EXERCISE

SCOPES OF VARIABLES WITH MVC (1)

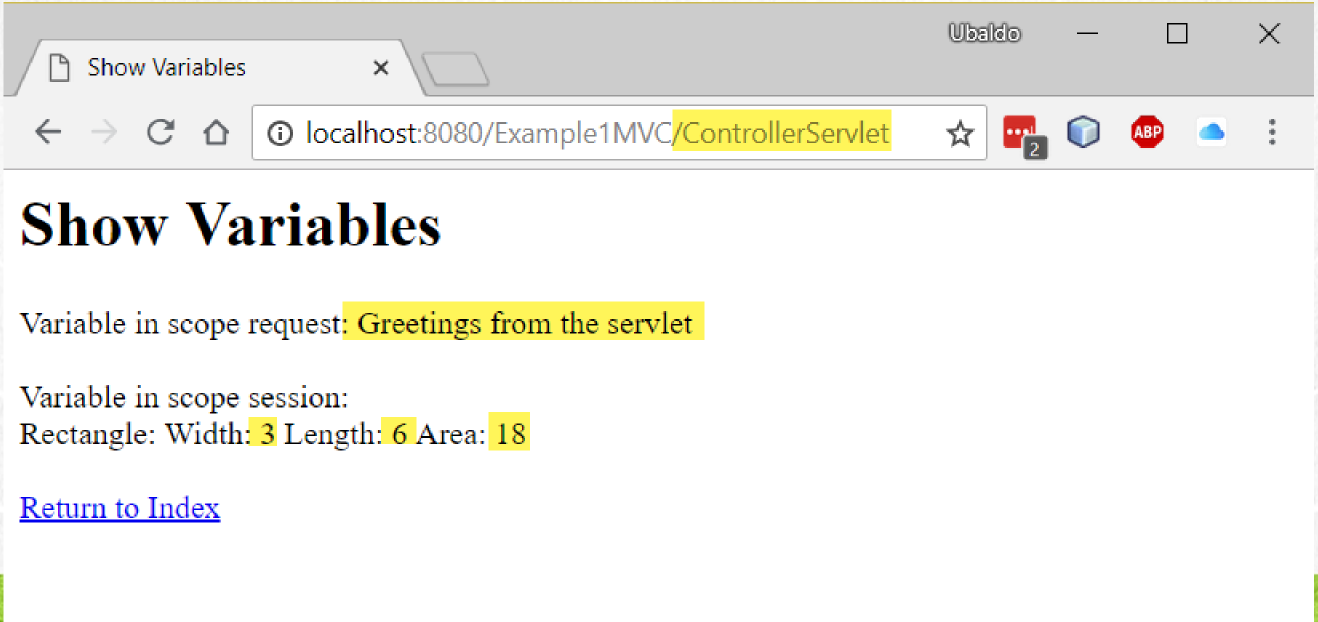
Global Mentoring

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx

EXERCISE OBJECTIVE

Create a program to manage the MVC design pattern. At the end we should observe the following:



Show Variables

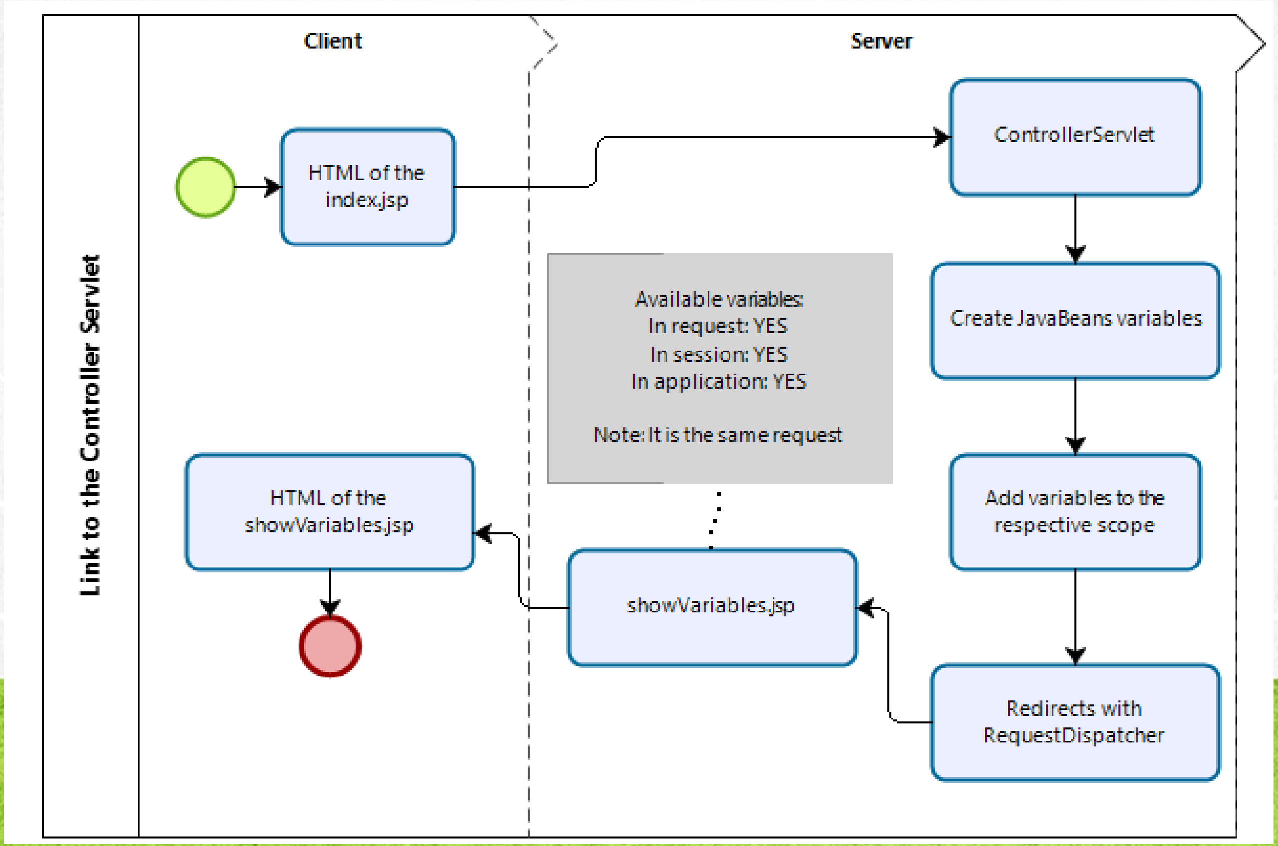
Variable in scope request: Greetings from the servlet

Variable in scope session:
Rectangle: Width: 3 Length: 6 Area: 18

Return to Index

DIAGRAM

Diagram that we will take as a basis to develop the exercise:



