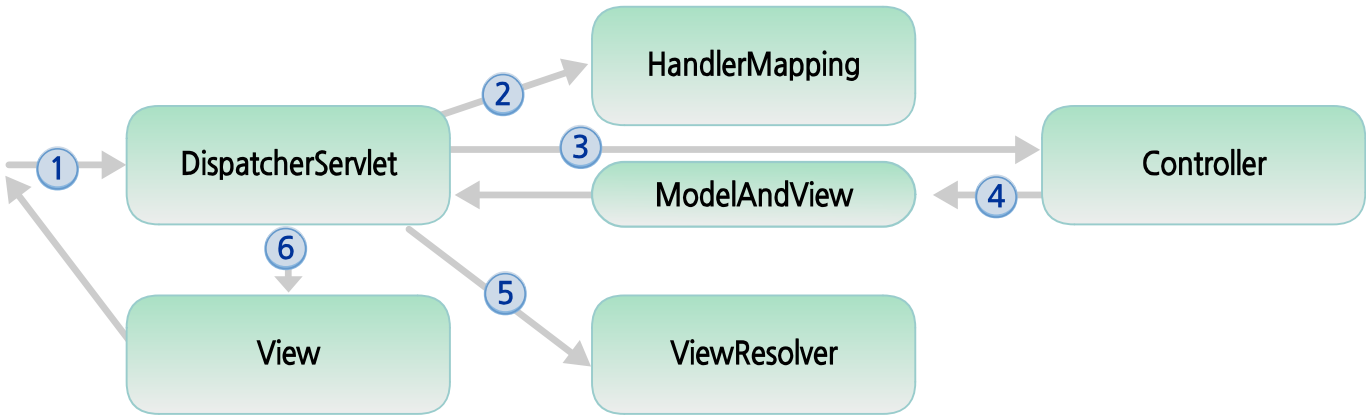


# 08 Spring MVC 기본

## Spring MVC 적용 및 구현

Spring MVC 처리 흐름



FrontController 역할을 수행하는 DispatcherServlet은 HandlerMapping이 리턴한 Controller를 호출하고, ViewResolver에 의해 View를 찾아 화면을 보여주는 구조임

Spring MVC 구성요소

구성 요소	설명
DspatcherServlet	클라이언트의 요청을 받는 FrontController임
HandlerMapping	클라이언트의 요청에 대한 Controller를 리턴함
Controller	클라이언트의 요청을 처리함
ViewResolver	Command 객체의 처리 결과를 보여줄 View를 결정함
View	Command 객체의 처리 결과를 보여줄 응답을 생성함

Spring MVC 구성요소

- Step 1

web.xml 파일에 DispatcherServlet 등록
- Step 2

클라이언트의 요청에 대한 Controller 작성
- Step 3

Spring 설정파일에 HandlerMapping, Controller, ViewResolver 등록
- Step 4

JSP 작성

# Controller 에서 Spring Bean 사용하기

- Listener
  - Listener를 등록하여 웹 초기화 시 Listener 클래스에 의해 나머지 XML 설정 파일이 초기화되도록 구현하면 웹 어플리케이션 내에서 다양한 Bean 들을 사용할 수 있게 됨
- Listener 등록
  - web.xml 파일에 <listener>라는 태그로 등록하며, 그 하위에 <listener-class>라는 태그를 이용하여 Spring MVC가 제공하는 Listener 클래스를 지정하면 됨

## Contorller 구현 기본

- Annotation 사용
  - Spring MVC 역시 XML 설정을 최소화할 수 있도록 Annotation 기반 설정을 지원하는데 기본 설정은 IoC 설정과 동일함

① context 관련 Namespace 선언



② <context:component-scan>이라는 태그를 사용하여 Controller Annotation 컴포넌트 스캔



③ 스캔할 클래스들의 패키지를 base-package속성을 이용해서 등록

- @Controller 사용
  - @Controller Annotation이 적용된 클래스는 어떤 Controller 인터페이스는 클래스를 상속할 필요가 없음
  - Controller 클래스는 POJO(Plain Old Java Object)로 구현될 수 있다는 의미로 메소드의 구현이 보다 자유로우며 메소드의 이름과 매개 변수 역시 명확함
- @RequestMapping
  - @RequestMapping을 Controller 클래스의 메소드에 설정할 때, 요청 URI와 적절하게 매핑하면 됨
  - @RequestMapping Annotation을 이용하면 클라이언트의 요청 방식(GET/POST)에 따라서 수행될 메소드를 다르게 설정할 수 있음

## 클라이언트 요청처리

- HTML <form> 처리
  - 대부분의 Controller는 사용자의 입력 정보를 추출하여 Model에 해당하는 컴포넌트의 메소드를 호출함
  - 복잡하고 길게 작성되는 login() 메소드를 자바빈을 이용해서 구현하면 간단하게 구현할 수 있음
    - ▶ login() 메소드의 매개변수로 사용자가 입력한 값을 매핑할 수 있는 자바빈을 등록하면 Spring 컨테이너가 자바빈을 생성하여 넘겨줌
    - ▶ 이때, 사용자가 입력한 값을 자바빈의 property에 자동으로 채워주기까지 함
    - 결과적으로 사용자 입력 정보 추출과 자바빈 생성 및 값 설정과정을 컨테이너에 의해서 자동으로 처리할 수 있는데, 이런 자바빈을 Command 객체라고 함
- @RequestParam 사용
  - @RequestParam을 이용해서 클라이언트에서 넘겨준 요청 파라미터 정보를 추출할 수 있음
- Servlet API 사용 : Controller 메소드의 매개변수로 다양한 Servlet API를 지원함
  - ServletRequest / HttpServletRequest
  - ServletResponse / HttpServletResponse
  - HttpSession
- Controller의 리턴타입

타입	설명
ModelAndView	Model과 View 정보가 저장된 객체임
String	View 정보를 정확한 이름으로 설정함
Model	View에 전달할 객체들을 담고 있는 객체임
Map	View에 전달할 정보가 저장된 Map 객체를 리턴함 - 이때 View 이름은 요청 URL을 통해서 자동으로 결정됨