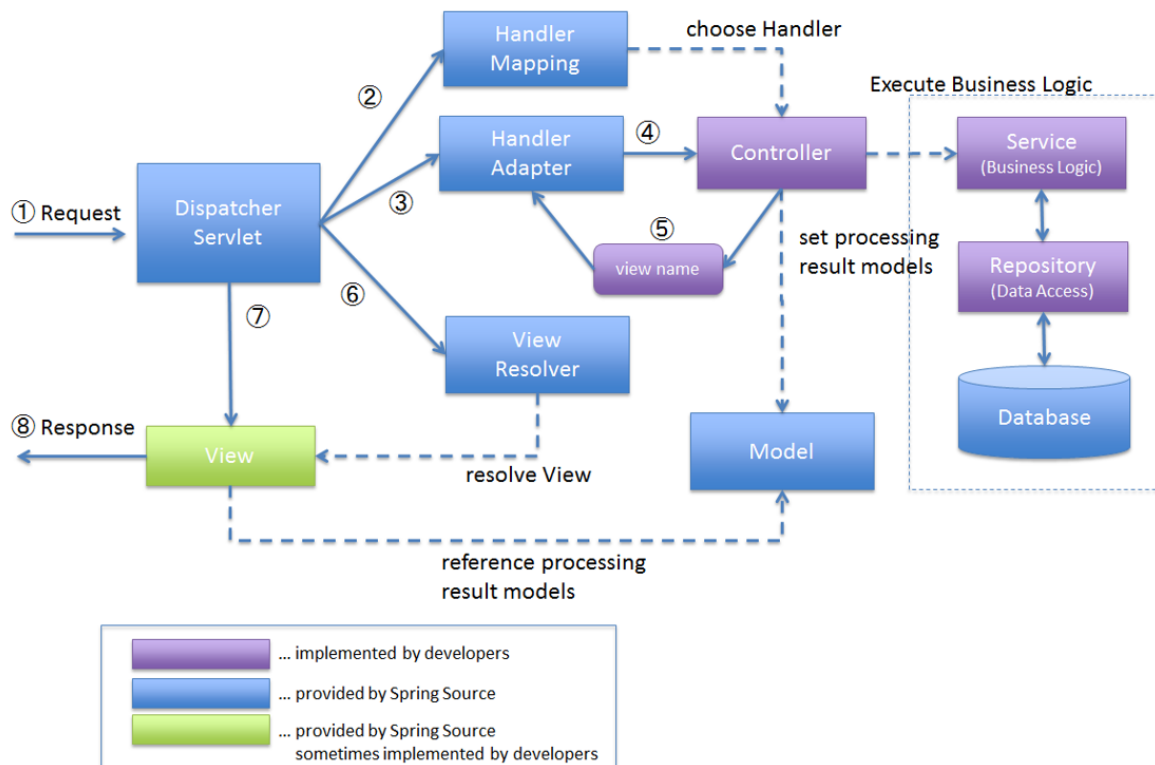
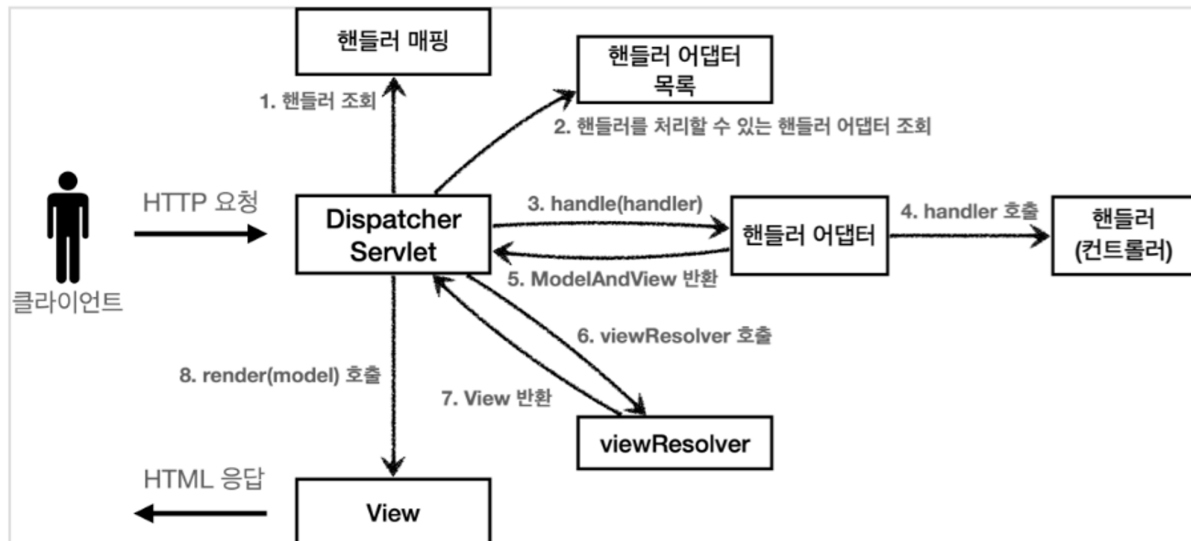


Spring MVC 흐름 정리



1. 클라이언트가 HTTP 요청을 보낸다

- GET : 브라우저 주소창 엔터

- 주소 뒤에 ?parameter=value 를 붙여서 파라미터를 보낼 수 있음
 - POST : form에 데이터를 입력해서 전송
2. 요청을 처리할 Controller(핸들러)를 찾는 과정을 거친 후, Controller는 결과를 반환한다.
- Service : 비즈니스로직 영역
 - DAO : DB에 접근하는 객체
 - DB : 데이터 저장소
3. Model and View 객체는 Controller가 처리한 결과와 View에 대한 정보를 담는다
4. View Resolver는 적절한 View를 선택하여 응답으로 보내준다
5. 클라이언트는 결과 화면을 보게 된다

각각의 과정은 담당하는 객체(Bean)가 역할을 수행함

Spring에 필요한 여러가지 설정들은 (.XML) Spring Bean을 등록하는 과정

- web.xml, servlet-context.xml, root-context.xml ...

@Component - 스프링 빈임을 알려주는 어노테이션

@Autowired - 자동으로 의존성 주입을 받기위한 어노테이션

@Controller - 컨트롤러 빈 객체

@Service - 서비스 빈 객체

@Qualifier - 의존성 주입 지정해서 받기

MyBatis

- Java 메소드와 SQL을 매핑해주는 프레임워크

sqlSessionFactory

- 데이터베이스와의 연결과 SQL의 실행을 담당하는 객체를 만들어주는 객체
- SQL Session을 만들어주는 공장
 - 참고 : 팩토리 패턴

```
<mybatis:scan base-package= "com.ssafy.hw.model.dao" />
```

- dao에 해당하는 mapper를 찾아줌

mapper

- SQL과 자바 메서드를 매핑해주는 객체