

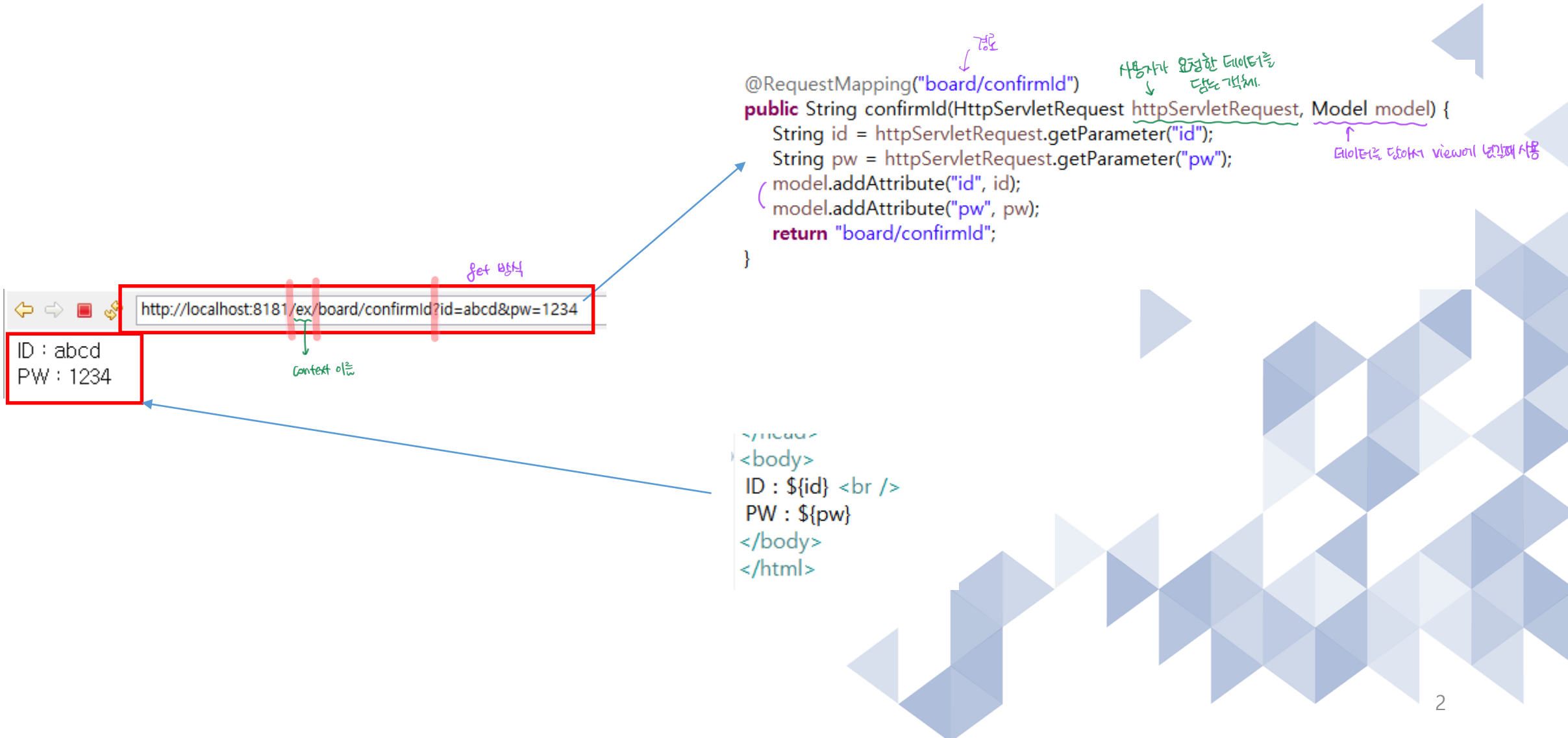
13강. Form 데이터

- HttpServletRequest 클래스
- @RequestParam 어노테이션
- 데이터(커맨드) 객체
- @PathVariable

Lecturer Kim Myoung-Ho
Nickname 불스
blogstudy@naver.com

13-1. HttpServletRequest 클래스

HttpServletRequest클래스를 이용해서 데이터를 전송하는 방법에 대해서 살펴 봅니다. (spring_13_1_ex1_springex)



↓ 넘길때 값을 반드시 넘겨야한다.
(값이 없을 경우 400 Error)

13-2. @RequestParam 어노테이션

@RequestParam 어노테이션을 이용해서 데이터를 전송하는 방법에 대해서 살펴 봅니다. (spring_13_1_ex1_springex)

```
@RequestMapping("board/checkId")
public String checkId(@RequestParam("id") String id, @RequestParam("pw") int pw, Model model) {
    model.addAttribute("identify", id);
    model.addAttribute("password", pw);
    return "board/checkId";
}
```

↑ id로 넘겨주면 'id'로 받는다.

http://localhost:8181/ex/board/checkId?id=abcdefg&pw=123456

ID : abcdefg
PW : 123456

```
</head>
<body>
    ID : ${identify} <br />
    PW : ${password}
</body>
</html>
```

13-3. 데이터(커맨드) 객체

데이터(커맨드) 객체를 이용하여 데이터 많을 경우 간단하게 사용 할 수 있습니다. (spring_13_1_ex1_springex)

기존 방법 : 다소 코드양이 많다.

```
@RequestMapping("/member/join")
public String joinData(@RequestParam("name") String name, @RequestParam("id") String id,
    @RequestParam("pw") String pw, @RequestParam("email") String email, Model model) {

    Member member = new Member();
    member.setName(name);
    member.setId(id);
    member.setPw(pw);
    member.setEmail(email);

    model.addAttribute("memberInfo", member);

    return "member/join";
}
```

개선 방법 : 코드양이 적다. Spring framework의 Member 객체를 넣는다.

```
@RequestMapping("/member/join")
public String joinData(Member member) {

    return "member/join";
}
```

http://localhost:8181/ex/member/join?name=홍길동&id=abc&pw=123&email=abc@abc.com

이름 : 홍길동
아이디 : abc
비밀번호 : 123
메일 : abc@abc.com

http://localhost:8181/ex/member/join?name=홍길동&id=abc&pw=123&email=abc@abc.com

이름 : 홍길동
아이디 : abc
비밀번호 : 123
메일 : abc@abc.com

13-4. @PathVariable

@PathVariable 어노테이션을 이용하면 경로(path)에 변수를 넣어 요청메소드에서 파라미터로 이용 할 수 있습니다.

