스프링 웹 개발기초

# 컨텐츠에 따른 분류

1. 정적 컨텐츠

변화가 없는 콘텐츠

HTML, CSS, JS와 같이 미리 서버에 저장해두고 서버가 요청을 받으면 그저 응답만 해주면 되는 것들로 구성되어 있습니다.

2. MVC와 템플릿 엔진

템플릿 엔진

서버에서 프로그래밍 후 HTML을 동적으로 만들어서 제공하는 것

Ex) JSP, PHP

MVC

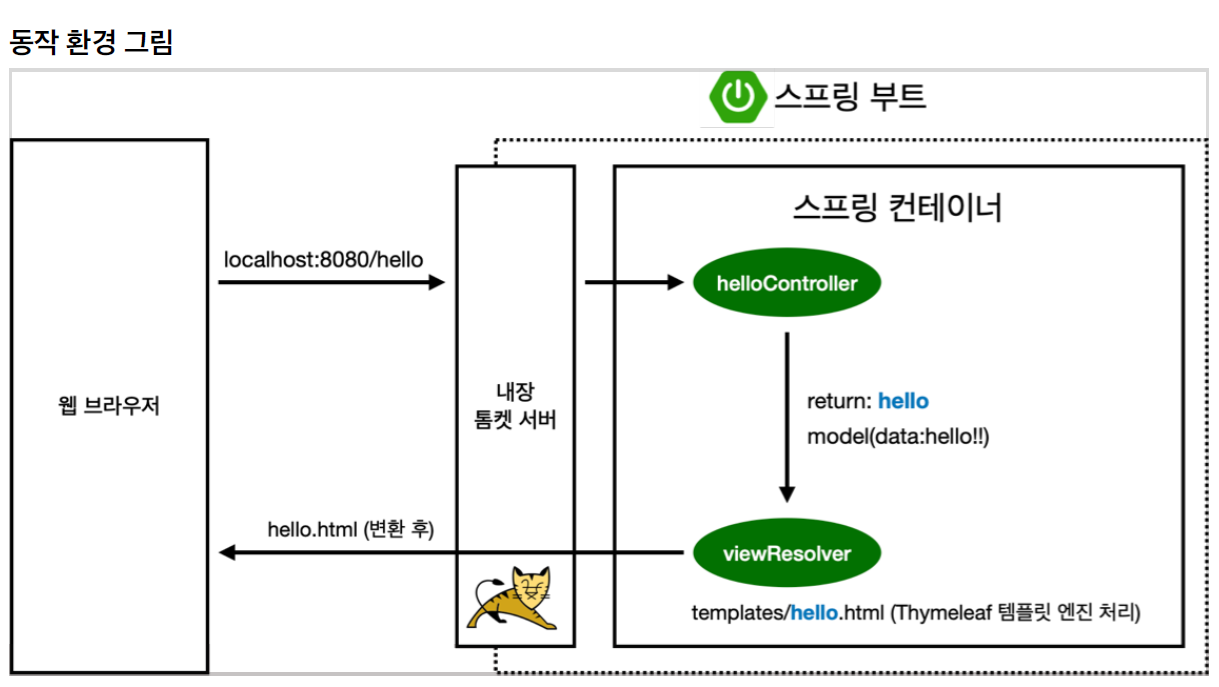
모델, 뷰, 컨트롤러 ----> 요즘 인기 많음.

3. API

JSON 데이터 구조포맷으로 클라이언트에게 데이터 전달하는 것

뷰나 리액트, 서버 간 통신 등에서도 API를 사용하기도 함.

동작환경 그림으로 살펴보기

1. 웹 브라우저 -> 서버(웹페이지 요청)
2. 톰캣 서버에서 ‘hello’ 컨트롤러 검사
3. 스프링은 hello 컨트롤러가 존재함
4. 스프링은 컨트롤러를 반환 ---> 웹페이지가 나타나게 됨
5. 만약에 hello 컨트롤러가 Spring에 없다면????
6. static 폴더에서 파일을 찾아보고 있다면 반환!   
   없다면 Error

# MVC 자세히 알아보기

과거에는 Model, View, Controller을 따로 나누지 않았다. ----> Model1 방식

MVC ----> Model, View, Controller, : 요즘은 이게 기본이다.

