

Creando la vista

Creando la vista	1
¿Qué aprenderás?	2
Introducción	2
La vista	2
Configurando la vista con MVC	7



¡Comencemos!

¿Qué aprenderás?

- Implementar y crear la o las vistas de un proyecto.
- Configurar las vistas jsp mediante el archivo application.properties.

Introducción

Como ya tenemos configurado nuestro Front Controller, necesitamos configurar y crear nuestras vistas del proyecto, las cuales corresponden a lo que los usuarios verán en su navegador o browser. En este capítulo, se explicará cómo crear nuestra vista para el proyecto y así lograr ver el resultado por la pantalla del navegador. Además, configuraremos nuestras vistas dentro de Spring MVC a través del archivo de propiedades de Spring.

¡Vamos con todo!



La vista

Las vistas, por lo general y en la práctica, corresponden a los archivos jsp (JavaServer pages), los cuales corresponden a archivos que se pueden programar y mezclar entre los lenguajes Java y HTML.

Cabe recordar, que la vista es la encargada de estructurar la forma en que se mostrarán los datos en la pantalla de los usuarios, por lo tanto el o los controladores se encargará de obtener los datos desde el modelo, para así poder pasarlos a las vistas correspondientes.

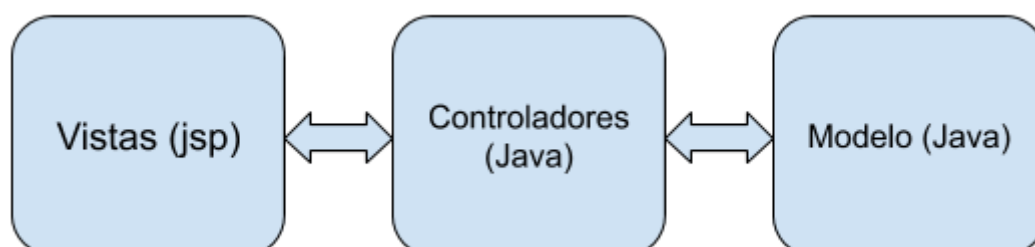


Imagen 1. Modelo MVC.
Fuente: Desafío Latam.

La vista, realiza una petición de datos al controlador, así mismo, el controlador resuelve y procesa la petición de la vista en cuestión, para así realizar la petición de los datos solicitados al modelo.

El modelo, procesa la petición del controlador, devolviéndole a este los datos solicitados.

El controlador, finalmente procesa los datos obtenidos desde el modelo, para así devolver los datos ya procesados y limpios a la vista, quien finalmente muestra dichos datos al usuario.

En base a lo anterior, vamos a crear la vista en nuestro proyecto. Esta será un archivo jsp que llamaremos hello.jsp

1. En la carpeta src/main/webapp/WEB-INF de nuestro proyecto, hacer click derecho, e ir al menú new y seleccionar Jsp File.