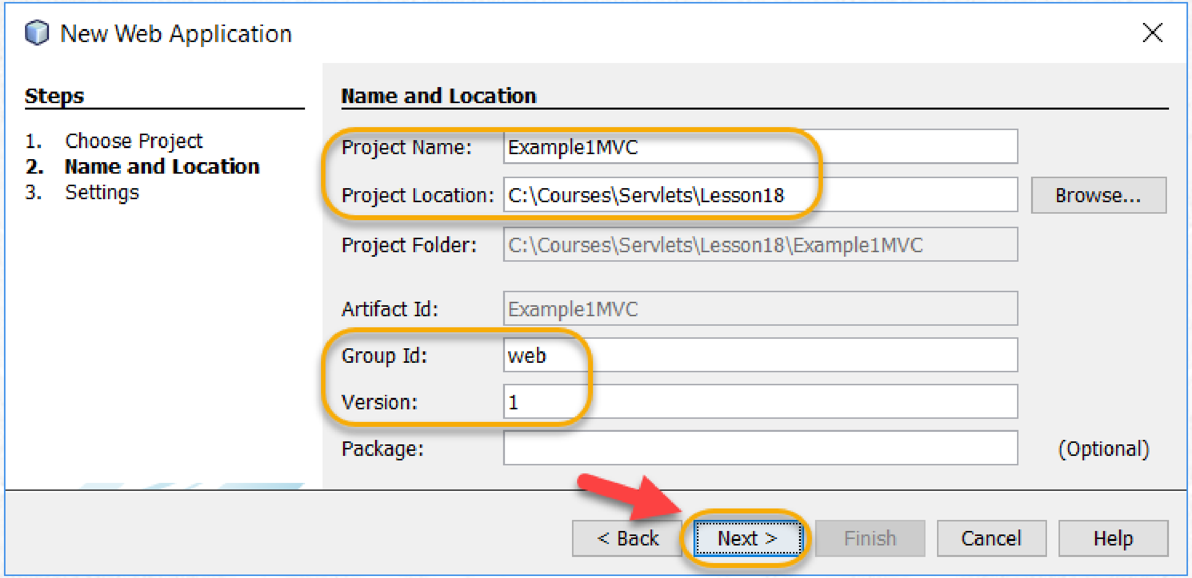
Link to the Controller Servlet
Client
Server
HTML of the index.jsp
ControllerServlet
Create JavaBeans variables
Add variables to the respective scope
Redirects with RequestDispatcher
showVariables.jsp
HTML of the showVariables.jsp

Available variables:
In request: YES
In session: YES
In application: YES

Note: It is the same request

1. CREATE A NEW PROJECT

Create a new project:



New Web Application

Steps

1. Choose Project
2. **Name and Location**
3. Settings

Name and Location

Project Name: Example1MVC

Project Location: C:\Courses\Servlets\Lesson18

Project Folder: C:\Courses\Servlets\Lesson18\Example1MVC

Artifact Id: Example1MVC

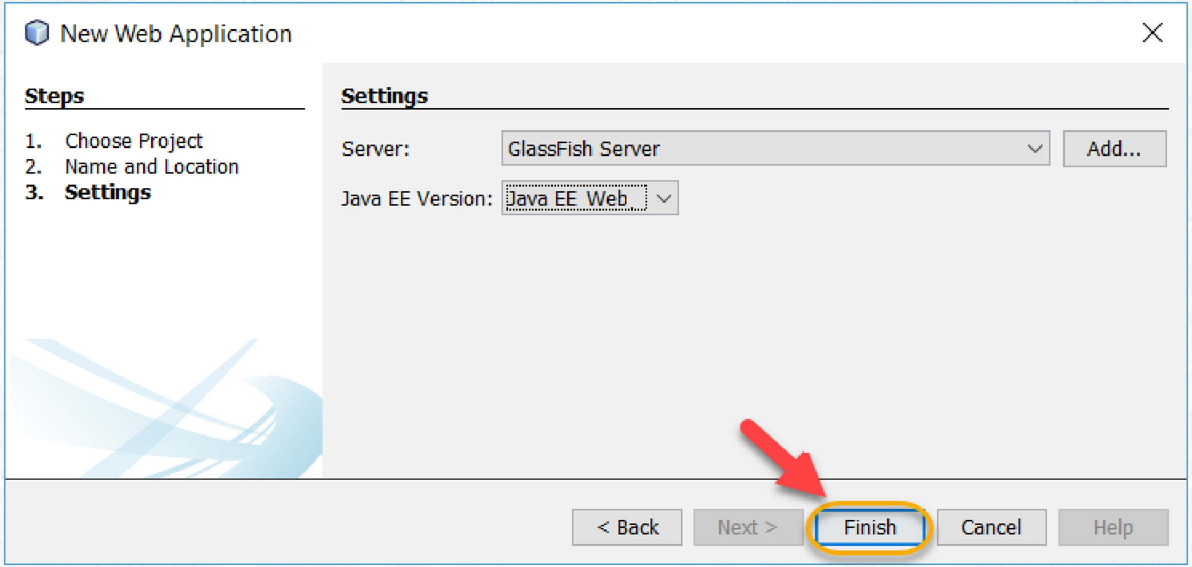
Group Id: web

Version: 1

Package:

1. CREATE A NEW PROJECT

Create a new project:



New Web Application

Steps

1. Choose Project
2. Name and Location
3. **Settings**

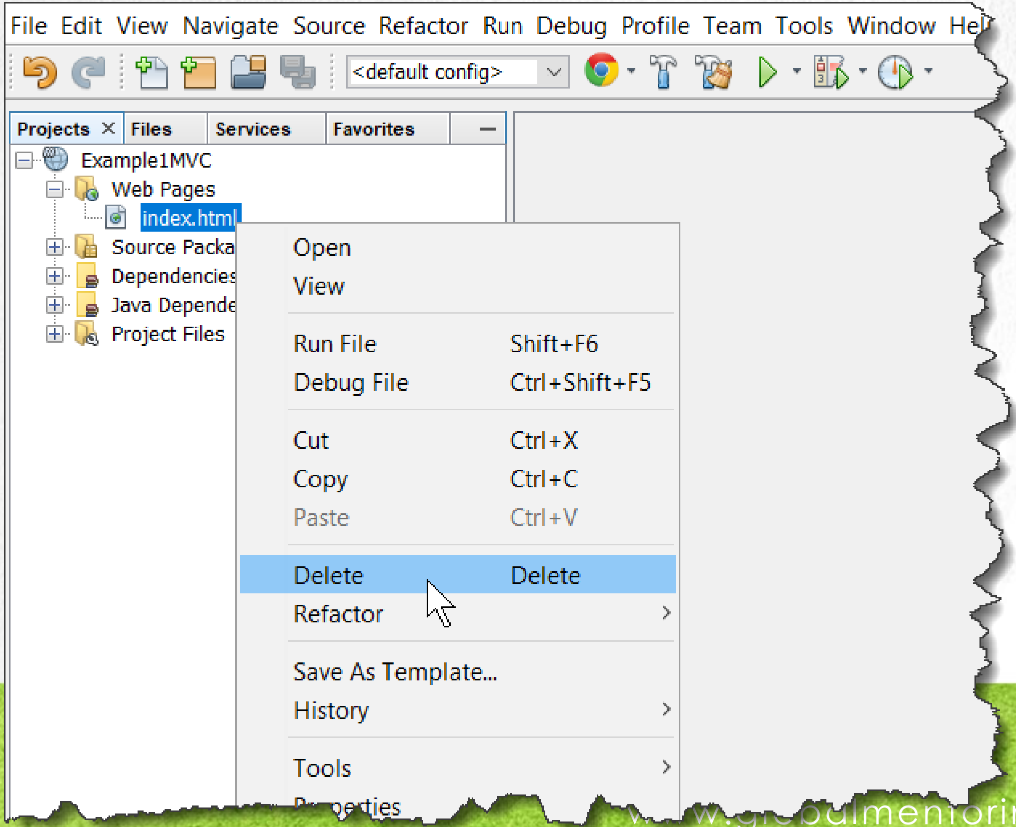
Settings

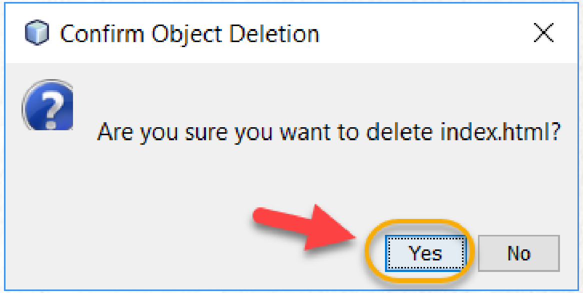
Server: GlassFish Server

Java EE Version: Java EE Web

2. DELETE THE INDEX.HTML FILE

Delete the index.html file:

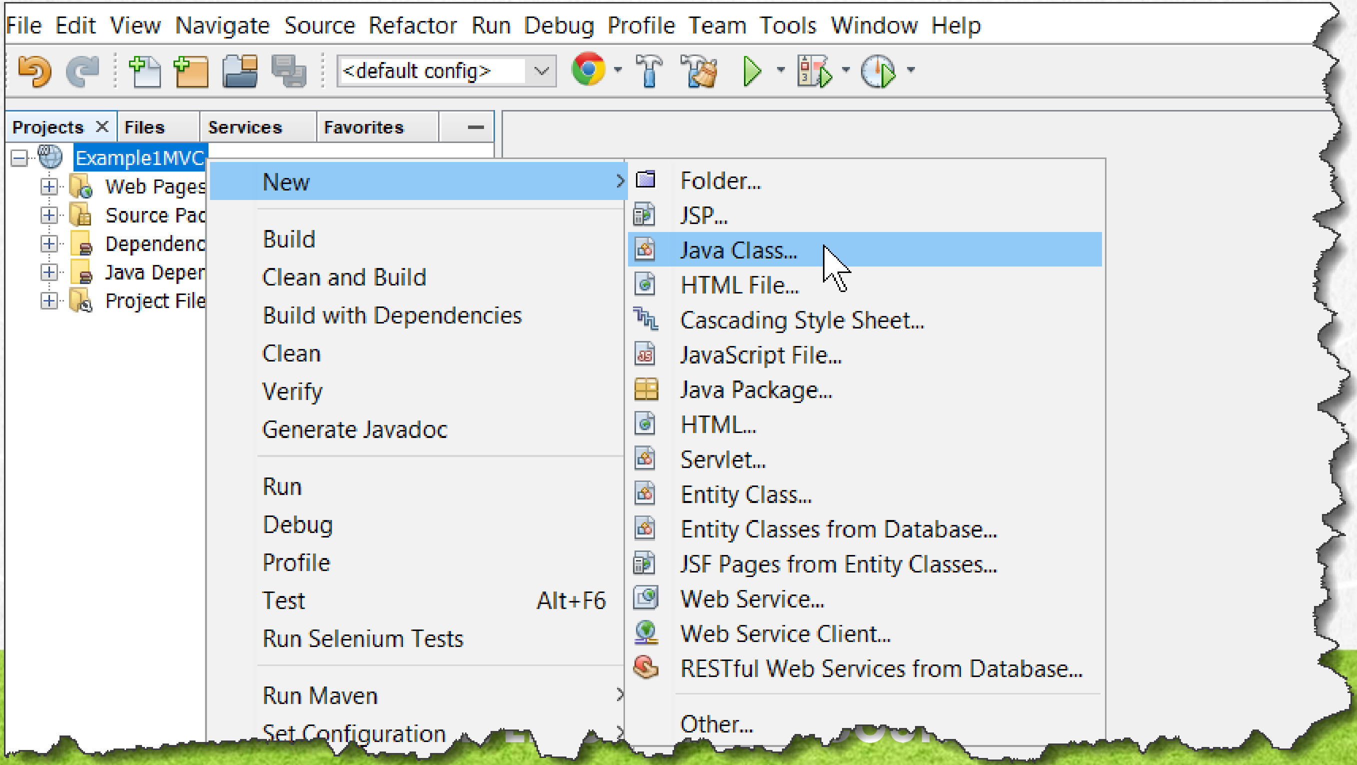




Are you sure you want to delete index.html?

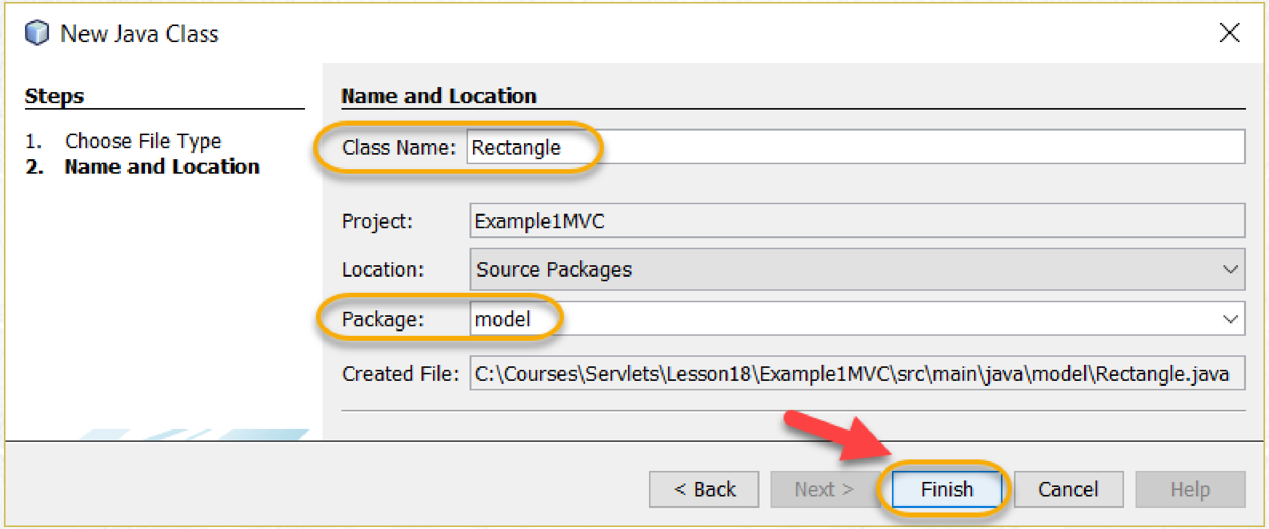
3. CREATE A NEW CLASS

Create the Rectangle.java class:



3. CREATE A NEW CLASS(CONT)

Create the Rectangle.java class:



New Java Class

Steps

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

Name and Location

Class Name: Rectangle

Project: Example1MVC

Location: Source Packages

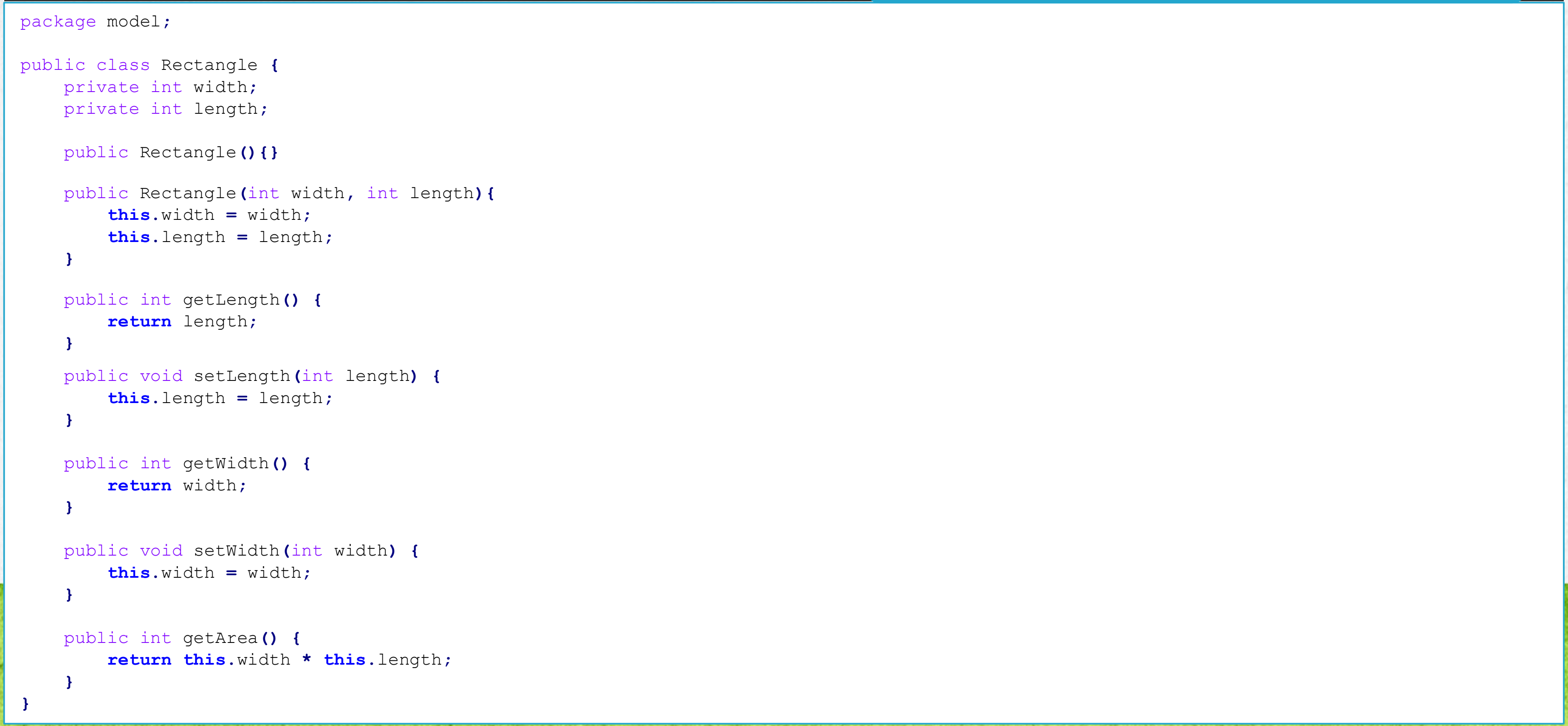
Package: model

Created File: C:\Courses\Servlets\Lesson18\Example1MVC\src\main\java\model\Rectangle.java

4. MODIFY THE FILE

Rectangle.java:

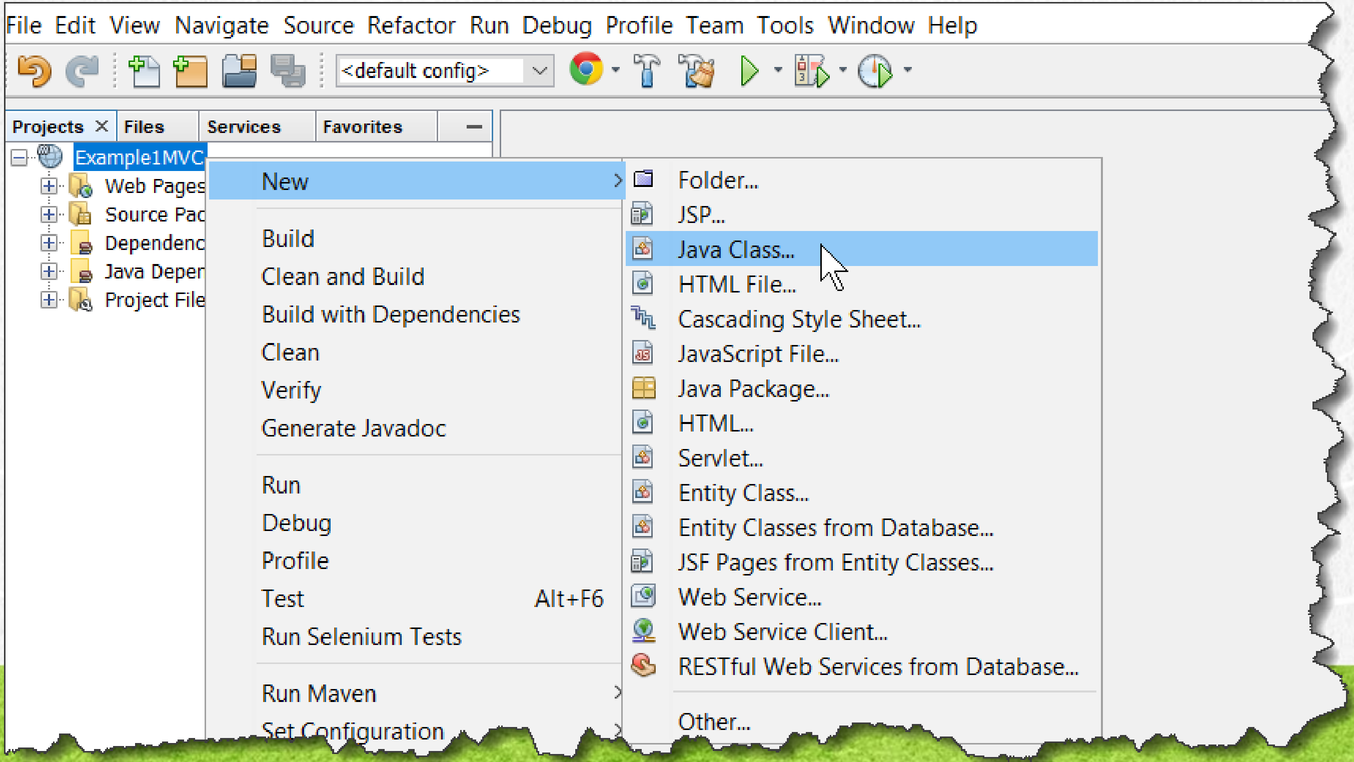
Click to download



```java
package model;

public class Rectangle {
    private int width;
    private int length;

    public Rectangle(){}

    public Rectangle(int width, int length){
        this.width = width;
        this.length = length;
    }

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

    public int getWidth() {
        return width;
    }

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

    public int getArea() {
        return this.width * this.length;
    }
}
```

