

CURSO DE SERVLETS Y JSPS

EJERCICIO

ALCANCES DE VARIABLES CON MVC 1



Experiencia y Conocimiento para tu vida

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

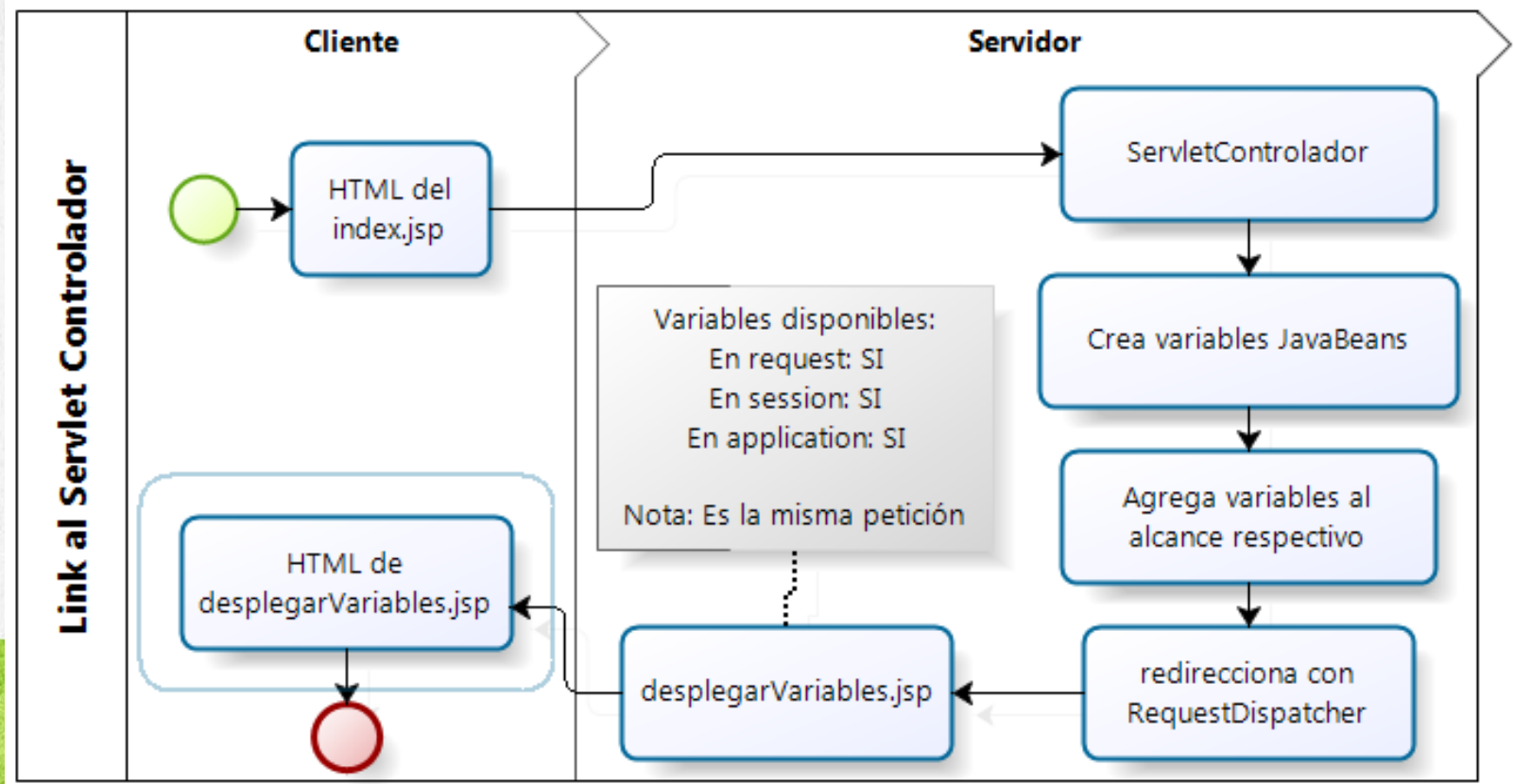
OBJETIVO DEL EJERCICIO

Crear un programa para el manejo básico de elementos de un JSP. Al finalizar deberemos observar lo siguiente:



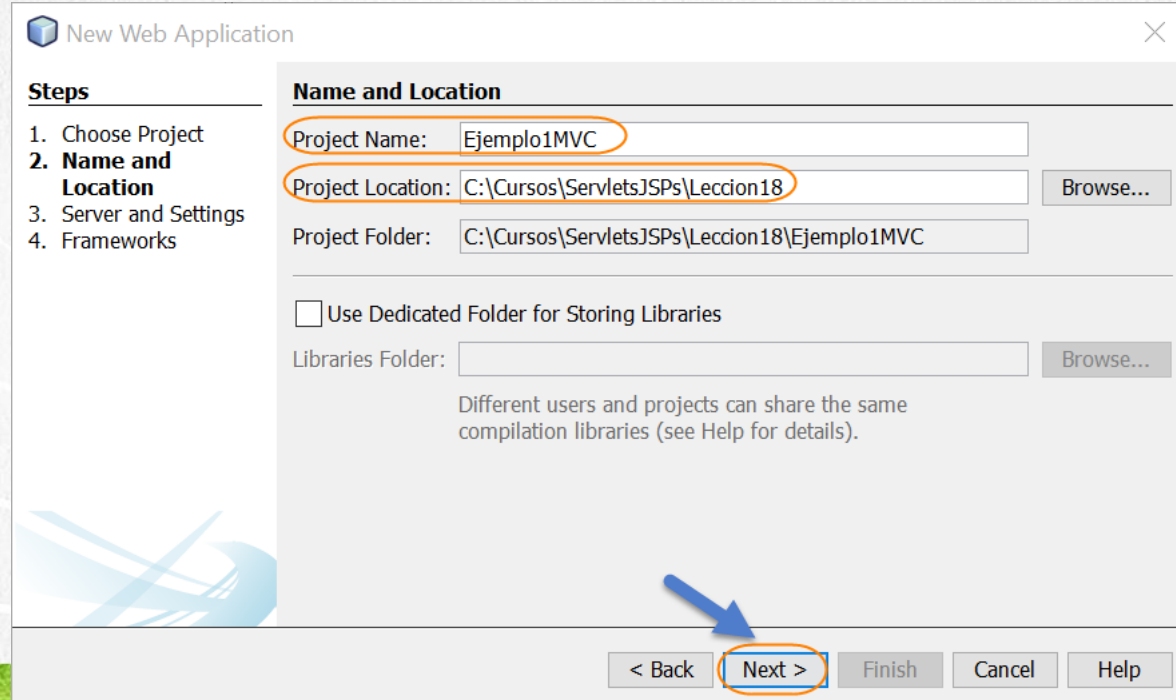
DIAGRAMA

Diagrama que tomaremos como base para desarrollar el ejercicio:



PASO 1. CREACIÓN DEL PROYECTO

Vamos a crear el proyecto:



New Web Application

Steps

1. Choose Project
2. **Name and Location**
3. Server and Settings
4. Frameworks

Name and Location

Project Name: Ejemplo1MVC

Project Location: C:\Cursos\ServletsJSPs\Leccion18 Browse...

Project Folder: C:\Cursos\ServletsJSPs\Leccion18\Ejemplo1MVC

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

Different users and projects can share the same compilation libraries (see Help for details).

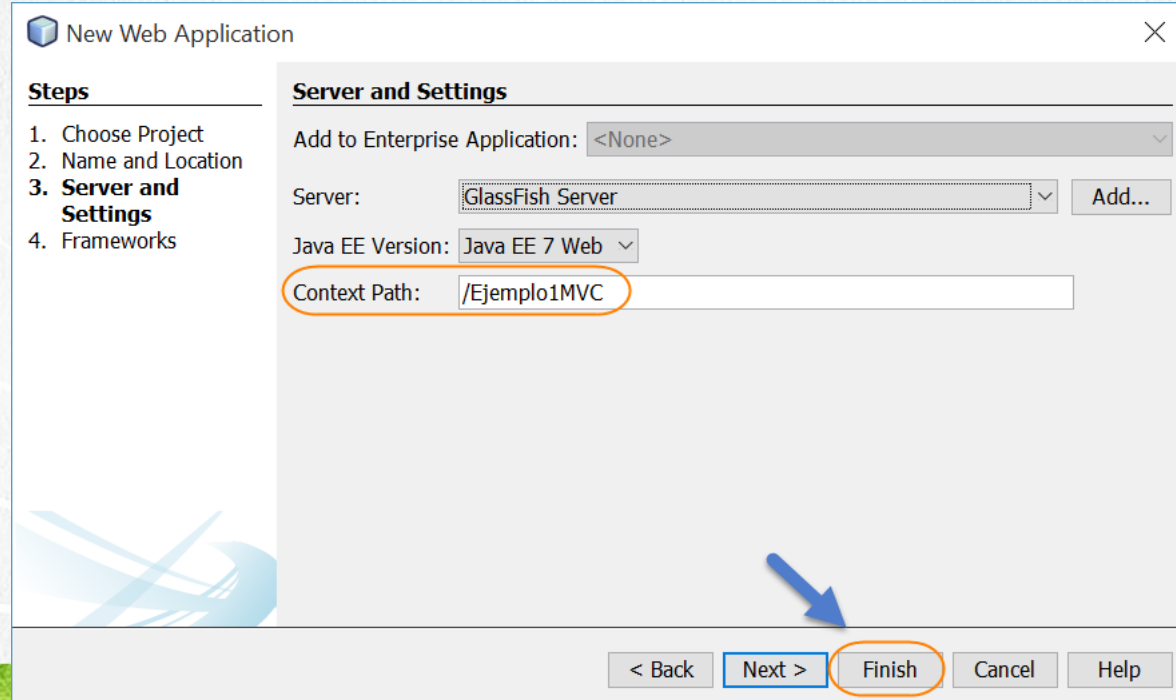
< Back Next > Finish Cancel Help

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

PASO 1. CREACIÓN DEL PROYECTO

Vamos a crear el proyecto:



The screenshot shows the 'New Web Application' wizard in the Eclipse IDE, specifically the 'Server and Settings' step. The wizard is titled 'New Web Application' and has a close button (X) in the top right corner. On the left, there is a 'Steps' panel with four steps: 1. Choose Project, 2. Name and Location, 3. **Server and Settings** (the current step), and 4. Frameworks. The main area of the wizard is titled 'Server and Settings' and contains the following fields and buttons:

- 'Add to Enterprise Application:' with a dropdown menu showing '<None>'.
- 'Server:' with a dropdown menu showing 'GlassFish Server' and an 'Add...' button.
- 'Java EE Version:' with a dropdown menu showing 'Java EE 7 Web'.
- 'Context Path:' with a text field containing '/Ejemplo1MVC'.

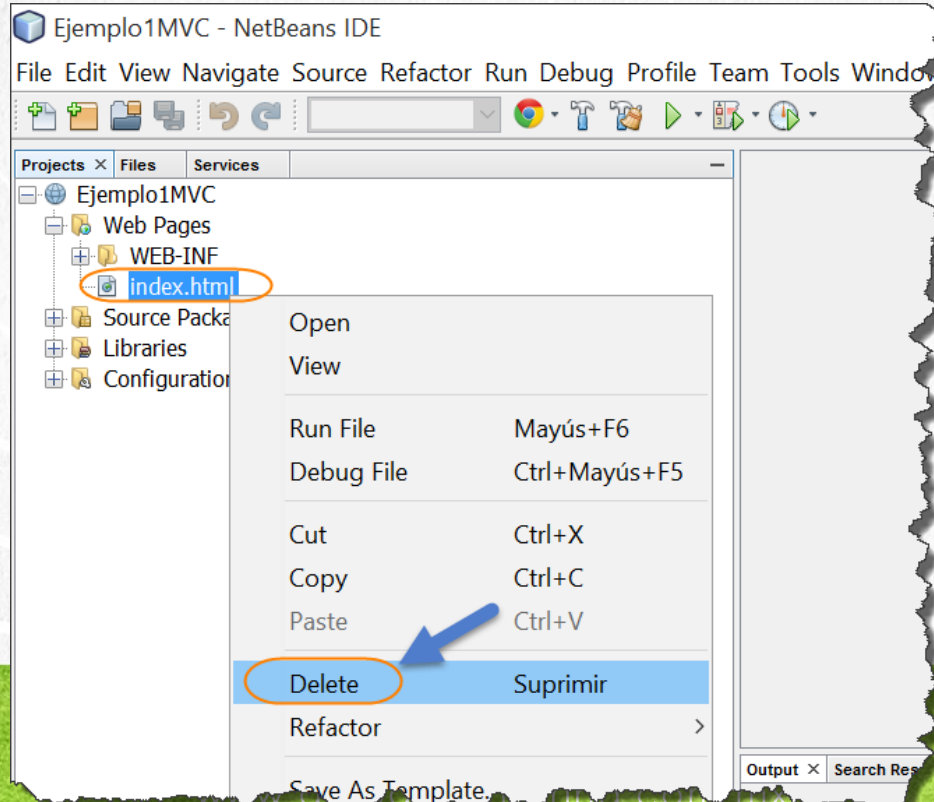
At the bottom of the wizard, there are five buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'. The 'Finish' button is highlighted with an orange circle, and a blue arrow points to it from the right.

