

Trabajo Práctico de presentación obligatoria.

Objetivo de aprendizaje:

- Creación de un proyecto spring boot.
- Uso de trazas con Log.
- Uso de @PostMapping
- Html – formularios
- Capa Service

PARTE 1

1.- Utilizando el IDE **Spring Tool Suite**, cree un proyecto de tipo **Spring Starter Project**. Coloque como nombre de proyecto **Apellido_Nombre_LU_TP2**. (donde **Apellido** es su apellido, **LU** es su número de libreta universitaria. Ej: **Diaz_Maria_1245_TP2**).

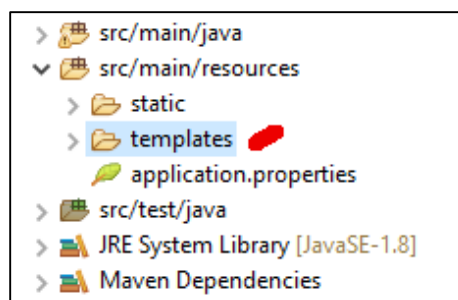
El nombre del paquete debe cumplir con lo siguiente:

`ar.edu.unju.fi.tp2`

2.- A continuación cree un nuevo paquete con el nombre **ar.edu.unju.tp2.model**, en dicho paquete agregue la siguiente clase java. La clase debe estar anotada con el estereotipo **@Component**, esto marcará a la clase como un bean de Spring.

Nombre de clase	atributos
Producto	codigo (int) nombre (String) precio (double) marca (String) stock(int)

3.- Dentro de la carpeta **templates**, cree los archivos html necesarios



En cada archivo html agregue el namespace que refiere al uso de thymeleaf

```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3   <head>
4     <meta charset="UTF-8">
5     <title>Aquí el título</title>
6   </head>
7   <body>
8
9   </body>
10 </html>
11
```

En un archivo html, por ejemplo nuevo.html confeccionar un formulario para poder dar de alta un nuevo producto, la acción del formulario es disparar una petición post para guardar el objeto producto en la lista de productos.

Luego de guardar muestre otra página html con el mensaje “El producto fue guardado con éxito” y agregue un enlace que solicite mostrar la página ultimoproducto.html, la cual mostrará el detalle del último producto agregado a la lista.

En la carpeta templates, debería tener las siguientes páginas html: **nuevo.html**, **resultado.html**, **ultimoproducto.html**

4.- Vamos a agregar un nuevo paquete con el nombre **ar.edu.unju.fi.tp2.controller**. En este paquete agregaremos las clases java anotadas con **@Controller**, se las denomina controladores porque son las encargadas de tomar las peticiones y controlar que respuesta debe ser presentada en la vista (página html) de nuestro proyecto.

Dentro del paquete agregaremos la siguiente clase:

ProductoController

Agregue los métodos necesarios para responder a las siguientes peticiones:

/producto: de tipo **Get**, muestra la página con el formulario para dar de alta un nuevo producto (nuevo.html)

/producto/guardar: de tipo **Post**, guarda el nuevo producto en la lista de productos que maneja la capa Service. Luego muestra la página resultado.html

/producto/ultimo: de tipo **Get**, muestra en la página ultimoproducto.html todos los datos del último producto que se agregó en la lista.

5.- Agregue dos nuevos paquetes **ar.edu.unju.fi.tp2.service** y **ar.edu.unju.fi.tp2.service.imp** en dichos paquetes declare la lógica necesaria para poder guardar objetos de tipo Producto en una lista (ArrayList). En el paquete service la interface **IProductoService** (en la interface solo la declaración de los métodos necesarios para guardar un producto en la lista y otro método para obtener el último producto de la lista) y en el paquete imp la clase que implementa la interface: **ProductoServiceImp** (la implementación de los métodos de la interface y la declaración de una lista de objetos Producto).

Nota: debe haber visto los videos 19 y 20 que refieren al uso de ArrayList.

6.- Agregue trazas de Log para poder identificar las acciones que se realizan en la capa Service. Ej: si se guardó un nuevo producto, podríamos mostrar en la traza el estado del objeto que se guardó.