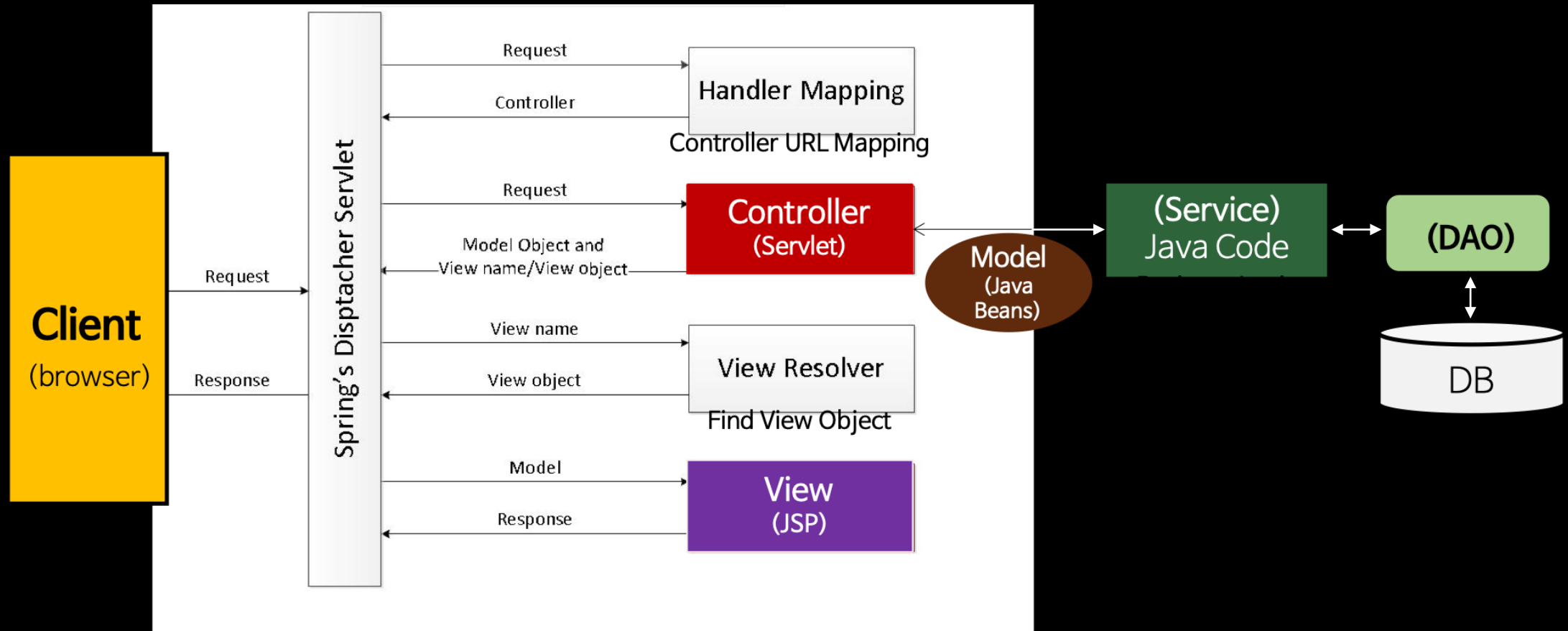


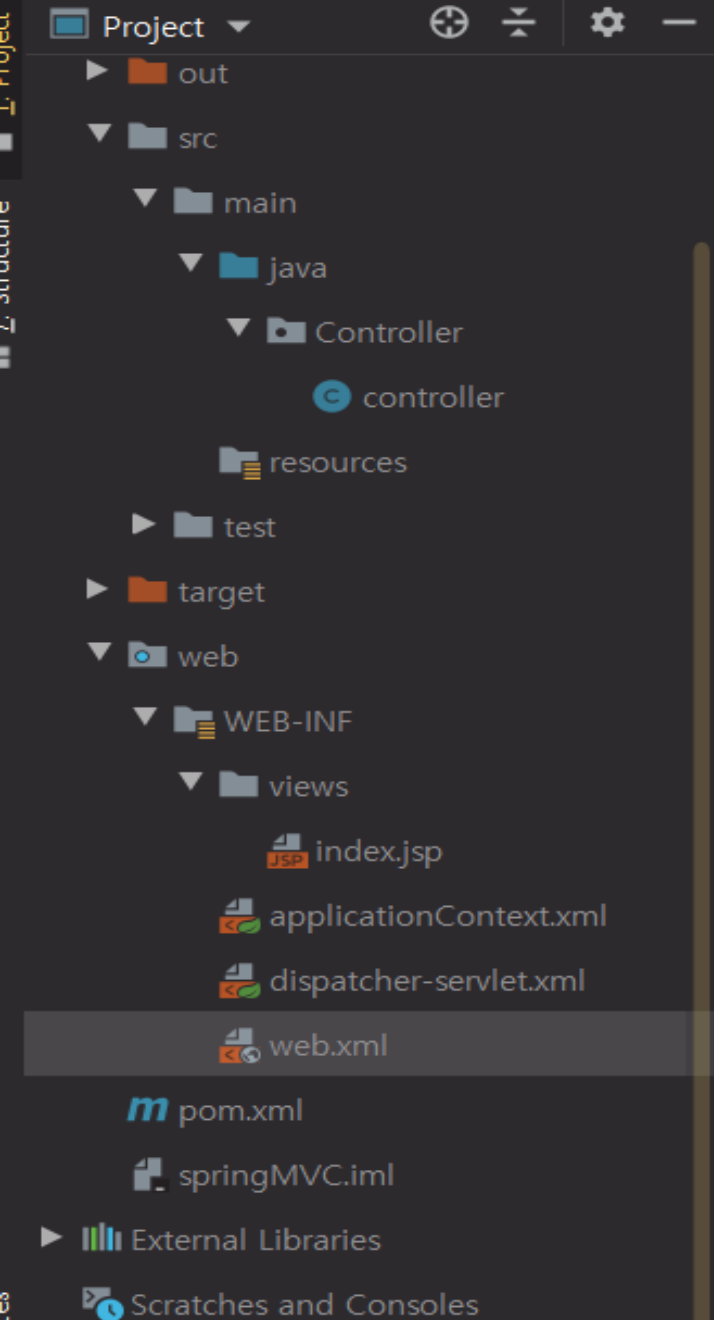
Spring MVC

MashUp 백엔드 9기 박미현

Spring MVC Architecture



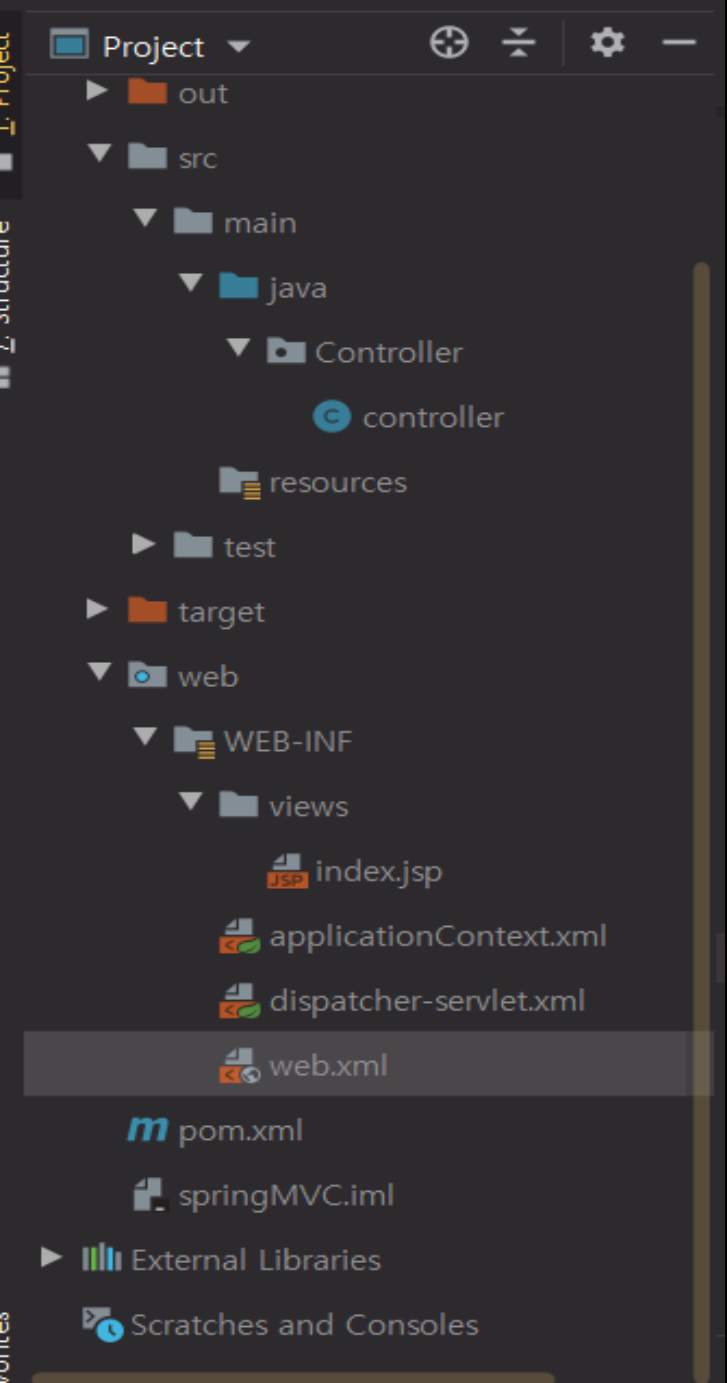
Spring MVC 흐름



```

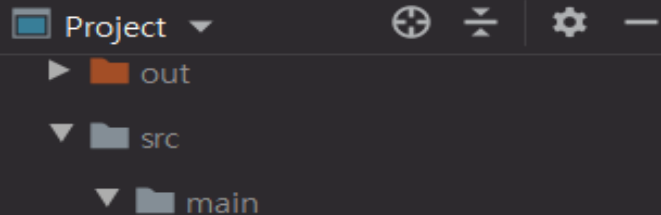
1  <?xml version="1.0" encoding="UTF-8"?>
2  <web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
3          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4          xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
5          version="4.0">
6      <context-param>
7          <param-name>contextConfigLocation</param-name>
8          <param-value>/WEB-INF/applicationContext.xml</param-value>
9      </context-param>
10     <listener>
11         <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
12     </listener>
13     <servlet>
14         <servlet-name>dispatcher</servlet-name>
15         <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
16         <load-on-startup>1</load-on-startup>
17     </servlet>
18     <servlet-mapping>
19         <servlet-name>dispatcher</servlet-name>
20         <url-pattern>/</url-pattern>
21     </servlet-mapping>
22 </web-app>
    
```