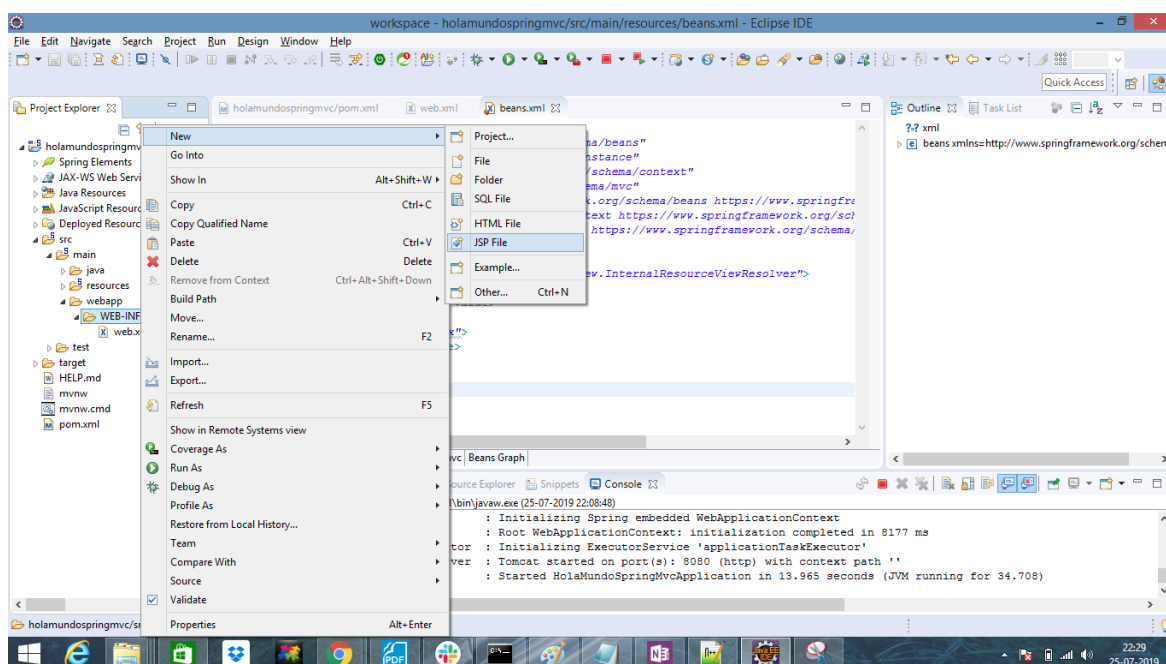


Imagen 2. Creando un nuevo archivo .JSP.

Fuente: Desafío Latam.

2. En la siguiente ventana, en File Name;, colocar el nombre de nuestro archivo. Para este caso hello.jsp.

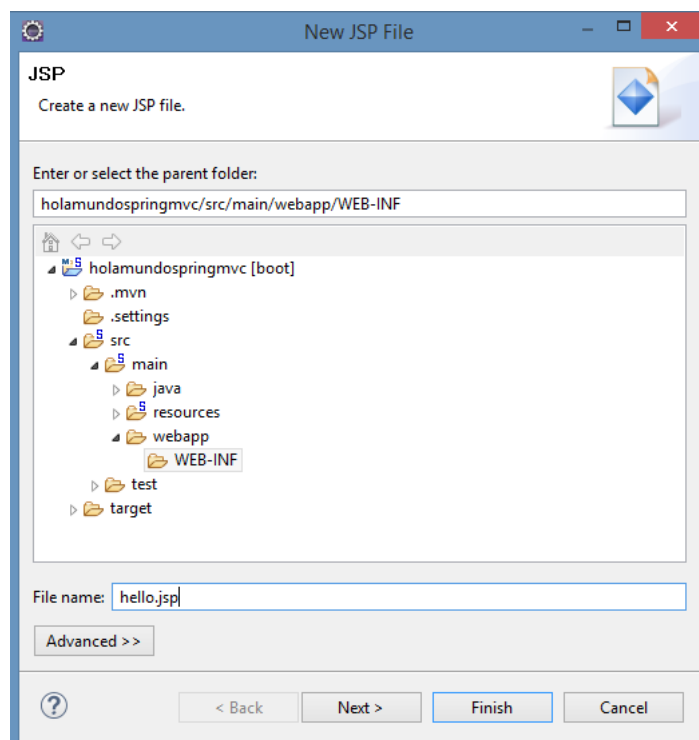


Imagen 3. Creando hello.jsp.
Fuente: Desafío Latam.

3. Luego, presionar el botón Finish, y nuestro archivo, se habrá creado exitosamente con el código jsp estándar para comenzar.

