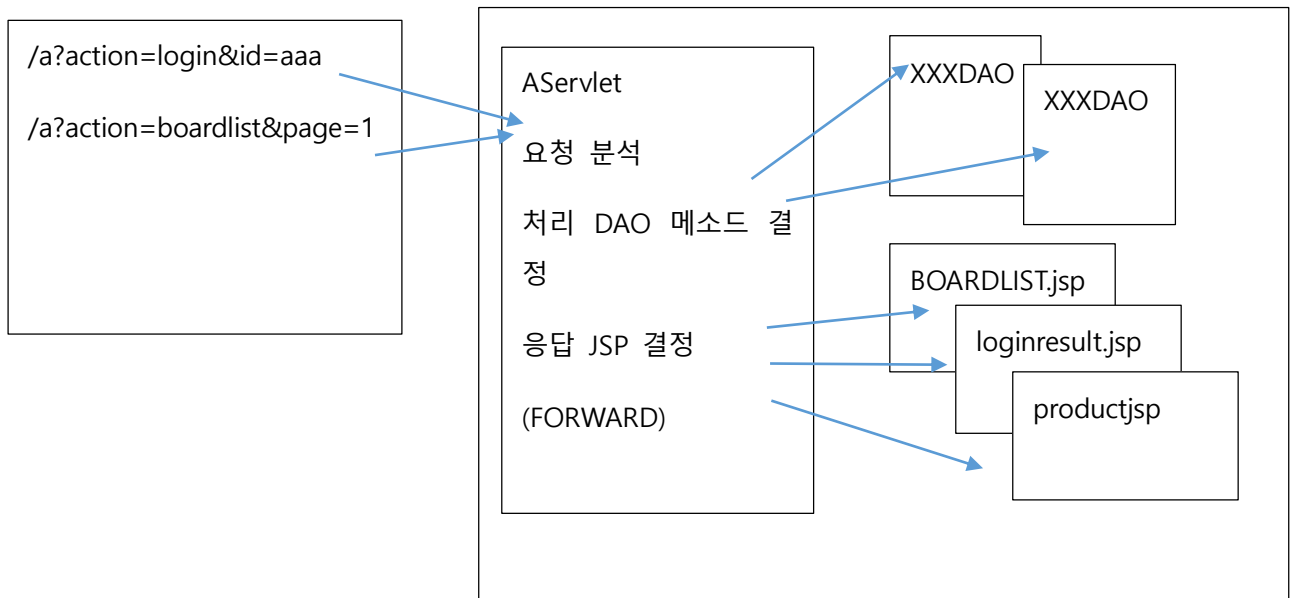


17장 모델 2 방식 개발- spring

웹서버 개발 형태=형식 = 모델1 방식=흐름 복잡하다



BoardServlet / boardlist.jsp / BoardDAO

BoardDTO

요청 파라미터 추출 - 분석 - 요청 따른 처리 -
결과 응답

서블릿 - controller

xxxdao, xxxDTO - model

JSP - VIEW

MVC방식 구현= MODEL 2

SERVLET + JSP MVC = 권고

SPRING MVC = 강제

18장 스프링

환경설정 복잡

개념, 용어 어렵다

spring framework =

반제품(내 + SPRING 제공) 구현 개발 + 규칙 + 나머지만 내가 개발 = 작업 자동화 도움

프로그램(다수 협업-규칙 약속) 구현 개발 = 기존 개발 수정 단축
기존 개발 수정 --->

BoardDAO 게시물 관련 기능 메소드 구현 클래스

+

MAIN / SERVLET

틀 정형 프로그램 패턴 (구현 이미-파라미터) + ????

==>

workspace_spring - spring/pom.xml - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- spring
 - src/main/java
 - multi/campus/spring
 - HomeController.java
 - src/main/resources
 - JRE System Library [JavaSE-1.8]
 - Maven Dependencies
 - Apache Tomcat v9.0 [Apache Tomcat v9.0]
 - src/test/java
 - src/test/resources
 - src
 - main
 - webapp
 - resources
 - WEB-INF
 - classes
 - spring
 - views
 - web.xml
 - test
 - target
 - pom.xml

Boot Dashboard

Type tags, projects, or working set names to match (incl. * and ? wildcards)

local

workspace_web - jsp/WebContent/eli/jstest3.jsp - Eclipse IDE

File Edit Navigate Search Project Run Window Help

Project Explorer

 - src
 - board
 - session
 - Libraries
 - Apache Tomcat v9.0 [Apache Tomcat v9.0]
 - JRE System Library [jdk9]
 - Web App Libraries
 - build
 - WebContent
 - actioninclude
 - bean
 - board
 - cart1
 - error
 - include
 - META-INF
 - session
 - WEB-INF
 - web.xml
 - jst-1.2.jar
 - first.jsp
 - loginprocess.jsp
 - membervar.jsp

Overview | Dependencies, Dep

Problems

0 errors, 3 warnings, 2 others

0 items selected

Description	Resource	Path	Location	Type
Warnings (3 items)				
Infos (2 items)				

31 pageContext.se
32 %>
33
34 <c:forEach ite
35 \${
36 </
37
38 <
39 \$
40
41
42 <
43
44
45 %>
46
47
48
49
50
51

받는 사람: 박소현 (DM) 파일 ...
최근에 메시지 입력...

Tomcat v9.0 Server at local

김승현
이재영

오후 3:56
2021-04-13

spring DI= DEPENDENCY INJECTION 강제화

IOC = INVERSION OF CONTROL – 제어 역행 – 역전 제어

자바클래스 내부 객체 생성 권한 없다 – 스프링 전달 객체들 사용

의존성 – 자바객체들

spring 의존성 결정

스프링은 ioc 기능을 가지며 ioc 구현하기 위해서 di 사용한다

```
class A{  
    B b1 ;  
    A(B b1){  
        this.b1 = b1;  
    }  
    setB(B b1){==> B 객체+B 하위클래스 객체  
        this.b1 = b1;  
    }  
}
```

```
class Main{  
    ...A a1 = new A();  
    a1.setB(new B())
```

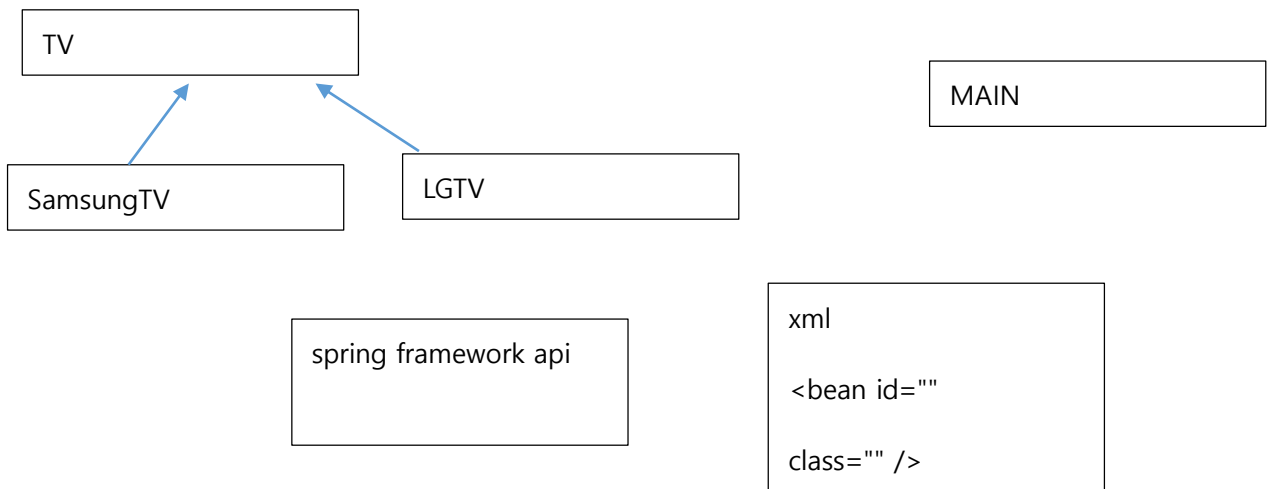
interface TV void powerOn(); void powerOff(); void soundUp(); void soundDown();	
class SamsungTV implements TV void powerOn(){ } void powerOff(){} void soundUp(){} void soundDown(){} 	class LGTV implements TV void powerOn(){.....} void powerOff(){} void soundUp(){} void soundDown(){}
TV tv = TVFactory.getTV("") ==> 의존 객체 수정시 같이 수정 부분 많으면 커플링 높다- 낮추기 위해	class TVFactory static TV getTV(String order)

interface BoardDAOInter { }	
class BoardDAOOracle implements BoardDAOInter { 구현-오라클 저장 / 조회 }	class BoardDAOMySQL implements BoardDAOInter { 구현-mYSQL 저장 / 조회 }

```

class A{
BoardDAOInter i ;
A(BoardDAOInter i ){
    this.i = i;
}
i.getBoardList()

```



bean= 자바객체

<bean

<property name="a" value="A 객체"

<constructor-arg

-

MemberMain

키보드입력

id : xxxx

암호 : xxxx

MemberService

MemberDAO dao;

setMemberDAO();

register();

login();

1> register / login 두가지 기능 필수=인터페이스

2> dao는 db sql 실행단위 메소드

3> 서비스는 실제서비스기능 단위의 메소드

MemberDAO

MemberVO vo;

setMemberVO(MemberVO)

selectMember()

insertMember()

MemberVO

id

pw

DAO = DATA

메소드 1개

메소드

register(){ 기능 단위

dao

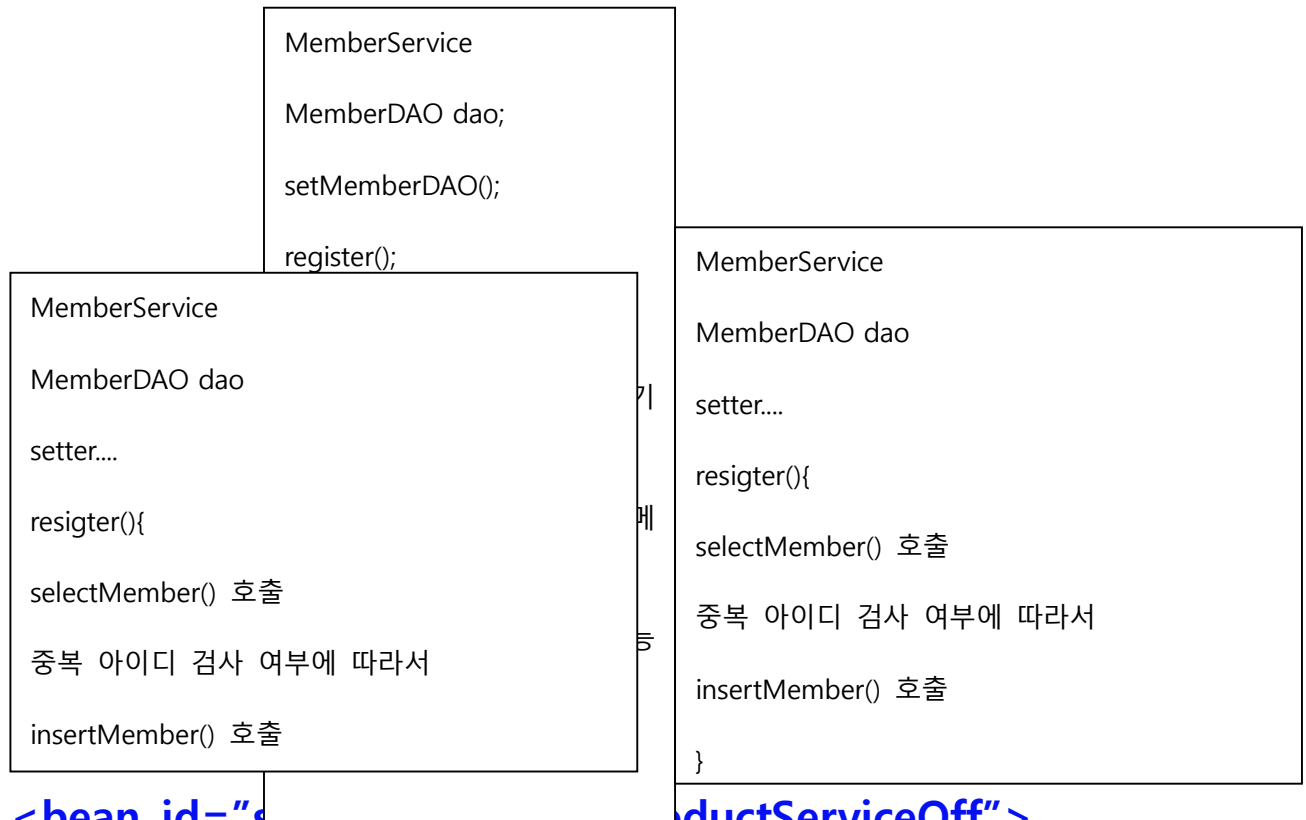
if(!selectMember()){

dao insertMember();

}

}

SERVLET,JSP -> SERVICE -> DAO VO



<bean id="service" class="...ProductServiceOff">

1> IOC 개념적 용어

2> DI 구현

<bean

<property -> setter di

<constructor -> 생성자 di

main - service : dao : vo

xml 파일 태그 설정

+

spring annotation

MemberMain

키보드입력

id : xxxx

암호 : xxxx

MemberService

register();

login();

@Service("service")

class MemberServiceImpl

@Autowired

MemberDAO dao;

@Repository("dao")

class MemberDAO

@Autowired

MemberVO vo;

selectMember()

insertMember()

@Component("vo")

class MemberVO

id

pw

@Service	서비스클래스위
@Repository	dao클래스위
@Component	모든클래스위
@Autowired	변수위, 메소드위 다른 객체 생성 자동 주입 setter injection const...injection
@Qualifier("이름")	변수위 @Autowired 함께 2개 이상 객체 생성시 지정 한 객체 선택

<bean id="" class="...ServiceImpl" />

new ServiceImpl();

xml	@ 26장
xml 파일 모든 설정 이해하기 쉽다 bean 생성 코드는 늘어간다 bean 수정-저장	자바소스 내부 설정 위치 정해져있다 bean 태그는 늘어나지 않는다. 수정 자바 소스만 저장 한번에 모든 설정 이해하기 힘들다 단순 간결한 표현

	<p>xml 파일 내부</p> <pre><context:component-scan base-package="패키지명" ></pre>
--	---

DAO = DATA ACCESS OBJECT

메소드 1개 sql 실행 단위 = 1개 메소드

register(){ 기능 단위

IOC + DI

3> AOP

4> MVC

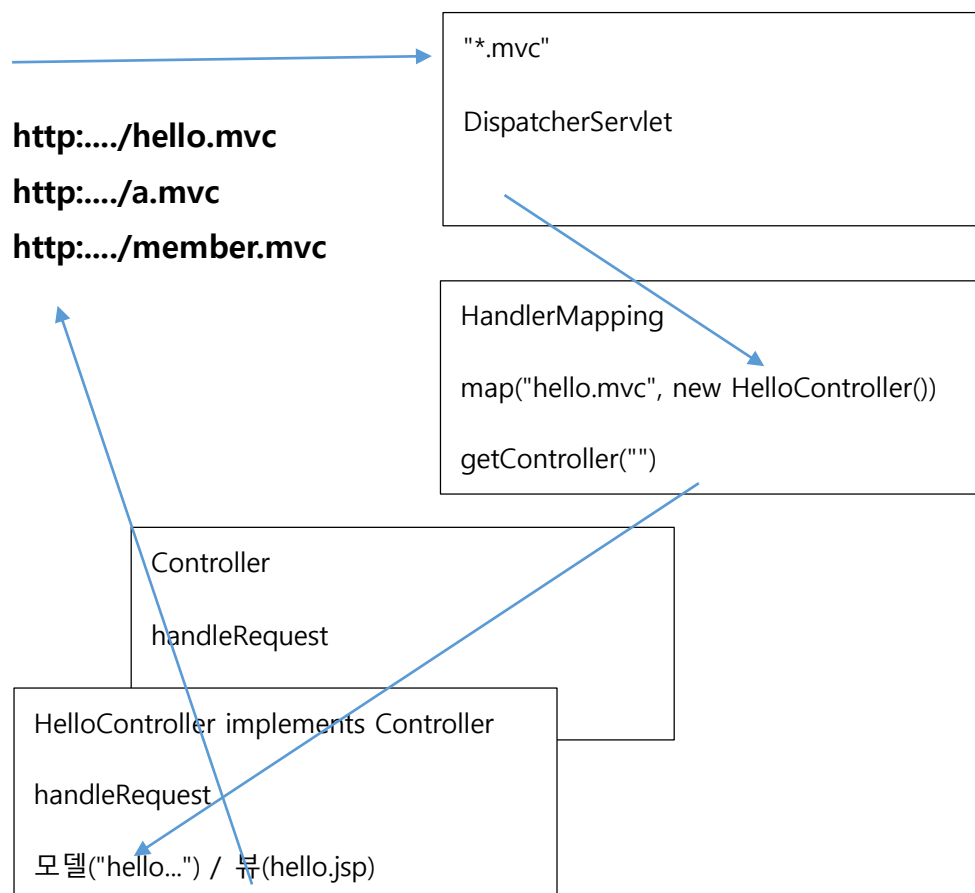
- MVC 실제 구현(스프링 API 처럼 ..)

DispatcherServlet -> 모든 요청 (대표) controller

요청 hello.mvc - 처리 - 응답--> m v c

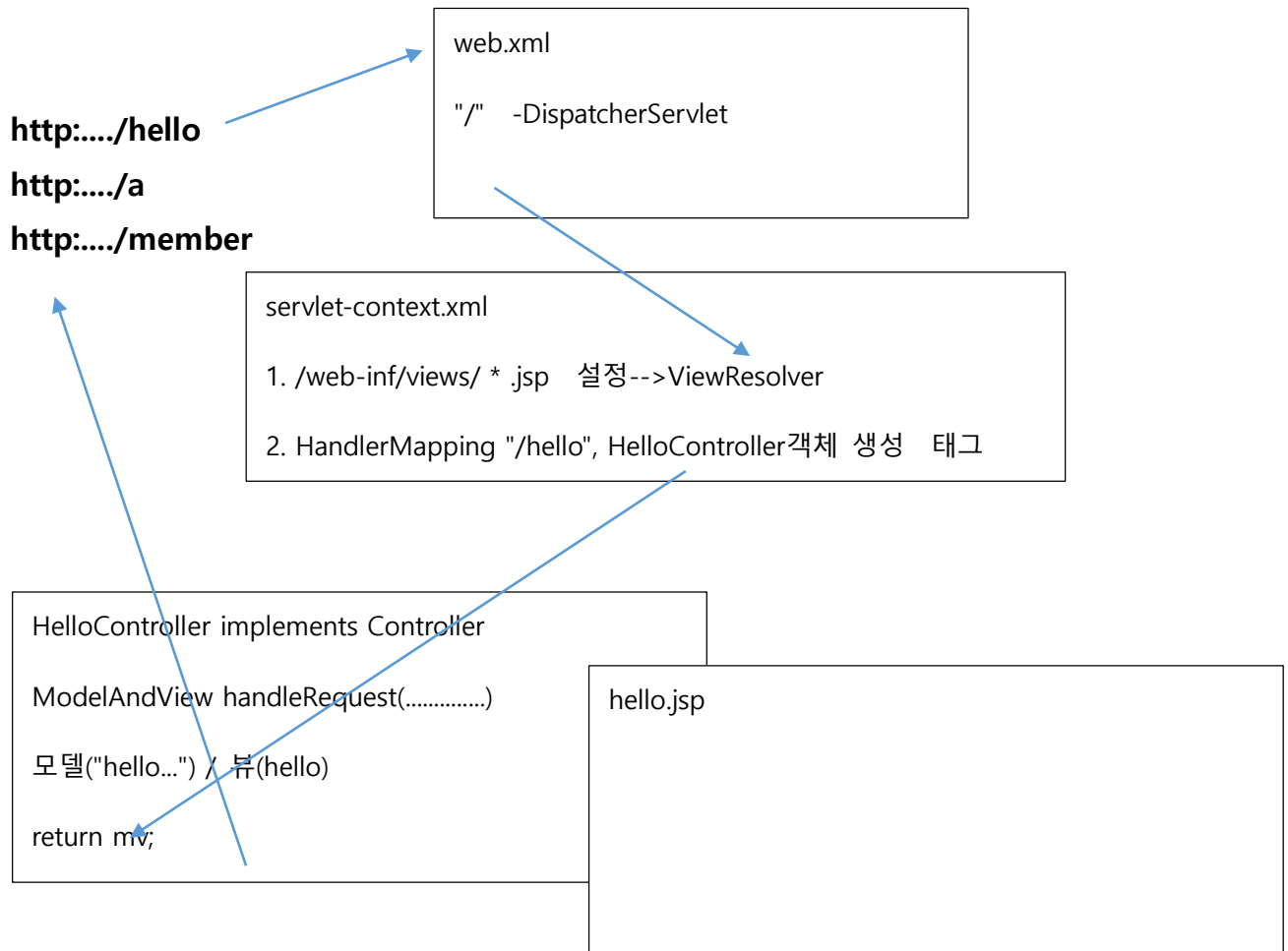
-xxxController(인터페이스 + 하위클래스)

HandlerMapping - (model , jsp)



- 스프링 방식 변경

1.XML



2.ANNOTATION

ioc / di

<bean --> @Service @Repository @Component

<property --> @Autowired @Qualifier

@Controller

@RequestMapping

@RequestParam

- 메소드 매개변수
- 메소드 리턴타입

5> JDBC