

Reqeust, Response

Request, Response

- Request는 요청으로 Spring Boot로 Client가 요청한 헤더와 데이터가 담겨있다.
- Response는 응답으로 Spring Boot가 Client의 요청에 따라 응답할 헤더와 데이터가 담겨있다.

예시1) Request, Response

- **HttpServletRequest, HttpServletResponse**

```
@RestController // Client에게 view(*.js, *html 등)가 아닌 data자체를 넘겨주기 위한 방법.
public class Controller {

    @GetMapping("/") // GET요청에 대한 응답을 매핑 시켜준다.
    public String hello() { return "Hello, Spring"; }

    @GetMapping("/person")
    public Person responsePerson
        (@RequestParam("name") String name,
         @RequestParam("age") int age,
         HttpServletRequest req,
         HttpServletResponse res) {

        System.out.println("req.Method : " + req.getMethod());
        System.out.println("req.URL : " + req.getRequestURL());
        System.out.println("req.QueryString : " + req.getQueryString());

        res.setStatus(404);

        return new Person(name, age);
    }
}
```

➤ HttpServletRequest

getMethod() : 요청 메소드를 String으로 반환 (Ex. GET, POST, PUT, etc..)
getRequestURL() : 요청 URL를 String으로 반환 (Ex. http://localhost:8080...)
getQueryString() : 요청 쿼리스트링을 String으로 반환

➤ HttpServletResponse

setStatus(int status) : 응답할 Http Status Code를 셋팅 (Ex. 200, 400, 404, 500, etc...)

- HttpServletRequest와 Response의 메소드에 대해서 더 알고 싶다면 아래의 참고문헌 링크를 참조하자.

참고문헌

- Oracle. <https://docs.oracle.com/javaee/6/api/javax/servlet/http/HttpServletRequest.html>. Request
- Oracle. <https://tomcat.apache.org/tomcat-5.5-doc/servletapi/javax/servlet/http/HttpServletResponse.html>. Response