5. 컨트롤러

1. 스프링 MVC 컨트롤러
2. 파라미터 수집과 변환

1.1 컨트롤러 어노테이션

- DispatcherServlet 이 인식하는 Controller 객체로 만들고 컨테이너에 빈 등록
    - @Controller

- URI 요청을 컨트롤러의 특정 메서드와 매핑
    - @RequestMapping
    - @PostMapping   @GetMapping   @DeleteMapping   @PutMapping

- 요청정보 받기
    - 커맨드 객체　　　　 : 질의문자열 "select?id=park&name=dong"
    - @RequestParam　　　 : 질의문자열
    - @RequestBody　　　　 : json 문자열 {"id":"park", "name":"dong"}
    - @RequestPart　　　 : 첨부파일(multipart)
    - @PathVariable　　　 : "select/{name}/{age}"
    - @ModelAttribute

1.2 컨트롤러 만들기

- servlet-context.xml

```
<component-scan />
<annotation-driven/>
```

- @Controller

```
@Controlle
public class HomeController {

  @RequestMapping("/")
  public string home(){
    model.addAttribute("today", new Date());
      return "home";
  }
}
```

2.1 요청정보 받기

▶ **커맨드 객체**   객체 → 커맨드 객체

mypage.do?name=choi&age=20

```
public UserVO {
    private String name;
    private Integer age;
}
```

```
@RequestMapping("/users")
public String process(UserVO users)  {
```

▶ **커맨드 객체**   객체 리스트 → 커맨드 객체

mypage.do?list[0].name=choi&list[0].age=20&list[1].name=park&list[1].age=30

mypage.do?list%5B0%5D.name=choi&list%5B1%5D.name=park   <=== 인코딩

```
public UserListVO {
    private List<UserVO> list;
}
```

```
@RequestMapping("/users")
public String process(UserListVO users)  {
```

2.1 요청정보 받기

▶ **@RequestParam** 질의문자열 → 기본데이터형

```
<form action ="users"  method="post">
    <input name="name" value="hong">
    <input name="age" value="20">
</form>
```

```
mypage.do?name=hong&age=20
```

```
@RequestMapping("/users")
public String process(UserVO vo)  {
```

```
@RequestMapping("/users")
public String rpocess(@RequestParam String name,
                      @RequestParam Integer age)  {
```

```
@RequestMapping("/users")
public String rpocess(@RequestParam Map<String, Object>
map)  {
```

▶ **@RequestParam** 리스트, 배열 → List<String>

```
mypage.do?name=choi&name=park&name=kim
```

```
@RequestMapping("/mypage.do")
public String login(@RequestParam  String[] name)  {
```

```
@RequestMapping("/mypage.do")
public String login(@RequestParam  List<String> name)  {
```

2.1 요청정보 받기

▶ **@PathVariable** uri 경로에 변수 → 기본타입변수

```
location.href ="users/hong"
```

```
@RequestMapping("/users/{id}")
public String process(@PathVariable String id)  {
```

▶ **@RequestBody** json 문자열 → 커맨드객체

```
$.ajax({
    contentType : "json",
    data : json.stringify(  {id : id}  )
})
```

```
@RequestMapping("/users")
public String process(@RequestBody UserVO vo)  {
```

2.1 요청정보 받기

▶ **첨부파일 업로드** → @RequestPart MultipartFile

```
<form action ="mypage.do"
        method="post"
        enctype="multipart/form-data">
    <input name="id" value="hong">
    <input name= "pic" type="file">
</form>
```

```
class UserVO {
  String id;
 MultipartFile pic;
}
```

```
@RequestMapping("/mypage.do")
public String login(UserVO vo)  {
```

```
class UserVO {
  String id;
}
```

```
@RequestMapping("/mypage.do")
public String login(UserVO vo ,
              @RequestPart MultipartFile pic)  {

}
```

7

2.1 요청정보 받기

▶ @ModelAttribute

클라이언트 요청: mypage.do?id=hong

컨트롤러

```
@RequestMapping("/mypage.do")
public String login(Model model, UserVO vo)  {
  model.addAttribute("profile", service.getUser(vo.getId()));
     return "mypage";
}
```

뷰 : mypage.jsp

```
<div>
${param.id}
${userVO.id}

${profile.id}
</div>
```

```
@RequestMapping("/mypage.do")
public String login(Model model ,
            @ModelAttribute("user") UserVO vo)  {
```

```
${param.id}
${user.id}
```

