

Internacionalización basada en ficheros

Curso intermedio de Thymeleaf

Algunos conceptos

- Locale: representa una región del mundo. Nos permite conocer el idioma, la moneda, la forma de puntuar los números... Nuestra JVM tiene uno por defecto.
- Mensajes: pensemos en ellos como etiquetas de texto (labels) que vamos a utilizar en nuestra aplicación para identificar títulos, errores, encabezados, ... y que suelen ser estáticos.

Spring Boot + Thymeleaf

- Admite el manejo de mensajes localizados (en función de un *Locale*).
- Por defecto, busca un fichero llamado `messages.properties` en la raíz del classpath.
- Thymeleaf ofrece funciones que nos permiten visualizar los mensajes asociados al *Locale* activo.

Configuración

- LocaleResolver: permite establecer el *Locale* activo.
 - Nos permite indicar una estrategia para escogerlo.
 - Session, Cookie, AcceptHeader
 - Podemos establecer un *Locale* por defecto
- LocaleChangeInterceptor: permite gestionar el cambio de Locale
 - Lo más fácil es a través de un parámetro en la URL.
 - Debemos registrar este interceptor.

messages_XX.properties

- Almacenados en /src/main/resources/
- El fichero con el idioma por defecto: messages.properties
- Resto de idiomas: messages_XX.properties
 - messages_en.properties
 - messages_it.properties
 - messages_fr.properties
 - ...

Thymeleaf

- Nos ofrece las expresiones `#{...}` para usar con `th:text`
- Reciben como argumento un mensaje definido anteriormente

```
<h3 th:text="#{titulo_principal}">Texto</h3>
```
- En función del Locale activo, se *inyecta* uno u otro mensaje.
- Si este mensaje no está definido en un idioma (por ejemplo inglés), se utiliza el mensaje por defecto.

Reto

- Plantillas: completa los cambios necesarios en el resto de plantillas
- Mensajes: añade un nuevo fichero de mensajes, con otro idioma diferente que conozcas.