

2024년 상반기 K-디지털 트레이닝

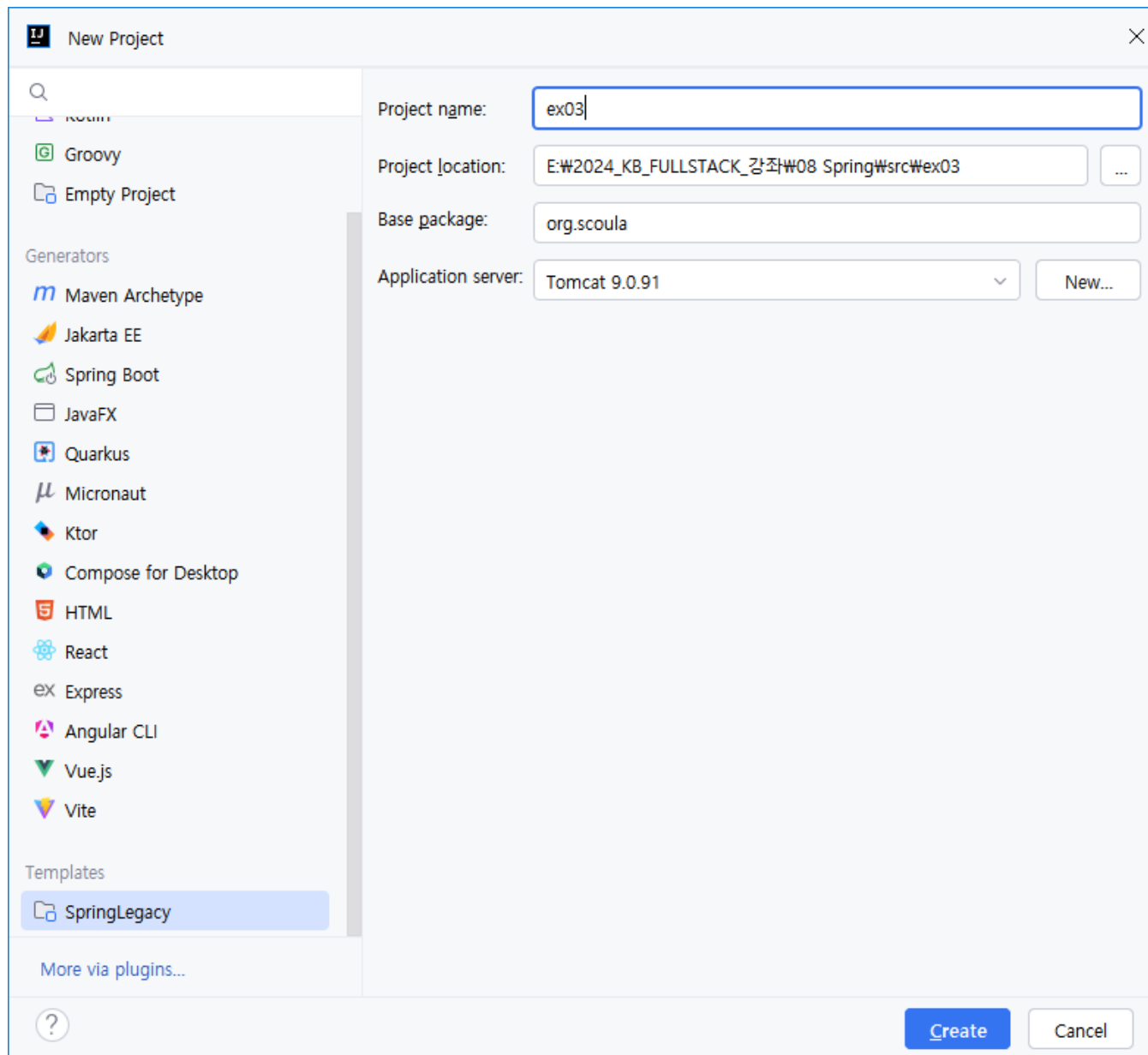
# 스프링 MVC의 Controller

---

[KB] IT's Your Life

## ✓ 프로젝트 만들기

- 템플릿: SpringLegacy
- 프로젝트명: ex03



# @Controller, @RequestMapping

## ServletConfig.java

```
package org.scoula.config;

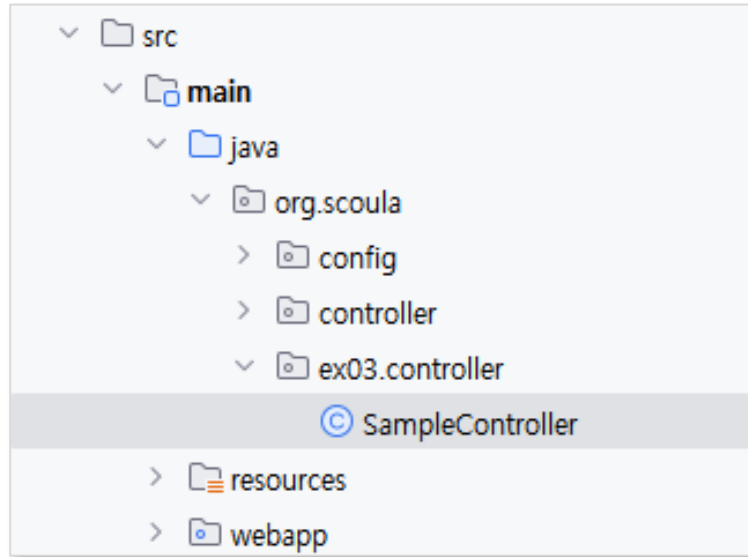
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.ComponentScan;
import org.springframework.web.servlet.config.annotation.EnableWebMvc;
import org.springframework.web.servlet.config.annotation.ResourceHandlerRegistry;
import org.springframework.web.servlet.config.annotation.ViewResolverRegistry;
