

SPRING

18강_Controller 객체 구현

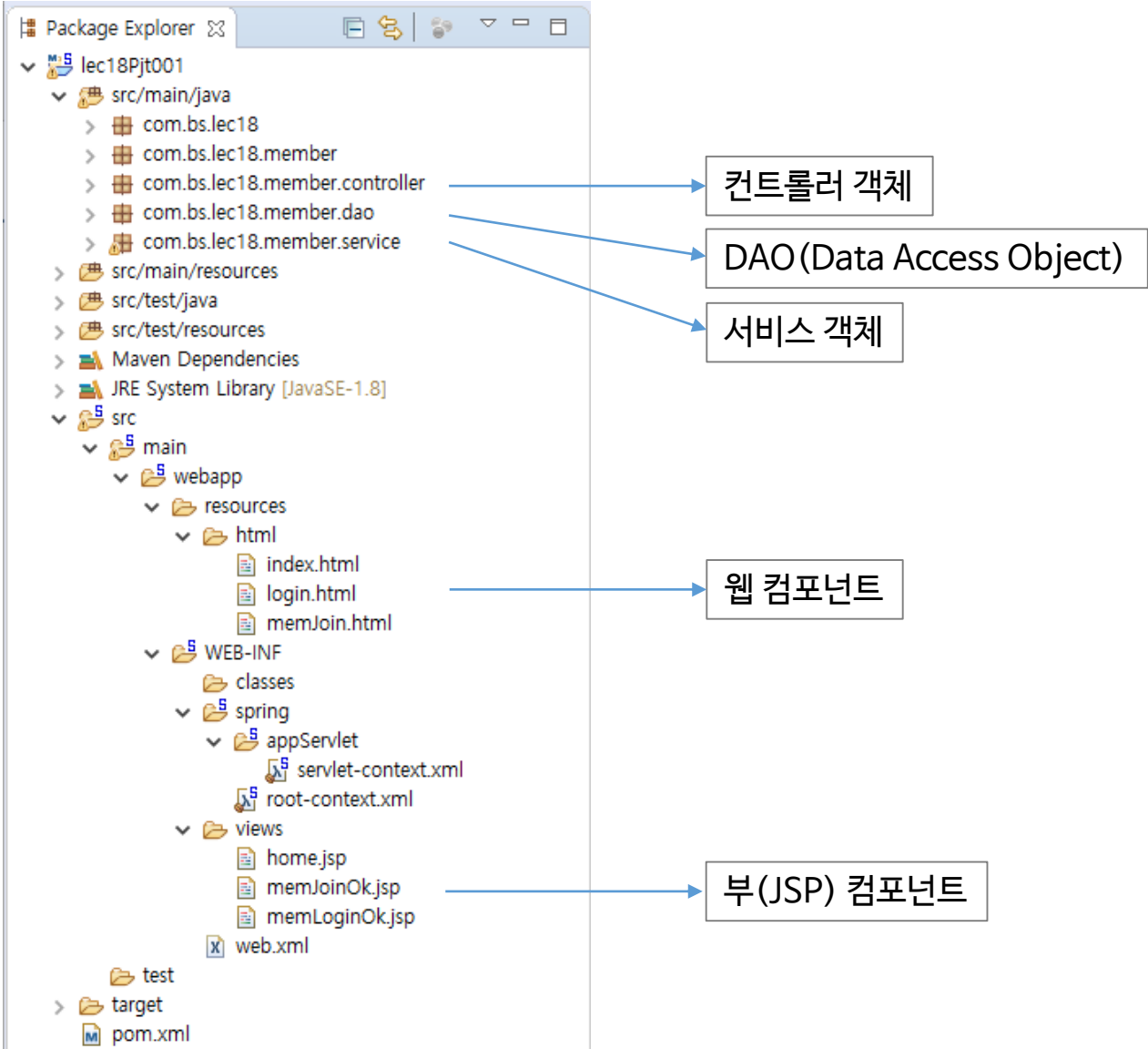
컨트롤러의 URL 맵핑과 파라미터 처리 방법에 대해서 학습합니다.