Spring MVC 흐름



```
controller.java x
1 package Controller;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.RequestMapping;
5
6 @Controller
7 public class controller {
8
9     @RequestMapping(value = "/")
10    public String test() { return "index"; }
11
12 }
13
14
```

Spring MVC 흐름



dispatcher-servlet.xml

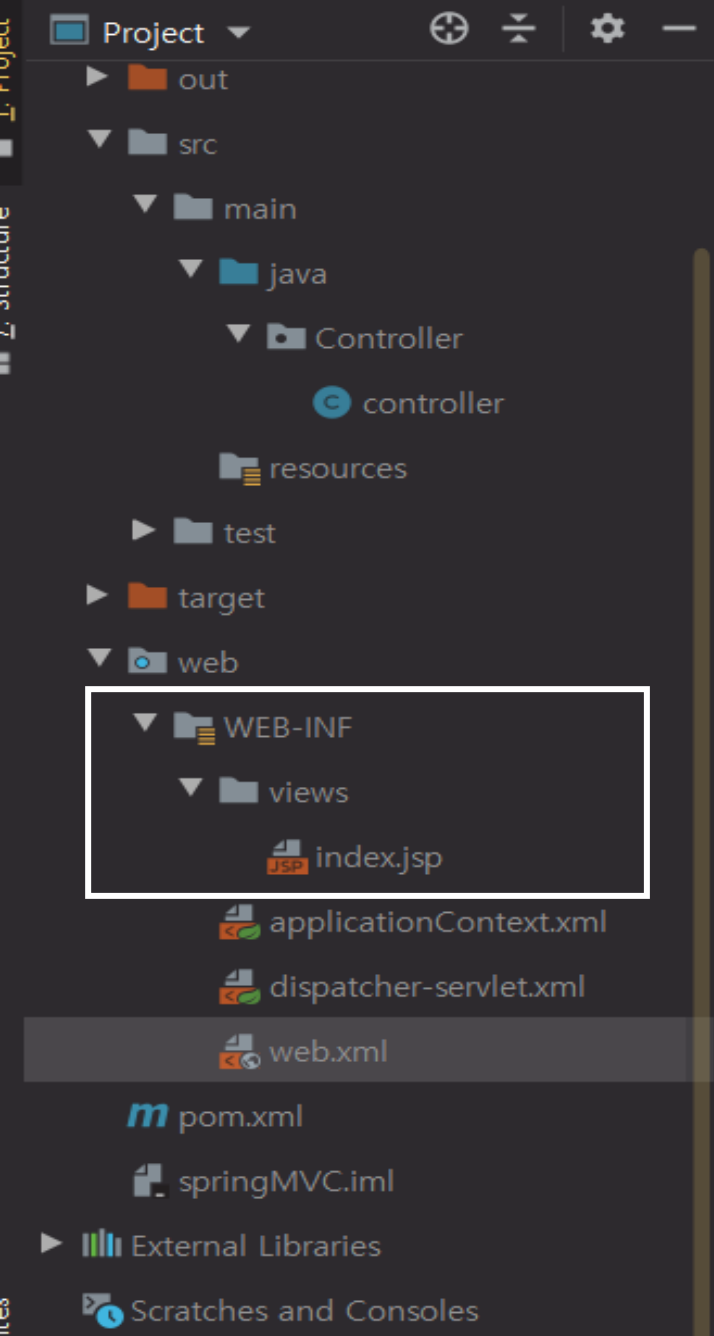
```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xmlns:context="http://www.springframework.org/schema/context"
5       xmlns:mvc="http://www.springframework.org/schema/mvc"
6       xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd
7                          http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.0.xsd
8                          http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc-4.0.xsd">
9
10    <mvc:annotation-driven/> <!--Annotation 활성화-->
11    <context:component-scan base-package="Controller"/> <!--Component 패키지 지정-->
12
13    <bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
14        <property name="prefix" value="/WEB-INF/views/"></property>
15        <property name="suffix" value=".jsp"></property>
16    </bean>
17 </beans>
```