import org.springframework.web.servlet.config.annotation.WebMvcConfigurer;
import org.springframework.web.servlet.view.InternalResourceViewResolver;
import org.springframework.web.servlet.view.JstlView;

@EnableWebMvc
@ComponentScan(basePackages = {
    "org.scoula.controller",
    "org.scoula.ex03.controller"
})
public class ServletConfig implements WebMvcConfigurer {
    ...
}
```

# 1 @Controller, @RequestMapping

## ✓ 컨트롤러 추가

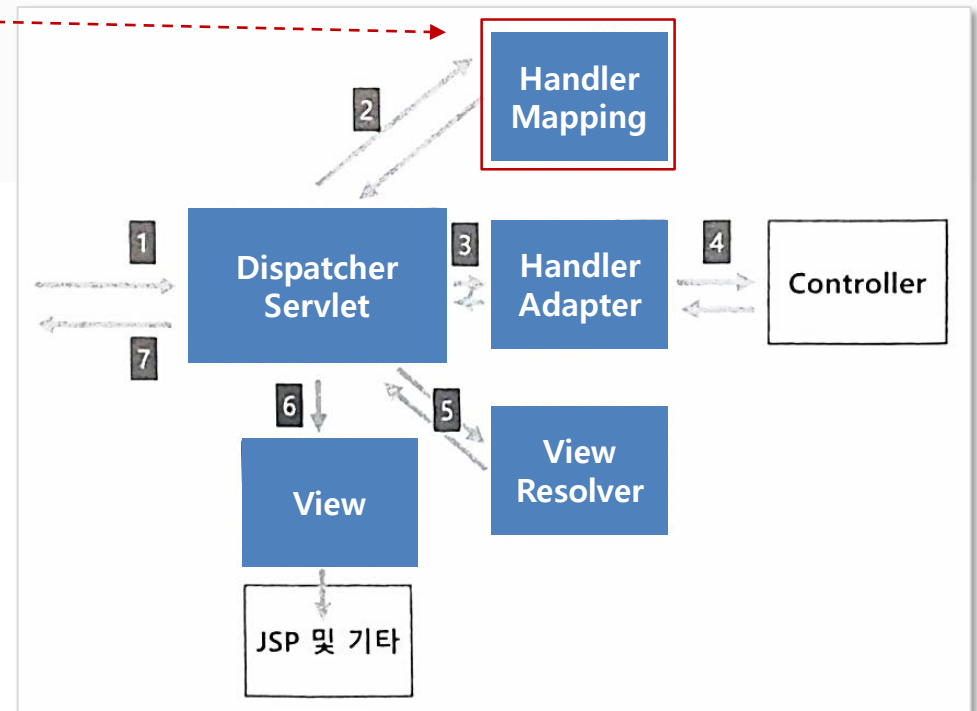
- org.scoula.ex03.controller.SampleController



