**11/30 김민식 OJT Day 21**

**Spring FrameWork 심화 교육 Day 03**

**16. MyBatis 개요**

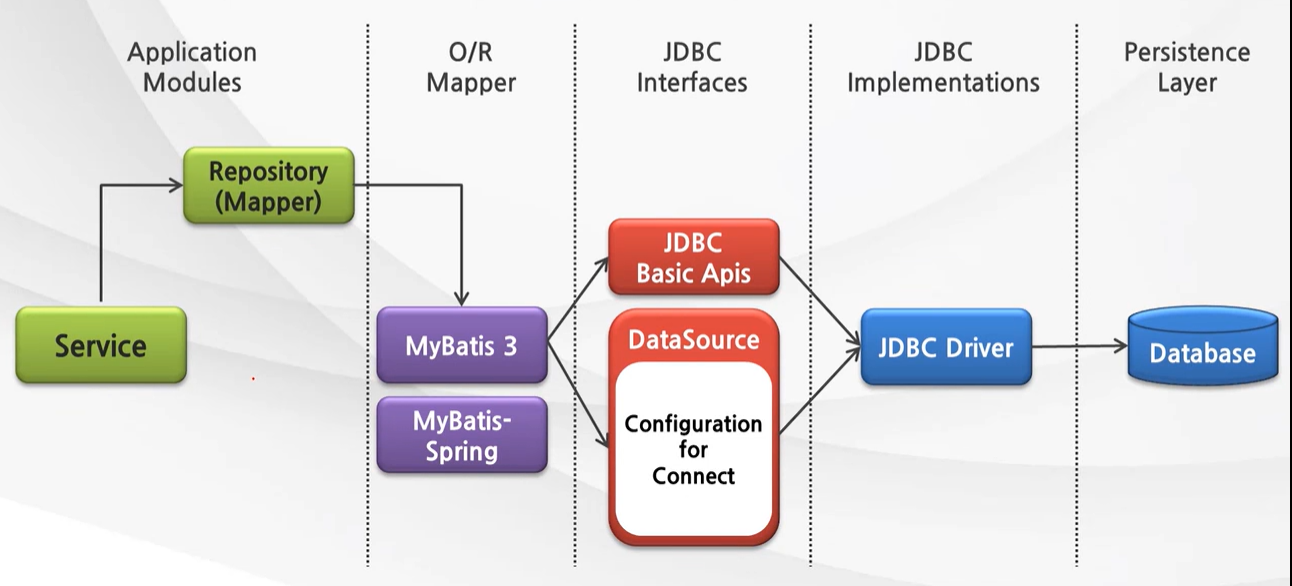
[Spring Framework Basic 16강 MyBatis 개요 | T아카데미 - YouTube](https://www.youtube.com/watch?v=9b5P4YiyqOY&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=16)

1. MyBatis 개요와 특징

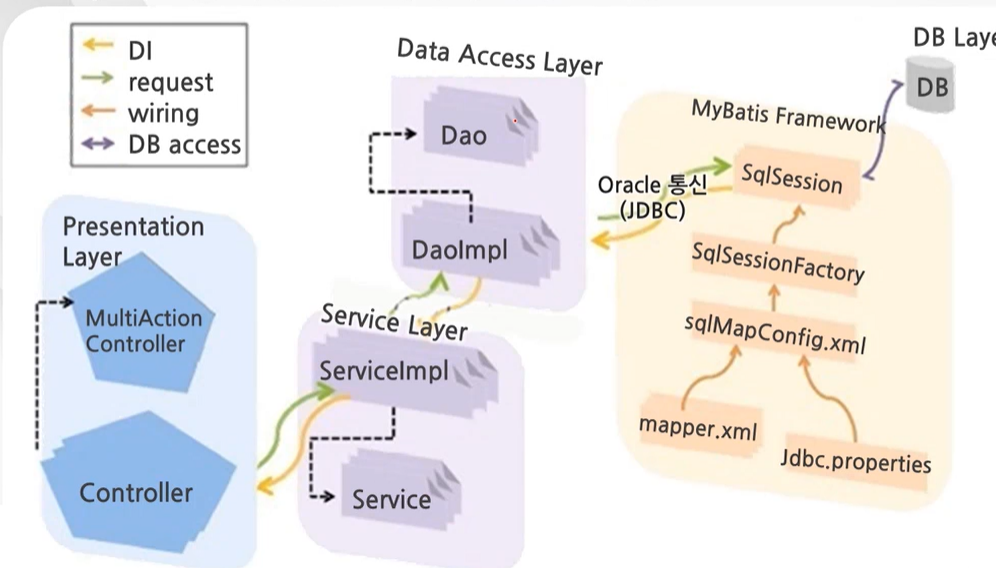
* MyBatis의 개요
  + ORM(Object Relational Mapping) 프레임워크
  + 자바 오브젝트와 SQL문 사이의 자동 Mapping 기능을 지원
* MyBatis 특징
  + 쉬운 접근성과 코드의 간결함
  + SQL문과 프로그래밍 코드의 분리
  + 다양한 프로그래밍 언어로 구현가능

