

Spring

I18N

Spring: I18N

- Spring provee una forma simple de lograr la internacionalización de nuestra aplicación.
- Para esto es necesario utilizar los archivos properties.
- Y crear un par de componentes:
 - Un bean para saber la localización desde donde se esta viendo la app.
 - Un interceptor, para cambiar la localización cada vez el usuario lo cambie.

```
public class MvcConfig {  
    @Bean  
    public LocaleResolver  
    localeResolver() {  
        SessionLocalResolver  
        sessionLR = new  
        SessionLocaleResolver();  
  
        sessionLR.setDefaultLocal  
        e(new Locale("es", "ES"));  
        return sessionLR;  
    }  
}
```

Spring: I18N

- Spring provee una forma simple de lograr la internacionalización de nuestra aplicación.
- Para esto es necesario utilizar los archivos properties.
- Y crear un par de componentes:
 - Un bean para saber la localización desde donde se esta viendo la app.
 - Un interceptor, para cambiar la localización cada vez el usuario lo cambie.

```
public class MvcConfig {  
    @Bean  
    ...  
    @Bean  
    public  
    LocaleChangeInterceptor  
    localeChangeInterceptor() {  
        LocaleChangeInterceptor  
        localeInterceptor = new  
        LocaleChangeInterceptor();  
  
        localeInterceptor.setParamN  
        ame("lang");  
        return localeInterceptor;  
    }  
}
```

Spring: I18N

- Y por ultimo, debemos registrar el interceptor en pila.
- Para esto hacemos un override del método `addInterceptor()`.

```
public class MvcConfig {  
    @Bean  
    ...  
    @Bean  
    ...  
    • @Override  
      public void  
      addInterceptor(Interceptor  
      Registry registry) {  
  
        registry.addInterceptor(this.  
        localeChangeInterceptor())  
        ;  
      }  
}
```