m

s

External Libraries

Scratches and Consoles



Spring MVC 흐름

```
index.jsp x
1  <%--
2      Created by IntelliJ IDEA.
3      User: hyoni
4      Date: 2020-05-16
5      Time: 오전 12:20
6      To change this template use File | Settings | File Templates.
7  --%>
8  <%@ page contentType="text/html; charset=UTF-8" language="java" %>
9  <html>
10 <head>
11     <title>Title</title>
12 </head>
13 <body>
14     Hello Spring World
15 </body>
16 </html>
```

```
@GetMapping("/")  
public String index(Model model, @LoginUser SessionUser user) {  
    model.addAttribute( attributeName: "posts", postsService.findAllDesc());  
  
    if (user != null) {  
        model.addAttribute( attributeName: "userName", user.getName());  
    }  
    return "index";  
}
```

indexController.java

Model

import ...

@RequiredArgsConstructor

@Controller

public class IndexController {

private final PostsService postsService;

@GetMapping("/")

public String index(Model model, @LoginUser SessionUser user) {

model.addAttribute(attributeName: "posts", postsService.findAllDesc());

if (user != null) {

model.addAttribute(attributeName: "userName", user.getName());

}

return "index";

}

@GetMapping("/posts/save")

public String postsSave() { return "posts-save"; }

@GetMapping("/posts/update/{id}")

public String postsUpdate(@PathVariable Long id, Model model) {

PostsResponseDto dto = postsService.findById(id);

model.addAttribute(attributeName: "post", dto);

return "posts-update";

}

}

Controller

@RequiredArgsConstructor

@Service

public class PostsService {

private final PostsRepository postsRepository;

@Transactional(readOnly = true)

public List<PostsListResponseDto> findAllDesc() {

return postsRepository.findAllDesc().stream()

.map(PostsListResponseDto::new)

.collect(Collectors.toList());

}

}

View

- 템플릿 엔진 (Thymeleaf, Groovy, Freemarker, etc)
- JSP
- JSTL

Spring MVC 이점

```
public class HelloServlet extends HttpServlet {  
    private HelloService helloService = new HelloService();  
  
    @Override  
    protected void doGet(HttpServletRequest request, HttpServletResponse response) {  
        //business(domain) logic  
        helloService.doStuff(request.getHelloInfo());  
        //view logic  
        response.getWriter().println("<html>");  
        response.getWriter().println("<head>");  
        response.getWriter().println("<body>");  
        response.getWriter().println("<h1>Hello Servlet</h1>");  
        response.getWriter().println("</body>");  
        response.getWriter().println("</head>");  
        response.getWriter().println("</html>");  
    }  
}
```