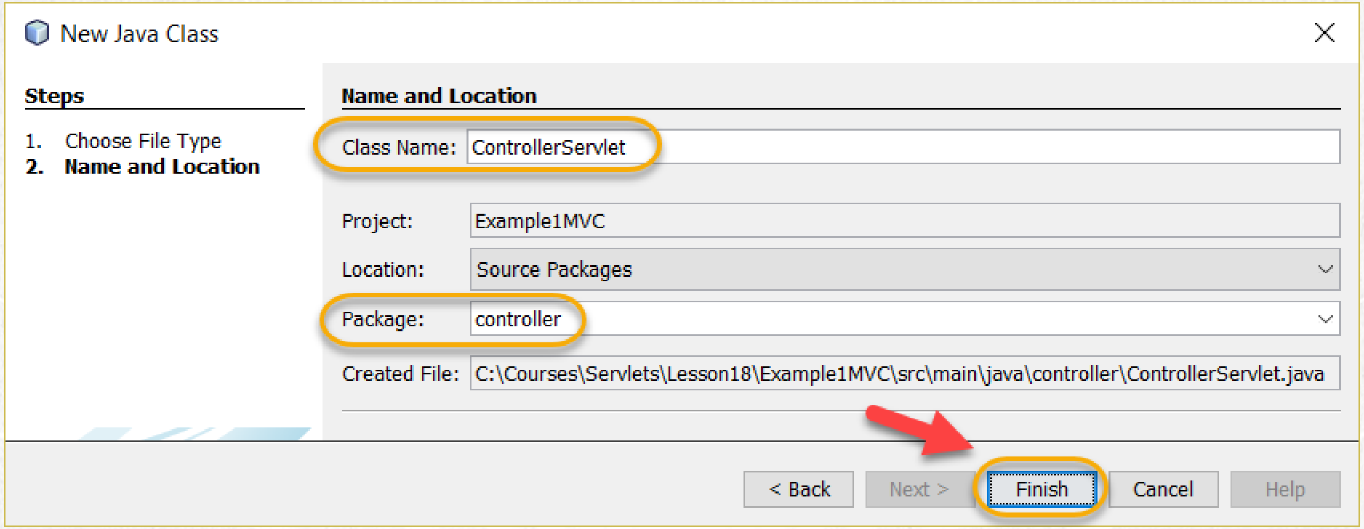
5. CREATE A NEW CLASS

Create a new class ControllerServlet.java:



5. CREATE A NEW CLASS(CONT)

Create a new class ControllerServlet.java:



New Java Class

Steps

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

Name and Location

Class Name: ControllerServlet

Project: Example1MVC

Location: Source Packages

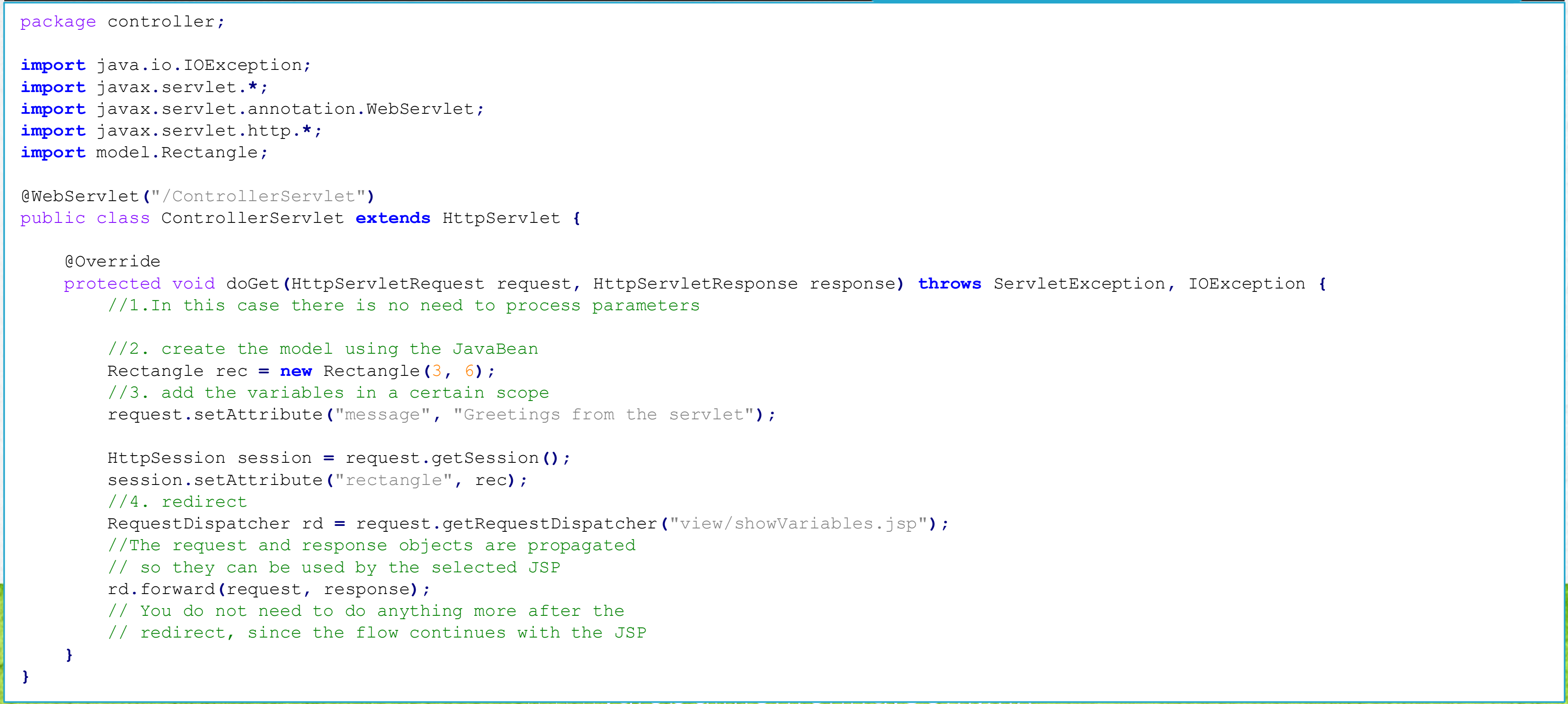
Package: controller

Created File: C:\Courses\Servlets\Lesson18\Example1MVC\src\main\java\controller\ControllerServlet.java

5. MODIFY THE CODE

ControllerServlet.java:

