

## 12강. 컨트롤러

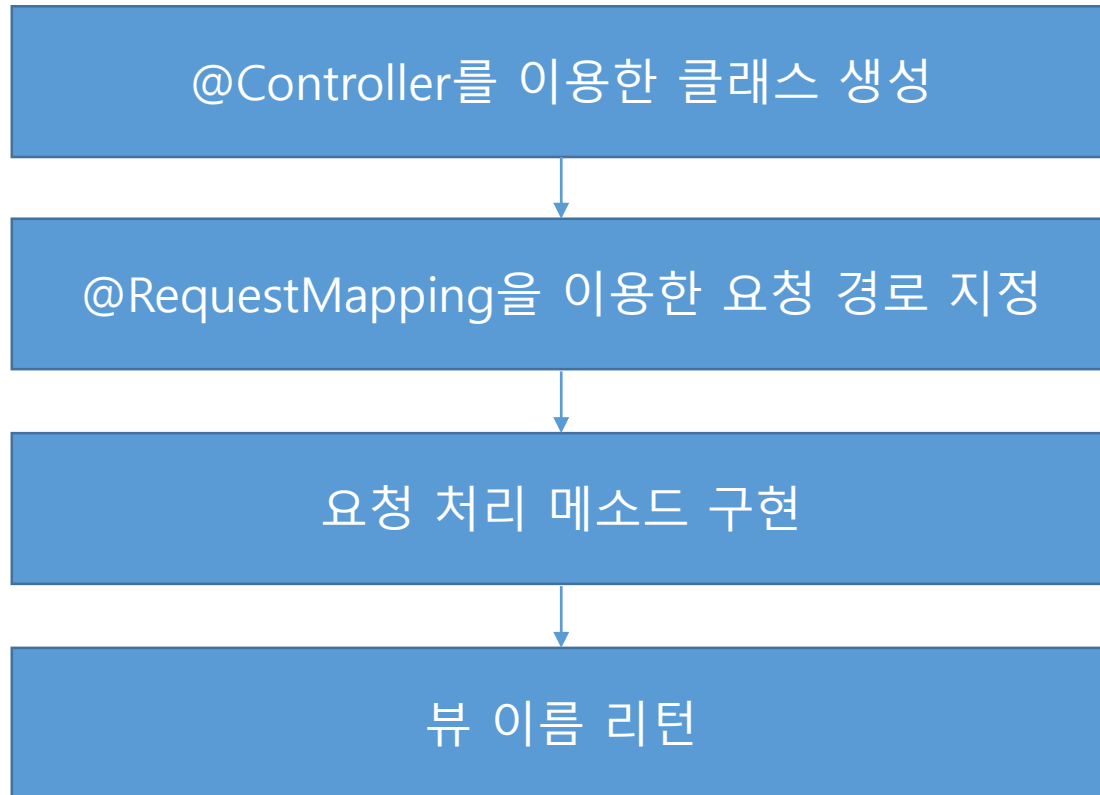
- 컨트롤러 클래스 제작
- 요청 처리 메소드 제작
- 뷰에 데이터 전달
- 클래스에 @RequestMapping 적용

Lecturer Kim Myoung-Ho  
Nickname 불스  
blogstudy@naver.com

## 12-1. 컨트롤러 클래스 제작

최초 클라이언트로부터 요청이 들어왔을 때, 컨트롤러로 진입하게 됩니다.  
그리고 컨트롤러는 요청에 대한 작업을 한 후 뷰쪽으로 데이터를 전달합니다.  
(spring\_12\_1\_ex1\_springex)

### 컨트롤로 클래스 제작 순서



Servlet-context.xml 파일내에서

```
<Context:component-scan base-package = "com.javawec.ex" />
```

우리 패키지를 지정하여 Controller를 찾는다.

```

3 @Controller
4 public class HomeController {
5
6     ...
7 }
  
```

## 12-2. 요청 처리 메소드 제작

클라이언트의 요청을 처리할 메소드를 제작 합니다. (spring\_12\_1\_ex1\_springex)

