**spring MVC 필기**

**Spring 프로젝트 생성.**

1. Dynamic web project를 원하는 이름으로 생성한다.

2. Project를 우클릭해서 configure -> Convert to Maven Project를 통해 Maven 프로젝트로 변경한다. (pom.xml에서 ctrl + A 로 문서 전체를 선택한 후 ctrl + shit + F 를 누른다. (들여쓰기 정리))

3. pom.xml에 다음 라이브러리를 설정한다.

servlet-api

jsp-api

jstl

springWebMVC

라이브러리 설정은 MVNRepository라는 곳에서 각각 검색하며 붙여넣기한다. (pom.xml 참고)

**URL주소**

경로1/경로2/경로3에서 첫번째 경로를 Context Path라고 한다. 한 서버에서 각각의 웹 어플리케이션을 구분하기 위한 이름이며 폴더의 이름 자체가 Context Path가 된다.

예를 들어, SpringTest1 프로젝트에서 작업을 하고 있다면 SpringTest1이 Context Path가 되는 것이다.

**URL mapping**

@RequestMapping(value="/test6",method=RequestMethod.***GET***)

**public** String test6() {

**return** "sub2/test6";

}

requestMapping의 value에 나오는 주소는 실제 주소가 요청을 받는 주소이다.

index.html에서 /test6으로 링크를 걸면 test6()이 실행되고 sub2/test6.jsp를 보여준다.

실제 주소는 return 하는 sub2/test6이므로 sub2폴더안에 test6이 있어야 실행된다.

**요청 방식**

RequestMapping은 요청 주소와 요청 방식을 지정 가능하다.

get 방식과 post 방식이 있는데 동시에 하는 방법은

@RequestMapping(value="/test5",method={RequestMethod.***GET***,RequestMethod.***POST***}) 이렇게 method에 둘 다 써주는 것이다.

**파라미터 추출하기**

클라이언트 요청으로 전달하는 파라미터 데이터를 전달 받는 여러 방식이 있다.

spring MVC는 필요한 데이터나 객체를 주입을 받아 사용한다.

HttpServletRequest : servlet, jsp에서 쓰는 방식 getParameter 등을 이용해서 데이터를 받아온다.

WebRequest : Spring MVC에서 제공하는 HttpServeletRequest를 확장한 것이다.

@PathVariable : 데이터가 요청 주소에 있을시 값을 주입받는 방식

Restful API 서버 프로그래밍에서 사용하는 방식

주소/값1/값2/값3과 같이 값이 온다..

@GetMapping("/test4/{data1}/{data2}")

**public** String test4(@PathVariable String data1,

@PathVariable String data2) {

System.***out***.println(data1 + " " + data2 + " " + data3);

**return** "result";

}

@RequestParam : 파라미터 데이터를 변수로 직접 주입받고 형 변환도 가능하다.

@GetMapping("/test5")

**public** String test5(@RequestParam **int** data1, @RequestParam **int** data2, @RequestParam **int**[] data3)

처럼 변수와 data 이름을 똑같이 받을 수도 있고

@RequestParam(value = "data1") **int** val1, 처럼 변수와 이름이 다르게 할 수도 있다.

**객체로 파라미터 주입받기.**

map이나 list로 Requestparam을 받으면 String으로만 받을 수 있다.

@ModelAttribute

파라미터를 객체로 받을 수 있다.

전달되는 파라미터의 이름과 동일한 프로퍼티에 자동으로 주입된다.

이런 객체를 커맨드 객체라고한다.

**public** String test2(@ModelAttribute DataBean bean)

이런식으로 사용하며 @ModelAttribute를 생략해서 DataBean bean만 써도 실행 가능하다.

DataBean은 입력으로 들어오는 매개변수와 같은 이름으로 멤버변수와 getter, setter가 있어야한다.