2.1 요청정보 받기

- @DateTimeFormat
  - 입력값 : 2022/05/10

```java
import java.util.date;
class UserVO {
  String id;
  private Date wdate;
}
```

  - 입력값 : 2022-05-10

```java
import java.util.date;
class UserVO {
  String id;
  @DateTimeFormat(pattern = "yyyy-MM-dd")
  private Date wdate;
}
```

```java
import java.sql.date;
class UserVO {
  String id;
  private Date wdate;
}
```

2.2 리턴타입

▶ **리턴 타입**

1. String
2. ModelAndView
3. void
4. 객체타입

```
@RequestMapping("/mypage")
public String login( )  {
    return "mypage";
}
```

```
@RequestMapping("/mypage")
public ModelAndView login( )  {
    return new ModelAndView("mypage");
}
```

```
@RequestMapping ("/mypage")
public void login( )  {      }
```

WEB-INF/views/mypage.jsp

```
@RequestMapping ("/mypage")
@ResponseBody
public UserVO login( )  {
    return vo;
}
```

{"name":"park", "age":20}

2.3 페이지이동

메서드 반환값 → `return "home";`

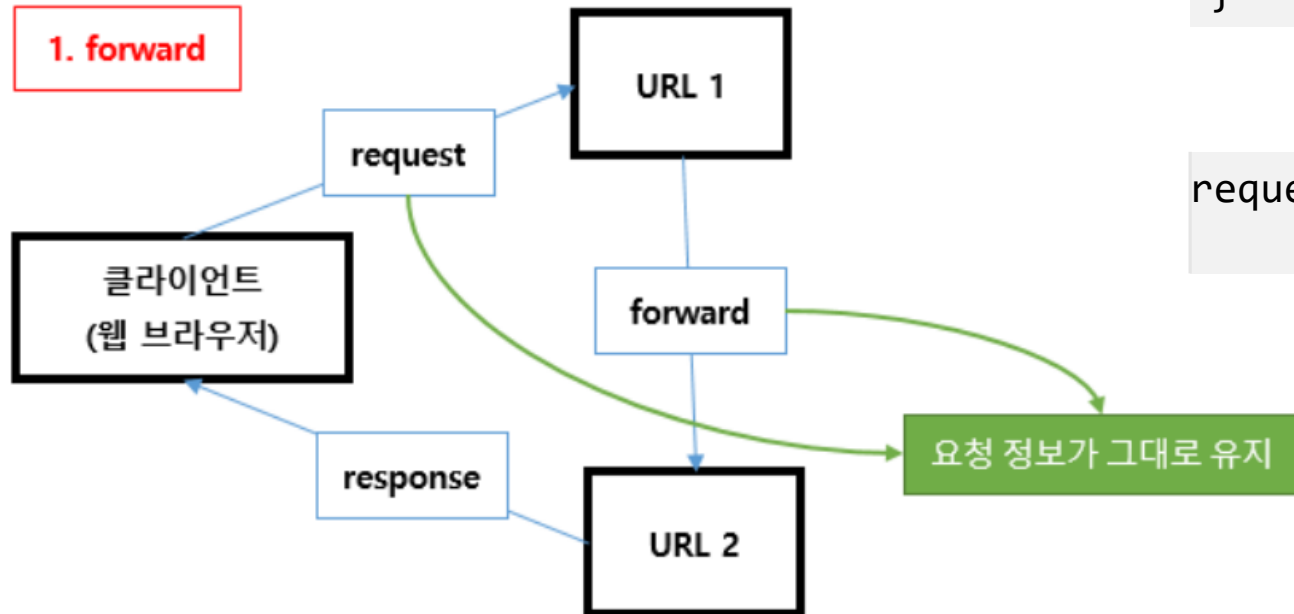JSP 파일 → `/WEB-INF/views/home.jsp`

사용자 응답 브라우저 →

```
servlet-context.xml    J HomeController.java    Home ✕
⇦ ⇨ ■ 🔍  http://localhost:8090/lec14/
```

Hello world!

The time on the server is 2018? 2? 3? (?) ?? 3? 38? 28?.

