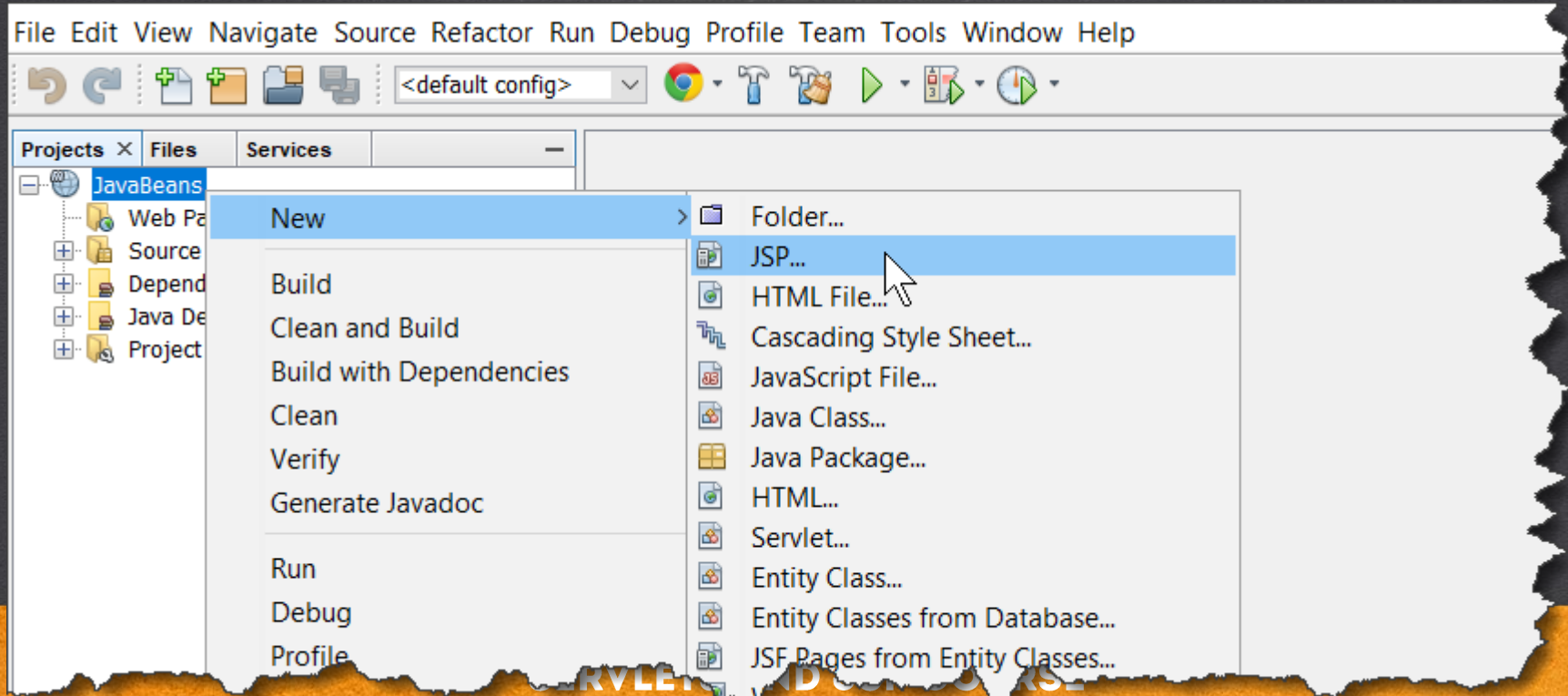
7. CREATE A NEW JSP

Create a new JSP:

8. MODIFY THE FILE

setterJsp.jsp:

Click to download

```html
<html>
    <head>
        <meta charset="UTF-8">
        <title>JSP that modifies a JavaBean</title>
    </head>
    <body>
        <h1>JSP that modifies a JavaBean</h1>

        <!-- Declare the JavaBean to be used -->
        <!-- The scope of the variable is of type session -->
        <jsp:useBean id="rectangle"
                    class="beans.Rectangle" scope="session" />
```

Archivo setterJsp.jsp:

Click to download

```jsp
        Modify the attributes:
        <br>
        <br>
        <%
            //Values in hard code
            int widthValue = 5;
            int lengthValue = 10;
        %>
        <jsp:setProperty name="rectangle" property="width"
        value="<%=widthValue%>" />

        New width value: <%=widthValue%>
        <br>

        <jsp:setProperty name="rectangle" property="length"
        value="<%= lengthValue%>" />

        New length value: <%= lengthValue%>

        <br>
        <br>
        <a href="index.jsp">Return to index</a>
    </body>
</html>
```