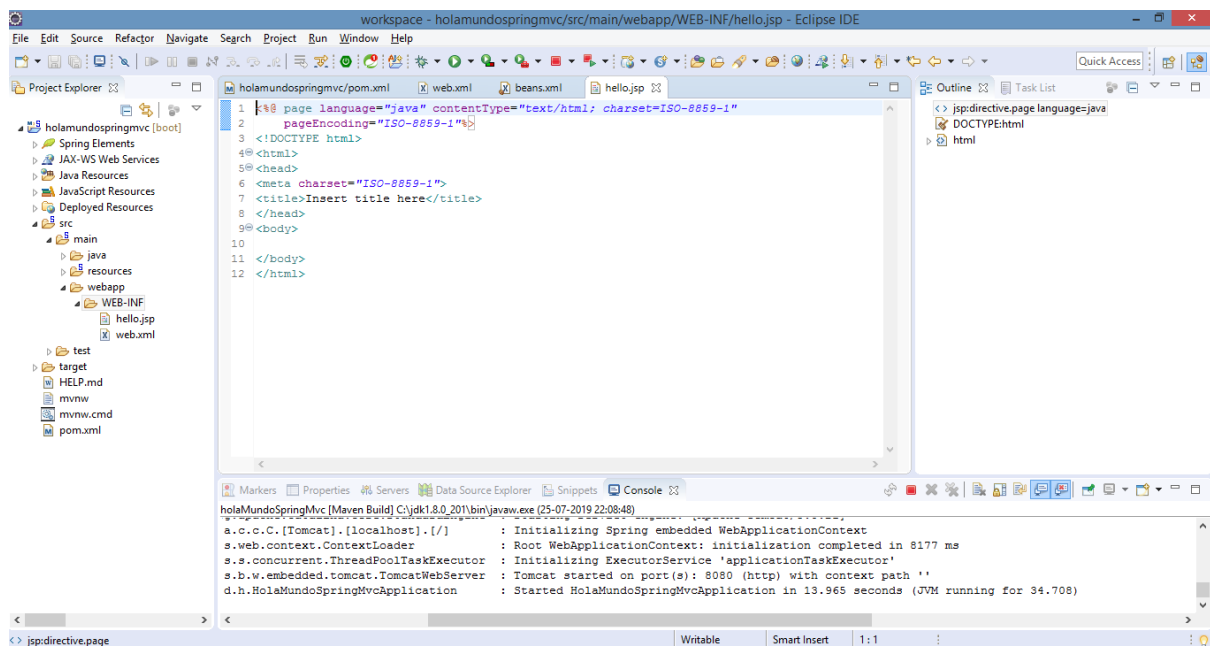


Imagen 4. Archivo creado.
Fuente: Desafío Latam.

4. Dentro de las etiquetas `<body></body>`, escribiremos "Hola Mundo Spring MVC !!!". Luego guardamos el archivo.

Pues bien, para que todo funcione a la perfección, nos quedaría crear y programar nuestro controlador, quien tendrá la responsabilidad de recibir la petición de la vista y mostrarla en pantalla. El archivo `hello.jsp`, quedaría de la siguiente forma:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
    Hola Mundo Spring MVC !!!
</body>
</html>
```

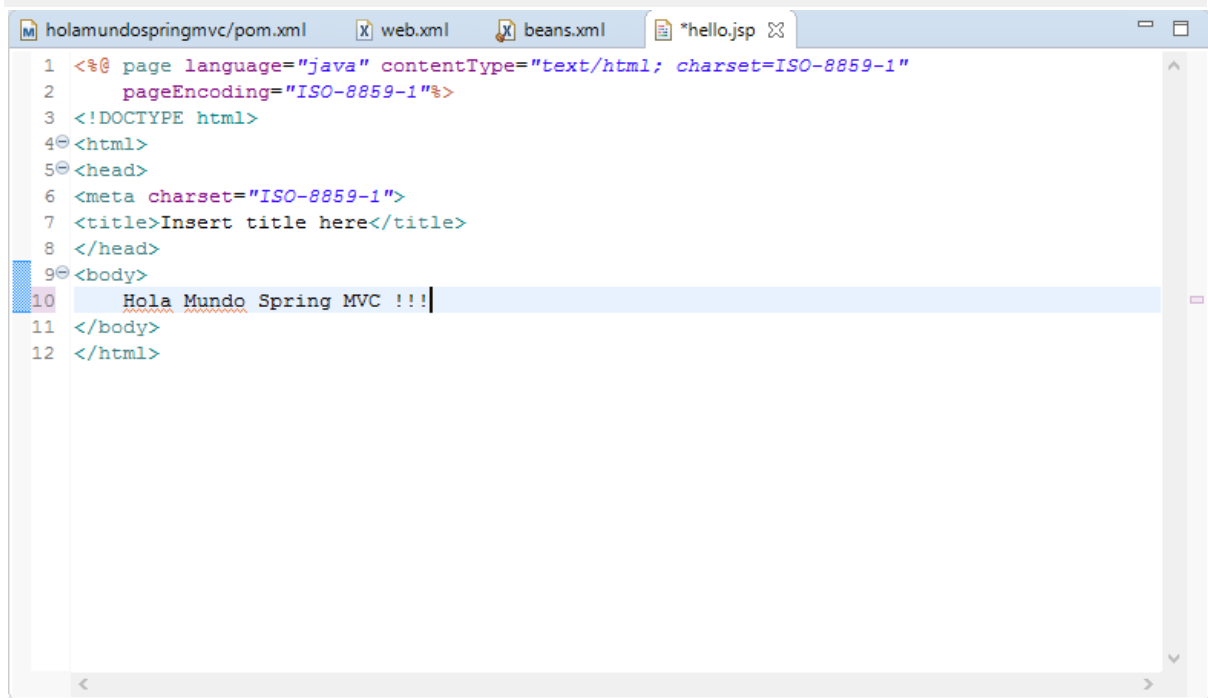


Imagen 5. Archivo `hello.jsp`.
Fuente: Desafío Latam.

Una vista válida también sería la que vimos en Spring MVC Form.

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
```