# @Controller, @RequestMapping

## SampleController.java

```
package org.scoula.ex03.controller;  
  
import org.springframework.stereotype.Controller;  
import org.springframework.web.bind.annotation.RequestMapping;  
  
@Controller  
@RequestMapping("/sample")  
public class SampleController {  
  
}
```



# @Controller, @RequestMapping

## SampleController.java

```
package org.scoula.ex03.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;

import lombok.extern.log4j.Log4j;

@Controller
@RequestMapping("/sample")
@Log4j
public class SampleController {

    @RequestMapping("")    // url : /sample
    public void basic() {
        log.info("basic.....");
    }
}
```

# 1 @Controller, @RequestMapping

## ✓ 확인

- <http://localhost:8080/sample>

```
INFO : org.springframework.web.servlet.DispatcherServlet - FrameworkServlet 'dispatcher': initialization completed in 910 ms
```

```
INFO : org.scoula.ex03.controller.SampleController - basic.....
```

```
...
```

# @Controller, @RequestMapping

## ✓ 확인

- <http://localhost:8080/sample>

- src/main/resources/log4j.xml

```
<root>
  <priority value="debug" />
  <appender-ref ref="console" />
</root>
```

```
DEBUG: _org.springframework.web.servlet.HandlerMapping.Mappings -
```

```
o.s.c.HomeController:
```

```
{GET [/]}: home()
```

```
DEBUG: _org.springframework.web.servlet.HandlerMapping.Mappings -
```

```
o.s.s.c.SampleController:
```

```
{[/sample]}: basic()
```

```
DEBUG: _org.springframework.web.servlet.HandlerMapping.Mappings - 'beanNameHandlerMapping' {}
```

```
DEBUG: _org.springframework.web.servlet.HandlerMapping.Mappings - 'resourceHandlerMapping'
```

```
{/resources/**=ResourceHttpRequestHandler [ServletContext [/resources/]]}
```

```
DEBUG: org.springframework.jndi.JndiTemplate - Looking up JNDI object with name  
[java:comp/env/spring.liveBeansView.mbeanDomain]
```

```
DEBUG: org.springframework.jndi.JndiLocatorDelegate - Converted JNDI name  
[java:comp/env/spring.liveBeansView.mbeanDomain] not found -
```



## @RequestMapping의 변환

### ✓ @RequestMapping

- 수용할 HTTP method 설정
- GET, POST, PUT, DELETE 등

`@RequestMapping(value="/basic", method= {RequestMethod.GET, RequestMethod.POST})`