9. CREATE A NEW JSP

Create a new JSP:
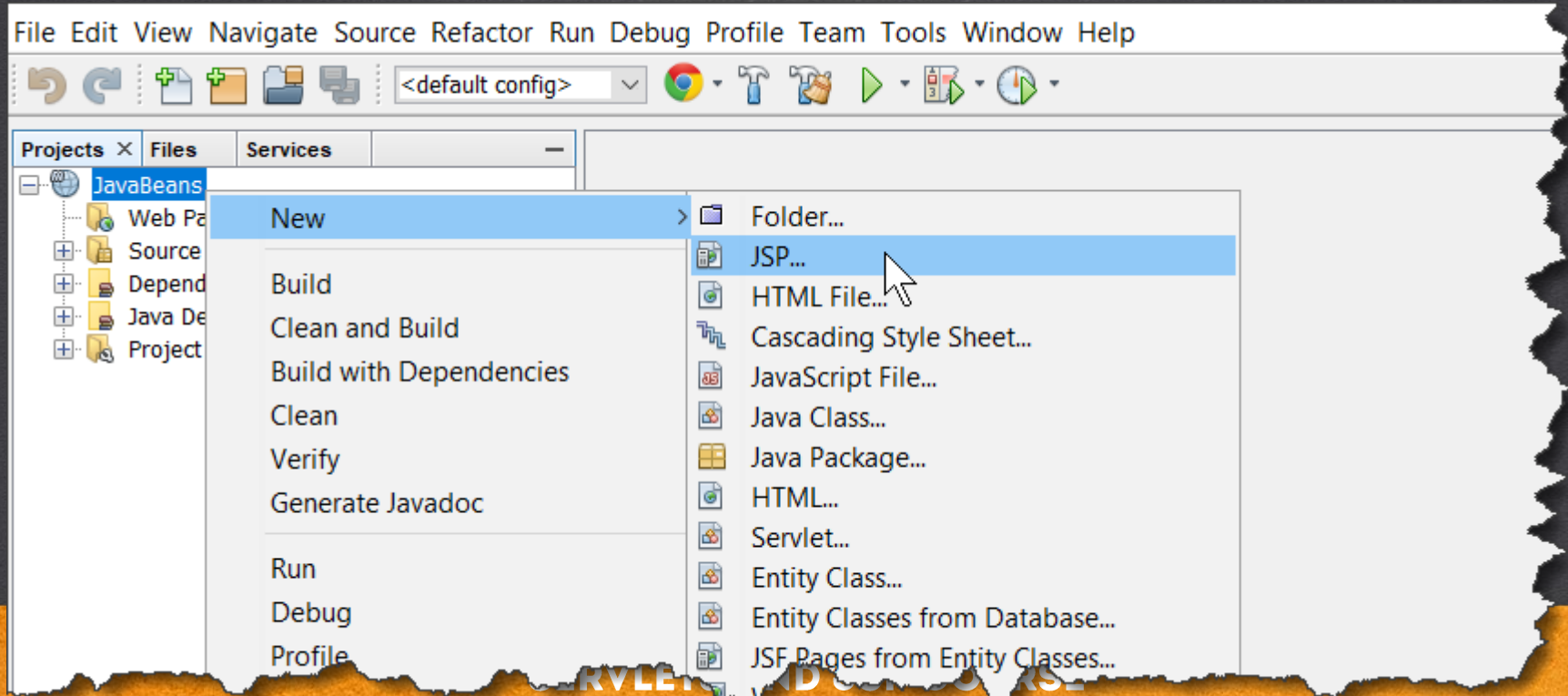
9. CREATE A NEW JSP

Create a new JSP:

setterParamJsp.jsp:

Click to download

```html
<html>
    <head>
        <meta charset="UTF-8">
        <title>JSP that modifies a JavaBean by parameters</title>
    </head>
    <body>
        <h1>JSP that modifies a JavaBean by parameters</h1>

        <!--declare the JavaBean to be used-->
        <jsp:useBean id="rectangle" class="beans.Rectangle" scope="session" />

        Modification of attributes:
        <br>
        <br>

        <jsp:setProperty name="rectangle"
                        property="width" param="widthParam" />

        New width value: <%= request.getParameter("widthParam")%>
        <br>

        <jsp:setProperty name="rectangle"
                        property="length" param="lengthParam" />
```

