

## | Background

- ✓ JSP
- ✓ Attribute Binding
- ✓ Page Navigation

## | Goal

- ✓ 서블릿과 JSP페이지를 연결할 수 있다.
- ✓ 서버상에서 정보를 저장하는 Attribute를 정확하게 활용할 수 있다.
- ✓ 개별적인 parameter 값들을 하나의 객체에 저장할 수 있다

## | Problem

KB Star 은행은 새로운 고객 관리 시스템을 구축하고 있다. Back-End 구축을 준비하면서 Servlet 과 함께 JSP 를 사용하려고 한다. 기존 코드 Servlet과 함께 html파일 대신에 JSP 를 활용한다. MVC 모델에 기반해서 서로의 역할을 분리한다.

1. Context Path( 프로젝트 명)은 hw\_BackEnd 로 한다.
2. 아래의 가이드에 따라 Servlet 과 JSP 파일을 작성 또는 수정한다.

### >>>> get\_param.jsp요청 흐름 정리

- 1) get\_param.jsp 는 제공되는 파일을 그대로 사용한다.
- 2) get\_param.jsp 의 요청 때 넘어온 값은 Map 객체에 담고 적절한 Attribute에 바인딩 한다. 서블릿 이름은 web.servlet.controller.ParameterServlet 으로 한다.
- 3) 서버에 저장된(바인딩) 데이터 내용은 결과페이지 getResult.jsp 로 forward 한다.

### >>>> post\_param.jsp요청 흐름 정리

- 1) post\_param.jsp는 제공되는 파일을 그대로 사용한다.
- 2) web.servlet.dto. ParameterDto.java :
  - post\_param.jsp 의 요청에 포함된 parameter 를 field로 가진다.
  - Encapsulation을 적용하고 toString()를 overriding 한다.
- 3) post\_param.jsp요청 때 넘어온 값은 ParameterDto객체에 담고 다시 서버상 적절한 Attribute에 바인딩 한다.  
서블릿 이름은 web.servlet.controller.ParameterServlet 으로 한다.
- 4) 서버에 저장된(바인딩) 데이터 내용은 결과페이지 postResult.jsp 로 forward 한다.

3. 각각 실행된 결과는 아래와 같다.

## \* get\_param.jsp 실행화면

localhost:8888/be\_ws02/get\_param.jsp

**By Link**

[get link](#)

**By Button & JavaScript**

Get 전송

localhost:8888/be\_ws02/params?job=insert&pageNo=10&searchWord=bank

**Get Result**

pageNo : 10

searchWord : bank

job : insert

## \* post\_param.jsp 실행화면

localhost:8888/be\_ws02/post\_param.jsp

**기본 정보**

이름  
길복순

e-mail  
kbs@google.com

**성별**

☐ 남 ☒ 여

**취미**

☐ 영화 ☒ 스포츠 ☒ 음악

**선호하는 재테크**

연금

**기타**

KBStar 연금상품 이율이 좋아요  
~~!!

Post 전송

localhost:8888/be\_ws02/params

**Post Result**

name	길복순
email	kbs@google.com
gender	F
hobby	sport, music
favorite	PS
desc	KBStar 연금상품 이율이 좋아요~~!!

4. Class Diagram은 다음과 같다.  
(패키지명은 앞 문서를 참고하세요)

🔍 servlet.ParameterServlet
▣ serialVersionUID: long
◆ doGet(request: HttpServletRequest, response: HttpServletResponse): void
◆ doPost(request: HttpServletRequest, response: HttpServletResponse): void

🔍 dto.ParameterDto
▣ name: String
▣ email: String
▣ gender: String
▣ hobby: String[]
▣ favorite: String
▣ desc: String
🔍 ParameterDto()
🔍 ParameterDto(name: String, email: String, gender: String, hobby: String[], favorite: String, desc: String)
🔍 getName(): String
🔍 setName(name: String): void
🔍 getEmail(): String
🔍 setEmail(email: String): void
🔍 getGender(): String
🔍 setGender(gender: String): void
🔍 getHobby(): String[]
🔍 setHobby(hobby: String[]): void
🔍 getFavorite(): String
🔍 setFavorite(favorite: String): void
🔍 getDesc(): String
🔍 setDesc(desc: String): void
🔍 toString(): String