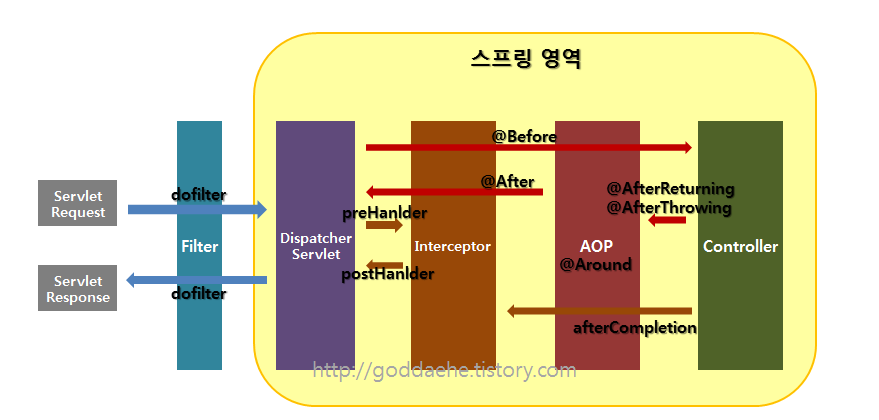
13. 파일 업로더, 인터셉터

인터셉터(Interceptor)란?

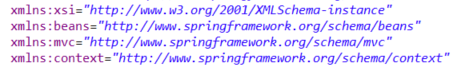


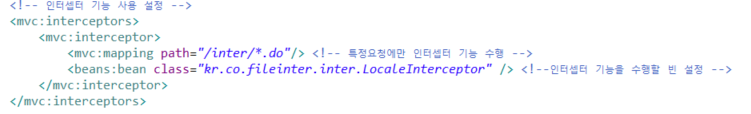
컨트롤러의 핸들러(Handler)를 호출하기 전과 후에 요철과 응답을 참조하거나 가공할

수 있는 일종의 필터

controller의 핸들러가 실행되기 전이나 후에 추가적인 작업을 원할 때 사용

인터셉터 설정





appServlet에 인터셉터 설정

<mvc:interceptors>

<mvc:interceptor>

<mvc:mapping path=”인터셉터를 실행할 url” />

<beans:bean class=”인터셉터 기능을 할 객체”>

inter/index.do를 요청 받을 경우 인터셉터 객체 실행



빈으로 등록한 객체에 HandlerInterceptorAdapter를 상속받아 인터셉터 구현

상속받은 HandlerInterceptorAdapter를 통해 override하는 메서드들

boolean preHandle()

컨트롤러 실행 전에 호출됨

리턴타입은 boolean

void postHandle()

컨트롤러 실행 후 DispatherServlet이 뷰로 보내기 전에 호출 됨

void afterCompletion()

