

스프링 정리 문제

1. 스프링을 왜 사용하는가?

많은 이유가 있겠지만,
편리한 MVC 구조, 의존성 주입 등 의존 관계 설정이 편리합니다.

2. 프레임워크란?

개발에서 기본이 되는 틀을 말한다!

3. 프레임워크의 종류 2가지?

JAVA의 프레임워크 : **SPRING**

Python의 프레임 워크 : **Django**

4. MVC2와 MVC1의 차이점

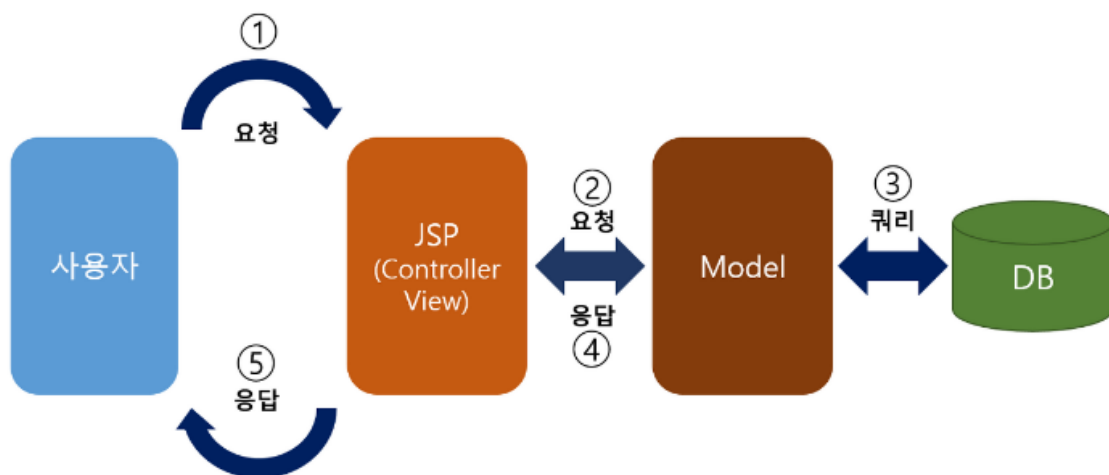
MVC 모델1 : jsp 중심, 완벽히 분리되지 못함.

Model(M) - 특정한 방법을 가지고 처리하는 부분.

View(V)- 브라우저에 보여지는 부분

Control(C) - 클라이언트의 요청을 받아서 서버에서 어떻게 처리할 지 제어하는 부분.

MVC 모델 1은 뷰와 컨트롤러의 역할이 합쳐져 있다.



위와 같이 JSP가 뷰와 컨트롤러의 역할을 모두 수행하면, JSP에 JAVA코드와 Html, css 등의

코드가 섞여 있어, 소스가 복잡해지고 읽기 어려워 **유지/보수가** 힘들다.

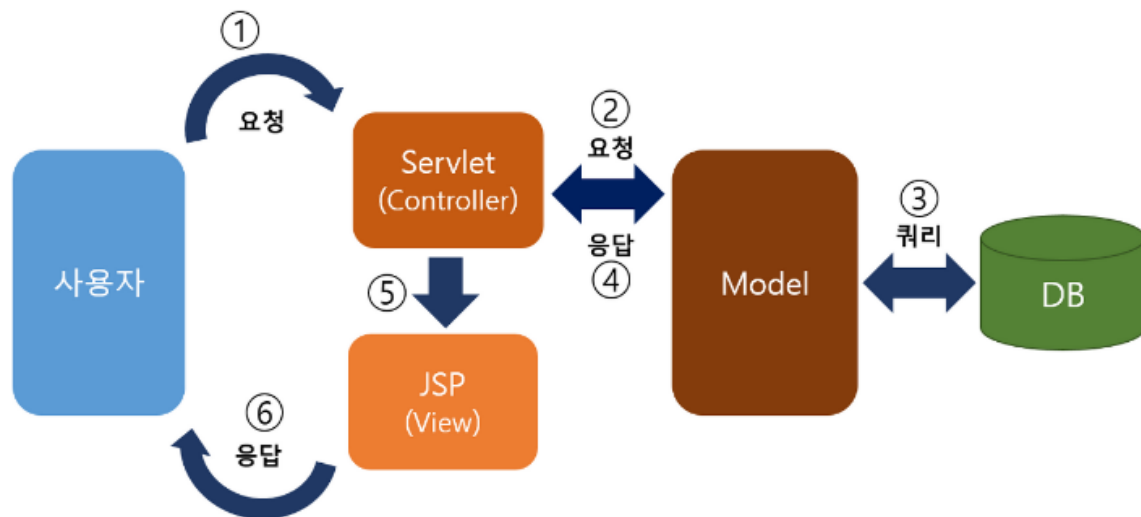
하지만 상대적으로 설계가 간단하여 **개발 속도가 빠르고 작은 프로젝트**에 알맞다.

MVC 모델2 는 MVC 모델 1의 유지/보수가 힘들다는 단점을 보완했다.

뷰와 컨트롤러의 역할을 모두 수행하던

JSP는 뷰의 역할만

컨트롤러의 역할을 Servlet이 수행한다.



html 과 java 코드가 분리되어 확장에 용이하고 **유지/보수가** 수월해진다.

5. 어노테이션(Annotation)의 역할 및 작업

Annotation(@)은 사전적 의미로는 **주석**이라는 뜻이다.

자바에서 Annotation은 코드 사이에 주석처럼 쓰이며 특별한 의미, 기능을 수행하도록 하는 기술이다.

즉, 프로그램에게 추가적인 정보를 제공해주는 **메타데이터**이다.

Annotation 종류

@ComponentScan

- @Component와 @Service, @Repository, @Controller, @Configuration이 붙은 클래스 **Bean**들을 찾아서 **Context**에 **bean**등록을 해주는 Annotation이다.

@Component

- 개발자가 직접 작성한 Class를 Bean으로 등록하기 위한 Annotation이다.

@Bean

- 개발자가 직접 제어가 불가능한 외부 라이브러리등을 Bean으로 만들려할 때 사용되는 Annotation이다.

@Autowired

- 속성(field), setter method, constructor(생성자)에서 사용하며 **Type에 따라 알아서 Bean을 주입** 해준다.

@Controller

- **Spring의 Controller를 의미**한다. Spring MVC에서 Controller클래스에 쓰인다.
- 컨트롤러로 등록 + 싱글톤 객체 생성