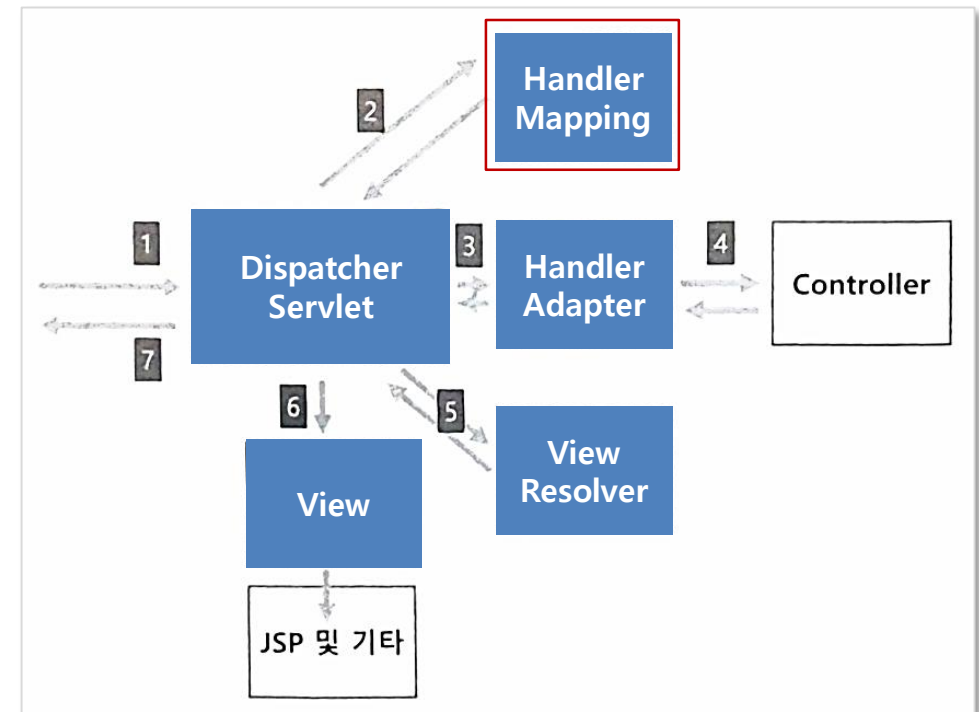
- 설정하지 않으면 모두 허용

### ✓ 메서드별 Mapping 어노테이션

- 일반적으로 한 가지 메서드로 매핑함
- @GetMapping
- @PostMapping
- @PutMapping
- @DeleteMapping

`@GetMapping("/basicOnlyGet")`

```
public void xxx() { ... }
```



## @RequestMapping의 변환

### SampleController.java

```
...
@Controller
@RequestMapping("/sample")
@Log4j
public class SampleController {

    @RequestMapping("") // url: /sample
    public void basic() {
        log.info("basic.....");
    }

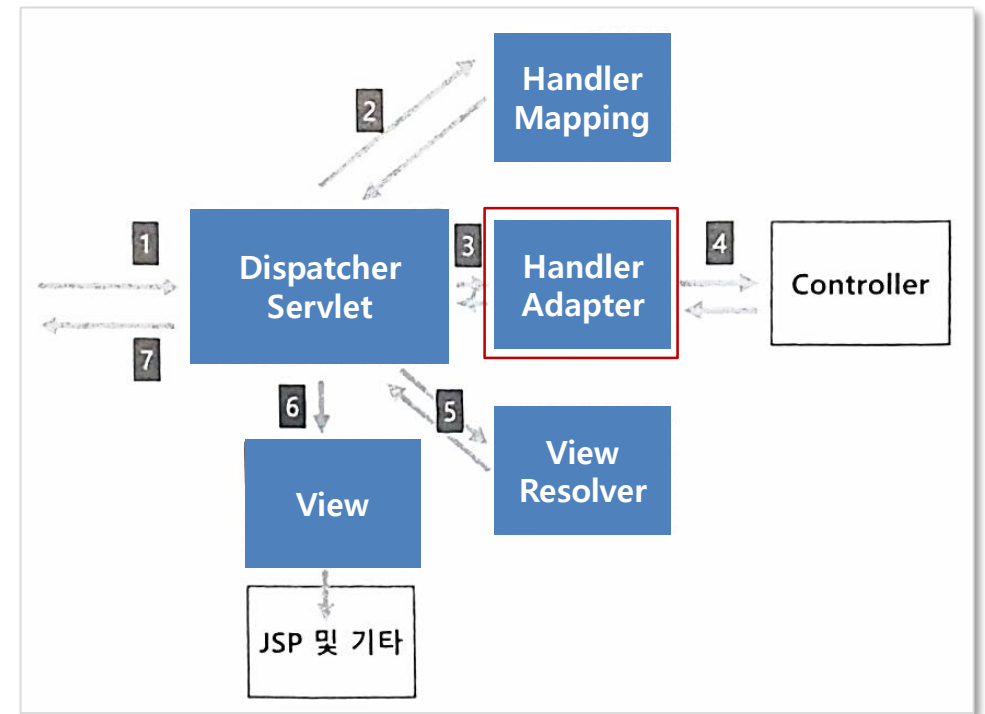
    @RequestMapping(value="/basic", method= {RequestMethod.GET, RequestMethod.POST}) // url: /sample/basic
    public void basicGet(){
        log.info("basic get.....");
    }

    @GetMapping("/basicOnlyGet") // url: /sample/basicOnlyGet
    public void basicGet2(){
        log.info("basic get only get.....");
    }
}
```