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

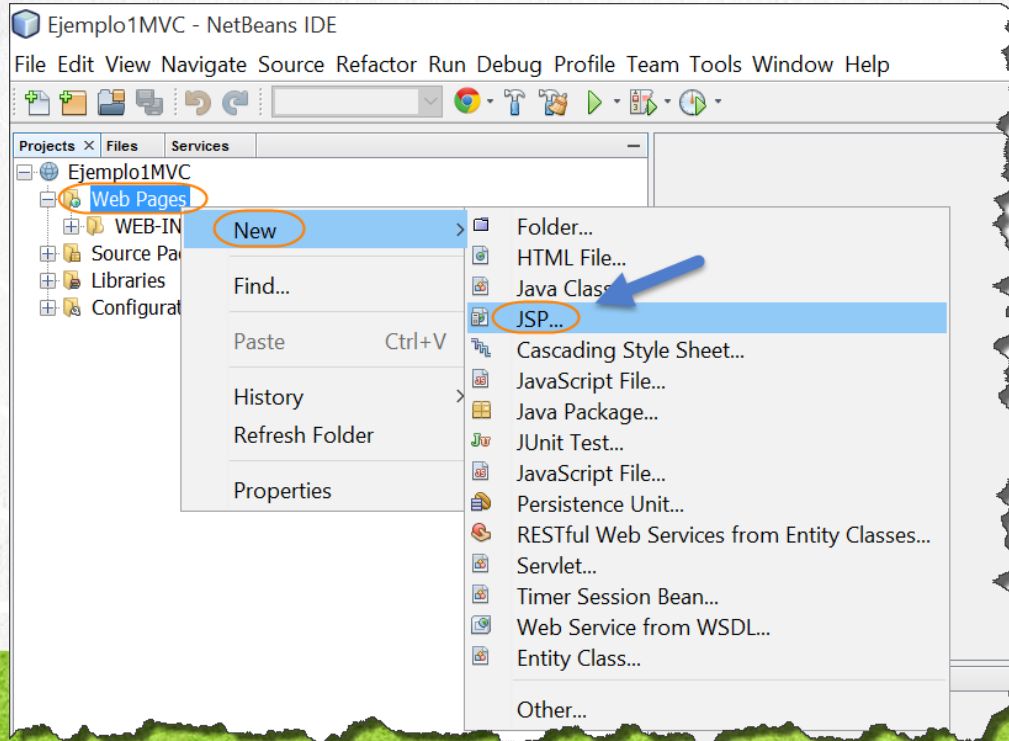
PASO 2. BORRAMOS UN ARCHIVO

Vamos a borrar el archivo index.html:



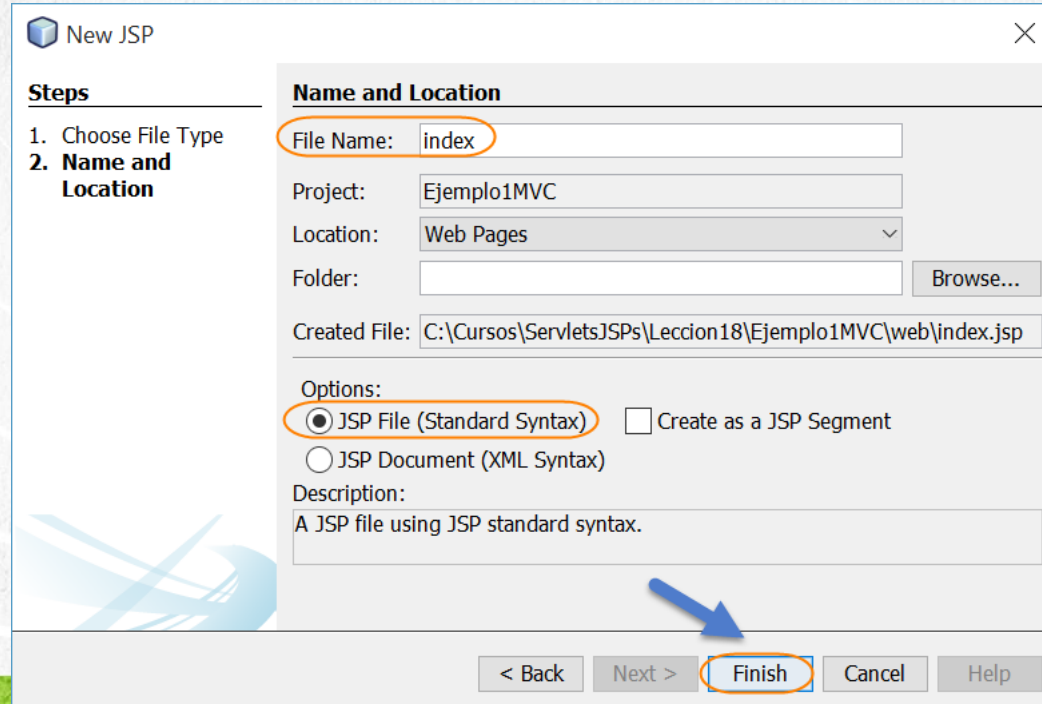
PASO 3. CREACIÓN DE UN JSP

Vamos a crear un nuevo JSP:



PASO 3. CREACIÓN DE UN JSP (CONT)

Vamos a crear un nuevo 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:

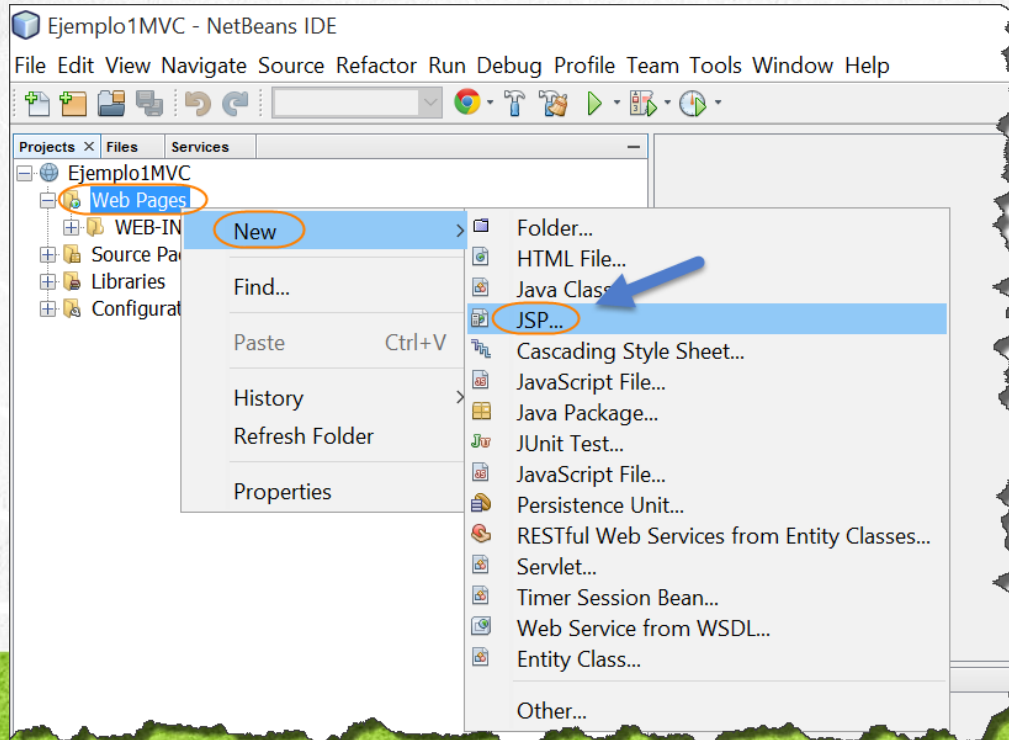
A JSP file using JSP standard syntax.

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

PASO 4. CREACIÓN DE UN JSP

Vamos a crear un nuevo JSP:



PASO 4. CREACIÓN DE UN JSP (CONT)

Vamos a crear un nuevo JSP:

New JSP

Steps

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

Name and Location

File Name: desplegarVariables

Project: Ejemplo1MVC

Location: Web Pages

Folder: vista

Browse...

Created File: tsJSPs\Leccion18\Ejemplo1MVC\web\vista\desplegarVariables.jsp

Options:

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

☐ JSP Document (XML Syntax)

Description:

A JSP file using JSP standard syntax.