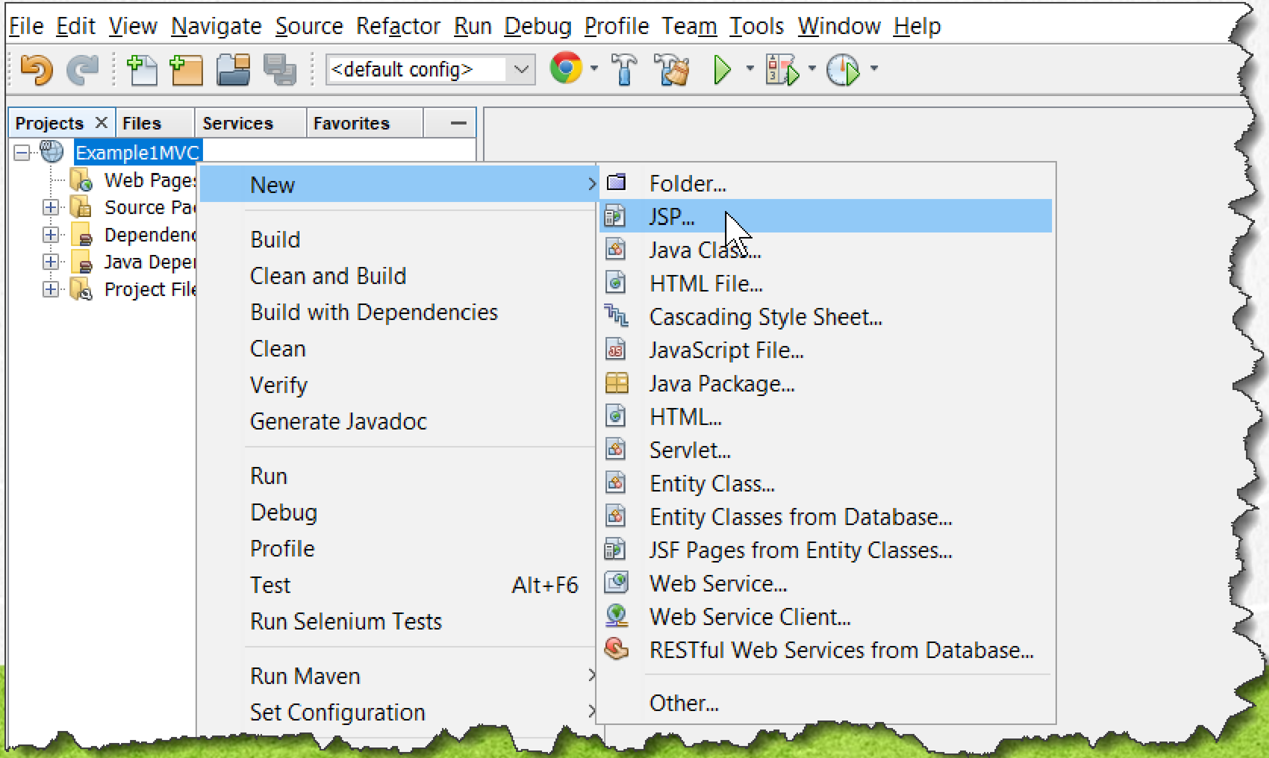
Click to download



```java
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 {
        //1.In this case there is no need to process parameters

        //2. create the model using the JavaBean
        Rectangle rec = new Rectangle(3, 6);
        //3. add the variables in a certain scope
        request.setAttribute("message", "Greetings from the servlet");

        HttpSession session = request.getSession();
        session.setAttribute("rectangle", rec);
        //4. redirect
        RequestDispatcher rd = request.getRequestDispatcher("view/showVariables.jsp");
        //The request and response objects are propagated
        // so they can be used by the selected JSP
        rd.forward(request, response);
        // You do not need to do anything more after the
        // redirect, since the flow continues with the JSP
    }
}
```

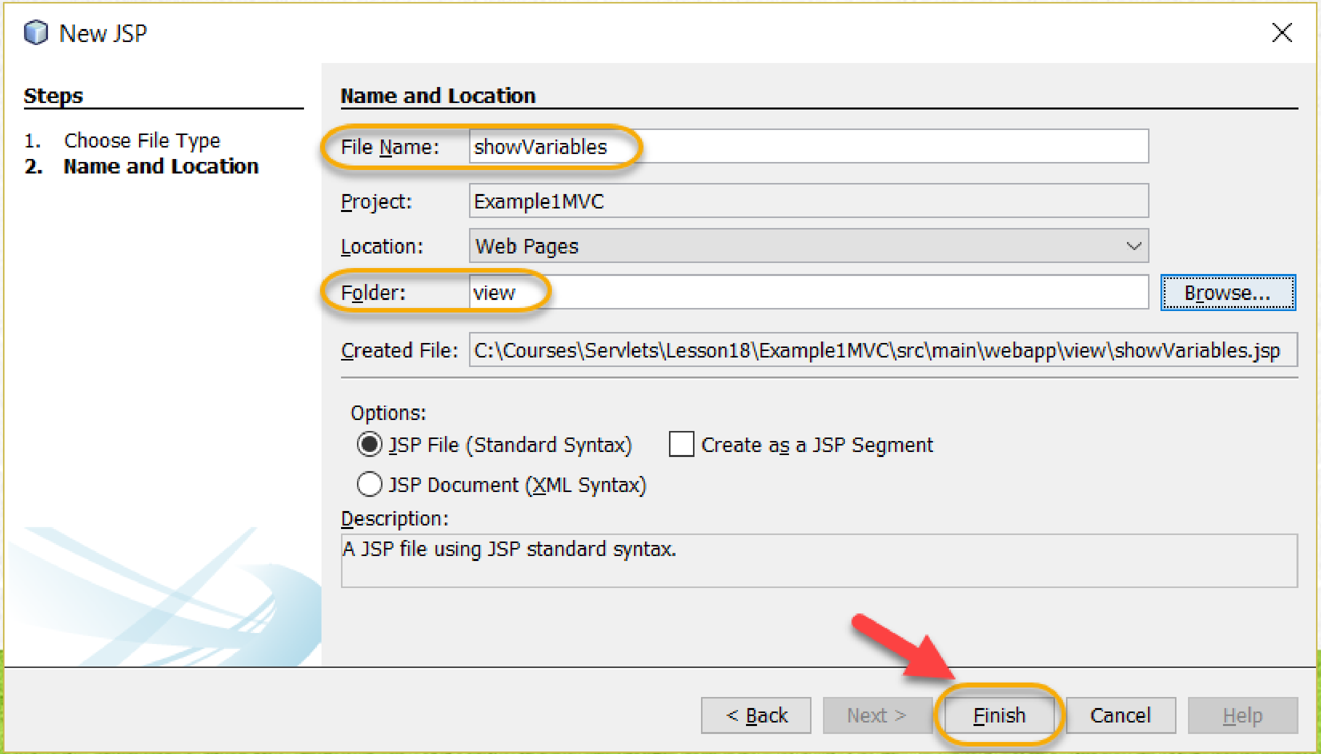
6. CREATE A NEW JSP

Create a new JSP:



6. 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: showVariables

Project: Example1MVC

Location: Web Pages

Folder: view

Created File: C:\Courses\Servlets\Lesson18\Example1MVC\src\main\webapp\view\showVariables.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

○ JSP Document (XML Syntax)

Description:

A JSP file using JSP standard syntax.