### 3 Controller의 파라미터 수집

#### ✓ 요청의 쿼리 파라미터 수집

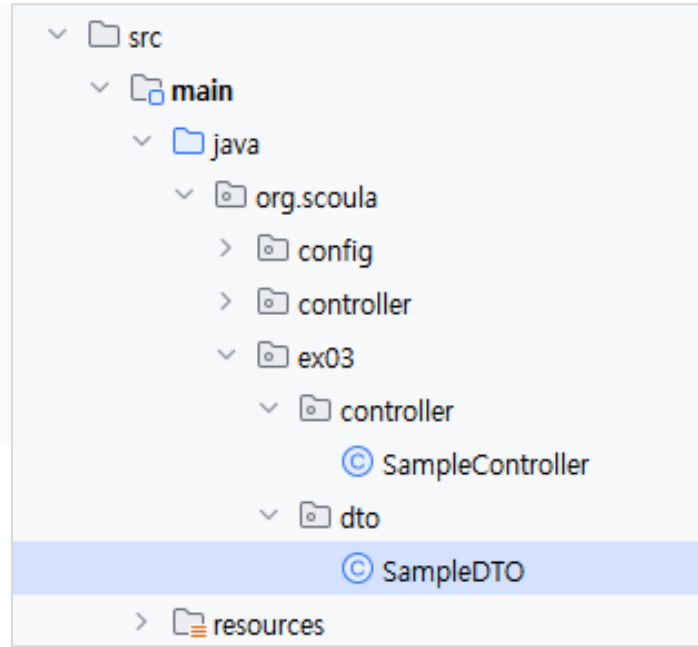
- `request.getParameter()`를 대체
- 컨트롤러 메서드의 매개변수에 필요한 쿼리 파라미터 매핑 지정
  - DTO 객체 사용
    - HandlerAdapter가 요청한 DTO 객체를 생성
    - Request의 파라미터/body에서 DTO의 프로퍼티를 설정해서 주입
  - `@RequestParam("파라미터명")` 어노테이션



### 3 Controller의 파라미터 수집

#### SampleDTO.java

```
package org.scoula.ex03.dto;  
  
import lombok.Data;  
  
@Data  
public class SampleDTO {  
    private String name;  
    private int age;  
}
```