1. MyBatis3과 MyBatis-Spring의 주요 컴포넌트

* MyBatis와 MyBatis-Spring을 사용한 DB 액세스 Architecture



* MyBatis를 사용하는 데이터 액세스 계층

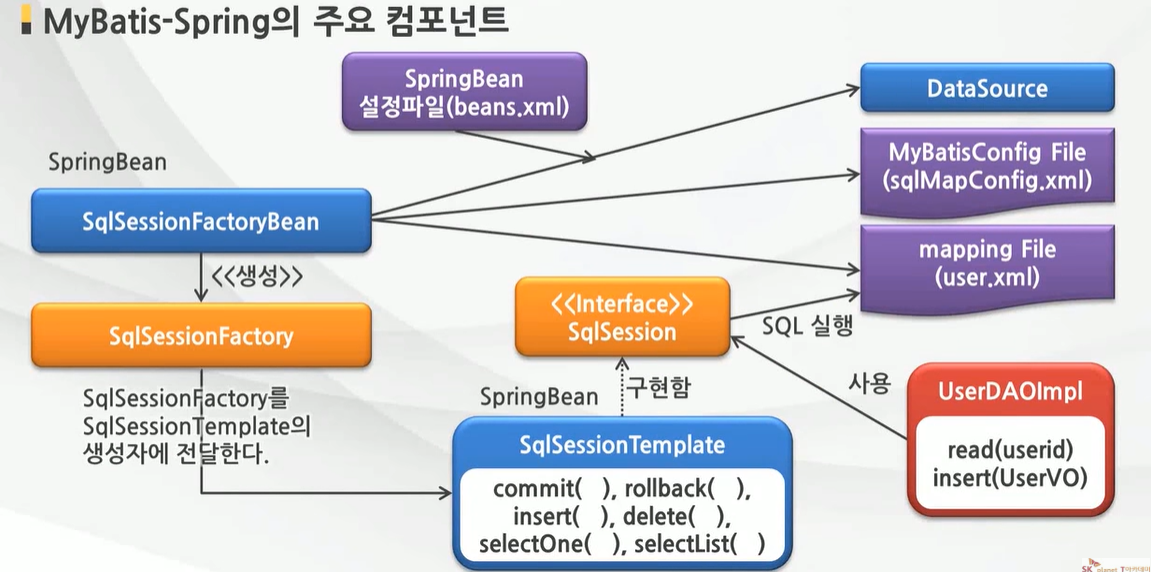


**17. MyBatis 애플리케이션 작성**

[Spring Framework Basic 17강 MyBatis 애플리케이션 작성(1) | T아카데미 - YouTube](https://www.youtube.com/watch?v=9pP0u0ybZSM&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=17)

1. MyBatis 파일 작성 및 MyBatis 설정

* MyBatis-Spring의 주요 컴포넌트



**18. MyBatis 애플리케이션 작성**

[Spring Framework Basic 18강 MyBatis 애플리케이션 작성(2) | T아카데미 - YouTube](https://www.youtube.com/watch?v=Wn_MNeNgG-Q&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=18)

1. Mapper 인터페이스 개념

* Mapper 인터페이스
  + SQL을 호출하기 위한 인터페이스

1. 여러 개의 Mapper 인터페이스 설정

* MapperScannerConfigurer의 사용
  + 지정한 패키지 아래 모든 인터페이스가 Mapper 인터페이스로 간주
  + Mapper인터페이스의 객체가 DI 컨테이너에 등록
* Marker 인터페이스와 Marker 어노테이션의 사용
  + 검색의 대상이 되는 Package 아래의 인터페이스들 중에서 Mapper로서 작성한 인터페이스로만 범위를 좁히려면 Marker 인터페이스와 Marker 어노테이션을 작성
* MapperScannerConfigurer에 Marker 어노테이션 지정

**19. Spring MVC 개요**

[Spring Framework Basic 19강 Spring MVC 개요 | T아카데미 - YouTube](https://www.youtube.com/watch?v=NN_fYbcHZX8&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=19)

