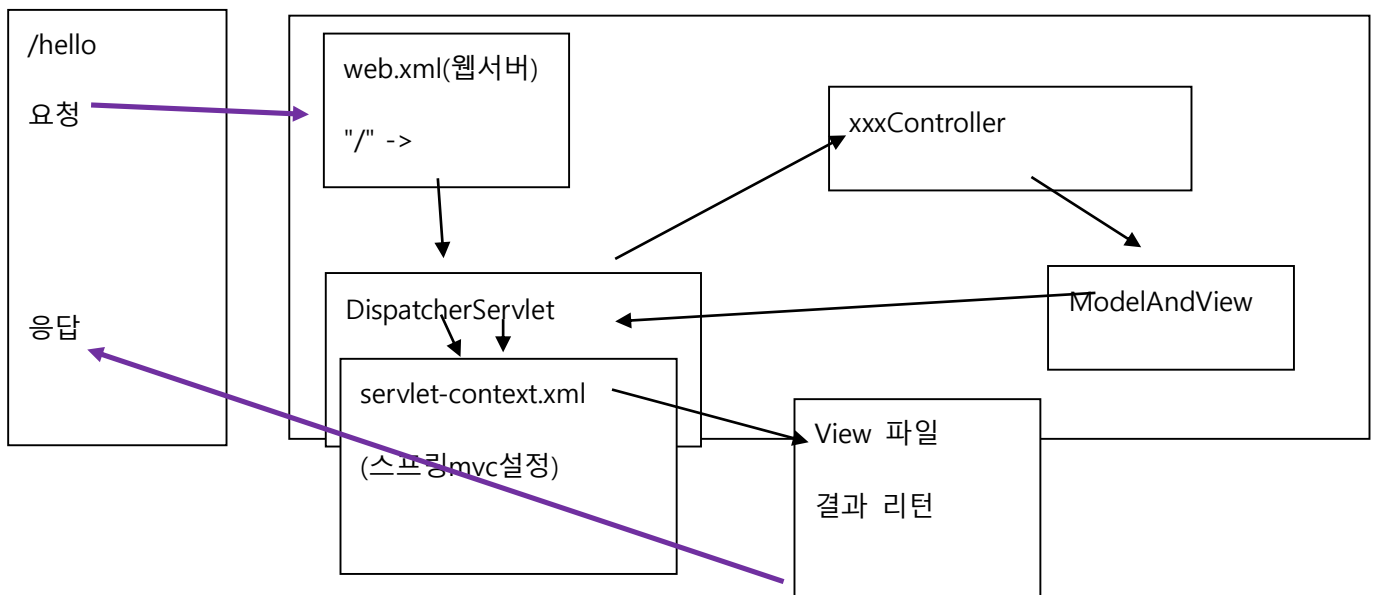
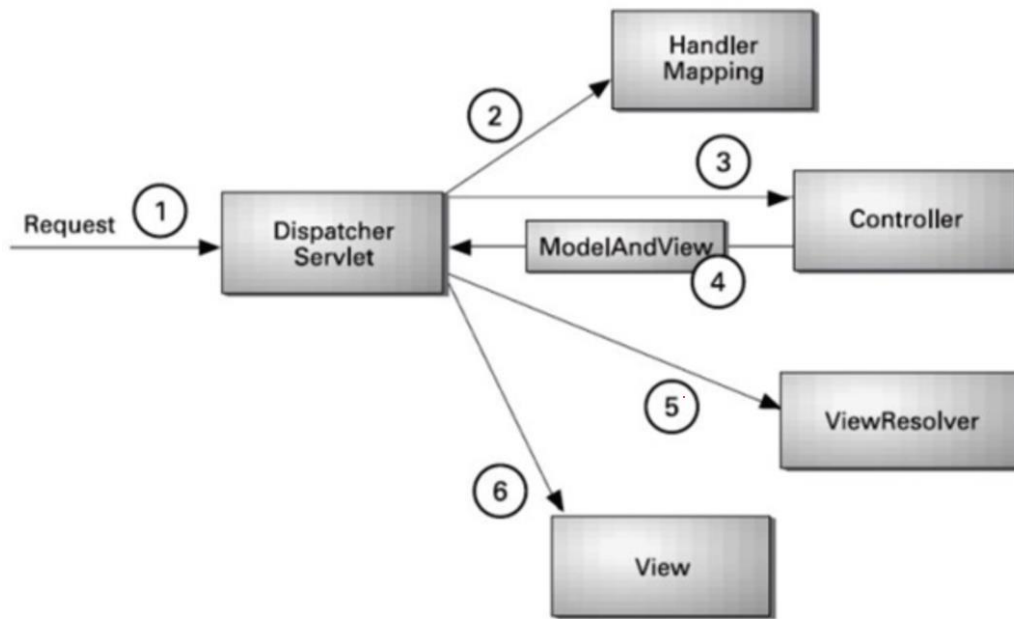
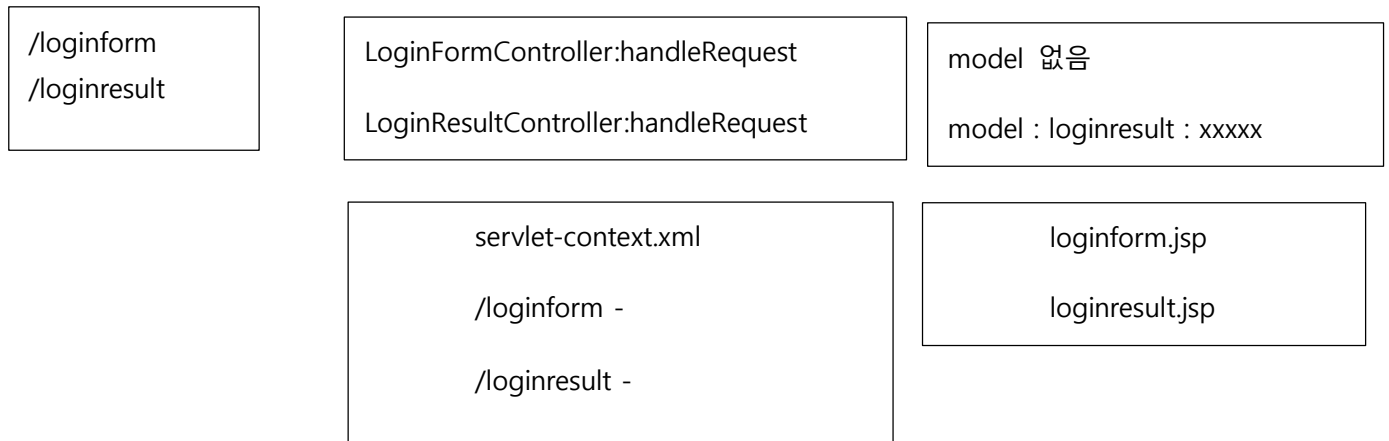


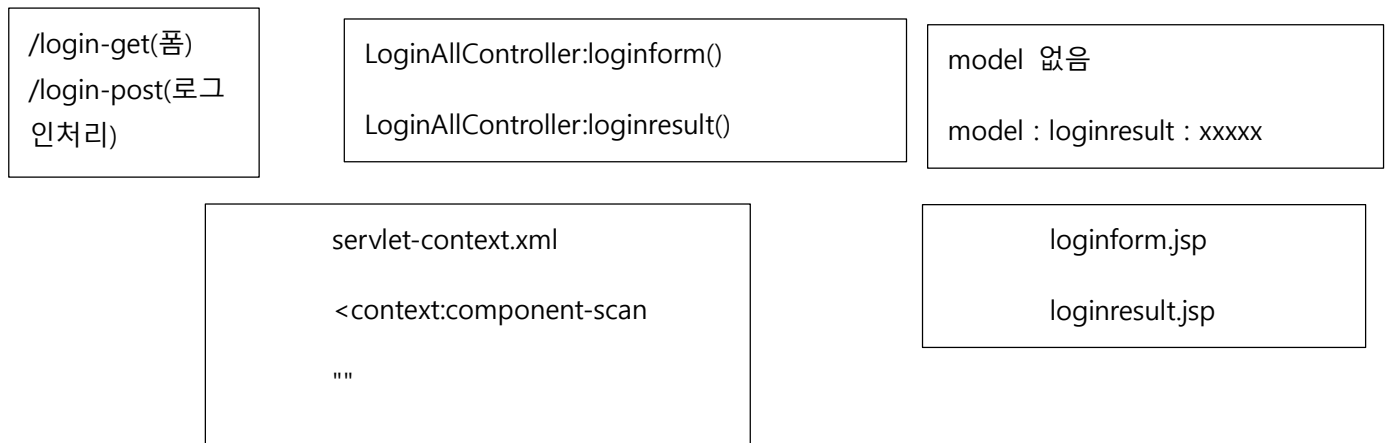
Understanding the flow of Spring Web MVC



- spring mvc 로그인처리(XML방식)



- spring mvc 로그인처리(ANNOTATION방식)



@Controller	컨트롤러 클래스 위 선언 @Component @Repository @Service @Controller
@RequestMapping("/a") – get 그리고 post	메소드 위 선언
@GetMapping("/a") - get	메소드 위 선언
@PostMapping("/a") - post	메소드 위 선언
@RequestParam("요청파라미터명") String 변수	매핑 메소드 매개변수앞 선언
@ModelAttribute("모델명") xxxDTO dto	매핑 메소드 매개변수앞 선언

- 컨트롤러 메소드 매개변수

서블릿 api	HttpServletRequest, HttpSession 등
자바 데이터 타입의 변수	요청 파라미터 이름과 동일 이름 변수 자동 저장. 타입 변환 자동. @RequestParam("요청파라미터이름")
자바 객체	@ModelAttribute("모델명") LoginVO

- 컨트롤러 메소드 리턴타입

ModelAndView	모델과 뷰 설정
String	"뷰이름" 설정(모델 없거나 @ModelAttribute("dto") 대신)
	"redirect:컨트롤러매핑url메소드호출"
void	뷰이름은 url이름 동일 뷰

19장 ioc di

21장 mvc

26장 = spring mvc+ di annotation

26장 annotation

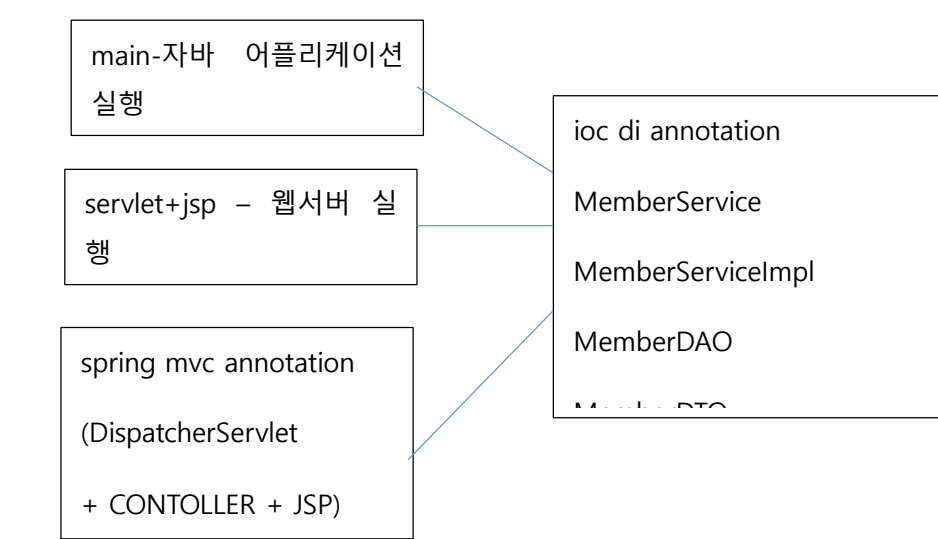
ioc di	@Component-클래스위 @Service-클래스위 @Repository-클래스위 @Autowired-변수 @Qualifier-변수
mvc	@Controller-컨트롤러클래스위 @RequestMapping-메소드위나 클래스위 @GetMapping @PostMapping @RequestParam-메소드매개변수앞 @ModelAttribute-메소드 객체타입 매개변수앞

- STS4 TOOL SPRING MVC 프로젝트의 XML파일들

pom.xml	프로젝트 root위치 spring 다양한 라이브러리 필요 mvc + di + jstl + servlet + jsp *.jar 파일들 독립적 수동 다운로드 관리 힘들다 자동 다운로드 관리 툴 이용 ==> maven 설정내용 알려주는 파일 스프링 필요 라이브러리 다운로드 설정 추가
web.xml	src/main/webapp/WEB-INF 위치 웹서버 설정 "/" --> DispatcherServlet 전송(servlet-context.xml설정 같이 읽어라) "/" -post - utf-8 인코딩 처리 --> DispatcherServlet

	+ 스프링설정파일 추가 알려주는 파일 member.xml 파일 추가
servlet-context.xml	src/main/webapp/WEB-INF/spring/appServlet 위치 spring mvc 설정 파일
스프링설정파일 추가	경로 / *.xml src/main/java/annotation/service/member.xml ==> classpath: annotation/service/member.xml

- spring mvc와 비지니스 로직 처리 연동



- spring mvc view에서 이미지, js, css 파일 사용

*.java(컨트롤러 service dto dao class)	src/main/java/패키지명.xxx
jsp	src/main/webapp/WEB-INF/views/*.jsp
html http://ip:port/mvc/resources/*.html css js gif jpg	src/main/webapp/resources/*.html

19 21 26장

ioc/di
mvc

- db
spring mvc + mybatis

자바 jdbc jdbc 기능	java.sql.*
spring jdbc 기능	22장
mybatis 프레임워크	23, 24장 db 연동 기능만. spring mvc / di / 다른프레임워크통합 - mybatis연동

hibernate 프레임워크	
	25 db 트랜잭션 기능 mysql 트랜잭션처리 자동

22장 – 25장

jdbc + mysql	mybatis + mysql
<pre> Class.forName("드라이버") Connection conn = DriverManager.getConnection("...", "", "") PreparedStatement pt = conn.prepareStatement("insert into emp values(?,?,?.."); pt.set타입(1, 값); ... pt.executeUpdate(); PreparedStatement pt = conn.prepareStatement("select * from emp"); ResultSet rs = pt.executeQuery(); EmpDTO = xecuteQuery(); rs-->XXDTO-->ArrayList </pre>	<pre> con.xml <dataSource <driverName <jdbcurl <user <password sql.xml <select id="allemp" resultType="EmpDTO" > select * from emp </select> 자바소스 1. con.xml 읽어와라 2. sql.xml 읽어와라 3.allemp 실행-EmpDTO 리턴 </pre>

mybatis framework

SPRING FRMEWORK

- 자바 객체 생성-관리(빈 singleton), AOP, MVC 등 도와주는 통합 기능의 자바 FRAMEWORK

Mybatis

자바 작성. 대표적 특징은 SQL문을 분리하는 SQL Mapper 프레임워크.

- JDBC 의 단점

DB 연결

SQL 전송

SQL 결과 리턴

DB 연결 해제

예외처리

1> 코드 반복과 사용 객체 해제 빈번- 재사용

2> 자바 언어와 SQL 다른 언어이다.- 분리

3> JDBC 모든 테이블 SELECT ResultSet -XXDTO 객체 변경 개발자 몫- 자동매핑

- MYBATIS 환경설정

1> SPRING MVC + MAVEN TOOL 관리

2> mvnrepository.com

설정태그==>pom.xml 붙여넣기

```
<!-- https://mvnrepository.com/artifact/org.mybatis/mybatis -->
```

```
<dependency>
```

```
    <groupId>org.mybatis</groupId>
```

```
    <artifactId>mybatis</artifactId>
```

```
    <version>3.5.6</version>
```

```
</dependency>
```

+

MYBATIS 태그

spring+mybatis 연동 태그

mysql driver 태그

2> 사이트 - 다운로드 - WEBCONTENT\WLIB\JSTL.JAR 위치 (개발자 수동)

- mybatis

자바 main insert / delete / update – dml 실행 – 임시 버퍼 저장 . 즉각 db 반영하지 않고 종료 SqlSession session = factory.openSession(); SqlSession session = factory.openSession(false); insert / delete / update – dml 실행 – 즉각 db 반영 SqlSession session = factory.openSession(true);	db연결.xml mybatis-config.xml	sql정의.xml sql-mapping.xml
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------	------------------------------

