

# 스프링 - Controller 02

---

## SampleController 분석

```
@Controller
@RequestMapping("/ex/*")
@Log4j
public class SampleController {

    @RequestMapping(value = "/basic", method = {RequestMethod.GET,
RequestMethod.POST})
    public void basic(HttpServletRequest req) {
        log.info("basic..." + req.getMethod());
    }

    @RequestMapping
    public void basic2() {
        log.info("basic2.....");
    }

    @GetMapping("/basicOnlyGet")
    public void basic3(){
        log.info("basic3.....only get");
    }

    /*
     * @GetMapping("/ex01") public void ex01(String name, int age) {
     * log.info("ex01..." + name + ", " + age); }
     */
    @GetMapping("/ex01")
    //외부에서 전달된 파라미터를 매개변수 필드명과 자동으로 매핑한다.
    public void ex01(InfoDTO infoDTO) {
        log.info("ex01..2.." + infoDTO.getName() + ", " +
infoDTO.getAge());
    }
}
```

```

    @GetMapping("ex02")
    //외부에서 전달된 파라미터 이름과 매개변수가 다를 경우 @RequestParam을
    사용하여 어디로 전달받을 지 알려준다
    public void ex02(@RequestParam("data1") String name,
    @RequestParam("data2") int age) {
        log.info("ex02... " + name + ", " + age);
    }

    @GetMapping("/ex03")
    public String ex03(@RequestParam("data") ArrayList<String> datas) {
        log.info("datas : " + datas);
        return "ex03";
    }
}

```

## Controller에서 화면까지 데이터 전달

```

@GetMapping("/ex04")
/*
 * //객체를 정보를 받으면 앞에를 소문자로 바꾼형태로 바로 전달된다. //getName
등을안쓰고 ${infoDTO.name} 형태로 가져다
 * 쓸 수 있다. //@ModelAttribute("dto") 이렇게 사용하면 ${dto.name} 형태로
쓴다.
public String ex04(@ModelAttribute("dto") InfoDTO infoDTO) {
    log.info("-----");
    log.info(infoDTO.toString());
    log.info("-----");
    return "ex04";
}
/*
 * 정리 : 만약 매개변수가 개체라면, 해당 클래스 타입의 앞글자만 소문자로 변경된
값이 화면에서 사용할 key값이다. 매개변수의
 * 타입이 InfoDTO라면, 화면에서 사용 시 key 값은 infoDTO가 된다. 만약에 key값을
수정하거나 매개변수가 많아진다면,
 * 직접 requestScope에 담아서 전달해야 한다. 이 때 request객체를 직접 불러오지
않고, Model이라는 데이터
 * 전달자를 사용하게 된다. 하지만 화면쪽에 전달할 데이터가 여러개가 아니라면,
@ModelAttribute()를 사용하여 화면에
 * 전달한다. @ModelAttribute(화면에 전달할 key값)

```

```

*/
@GetMapping("/ex05")
public void ex05(InfoDTO infoDTO, @ModelAttribute("gender") String gender) {
    log.info("ex05.....");
    log.info(infoDTO.toString());
    log.info("gender : " + gender);
}

@GetMapping("/ex06")
//Model 객체는 파라미터로 request 객체를 받는다.
// 따라서 여러 개의 데이터를 화면에 전달할 때
// addAttribute(KEY, VALUE)을 사용한다.
// 화면에서는 model에 설정한 KEY로 VALUE를 사용할 수 있다.
public String ex06(InfoDTO infoDTO, String gender, Model model) {
    log.info("ex06.....");
    log.info(infoDTO.toString());
    log.info("gender : " + gender);

    model.addAttribute("gender", gender);
    model.addAttribute("dto", infoDTO);

    return "ex/ex06";
}

```