View : 화면을 그리는데 집중

Model, Controller : 비즈니스 로직이나 내부적 처리에 집중

# Annotation

## @Controller

클래스 단위로 사용하는 어노테이션.

Controller 어노테이션이 붙은 클래스는 웹 요청을 처리하는 컨트롤러로 사용가능하다.

## @GetMapping(String)

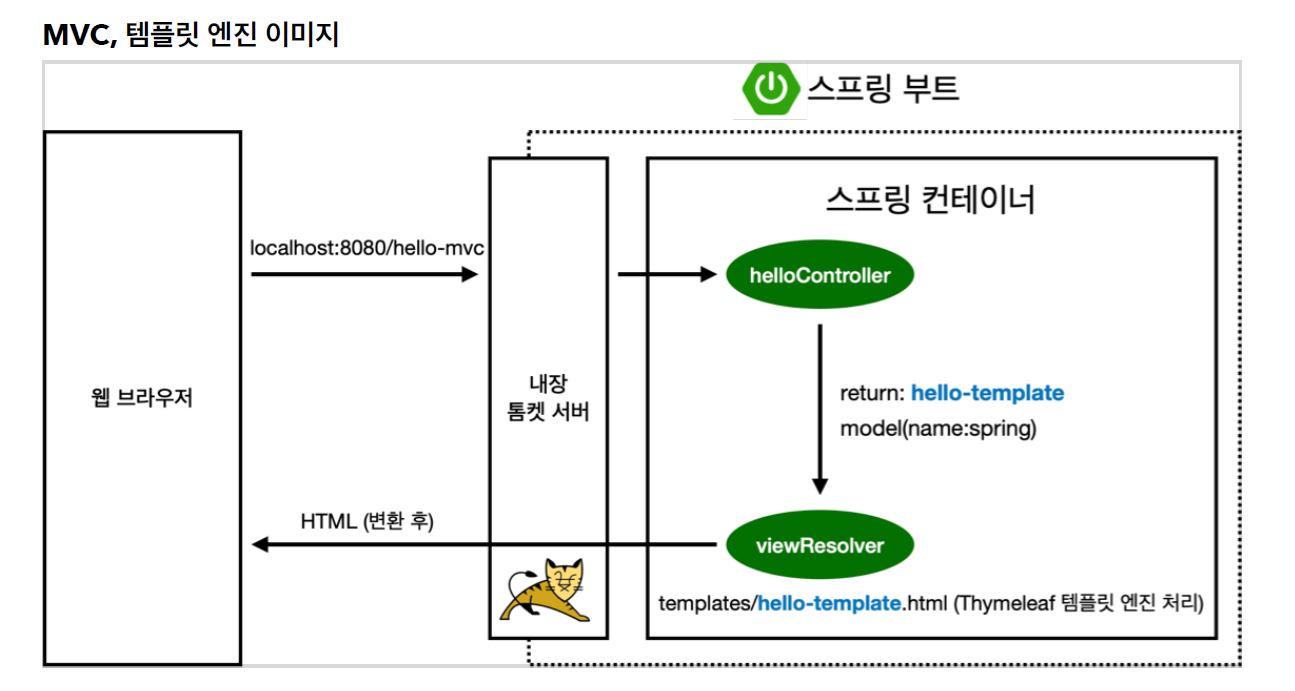
메소드 수준으로 사용하는 어노테이션

GetMapping을 통해 넘어온 String 매개변수는 클라이언트가 요청한 주소

이 노테이션이 붙은 메서드는 클라이언트가 해당 주소로 접근했을 경우 실행된다.

이 어노테이션이 붙은 메서드는 String값을 반환하며, 컴퓨터는 반환된 String값과 이름이 같은 파일을 templates 폴더 안에서 찾아 HTML파일로 변환 후 클라이언트에게 반환해준다.

Ex) @GetMaaping(“hello”)라는 어노테이션이 붙은 메서드의 경우

IP주소:포트번호/hello로 접속할 경우, 실행된다.

## @RequestParam(String)

@GetMapping 어노테이션이 붙은 메서드의 매개변수에 붙이는 어노테이션

해당매개변수에 웹페이지로부터 전달된 파라미터 값을 대입한다.

Ex) public String hello(@RequestParam(“name”) name) { }

위와 같은 메서드의 경우 웹페이지를 통해 전달된 name값이 메서드의 매개변수 name에 대입된다.