7. MODIFY THE FILE

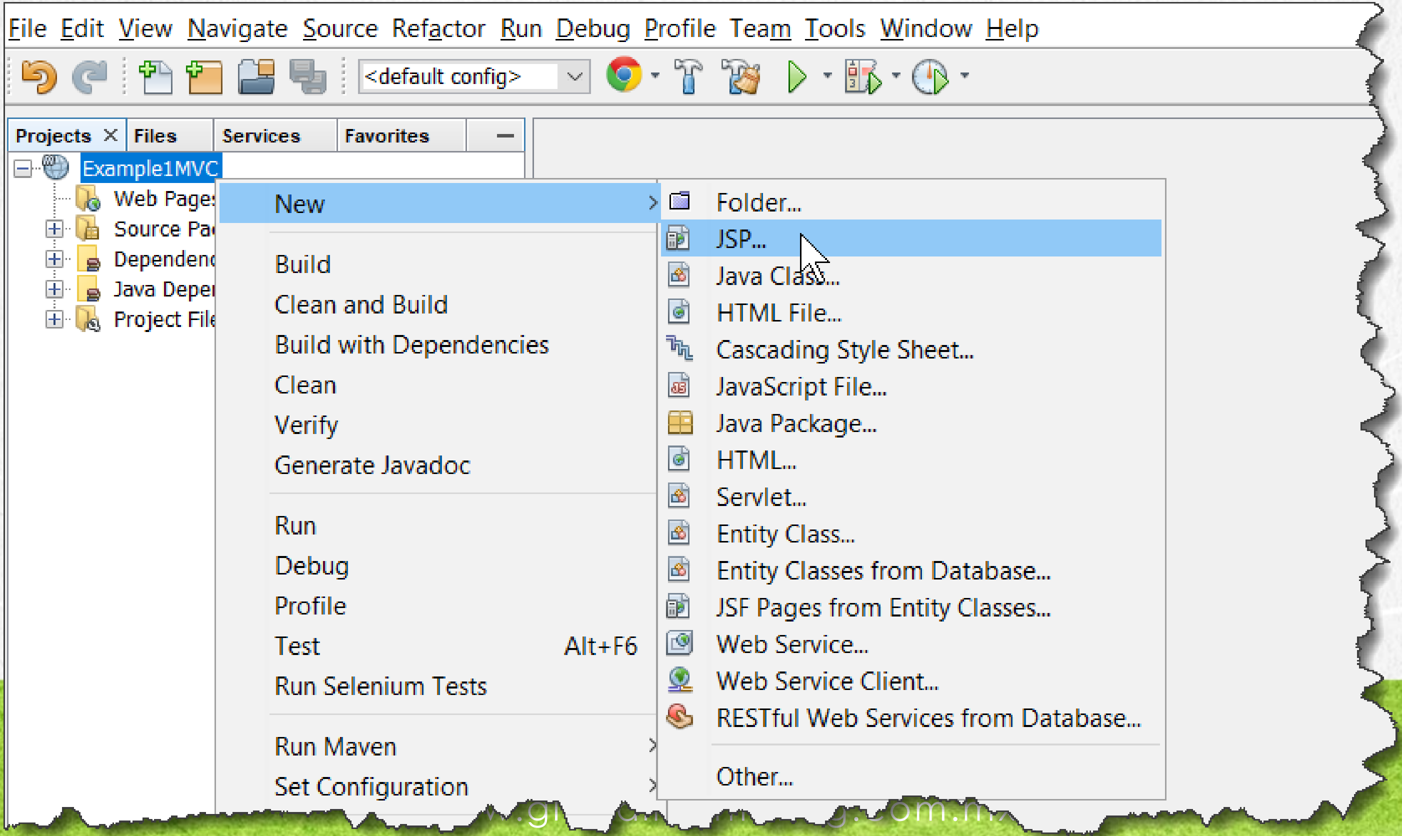
showVariables.jsp:

Click to download

```
<html>
    <head>
        <meta charset="UTF-8">
        <title>Show Variables</title>
    </head>
    <body>
        <h1>Show Variables</h1>

        Variable in scope request:
        ${message}

        <br>
        <br>

        Variable in scope session:
        <br>
        Rectangle:
        Width: ${rectangle.width}
        Length: ${rectangle.length}
        Area: ${rectangle.area}

        <br>
        <br>
        <a href="${pageContext.request.contextPath}/index.jsp">
            Return to Index
        </a>
    </body>
</html>
```

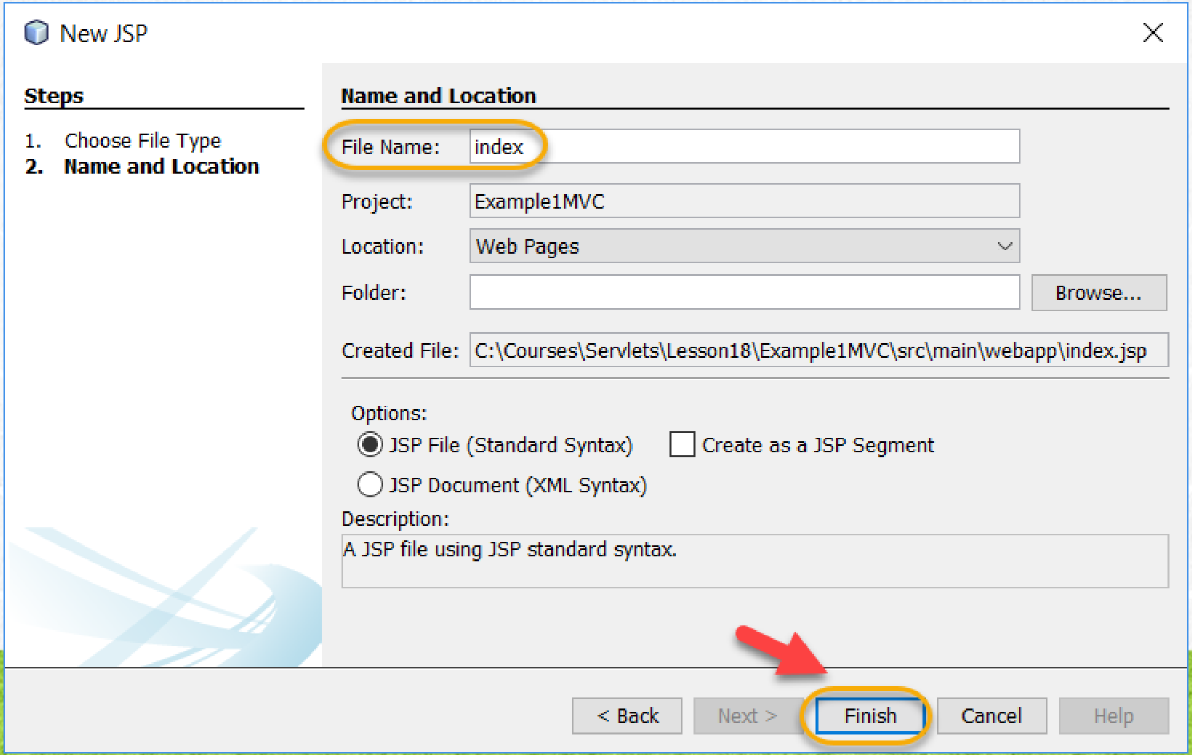
8. CREATE A NEW JSP

Create a new JSP:



8. 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: index

Project: Example1MVC

Location: Web Pages

Folder:

Created File: C:\Courses\Servlets\Lesson18\Example1MVC\src\main\webapp\index.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

○ JSP Document (XML Syntax)

Description:

A JSP file using JSP standard syntax.