18-1 웹 어플리케이션 준비

18-2 @RequestMapping을 이용한 URL 맵핑

18-3 요청 파라미터

18-1 : 웹 어플리케이션 준비

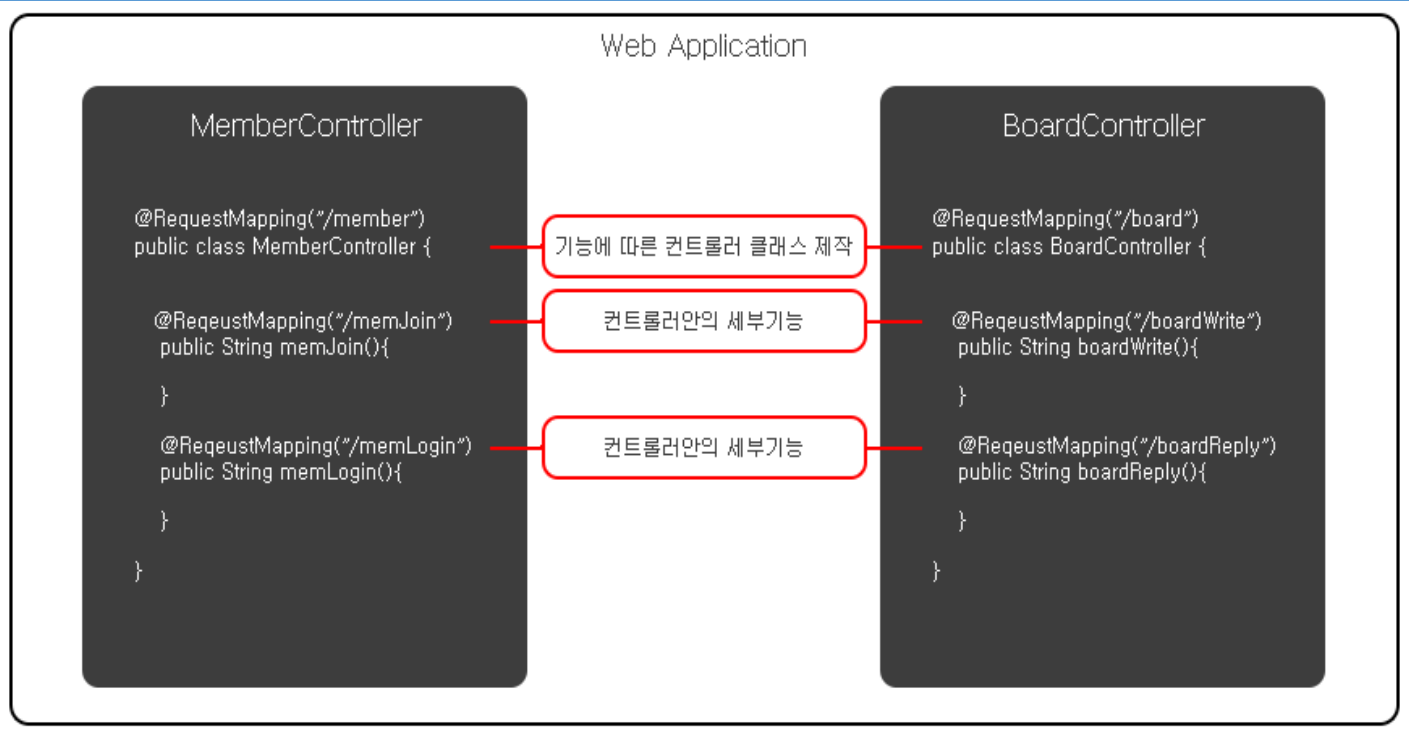


18-2 : @RequestMapping을 이용한 URL맵핑

메소드에 @RequestMapping 적용

http://localhost:8090/lec18/memJoin → memJoin() 실행

클래스에 @RequestMapping 적용



18-2 : @RequestMapping을 이용한 URL맵핑

데이터 전송방식에 따른 method 속성 설정

<form

method="post">



```
@RequestMapping(value="/memJoin", method=RequestMethod.GET)
public String memJoinByGet(Model model, HttpServletRequest request) {

    return "memJoinOk";
}
```

```
@RequestMapping(value="/memJoin", method=RequestMethod.POST)
public String memJoinByGet(Model model, HttpServletRequest request) {


    return "memJoinOk";
}
```

18-3 : 요청 파라미터

