

# Spring MVC : Exception 처리

🕒 생성일	@2022년 8월 21일 오전 3:21
☰ 태그	

## 1. Exception Controller

어노테이션 `@ControllerAdvice` 사용

- 해당 클래스가 스프링의 컨트롤러에서 발생하는 예외를 처리하는 존재임을 명시

어노테이션 `@ExceptionHandler`

- Exception Controller Handler에서 어떤 타입의 Exception을 처리할 것인지 명시
- 어노테이션 속성으로 Exception 타입의 클래스가 들어간다.

리턴값은 JSP의 경로가 된다.

- 기본경로 `/WEB-INF/views` + 리턴값 (문자열)
- 기본경로는 `servlet-context.xml`에 `view-resolvers`에 지정

```
@Log4j2
@NoArgsConstructor

@ControllerAdvice
public class CommonExceptionHandler {

    public String handleControllerException(Exception e, Model model) {
        log.trace("handleControllerException() invoked.");

        model.addAttribute("_EXCEPTION_", e);

        return "errorPage";
    } // handleControllerException

} // end class
```

## 2. 404 에러

보통 Client입장에서, 페이지가 존재하지 않을때 발생하는 Error이다.

@ControllerAdvice는 특정 Exception들을 공통적으로 핸들링 할 수 있지만, 404 에러같은 경우는 DispatcherServlet에서 바로 처리해버린다.

- 즉, @ControllerAdvice까지 전달되지 못하고, DispatcherServlet에 의해 바로 처리되어버린다.
- 처리의 의미는 바로 404에러페이지로 전달 되어버린다.

404에러 페이지로 넘기지 않고, Exception을 발생시킬려면 초기파라미터값을 설정한다.  
(DispatcherServlet에 초기화 파라미터로 지정한다.)

```
<init-param>
    <param-name>throwExceptionIfNoHandlerFound</param-name>
    <param-value>true</param-value>
</init-param>
```

### NoHandlerFoundtException 예외를 처리하는 핸들러 작성

```
@ExceptionHandler(NoHandlerFoundException.class)
public String handleNoHandlerFoundException(Exception e, Model model) {

    model.addAttribute("_EXCEPTION_", e);

    return "/error/errorPage";
} // handleNoHandlerFoundException
```

만약, web.xml에 404에러에 대해, DispatcherServlet의 초기화파라미터로 등록하지 않았으면, 아래와 같이 DispatcherServlet에 의해 바로 처리 되어 버린다.

## HTTP 상태 404 – 찾을 수 없음

---

타입 상태 보고

**설명** Origin 서버가 대상 리소스를 위한 현재의 representation을 찾지 못했거나, 그것이 존재하는지를 밝히려 하지 않습니다.

---

Apache Tomcat/9.0.64

하지만, Controller로 예외가 전달 될 수 있도록 설정을 한다면, 핸들러로 지정한 페이지로 이동된다.

## 해당 URL은 존재하지 않습니다.

---

org.springframework.web.servlet.NoHandlerFoundException: No handler found for GET /errorPage