

RequestBody, ModelAttribute, RequestParam

목차

- RequestBody, ModelAttribute, RequestParam
 - 목차
 - RequestBody, ModelAttribute, RequestParam이란?
 - RequestBody, ModelAttribute, RequestParam 활용 예시
 - Model 객체
 - Spring에서 활용 예시

RequestBody, ModelAttribute, RequestParam이란?

- @RequestParam
 - 1개의 HTTP 요청 파라미터를 받기 위해 사용
 - 필수여부가 TRUE이므로 반드시 해당 파라미터가 전송되어야함
 - 없으면 400 에러
 - 반드시 필요한 값이 아니라면, required를 false로 설정
 - 또는 defaultValue
- @RequestBody
 - 클라이언트가 전송하는 HTTP body를 변환
 - Json(application/json)을 Java 객체로 변환
 - RequestBody로 받는 데이터는 Spring에서 관리하는 MessageConverter들 중 하나인 MappingJackson2HttpMessageConverter를 통해 객체로 변환 => ObjectMapper
 - 데이터 형식이 Json이 아닐 수 도 있다.
 - ObjectMapper 동작과정 [중요하진 않음]
 - Json 메시지를 자바 객체로 변환하는 과정에서 객체의 기본생성자를 통해 객체를 생성
 - 내부적으로 Reflection을 사용
 - 그래서 반드시 @Setter가 필요한 것은 아닌데, @Getter나 @Setter 혹은 @JsonInclude 등 필드에 있는 변수들의 이름을 찾기 위한 메소드들을 필요로 한다. 그러므로 기본생성자 + @Getter로 클래스를 구현해주면 @Setter가 없어도 값이 바인딩된다.
- @ModelAttribute
 - 클라이언트가 전송하는 form() 형태의 HTTP body 요청 파라미터들을 생성자나 Setter로 바인딩하기 위해 사용
 - 파라미터 타입과 객체타입이 일치하는지 검증작업이 진행
 - @ModelAttribute('writer') String writer {writer:"최강현"}
 - ModelAttribute 동작과정
 - 먼저 Reflection을 통해 필드를 인자로 받는 생성자가 있는지 검사
 - 있다면 해당 생성자를 통해 값을 세팅하고
 - 모든 필드를 받는 생성자가 있다면 생성자로 값을 주입
 - 없다면 Setter로 값을 세팅
 - 모든 필드를 받는 생성자가 없다면, 기본생성자를 통해 객체를 생성하고
 - Setter로 남은 변수들 값 할당

```

@Getter
@Setter
@ToString
public class Board {

    private int index;
    private String writer;
    private String contents;

    public Board(int index) {
        this.index = index;
    }

    public Board(int index, String writer) {
        this.index = index;
        this.writer = writer;
    }
}

```

- Board(1,"글쓴이")를 통해 객체 생성
- Setter로 Contents에 값 주입

RequestBody, ModelAttribute, RequestParam 활용 예시

Model 객체

```

@Getter
@Setter
@ToString
public class Board {

    private int index;
    private String writer;
    private String contents;

}

```

Spring에서 활용 예시

```

@RestController
@RequestMapping("/board")
public class BoardController {

    @PostMapping("/requestBody")
    public ResponseEntity<Board> requestBody(@RequestBody Board board) {
        // @RequestBody는 MessageConverter를 통해 Json 형태의 HTTP Body를 Java
        객체로 변환시킨다.
    }
}

```

```
        return ResponseEntity.ok(board);
    }

    @PostMapping("/modelAttribute")
    public ResponseEntity<Board> modelAttribute(@ModelAttribute Board
board) {
        // @ModelAttribute는 폼(form) 형태의 HTTP Body와 요청 파라미터들을 객체에 바
인딩시킨다.
        return ResponseEntity.ok(board);
    }

    @GetMapping("/list")
    public ResponseEntity<List<Board>> requestParam(
        @RequestParam(value = "searchKeyword1", required = false,
defaultValue = "MangKyu") String searchKeyword) {

        // searchKeyword는 required가 false이기 때문에 없을 수도 있으므로, 없다면 기본
값이 할당된다.
        return
ResponseEntity.ok(boardService.getBoardList(searchKeyword1));
    }
}
```