HttpServletRequest 객체를 이용한 HTTP 전송 정보 얻기

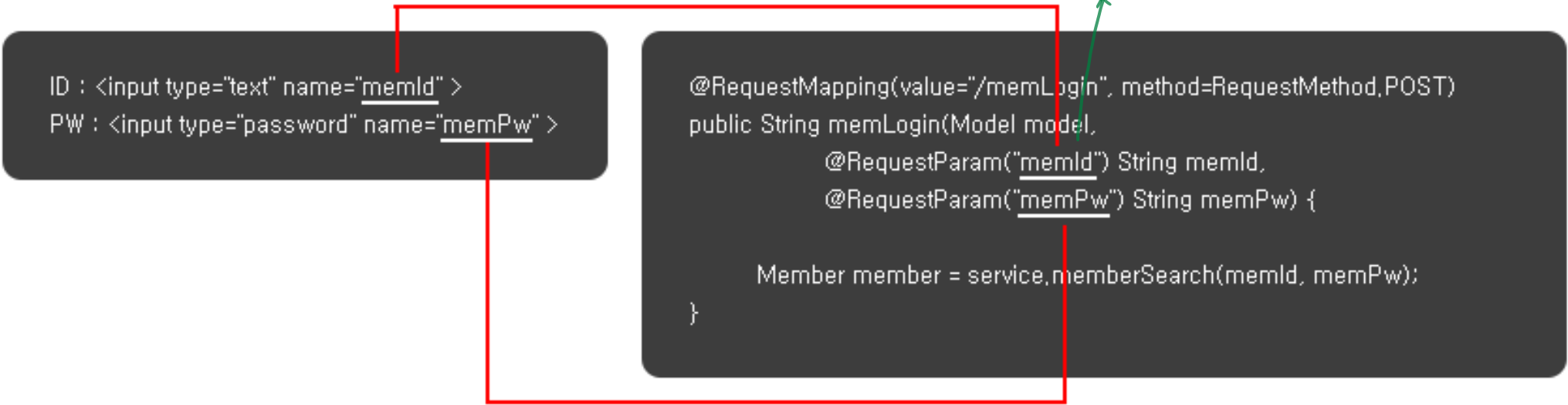
```
ID : <input type="text" name="memId" >  
PW : <input type="password" name="memPw" >
```

```
@RequestMapping(value="/memLogin", method=RequestMethod.POST)  
public String memLogin(Model model, HttpServletRequest request) {  
  
    String memId = request.getParameter("memId");  
    String memPw = request.getParameter("memPw");  
  
}
```



18-3 : 요청 파라미터

@RequestParam 어노테이션을 이용한 HTTP 전송 정보 얻기



@RequestParam (value = "memPw", required = true),
 required = false
 default value = "1234" → 값이 없으면 기본값으로 처리.

→ 파라미터 반드시 넣어야 한다.
 넣지 않으면 에러 발생

→ 넣지 않아도 에러 발생 X

18-3 : 요청 파라미터

→ 가장 많이 쓰인다. (코드인 양이 가장 많이 줄어든다)

커맨드 객체를 이용한 HTTP전송 정보 얻기

