



Desarrollo web en entorno servidor
Acceso a datos

Actividad 1.2 / 3.1
Desarrollo de una aplicación Web con Spring Boot
Cuarta parte – Listado de productos

CONTENIDO

1.- Objetivos	2
2.- Requisitos	2
3.- Tareas a realizar	2
3.1.- Definir la clase producto	2
3.2.- Crear el repositorio para productos.	2
3.3.- Crear el servicio para las categorías.....	2
3.4.- Cargar datos de productos en el repositorio	3
3.5.- Crear el controlador de productos	3
3.6.- Crear el listado de TODOS los productos	3
3.7.- Crear el listado de los productos de una categoría	3
3.8.- Enlazar el listado de categorías con el listado de productos por categoría	3

1.- Objetivos

Definir la clase “producto”.

Crear las interfaces y clases para el almacenamiento de productos usando estructuras de datos en memoria.

Realizar un listado de TODOS los productos.

Realizar un listado de productos por categoría. Hacer que el listado de categorías enlace con el listado de productos por categoría.

2.- Requisitos

Haber completado la tercera parte de la actividad, en la que ya se tiene una aplicación Spring funcional, pero con una página que muestra el listado de categorías.

3.- Tareas a realizar

3.1.- Definir la clase producto

Para definir esta clase hay que tener en cuenta:

- Se debe elegir un nombre adecuado.
- Hay que definir un identificador de categoría (la clave primaria). Debe ser obligatoriamente de tipo “Long” y debe llamarse “id” o “nombreDeClaseId”.
- Definir el resto de los atributos necesarios para la clase. Como mínimo, será necesario el nombre del producto y una descripción de texto más larga.
- Todos los productos tendrán, obligatoriamente, una imagen asociada.
- Las imágenes de productos, igual que se hizo con las de las categorías, se almacenarán en algún directorio dentro de resources/static.

3.2.- Crear el repositorio para productos.

Siguiendo el esquema de repositorio genérico basado en Map, que ya se habrá usado para las categorías, crear la interfaz y la implementación del repositorio de productos.

Se recuerda que en cualquier parte del programa que necesite un repositorio de productos se debe inyectar un objeto de la interfaz, no de la implementación. El contenedor de IoC de Spring buscará la implementación adecuada para la dependencia inyectada.

3.3.- Crear el servicio para las categorías

Siguiendo el esquema de servicio, que ya se habrá usado para las categorías, crear la interfaz y la implementación de un servicio para la lógica asociada a los productos.

No añadir al servicio ningún método innecesario. El servicio sólo tendrá los métodos que se deban usar desde los controladores. Por ejemplo, si en ningún sitio se tiene que obtener el número total de productos, el servicio no tendrá un método count(), aunque el repositorio sí que lo tenga. Sin embargo, si se necesita obtener un listado de todos los productos, sí que tendrá un método findAll, que llamará al método del repositorio.

Al igual que con los repositorios, siempre que se necesite un servicio de productos se debe inyectar un objeto de la interfaz, no de la implementación. El contenedor de IoC de Spring buscará la implementación adecuada para la dependencia inyectada.

3.4.- Cargar datos de productos en el repositorio

Crear un bean CommandLineRunner para inicializar los datos en el repositorio de productos. Se deben crear, al menos, productos en dos categorías. Así se podrán listar los dos tipos de listados que se van a desarrollar.

Si la inicialización de datos en categorías se está realizando de otra forma (sin CommandLineRunner), se puede hacer de la misma forma para los productos.

3.5.- Crear el controlador de productos

Crear un controlador (paquete controllers) para atender las peticiones asociadas a los productos. En este controlador se crearán los métodos para el listado de todos los productos, y el listado de productos por categoría.

3.6.- Crear el listado de TODOS los productos

Crear un método en el controlador de productos, y una vista asociada, para mostrar el listado de todos los productos de la tienda, sin filtrar por categoría.

Puede que, en función de las características / estructura de la tienda, en el listado de productos también aparezca el listado de categorías. En este caso, en el modelo que se pase a la vista, además de los productos, habrá que pasar las categorías de la tienda. Valorar si es útil utilizar @ModelAttribute para incluir este listado de categorías en todos los modelos del controlador.

3.7.- Crear el listado de los productos de una categoría

Crear un método en el controlador de productos, y una vista asociada, para mostrar el listado de todos los productos de la tienda, filtrando por categoría. Este método recibirá el id de categoría a través de la URL de la petición, con @PathVariable.

El controlador, para obtener los datos tendrá que utilizar un método del servicio de productos, que recibirá el id de la categoría, y devolverá todos los productos de esa categoría. Opcionalmente, si se desea, se puede añadir un método al repositorio de productos para que el filtrado se haga a nivel de repositorio, pero no es obligatorio.

En la página generada tendrá que aparecer, de alguna forma, el nombre de la categoría que se está listando.

Además, si en la página aparece un listado de categorías, la categoría listada tiene que aparecer destacada de alguna forma respecto a las demás. Puede ser negrita, otro color, subrayado, otro tamaño, etc. Se recomienda usar una clase CSS para la categoría destacada.

Al igual que el apartado anterior, puede que, en función de las características / estructura de la tienda, en el listado de productos también aparezca el listado de categorías. En este caso, en el modelo que se pase a la vista, además de los productos, habrá que pasar las categorías de la tienda. Valorar si es útil utilizar @ModelAttribute para incluir este listado de categorías en todos los modelos del controlador.

3.8.- Enlazar el listado de categorías con el listado de productos por categoría

Modificar el listado de categorías que se hizo en la anterior actividad para que cada categoría del listado enlace con el listado de productos de una categoría.

Si no la hay ya, añadir, en algún punto de la web, un enlace para poder acceder al listado completo de productos. Puede añadirse, entre otros sitios, como una opción adicional en un listado de categorías.