



스프링 프레임워크란?

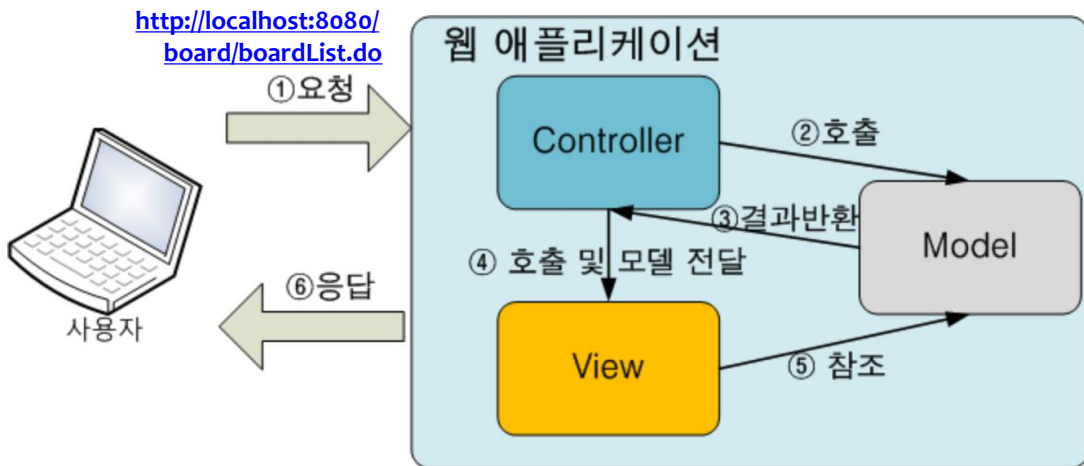


프로젝트 이름	설명
스프링 프레임워크	현대 자바 기반의 애플리케이션을 개발하는 데 기반이 되는 프레임워크, 스프링의 핵심은 애플리케이션 기반을 제공 함으로써 개발자들은 애플리케이션의 비즈니스 로직의 개발에만 집중할 수 있게 하는데 있다.
스프링 부트	스프링 프레임워크를 기반으로 바로 실행가능한 애플리케이션을 쉽게 만들도록 도와 준다. 대부분의 복잡한 스프링 관련 설정을 자동으로 처리하고 개발자는 최소한의 설정만 진행하면 된다.
스프링 데이터	스프링 애플리케이션에서 다양한 데이터 베이스, JPA 등의 데이터 접근 기술을 쉽게 사용할 수 있도록 도와준다. 스프링 데이터는 하나의 상위 프로젝트로 세부적으로는 데이터베이스 종류에 따라서 수많은 하위 프로젝트가 존재한다.
스프링 시큐리티	자바 애플리케이션에 인증(Authentication) 및 권한(Authorization)에 특화된 프레임워크 이다.
스프링 소셜	페이스북, 트위터, 링크드인과 같은 소셜 서비스 API와 쉽게 연동할 수 있게 도와준다.
※ 이외에도 스프링 클라우드, 스프링 배치, 스프링 모바일 등 다양한 프로젝트가 존재한다.	



Spring Boot 웹(MVC)

- ▶ MVC 패턴을 사용하면 사용자 인터페이스와 비즈니스 로직을 분리하여 개발할 수 있다.
- ▶ 서로 영향을 최소화하여 개발 및 변경이 쉬운 애플리케이션을 만들 수 있다.
- ▶ MVC 모델
 - ▶ Model : 어플리케이션의 정보, 즉 데이터(DB)
 - ▶ View : 사용자에게 보여질 화면(html)
 - ▶ Controller : 모델과 뷰의 중계역할(html과 데이터 결합)



```
package com.example.demo;
```

```
import ...
```

```
@RestController
```

```
public class HelloController {
```

```
    @RequestMapping("/board/boardList.do")
```

```
    public ModelAndView openBoardList () throws Exception {  
        ModelAndView mv = new ModelAndView("board/boardList");
```

```
        List<BoardDto> list = boardService.selectBoardList();  
        mv.addObject("list", list);
```

```
        return mv;
```

```
    }
```

```
}
```

Spring Boot 웹(MVC)

▶ MVC 구조

- ▶ **View** : 사용자가 보는 화면 또는 결과를 의미합니다.
- ▶ **Controller** : 사용자가 웹 브라우저를 통해서 어떠한 요청을 하면 그 요청을 처리할 컨트롤러를 호출하게 됩니다. 컨트롤러는 사용자의 요청을 처리하기 위한 비즈니스 로직을 호출하고 그 결과값을 사용자에게 전달해 주는 역할을 합니다.
- ▶ **Service** : 사용자의 요청을 처리하기 위한 비즈니스 로직이 수행됩니다. 일반적으로 서비스 영역은 서비스 인터페이스와 이 인터페이스의 구현체로 나뉩니다.
- ▶ **DAO** : Data Access Object 의 약자로 데이터베이스에 접속해서 비즈니스 로직 실행에 필요한 쿼리를 호출합니다.
- ▶ **DB** : 데이터베이스를 의미합니다. 데이터베이스에는 애플리케이션에서 발생한 모든 정보가 저장되어 있습니다.