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@taglib prefix="form" uri="http://www.springframework.org/tags/form"
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Inicio</title>
</head>
<body>
  <p>Pagina de <c:out value="${page_name}" /></p>
  <c:set var="edad" value="${18}"/>
  <c:if test = "${edad > 17}">
    <p>Mayor de edad<p>
  </c:if>
  <form:form id="frmAcciones" modelAttribute="contact"
action="/addContact">
    <div>
      <fieldset>
        <legend>Mantenedor de Contactos</legend>
        <table border="1">
          <tr>
            <td>Nombre: </td>
            <td><input type="text" id="idNombre"
name="nombre"/></td>
            <td>Apellido Paterno: </td>
            <td><input type="text" id="idApellidoPaterno"
name="apellidoPaterno"/></td>
          </tr>
          <tr>
            <td>Dirección: </td>
            <td><input type="text" id="idDireccion"
name="direccion"/></td>
            <td>Teléfono: </td>
            <td><input type="text" id="idTelefono"
name="telefono"/></td>
          </tr>
          <tr>
            <td colspan="4">
              <input type="button" value="Buscar" />
              <input type="button" value="Agregar"
onclick="submitFormAction('/contactManager/addContact', 'post')"/>
              <input type="button" value="Eliminar" />
            </td>
          </tr>
        </table>
      </fieldset>
    </div>
  </form>
</body>
</html>
```

```
        </tr>
      </table>
    </fieldset>
  </div>
</form:form>
</body>
</html>
```

Configurando la vista con MVC

Recapitulando hasta el momento, en nuestro proyecto hemos logrado lo siguiente:

- Creación Proyecto Maven con Spring Boot.
- Configuración del dispatcher en el archivo web.xml.
- Creación de la Vista.

Pues bien, estamos a punto de llegar a la meta, solo nos queda lo siguiente:

- **Configurar la Vista:** Es necesario indicarle a Spring donde se encuentran las vistas de nuestro programa y qué formato tendrán (html, jsp, otros). Para esto, Spring Boot provee a los desarrolladores un archivo de configuración llamado application.properties, el cual nos servirá para configurar todo lo relacionado con Spring Boot.
- **Creación y desarrollo del controlador:** Este, corresponde a una clase controller, la cual recibirá las peticiones desde el cliente.

De forma más práctica, configuraremos la vista, para nuestro controlador, de la siguiente manera:

1. Abrir el archivo `application.properties` de nuestro proyecto, ubicado en `src/main/resources`.

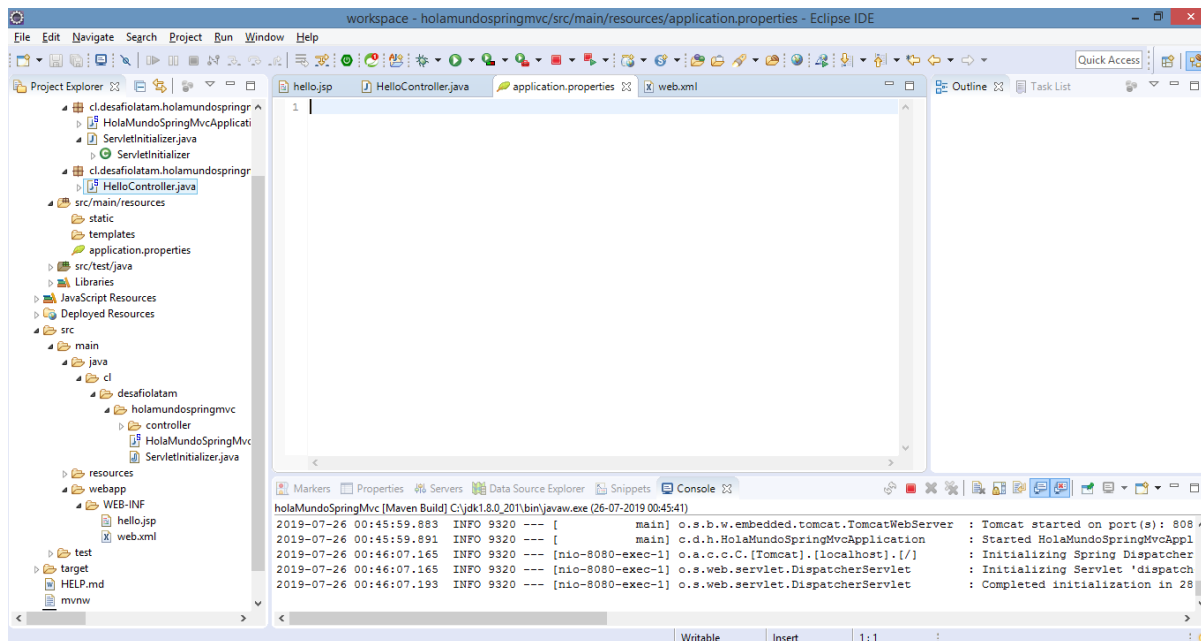


Imagen 6. `application.properties`.

Fuente: Desafío Latam.

2. El archivo `application.properties`, como se puede apreciar, se encuentra vacío y listo para comenzar a configurar la aplicación, en este caso la vista. En dicho archivo, colocaremos las siguientes líneas:

```
spring.mvc.view.prefix=/WEB-INF/
```

Esta línea, indica en qué lugar de nuestro proyecto, Spring debe buscar las vistas.

```
spring.mvc.view.suffix=.jsp
```

La línea anterior, indica a Spring el formato de la vista. Para nuestro caso, la vista es un jsp, como se ha definido en el apartado anterior.

El archivo application.properties, debiese quedar como se muestra a continuación:

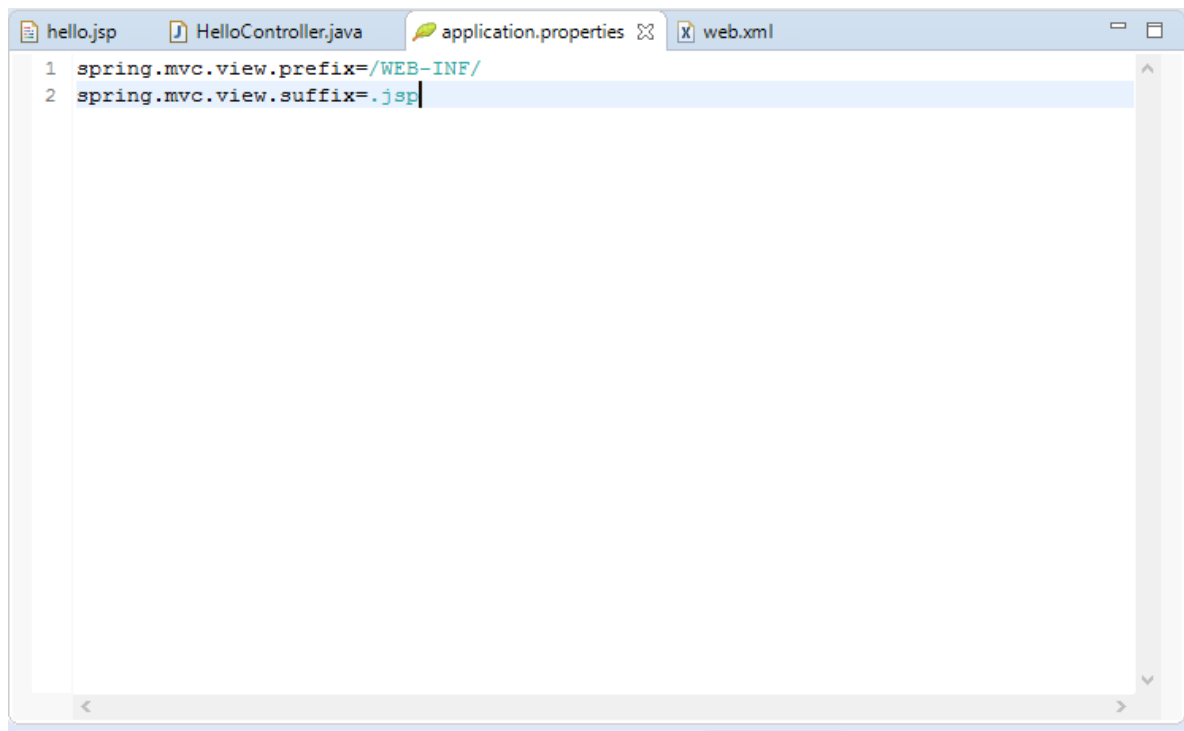


Imagen 7. Archivo application.properties.
Fuente: Desafío Latam.