2.3 페이지이동

```
<beans:bean class= "...view.InternalResourceViewResolver">
    <beans:property name= "prefix" value="/WEB-INF/views/" />
    <beans:property name= "suffix" value=".jsp" />
</beans:bean>
```

뷰 :   /WEB-INF/views/users/login.jsp

```
@Controller
public class LoginController{
    @RequestMapping("/login.do")
    public String login()  {
        return "users/login";
    }
}
```
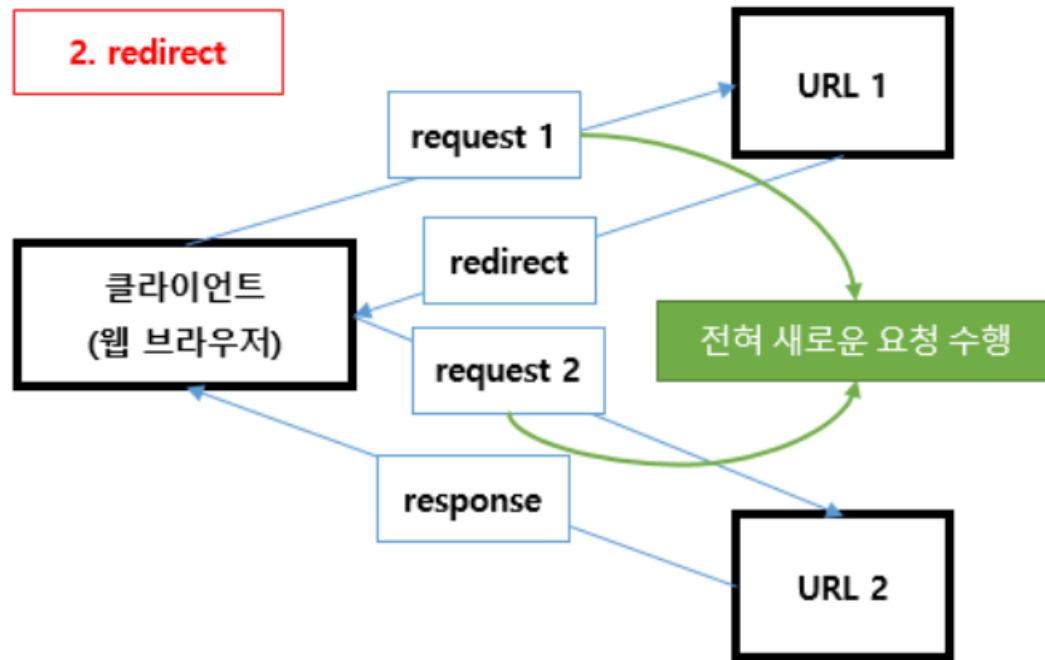
2.3 페이지이동

- forward


```
@RequestMapping("/mypage.do")
public String insert() {
    return "list";
}
```

```
request.getRequestDispatcher("/list.jsp")
        .forward(request, response);
```

2.3 페이지이동

- sendRedirect


```
@RequestMapping("/mypage.do")
public String insert() {
    return "redirect:list.do";
}
```

```
response.sendRedirect("main.jsp");
```

2.4 응답결과 보내기 – forward, Model

컨트롤러

```
@RequestMapping("/mypage.do")
public String login(Model model, UserVO vo)  {
    model.addAttribute("profile", service.getUser(vo.getId()));
    return "mypage";
}
```

뷰 : mypage.jsp

```
<div>
${profile.id}
</div>
```

2.4 응답결과 보내기 – redirect, RedirectAttributes

▶ RedirectAttributes : redirect 될 때 데이타가 여러 개인 경우에 유용

컨트롤러

```java
@PostMapping("/insert.do")
public String insert(BoardVO vo,
                RedirectAttributes rttr,
                @RequestParam String page) {
  service.insert(vo);
  rttr.addFlashAttribute("msg","등록완료");
  rttr.addAttribute("page", page);
  return "redirect:list.do";
}
```

redirect

```java
@RequestMapping("/list.do")
public String list(Model model,
                @RequestParam String page) {

  model.addAttribute("boards", service.select(page));

  Map<String, ?> flashMap =
      RequestContextUtils.getInputFlashMap(request);

  if(flashMap!=null) {
      System.out.println(flashMap.get("msg"));
  }

  return "list";
}
```

list.jsp

```html
<script type="text/javascript">
  var msg = '${msg}';
  if( msg !='') {
      alert("게시물이 등록되었습니다!");
  }
</script>
<body>
 <c:forEach items="${boards}" >
```

forward

2.5 세션 정보 조회