

UT 1.2 – Ejercicios

Ejercicio 07 – Inyección de dependencias en Spring

Copia el proyecto del ejercicio 06, y modifica los nombres de artefactos, paquetes y clases que consideres adecuado, para que se refleje que este es un ejercicio distinto.

Si lo prefieres, puedes ampliar el proyecto del ejercicio 06, para añadir lo que se plantea en este.

Queremos cambiar el comportamiento de las clases `ProcesoA`, `ProcesoB` y `ProcesoC` para que, además de mostrar el nombre cualificado de la clase, muestren una frase de una película.

Cada vez que se ejecute el proceso, generará una frase aleatoria de una película, usando un objeto `Faker`:

- `ProcesoA` –Regreso al futuro (provider `BackToTheFuture` en `Faker`)
- `ProcesoB` – Big Bang Theory (provider `BigBangTheory` en `Faker`)
- `ProcesoC` – Juego de Tronos (provider `GameOfThrones` en `Faker`)

Importante: el objeto `faker` tiene que ser un bean que se inyecte en las clases `ProcesoA`, `ProcesoB` y `ProcesoC` como dependencia. Esta dependencia se tiene que añadir a las que ya tengan cada una de estas clases, como un nuevo parámetro en constructor.

Ayuda:

Como la clase `Faker` no es parte de nuestro código, no podemos modificarla para añadir la anotación “`@Component`”. En este caso habrá que declararla como `@Bean`, y hacer un método que devuelva el objeto `Faker`. Para esto:

- Crear una clase “`BeansConfiguration`” en un paquete “`config`” de la aplicación
- Anotar esta clase con `@Configuration`
- Crear un método, con el nombre que se quiera (no es muy importante) que cree un objeto `Faker` y lo devuelva.
- Anotar el método con `@Bean`.

Probar la aplicación y verificar que se genera el nuevo mensaje.