9. MODIFY THE CODE

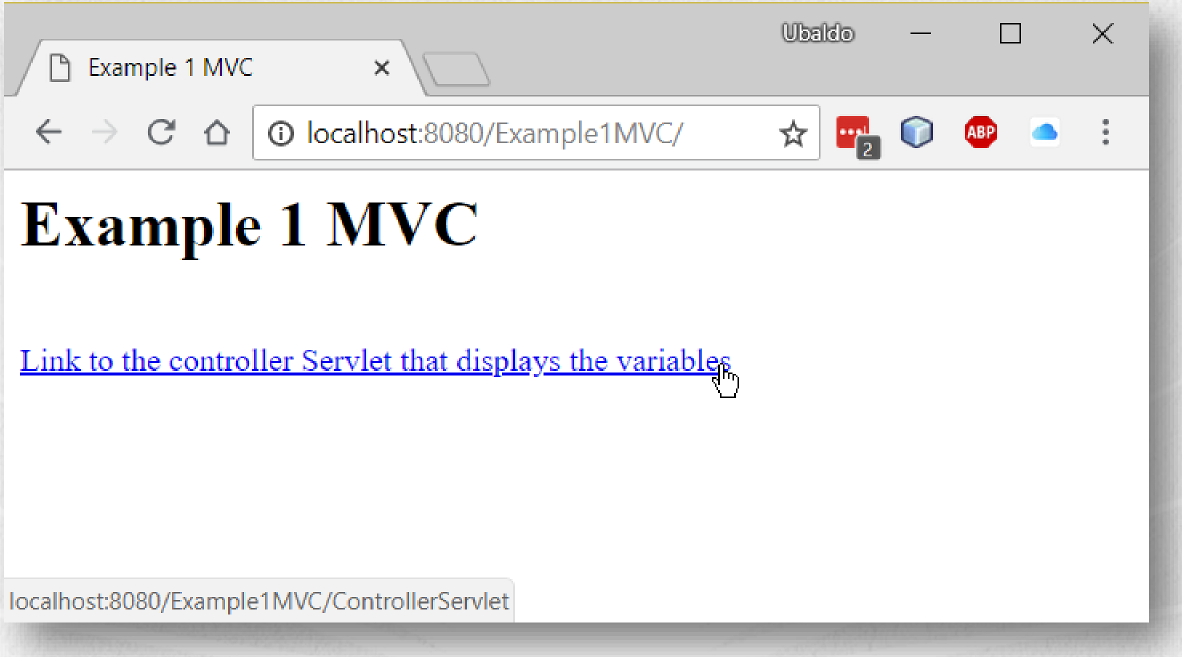
index.jsp:

Click to download

```
<html>
    <head>
        <meta charset="UTF-8">
        <title>Example 1 MVC</title>
    </head>
    <body>
        <h1>Example 1 MVC</h1>
        <br>
        <a href="${pageContext.request.contextPath}/ControllerServlet">
           Link to the controller Servlet that displays the variables
        </a>
    </body>
</html>
```

10. EXECUTE THE PROJECT

The result is as follows:



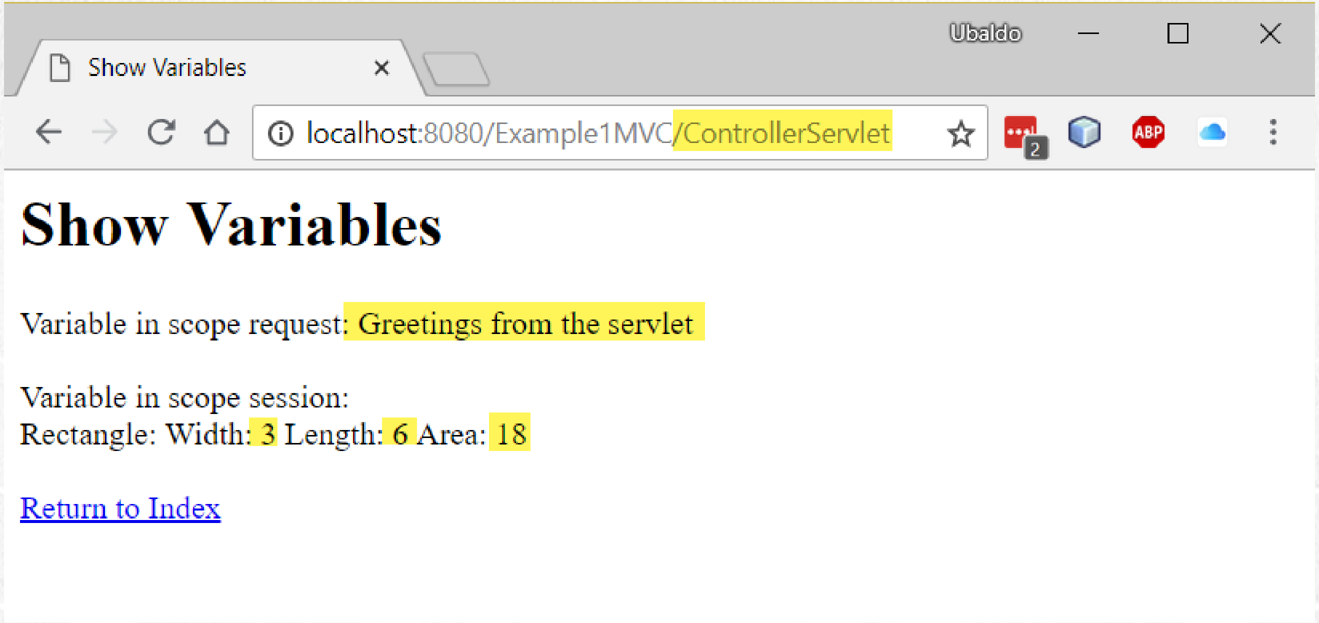
Example 1 MVC

Link to the controller Servlet that displays the variables

localhost:8080/Example1MVC/ControllerServlet

10. EXECUTE THE PROJECT (CONT)

The result is as follows:



Show Variables

Variable in scope request: Greetings from the servlet

Variable in scope session:
Rectangle: Width: 3 Length: 6 Area: 18

Return to Index

EXERCISE CONCLUSION

- With this exercise we have put into practice the basics of the MVC design pattern.

- We saw how to create each of the classes and components that are part of the pattern, both the controller (Servlet), the model (JavaBean) and the view (JSP). Then we will create another exercise to further expand this concept.

ONLINE COURSE

SERVLETS & JSP' S

By: Eng. Ubaldo Acosta

Global Mentoring

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx