

SERVLETS AND JSP COURSE

EXERCISE

SCOPES OF VARIABLES WITH MVC (2)



Experiencia y Conocimiento para tu vida

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

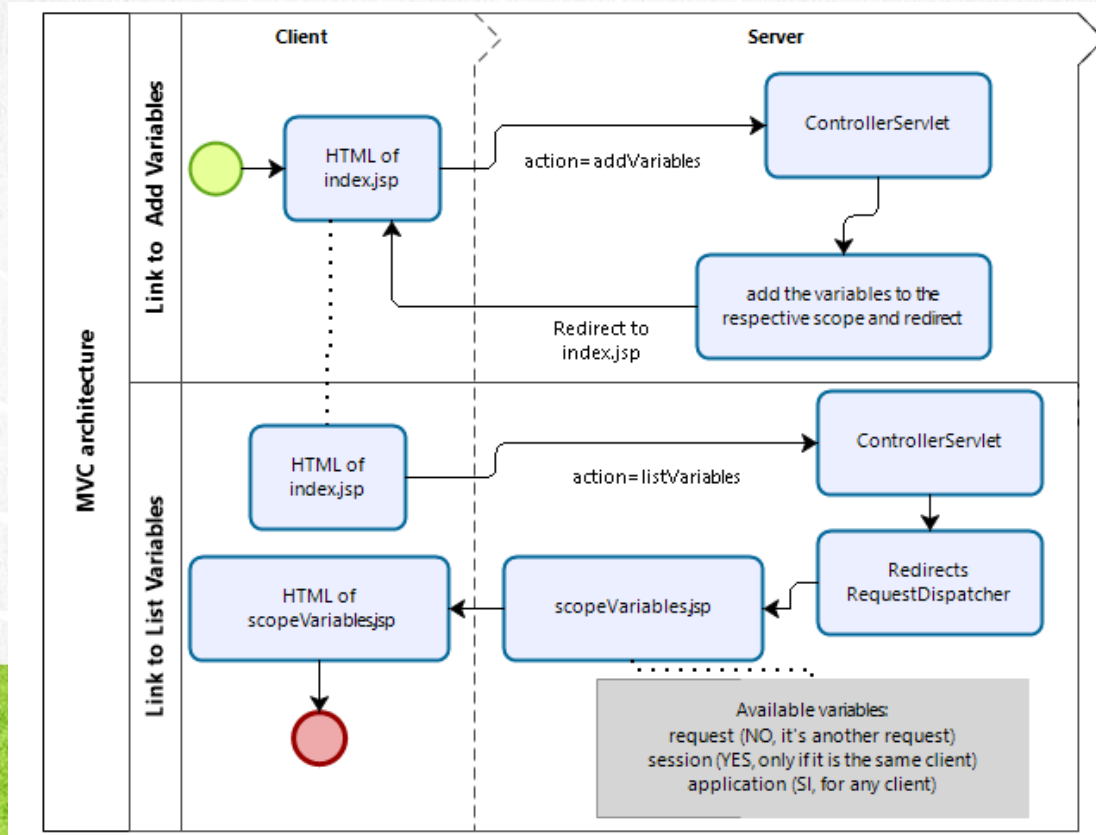
EXERCISE OBJECTIVE

Create a program for the basic management of elements of a JSP. At the end we should observe the following:



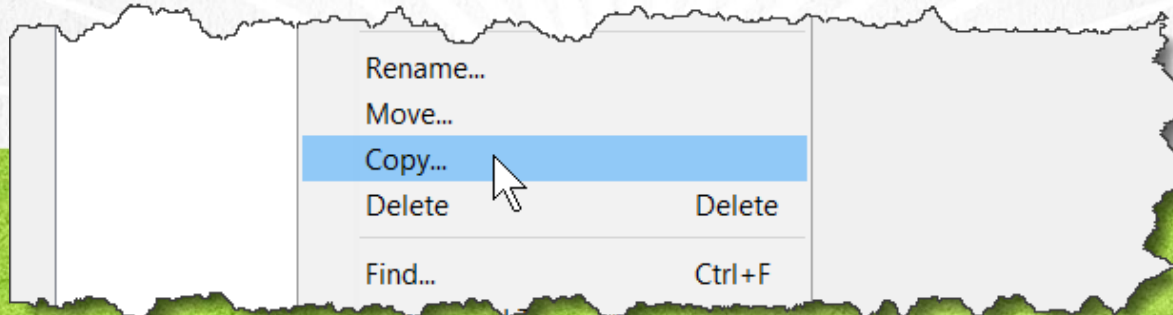
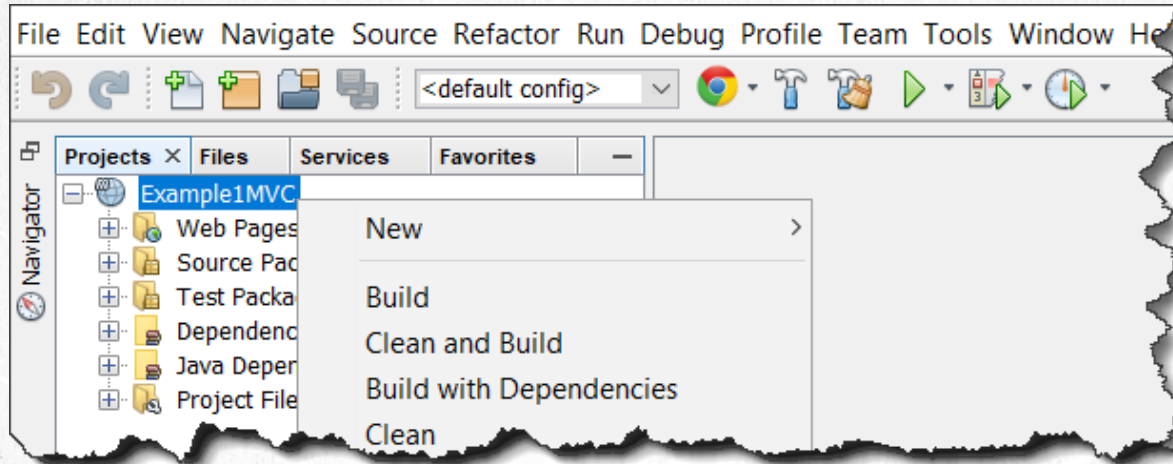
DIAGRAM

This is the diagram that we will take as a basis to develop the exercise:



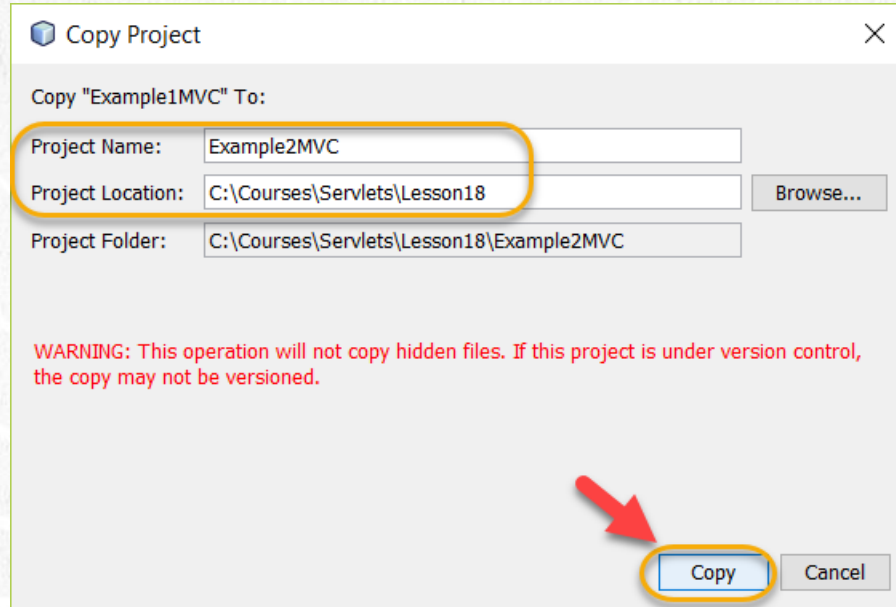
1. COPY THE PROJECT

Copy the previous project:



1. COPY THE PROJECT

Copy the project:



Copy Project

Copy "Example1MVC" To:

Project Name:

Project Location:

Project Folder:

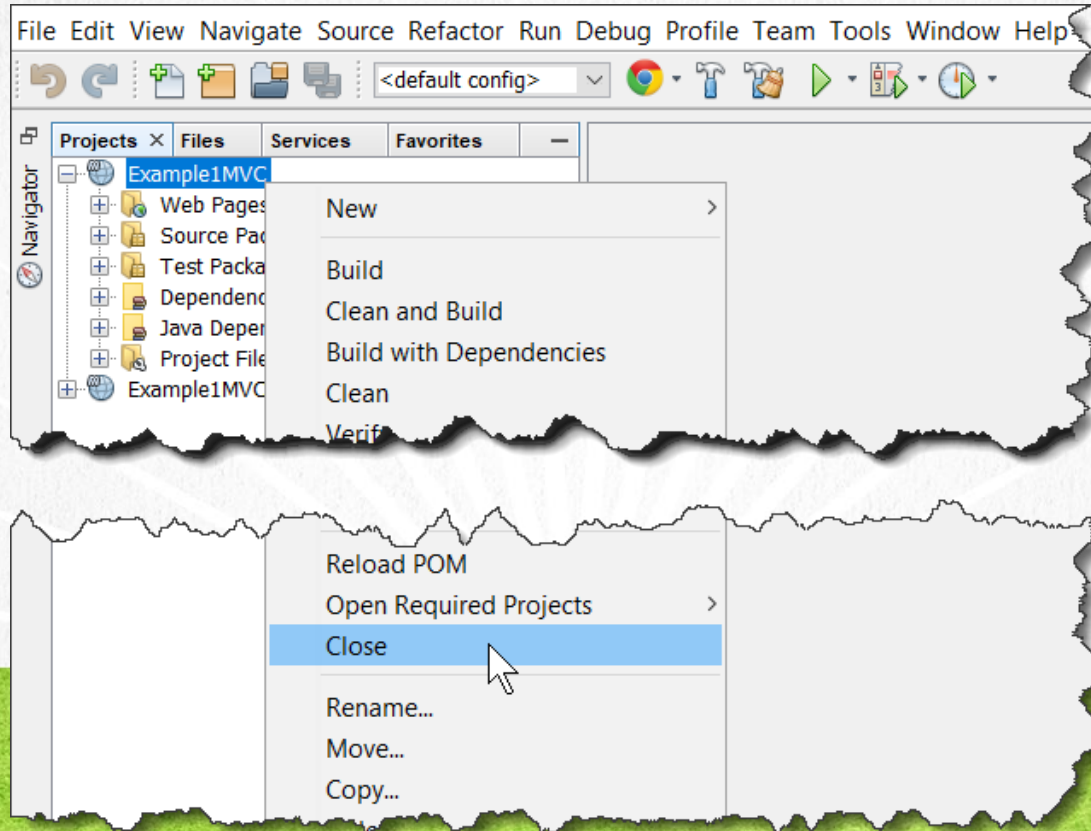
WARNING: This operation will not copy hidden files. If this project is under version control, the copy may not be versioned.

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

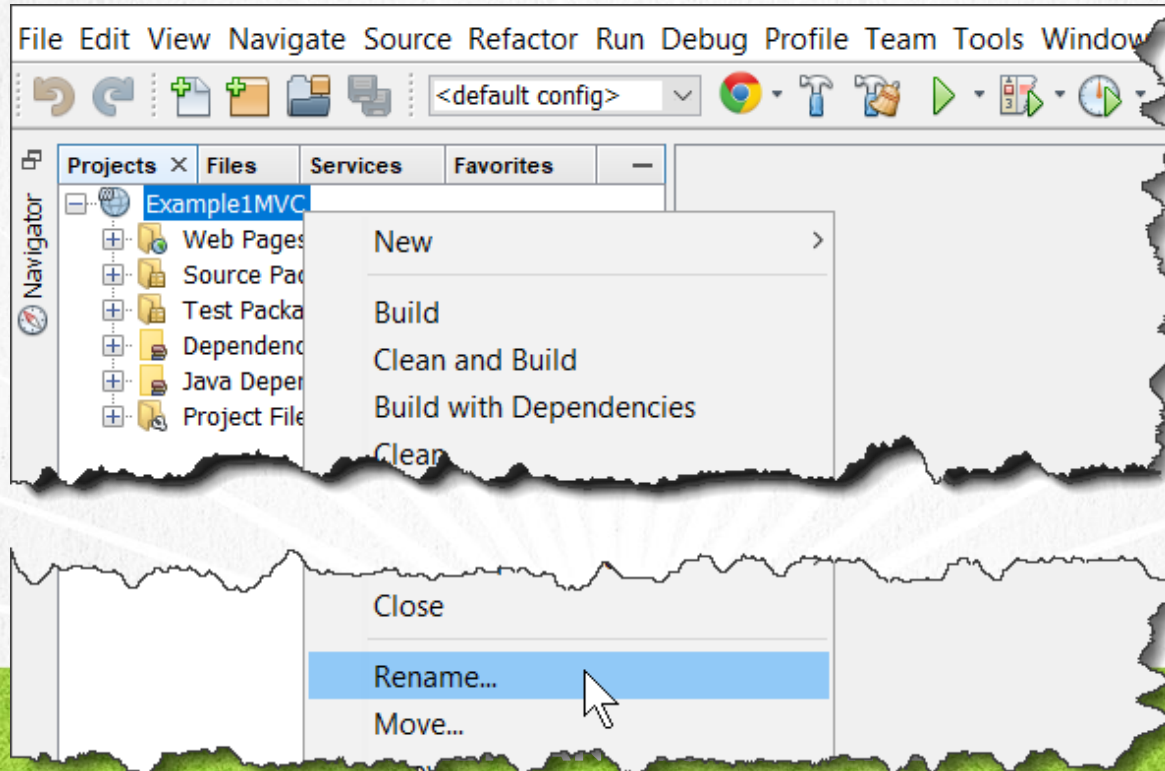
2. CLOSE THE PREVIOUS PROJECT

Close the previous project:



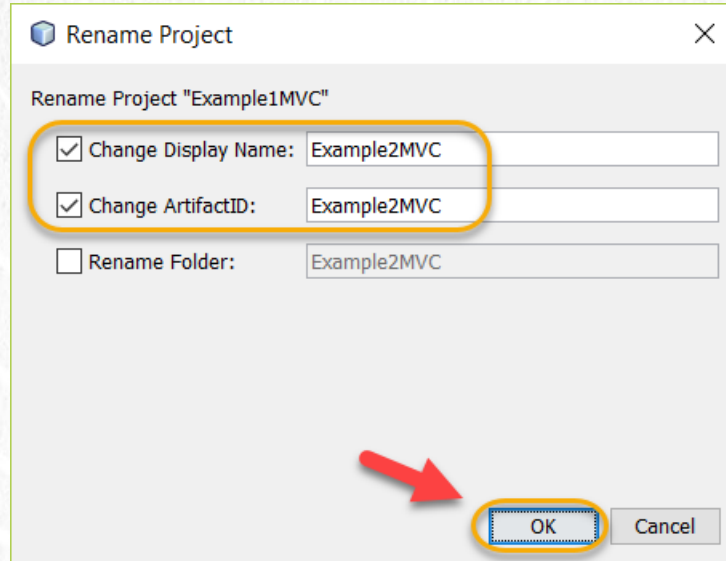
3. RENAME THE NEW PROJECT

Rename the new project:



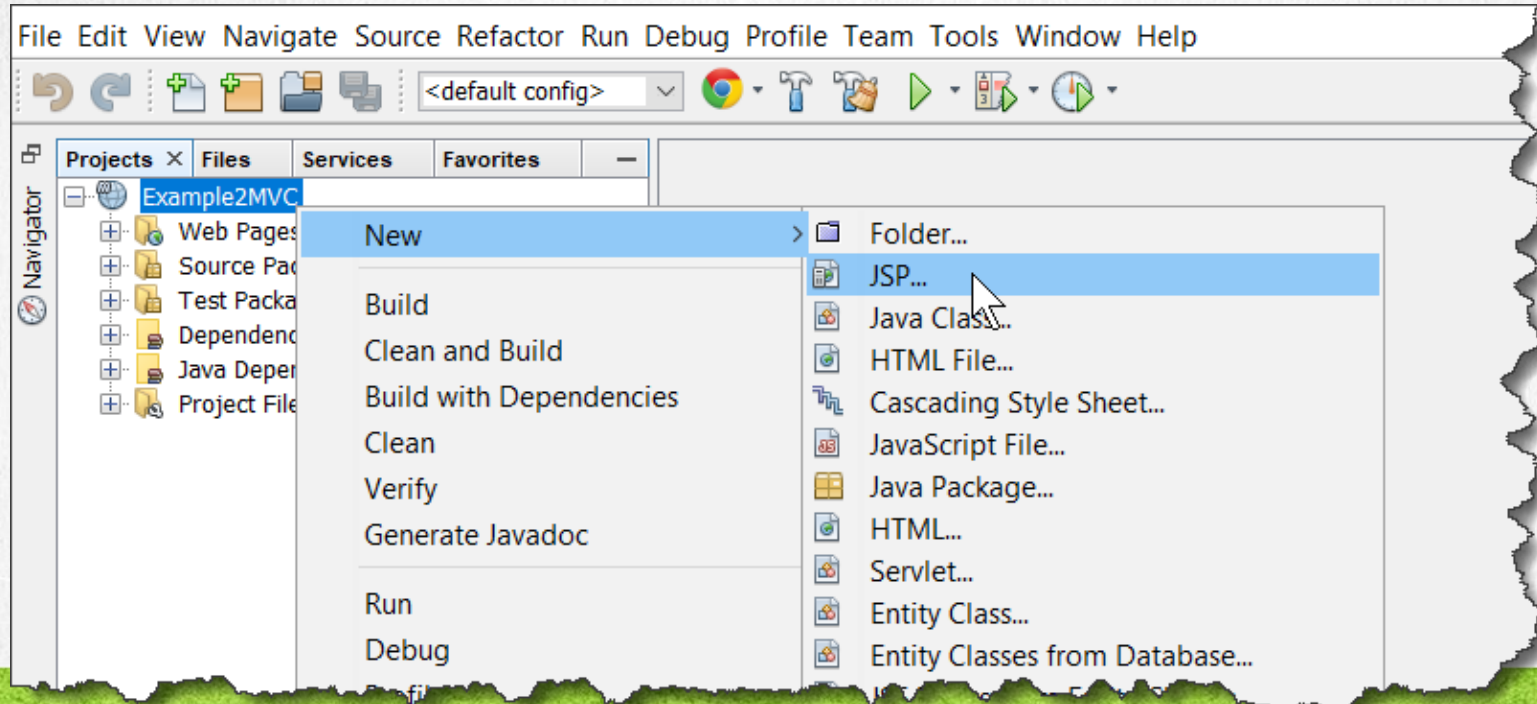
3. RENAME THE NEW PROJECT

Rename the new project:



4. CREATE A NEW JSP

Create a new JSP:



SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

4. CREATE A NEW JSP (CONT)

Create a new JSP:

New JSP

Steps

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

Name and Location

File Name:

Project:

Location:

Folder:

Created File:

Options:

☒ JSP File (Standard Syntax) ☐ Create as a JSP Segment

☐ JSP Document (XML Syntax)

Description:

< Back Next > **Finish** Cancel Help

5. MODIFY THE FILE

[scopeVariables.jsp](#):

Click to download

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Scope Variables</title>
</head>
<body>
  <h1>Scope Variables</h1>
  <br>
  Scope Variable Request:
  Width: ${rectangleRequest.width}
  Length: ${rectangleRequest.length}
  Area: ${rectangleRequest.area}
  <br><br>
  Scope Variable Session:
  Width: ${rectangleSession.width}
  Length: ${rectangleSession.length}
  Area: ${rectangleSession.area}

  <br><br>
  Scope Variable Application:
  Width: ${rectangleApplication.width}
  Length: ${rectangleApplication.length}
  Area: ${rectangleApplication.area}
  <br><br>
  <a href="${pageContext.request.contextPath}/index.jsp">
    Return to index
  </a>
</body>
</html>
```


6. MODIFY THE CODE

[index.jsp:](#)

Click to download

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Example 2 MVC</title>
</head>
<body>
  <h1>Example 2 MVC</h1>
  <div style="color:red"> ${message} </div>
  <br>
  <a href="${pageContext.request.contextPath}/ControllerServlet">
    Link to the Controller Servlet without parameters
  </a>
  <br><br>
  <a href="${pageContext.request.contextPath}/ControllerServlet?action=addVariables">
    Link to the Controller Servlet to add the variables
  </a>
  <br><br>
  <a href="${pageContext.request.contextPath}/ControllerServlet?action=listVariables">
    Link to the Servlet Controller to list the variables
  </a>
</body>
</html>
```

7. MODIFY THE CODE

ControllerServlet.java:

[Click to download](#)

```
package controller;

import java.io.IOException;
import javax.servlet.*;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;
import model.Rectangle;

@WebServlet("/ControllerServlet")
public class ControllerServlet extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        //Process the request
        //1. Process the parameters
        String accion = request.getParameter("action");

        //2. Create the JavaBeans to share
        Rectangle recRequest = new Rectangle(1, 2);
        Rectangle recSession = new Rectangle(3, 4);
        Rectangle recApplication = new Rectangle(5, 6);
```

7. MODIFY THE CODE

ControllerServlet.java:

[Click to download](#)

```
if ("addVariables".equals(accion)) {  
    //3. We share the variables in the scope  
    //selected  
  
    //request scope  
    request.setAttribute("rectangleRequest", recRequest);  
  
    //session scope  
    HttpSession session = request.getSession();  
    session.setAttribute("rectangleSession", recSession);  
  
    //application scope  
    ServletContext application = this.getServletContext();  
    application.setAttribute("rectangleApplication", recApplication);  
  
    //add a message  
    request.setAttribute("message", "The variables were added");  
  
    //4. Redirect to the selected JSP  
    request.getRequestDispatcher("index.jsp").forward(request, response);  
}
```

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

7. MODIFY THE CODE

ControllerServlet.java:

[Click to download](#)

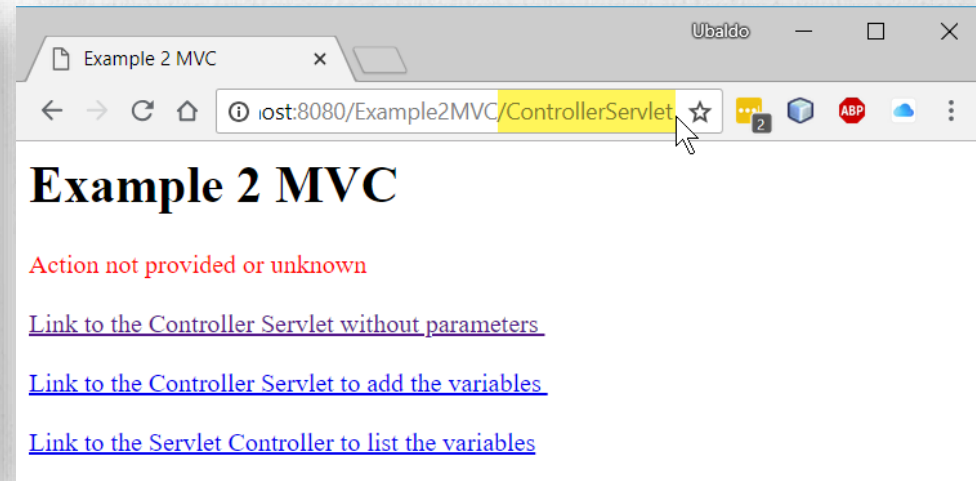
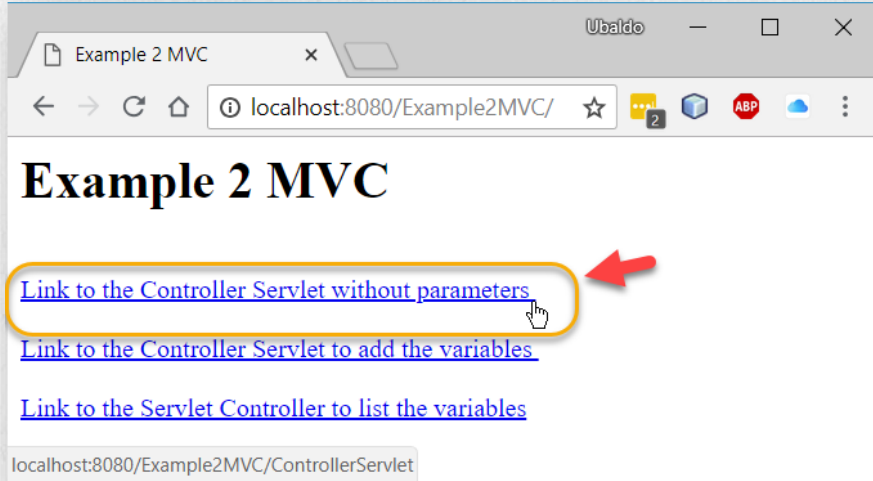
```
} else if ("listVariables".equals(accion)) {  
    //4. We only redirect, we no longer add variables  
    // and only those that remain available will be displayed  
    // according to the scope of each variable  
    request.getRequestDispatcher("/WEB-INF/scopeVariables.jsp").forward(request, response);  
  
} else {  
    //4. Redirect to the index page  
    // add a message  
    request.setAttribute("message", "Action not provided or unknown");  
    request.getRequestDispatcher("index.jsp").forward(request, response);  
}  
}
```