1. MVC(Model-View-Controller) 패턴의 개념과 모델2 아키텍쳐

* MVC 패턴의 개념
  + 소프트웨어 공학에서 사용되는 아키텍쳐 패턴
  + 주 목적은 Business logic과 Presentation logic을 분리
  + Model : 애플리케이션의 정보(데이터, business Logic)
  + View : 사용자에게 제공할 화면
  + Controller : Model과 View 사이의 상호 작용을 관리
* 각각의 MVC 컴포넌트의 역할
  + 모델(Model) 컴포넌트
    - 데이터 저장소와 연동, 데이터나 사용자에게 출력할 데이터를 다루는 일을 함
    - DAO 클래스, Service 클래스에 해당
  + 뷰(View) 컴포넌트
    - 모델이 처리한 데이터나 그 작업 결과를 가지고 사용자에게 출력
    - HTML과 JSP를 사용하여 작성
  + 컨트롤러(Controller) 컴포넌트
    - 클라이언트의 요청에 모델컴포넌트를 호출
    - Servlet과 JSP를 사용해서 작성
* 모델2 아키텍쳐 개념
  + 모델 1 아키텍쳐 : Controller 역할을 JSP가 담당 – 복잡
  + 모델 2 아키텍쳐 : Controller 역할을 Servlet이 담당 – 간단
* Front Controller 아키텍쳐
  + Front Controller 프로세스
    - Front Controller는 클라이언트가 보낸 요청을 받아서 공통적인 작업을 먼저 수행
    - 적절한 세부 controller에게 작업을 위임

1. Spring MVC의 개념

* Spring MVC의 특징
  + DI나 AOP같은 기능 뿐만 아니라, 서블릿 기반의 웹 개발을 위한 MVC 프레임워크를 제공
  + 모델2 아키텍쳐와 Front Controller 패턴을 적용
* Spring MVC와 Front Controller 패턴
  + Front Controller 역할을 하는 DispatcherServlet이라는 클래스를 계층에 맨 앞단에 놓고 서버로 들어오는 모든 요청을 받아서 처리
* DispatcherServlet 클래스
  + Front Controller 패턴 적용
  + Web.xml에 설정
  + 클라이언트로부터의 모든 요청을 전달 받음

**21. Spring MVC 웹어플리케이션 작성**

[Spring Framework Basic 21강 Spring MVC 웹어플리케이션 작성(1) | T아카데미 - YouTube](https://www.youtube.com/watch?v=xH5Vsq9thZU&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=21)

1. EL과 JSTL

* EL(Expression Language)의 개요
  + <% %> 와 같은 스크립팅 태그를 JSP에서 없앨 수 있다.
  + {}로 묶고 앞에 $기호를 붙인다.
  + 기본적으로 4가지 Scope(page, request, session, application)의 객체에 접근
  + null이거나 공백일 경우 아무 내용도 표시 X
  + EL은 JSP에서 기본으로 지원, JSTL은 따로 설치
* JSTL (Java Standard Tag Library) 개요
  + Jsp 표준 라이브러리
  + 스크립팅을 사용하지 않으면서 , 루프를 돌리거나 조건문을 실행할 수 있도록
  + 자주사용하는 반복과 조건 등 Custom Tag Library 모음
  + EL을 사용하여 표현
* JSTL Core 태그
  + 라이브러리 중 가장 많이 사용하는 태그
    - 변수지원 : set, remove
    - 흐름제어 : if, choose, forEach, forTokens
    - URL 처리 : import, redirect, url
    - 기타태그 : catch, out

1. Hello 컨트롤러 작성

* Controller와 JSP 구현 절차
  + POJO 형태의 HelloController 클래스 작성
  + @Controller 어노테이션을 선언
  + @RequestMapping 어노테이션을 선언
  + JSP를 이용한 View 영역의 코드를 작성
  + Browser 상에서 JSP를 실행

1. 사용자 목록 조회 컨트롤러 작성

* Controller와 JSP 구현

**22. Spring MVC 웹어플리케이션 작성**

[Spring Framework Basic 22강 Spring MVC 웹어플리케이션 작성(2) | T아카데미 - YouTube](https://www.youtube.com/watch?v=mruh05xc2ME&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=22)

1. ViewResolver 설정

* Controller의 실행 결과를 어떤 view에서 보여줄 것인지 결정하는 기능을 제공

1. 특정 사용자 조회 기능 구현

* Controller를 위한 핵심 어노테이션
  + @RequestParam : HTTP 요청에 포함된 파라미터 참조 시 사용
* View에 데이터와 화면정보를 전달하는 ModelAndView 클래스

1. 사용자 등록 화면 기능 구현

* @ModelAttribute : HTTP 요청에 포함된 파라미터를 모델 객체로 바인딩

**23. Spring MVC 웹어플리케이션 작성**

[Spring Framework Basic 23강 Spring MVC 웹어플리케이션 작성(3) | T아카데미 - YouTube](https://www.youtube.com/watch?v=bcRHf2JW_5E&list=PL9mhQYIlKEhfYqQ-UkO2pe2suSx9IoFT2&index=23)

1. 사용자 수정, 삭제

* @PathVariable : 파라미터를 URL 형식으로 받을 수 있도록 해줌

1. Spring MVC의 예외 처리

* @ExceptionHandler 어노테이션의 사용
  + 예외 처리 가능
  + 예외가 발생했을 때, 예외 Type과 Message를 보여주는 JSP페이지를 작성 필요
* viewError.jsp 작성

질문사항

1. 엑셀파일 활용