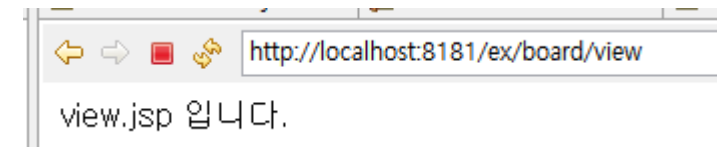
```
@RequestMapping("/board/view")
public String view() {
```

요청 경로(path)

```
    return "board/view";
}
```

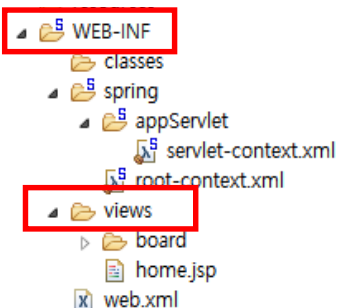
뷰페이지 이름

결과 화면



뷰페이지 이름 생성(조합) 방법

뷰페이지 이름 = prefix + 요청처리 메소드 반환값 + suffix



```
<!-- Resolves views selected for rendering by @Controllers to .jsp resources in the /WEB-INF
<beans:bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <beans:property name="prefix" value="/WEB-INF/views/" />
    <beans:property name="suffix" value=".jsp" />
</beans:bean>
```

### 12-3. 뷰에 데이터 전달

컨트롤러에서 로직 수행 후 뷰페이지를 반환 합니다. 이때 뷰에서 사용하게 될 데이터를 객체로 전달 할 수 있습니다.

#### Model 클래스를 이용한 데이터 전달

```
@RequestMapping("/board/content")  
public String content(Model model) {
```

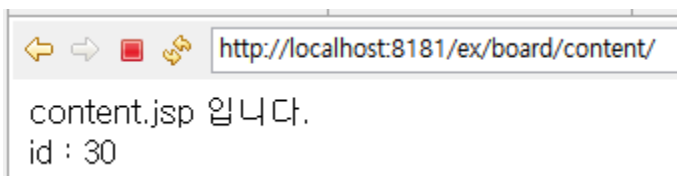
Model 객체를 파라미터로 받음

```
    model.addAttribute("id", 30);  
    return "board/content";  
}
```

Model 객체에 데이터를 담음

```
</head>  
<body>  
    content.jsp 입니다. <br />  
    id : ${id}  
</body>  
</html>
```

컨트롤러에서 전달 받은  
Model객체의 속성을 이용함.



## 12-3. 뷰에 데이터 전달

### ModelAndView 클래스를 이용한 데이터 전달

```
@RequestMapping("/board/reply")
public ModelAndView reply0 {
```

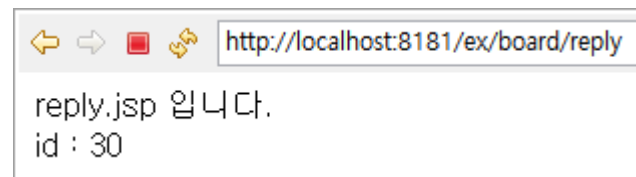
```
    ModelAndView mv = new ModelAndView();
    mv.addObject("id", 30);
    mv.setViewName("board/reply");
```

```
    return mv;
}
```

ModelAndView 객체 생성

Model 객체에 데이터를 담음

뷰이름 설정



## 12-4. 클래스에 @RequestMapping 적용

메서드에 @RequestMapping 어노테이션을 적용하여 요청경로를 얻습니다.  
 @RequestMapping 어노테이션을 클래스에 적용하여 요청경로를 얻는 방법에 대해서 살펴 봅니다.  
 (spring\_12\_4\_ex1\_springex)

```

7 @Controller
8 @RequestMapping("/board")
9 public class HomeController {
10

```

클래스에 @RequestMapping 적용



/board

```

@RequestMapping("/write")
public String write(Model model) {

    model.addAttribute("id", 30);

    return "board/write";
}

```

메서드에 @RequestMapping 적용



/write

조합된 요청 경로 : /board/write

http://localhost:8181/ex/board/write

write.jsp 파일 입니다.