10. MODIFY THE CODE

setterParamJsp.jsp:

```
        New length value:<%= request.getParameter("lengthParam")%>

        <br>
        <br>
        <a href="index.jsp">Return to index</a>
    </body>
</html>
```
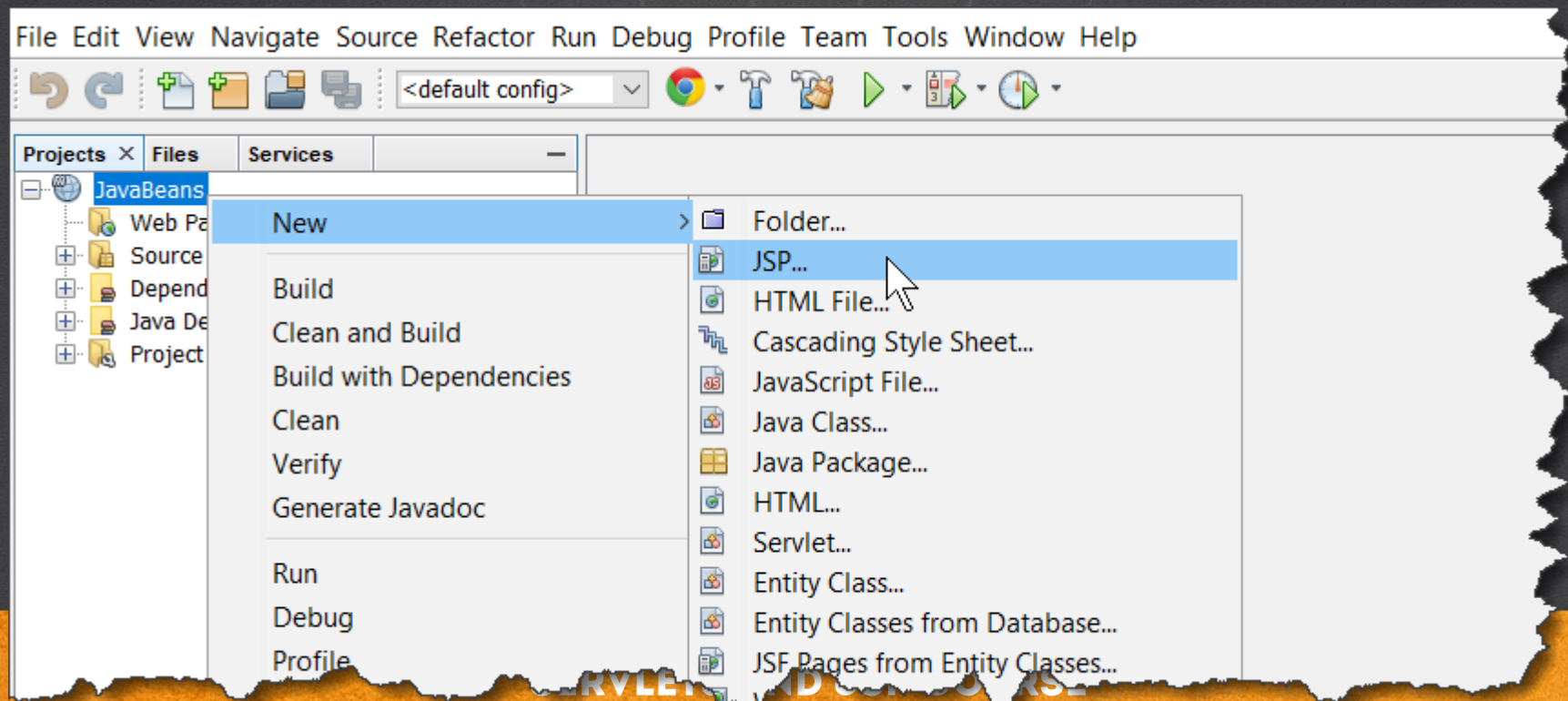
11. CREATE A NEW JSP

Create a new JSP:

11. CREATE A NEW JSP

Create a new JSP:

setterAllParamsJsp.jsp:

Click to download

```html
<html>
    <head>
        <meta charset="UTF-8">
        <title>JSP that modifies a JavaBean by Parameters</title>
    </head>
    <body>
        <h1>JSP that modifies a JavaBean by Parameters</h1>

        <!--declare the JavaBean to be used-->
        <jsp:useBean id="rectangle" class="beans.Rectangle"
                     scope="session" />

        <!--We modify all the properties whose parameter
         match the name of the property-->
        <jsp:setProperty name="rectangle" property="*" />

        All attributes were modified:
        <br>
        <br>

        New width value: <%= request.getParameter("width")%>
        <br>
        New length value: <%= request.getParameter("length")%>
        <br>
        <br>
        <a href="index.jsp">Return to index</a>
    </body>
</html>
```
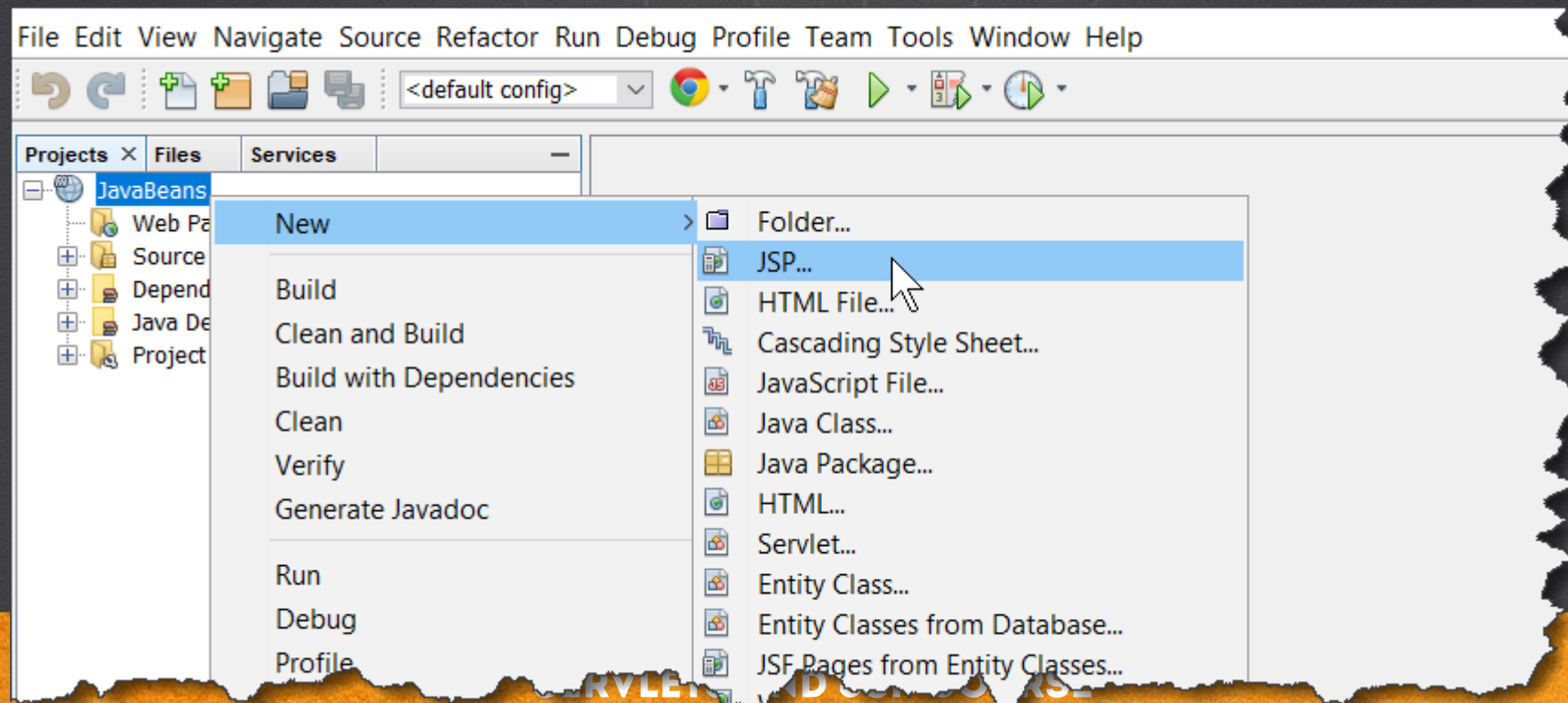
13. CREATE A JSP FILE

Create a JSP file:

13. CREATE A JSP FILE

Create the index.jsp file:

14. MODIFY THE CODE

index.jsp:

```html
<html>
    <head>
        <meta charset="UTF-8">
        <title>JavaBeans Example</title>
    </head>
    <body>
        <h1>JavaBeans Example</h1>
        <!-- JSP to modify values -->
        <a href="setterJsp.jsp">Modify JavaBean values</a>
        <br>
        <!-- JSP to read JavaBean values -->
        <a href="getterJsp.jsp">Read JavaBean values</a>
        <br>
        <br>
```
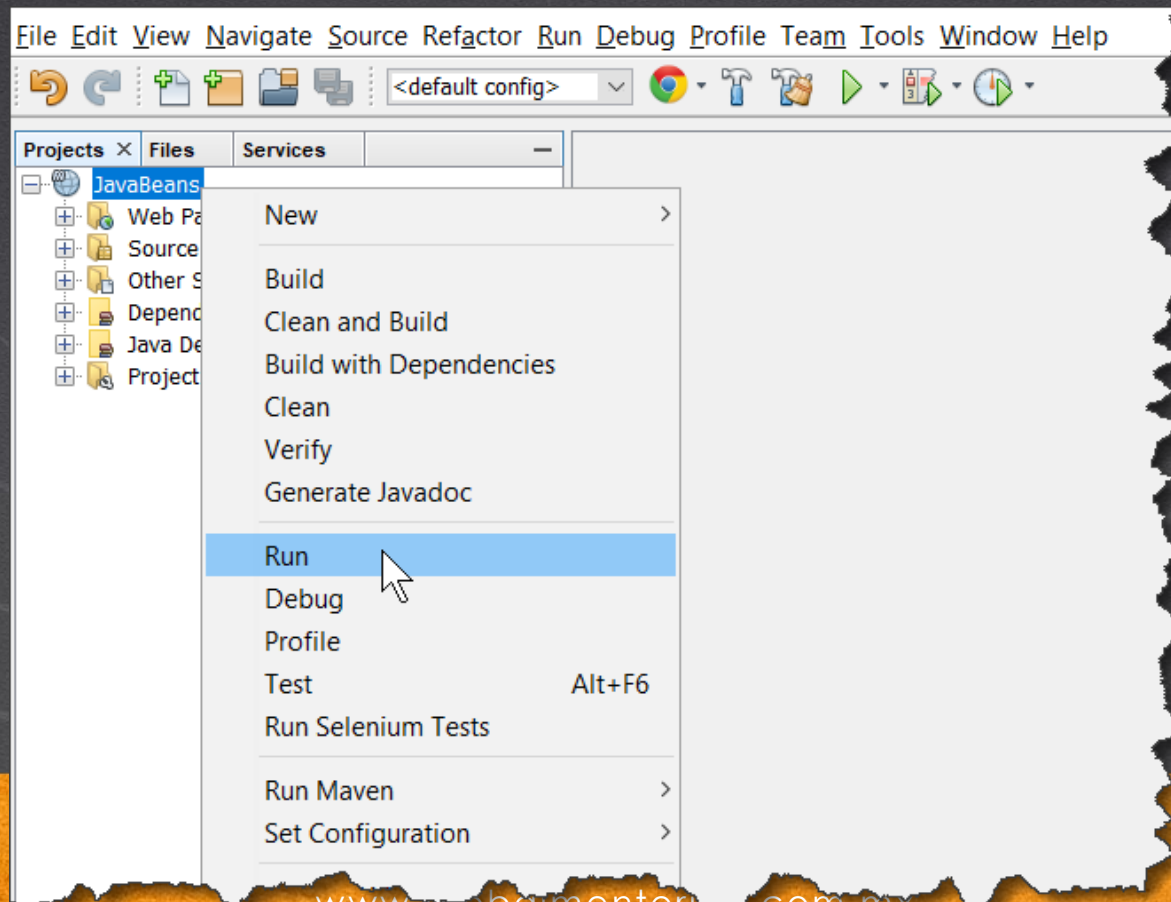
index.jsp:

```html
<!-- JSP to retrieve parameters of a form-->
<!--Form1 -->
Form 1:
<br>
<form name="form1" action="setterParamJsp.jsp">
    Width:<input type="text" name="widthParam">
    <br>
    Length: <input type="text" name="lengthParam">
    <input type="submit" value="Enviar" >
</form>

<br>
Form 2:
<br>
<form name="form2" action="setterAllParamsJsp.jsp">
    Width: <input type="text" name="width" >
    <br>
    Length: <input type="text" name="length" >
    <input type="submit" value="Send">
</form>
</body>
</html>
```

15. EXECUTE THE PROJECT

15. EXECUTE THE PROJECT

EXTRA TASKS OF THE EXERCISE

- Test with each of the options of the project.

- Verify step by step each of the options to clarify any flow or detail that was needed.

EXERCISE CONCLUSION

- With this exercise we have seen the use of JavaBeans and their integration with JSPs.

- This concept, although it seems simple, will be one of the most used throughout our Web applications, not only with the Servlets and JSPs technology, but practically with any Web framework.

Global Mentoring

ONLINE COURSE

SERVLETS & JSP'S

By: Eng. Ubaldo Acosta

Global Mentoring