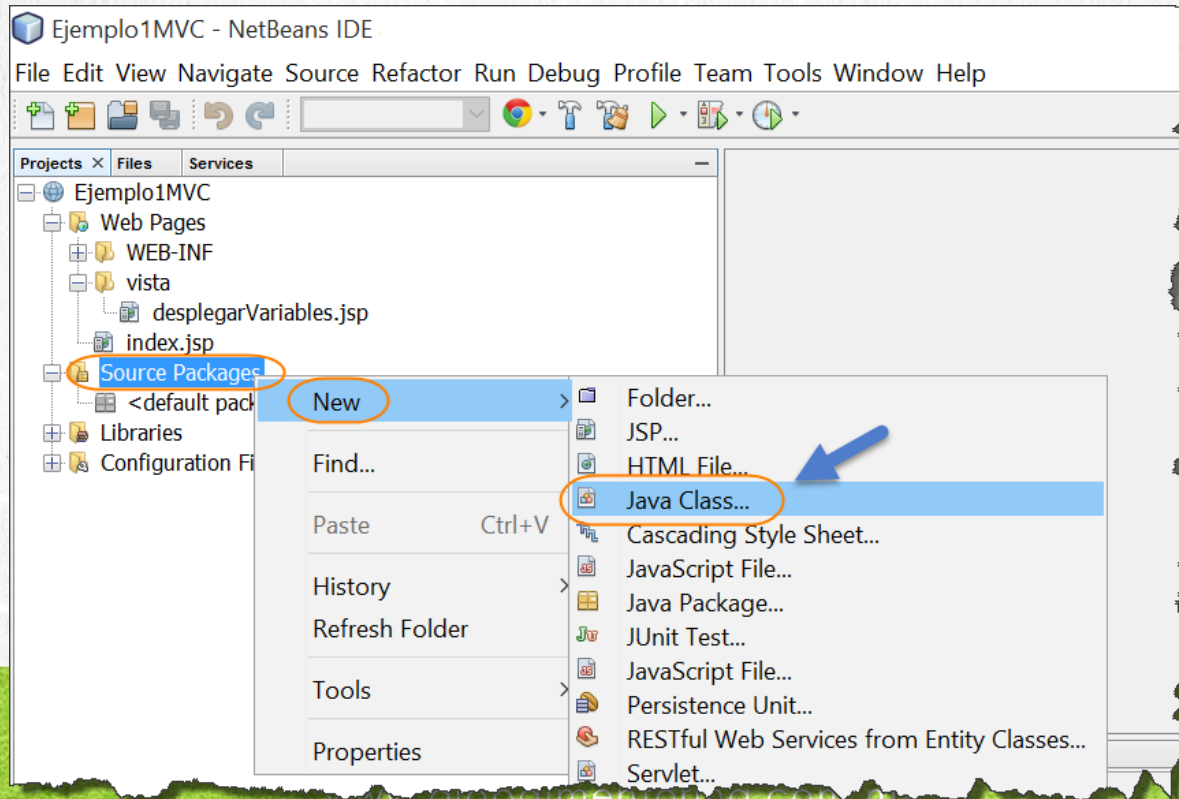
< Back Next > **Finish** Cancel Help

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

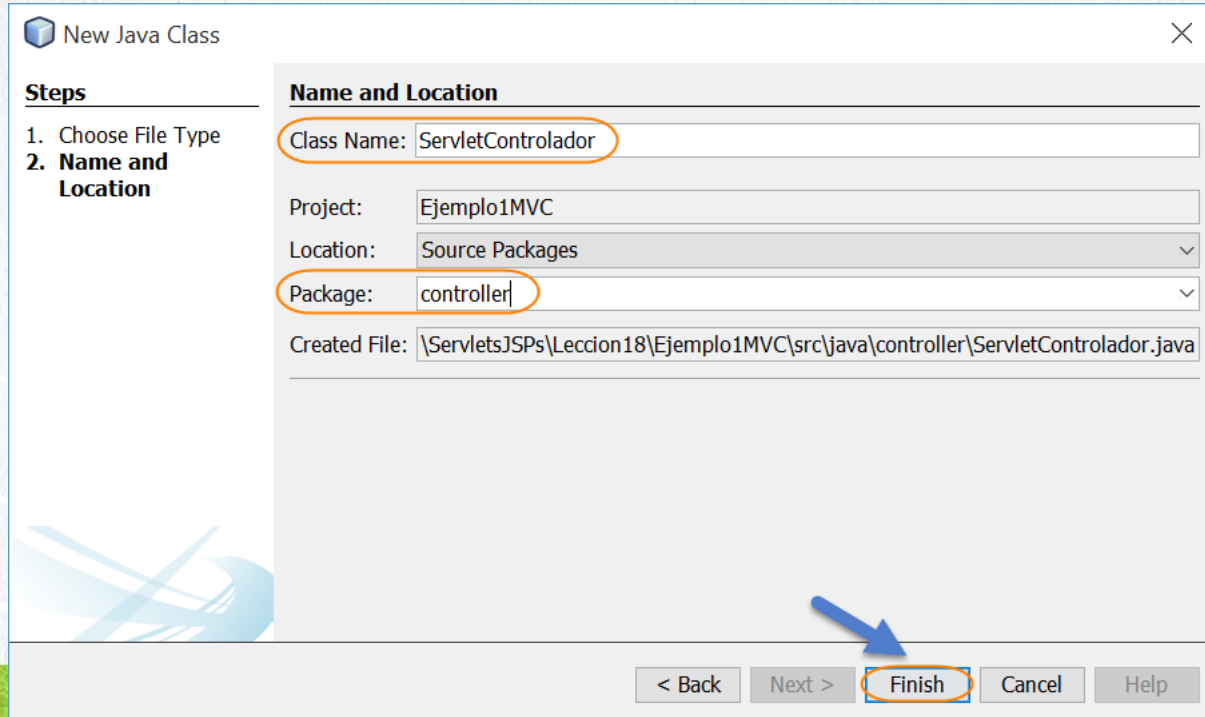
PASO 5. CREACIÓN DE UNA CLASE

Vamos a crear la clase ServletControlador.java:



PASO 5. CREACIÓN DE UNA CLASE (CONT)

Vamos a crear la clase ServletControlador.java:



New Java Class

Steps

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

Name and Location

Class Name:

Project:

Location:

Package:

Created File:

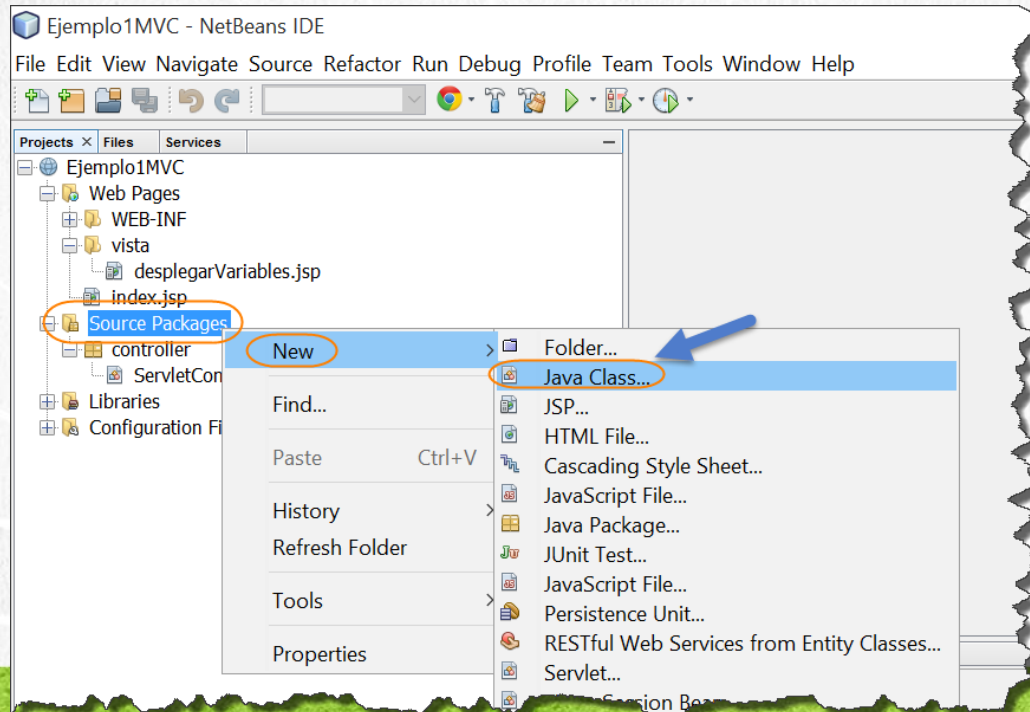
< Back Next > **Finish** Cancel Help

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

PASO 6. CREACIÓN DE UNA CLASE

Vamos a crear la clase ServletControlador.java:

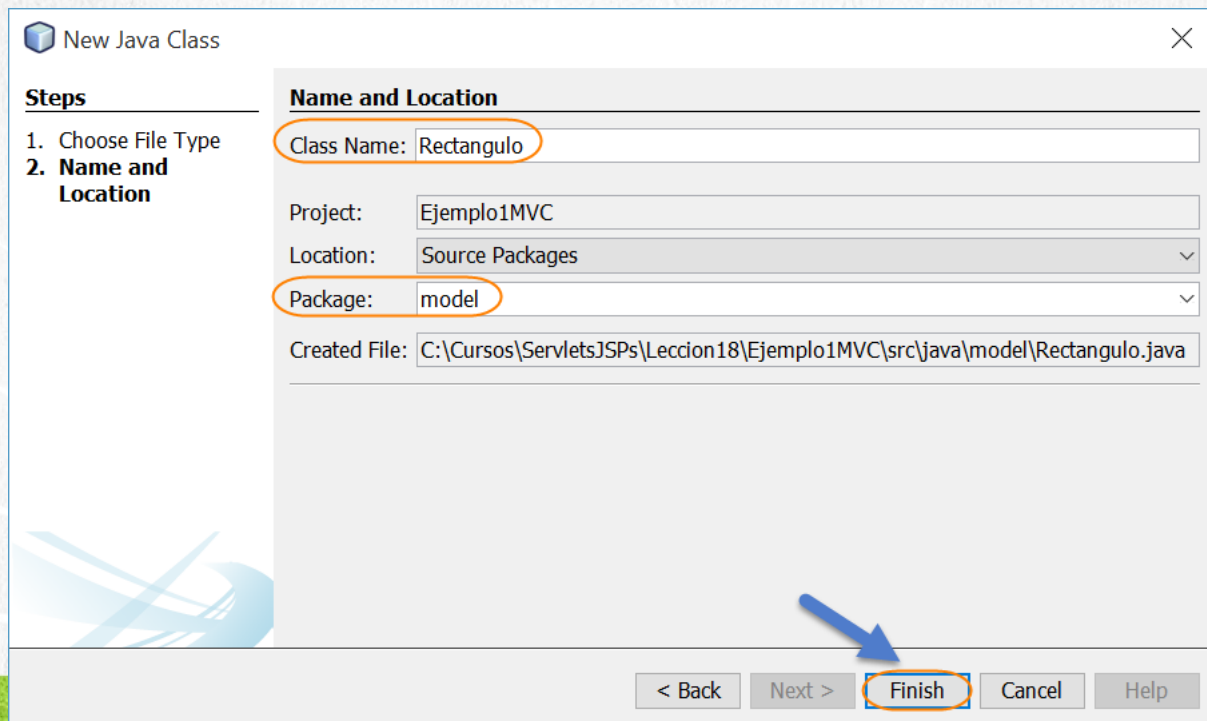


CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

PASO 6. CREACIÓN DE UNA CLASE (CONT)

Vamos a crear la clase ServletControlador.java:



New Java Class

Steps

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

Name and Location

Class Name: Rectangulo

Project: Ejemplo1MVC

Location: Source Packages

Package: model

Created File: C:\Cursos\ServletsJSPs\Leccion18\Ejemplo1MVC\src\java\model\Rectangulo.java

< Back Next > **Finish** Cancel Help

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

PASO 7. MODIFICAMOS EL CÓDIGO

Archivo index.jsp:

Dar click para ir al código

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>Ejemplo MVC</title>
  </head>
  <body>
    <h1>Ejemplo MVC</h1>
    <br>
    <a href="${pageContext.request.contextPath}/ServletControlador">
      Link al servlet controlador que despliega las variables
    </a>
  </body>
</html>
```

PASO 8. MODIFICAMOS EL CÓDIGO

Archivo desplegarVariables.jsp:

Dar click para ir al código

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

  Variable en alcance request:
  ${mensaje}

  <br>
  <br>

  Variable en alcance session:
  <br>
  Rectangulo:
  Base: ${rectangulo.base}
  Altura: ${rectangulo.altura}
  Area: ${rectangulo.area}

  <br>
  <br>
  <a href="${pageContext.request.contextPath}/index.jsp">
    Regresar al Inicio
  </a>
</body>
</html>
```

PASO 9. MODIFICAMOS EL CÓDIGO

Archivo ServletControlador.java:

Dar click para ir al código

```
package controller;

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

@WebServlet(name = "ServletControlador", urlPatterns = {"/ServletControlador"})
public class ServletControlador extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        //1. En este caso no hay necesidad de procesar parametros

        //2. Creamos los JavaBeans
        Rectangulo rec = new Rectangulo(3, 6);
        //3. Agregamos las variables en cierto alcance
        request.setAttribute("mensaje", "Saludos desde el servlet");

        HttpSession session = request.getSession();
        session.setAttribute("rectangulo", rec);
        //4. Redireccionamos
        RequestDispatcher rd = request.getRequestDispatcher("vista/desplegarVariables.jsp");
        //Se propagan los objetos request y response
        //para que puedan ser utilizados por el JSP seleccionado
        rd.forward(request, response);
        //Ya no se necesita hacer nada mas despues del
        //redireccionamiento, ya que el flujo continua
        //con el JSP
    }
}
```


PASO 10. MODIFICAMOS EL CÓDIGO

Archivo Rectangulo.java:

Dar click para ir al código

```
package model;

public class Rectangulo {
    private int base;
    private int altura;

    public Rectangulo(){}
    public Rectangulo(int base, int altura){
        this.base = base;
        this.altura = altura;
    }

    public int getAltura() {
        return altura;
    }

    public void setAltura(int altura) {
        this.altura = altura;
    }

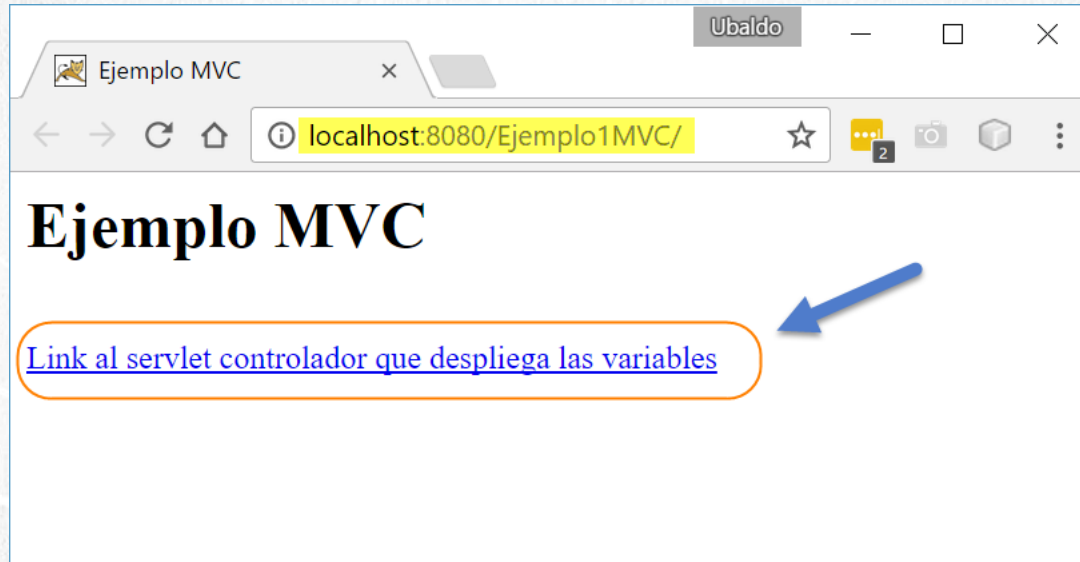
    public int getBase() {
        return base;
    }

    public void setBase(int base) {
        this.base = base;
    }

    public int getArea() {
        return this.base * this.altura;
    }
}
```

PASO 11. EJECUTAMOS EL PROYECTO

El resultado es como sigue:



PASO 11. EJECUTAMOS EL PROYECTO (CONT)

El resultado es como sigue:



CONCLUSIÓN DEL EJERCICIO

- Con este ejercicio hemos puesto en práctica lo básico del patrón de diseño MVC.
- Vimos como crear cada una de las clases y componentes que forman parte del patrón, tanto el controlador (Servlet), el modelo (JavaBean) y la vista (JSP). A continuación crearemos otro ejercicio para ampliar más este concepto.



Experiencia y Conocimiento para tu vida

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx

CURSO ONLINE

SERVLETS & JSP' S

Por: Ing. Ubaldo Acosta



Experiencia y Conocimiento para tu vida

CURSO DE SERVLETS Y JSPS

www.globalmentoring.com.mx