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

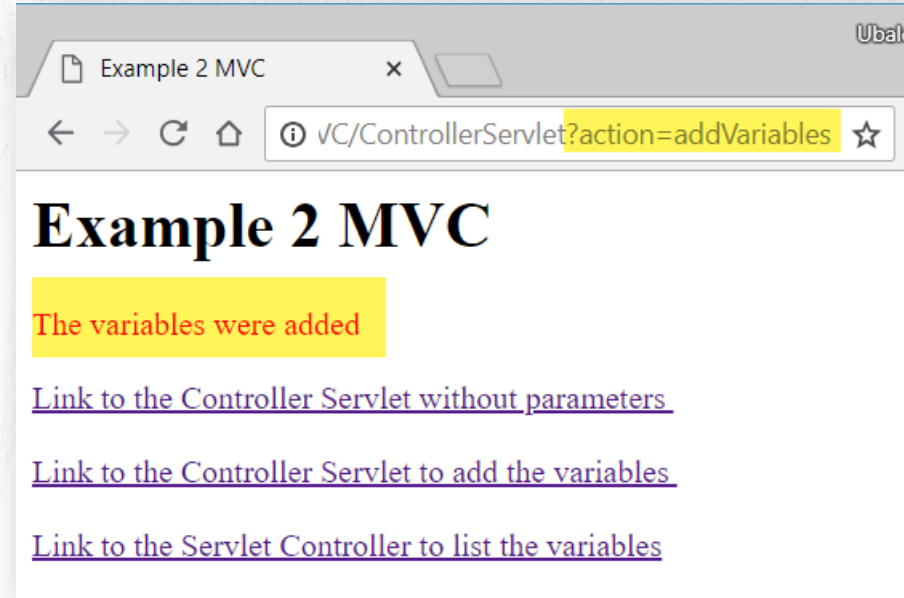
8. EXECUTE THE PROJECT

The result is as follows:



8. EXECUTE THE PROJECT

The result is as follows:



SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

8. EXECUTE THE PROJECT

The result is as follows:



SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

EXERCISE CONCLUSION

- With this exercise we have put in combination more subjects applying the scope of variables and the MVC design pattern.
- With this we conclude this topic.



SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

ONLINE COURSE

SERVLETS & JSP' S

By: Eng. Ubaldo Acosta



SERVLETS AND JSP COURSE

www.globalmentoring.com.mx