뷰까지 수행하고 나서 호출됨

Locale

Locale이란

어떤 언언어를 출력할 것인가를 결정하게 해주는 수단

spring이 제공하는 <spring:message> 커스텀 태그는 웹 요청과 관련된 언어 정보를

이용해서 알맞은 언어의 메시지를 출력한다

Spring MVC는 LocaleResolver를 이용해 웹 요청과 관련된 Locale을 추출하고 알맞은

언어의 메시지를 선택

<spring:message>

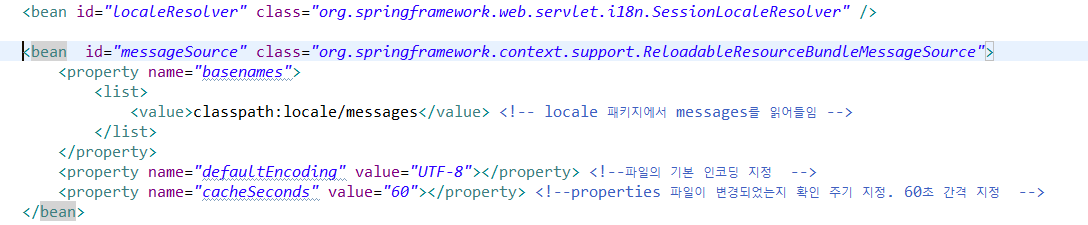
문자열을 담은 메시지 파일을 작성한다

메시지 파일에서 값을 읽어오는 MessageSource을 설정

JSP 코드에서 <spring:message>을 이용해 메시지를 출력

Locale 설정

servlet



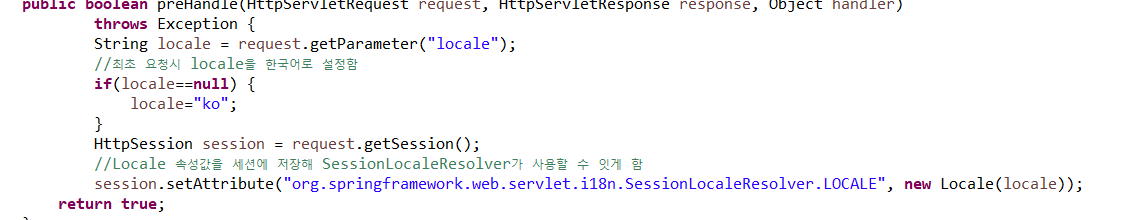
LocaleResolver 설정

sessionLocaleResolver는 세션 영역에 locale 정보를 저장

ReloadableResourceBundleMessageSource 타입의 빈 생성

classpath:를 통해 messages 번들을 locale 패키지에 읽어들임

interceptor

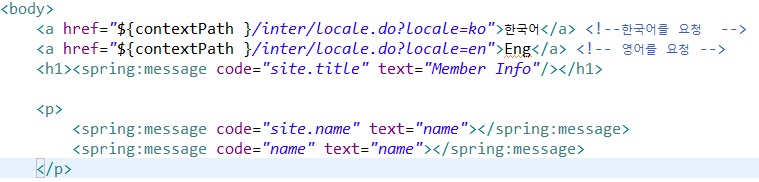


locale 값을 session에 저장

jsp



<spring:message>를 위한 태그 선언



<spring:message code=”” text=””>

code: 룩업할 메세지 키 지정

text : 찾지 못할 경우 출력할 문자열, 없으면 null로 출력

REST API(Representational State Transfer)

Rest란

자원을 자원의 표현(representation)으로 구분하여 해당 자원(resource)의

상태(state)를 주고 받는 사이트 구성 원리

Rest,Restful한 API는 이런 사이트 구성원리를 가진 소프트웨어간 통신 방식

다양한 클라이언트(브라우저, 안드로이드, ios)간 통신이 원할한 API 위해

=> rest는 멀티 플랫폼에 대한 지원을 위해 자원에 대한 아키텍처를 세우고

모색한 결과이다.

자원의 표현

자원

해당 소프트웨어가 관리하는 모든 것

ex) 문서, 그림, 데이터

rest는 자원을 URI로 표현

ex) game/book/non-fiction

자원의 표현

그 자원을 표현하기 위한 이름

자원의 상태에 대한 조작을 요청하는 server는 이에 응답

rest에서 하나의 자원은 JSON, XML,TEXT,RSS등 여러 형태로

표현됨, JSON과 XML이 일반적

그림 => jpg,png…

표현방식은 헤더에 담아 요청한다

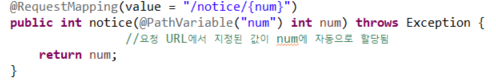
상태

Http 의 메서드(get,post,patch,delete)

@PathVariable

파라미터를 url형식으로 받음

url 자체에서 매개변수를 가져올 수 있다



@RequestMapping(value = “/notice/{num}”)

{매개변수로 사용할 값}

@PathVariable(“num”) int num)

@PathVariable(“매개변수로 받을 값”) 저장할 타입, 객체

=>

url의 notice/{num} 부분이 리턴됨