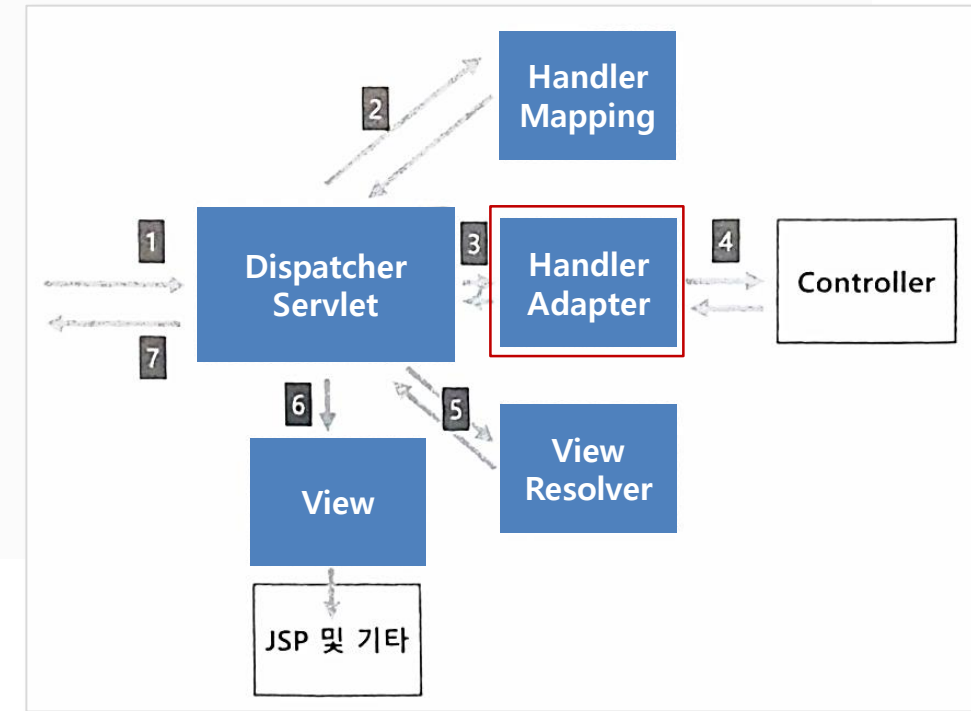


### 3 Controller의 파라미터 수집

#### SampleController.java

```
...  
public class SampleController {  
    ...  
  
    @GetMapping("/ex01")  
    public String ex01(SampleDTO dto) {  
        log.info("" + dto);  
  
        return "ex01";  
    }  
}
```

- HandlerAdapter가 SampleDTO의 디폴트 생성자를 이용해 생성
  - DTO 객체는 반드시 기본 생성자가 있어야 함.
- SampleDTO의 setter 메서드에서 프로퍼티명 유추
  - setName() → name, setAge() → age
- 프로퍼티와 동일한 이름의 요청 파라미터를 추출
  - request.getParameter("name"), request.getParameter("age")
- 해당 파라미터가 존재하면 setter를 이용해 DTO의 프로퍼티 설정
- 컨트롤러의 요청 처리 메서드의 인자로 전달



### 3 Controller의 파라미터 수집

#### ✓ 확인

`http://localhost:8080/sample/ex01`

INFO : org.scoula.ex03.controller.SampleController - SampleDTO(name=null, age=0)

`http://localhost:8080/sample/ex01?name=AAA&age=10`

INFO : org.scoula.ex03.controller.SampleController - SampleDTO(name=AAA, age=10)

### 3 Controller의 파라미터 수집

#### SampleController.java

```
...
public class SampleController {
    ...

    @GetMapping("/ex02")
    public String ex02(
        @RequestParam("name") String name,
        @RequestParam("age") int age) {
        log.info("name: " + name);
        log.info("age: " + age);

        return "ex02";
    }
}
```

http://localhost:8080/sample/ex02?name=AAA&age=10

INFO : org.scoula.ex03.controller.SampleController - name: AAA  
INFO : org.scoula.ex03.controller.SampleController - age: 10

## 3 Controller의 파라미터 수집

### ✓ 리스트 배열 처리

- 동일한 이름의 파라미터가 여러 개 전달되는 경우
- ArrayList<> 나 배열 사용



### 3 Controller의 파라미터 수집

#### SampleController.java

```
...
public class SampleController {
    ...

    @GetMapping("/ex02List")
    public String ex02List(@RequestParam("ids") ArrayList<String> ids) {
        log.info("ids: " + ids);

        return "ex02List";
    }
}
```

http://localhost:8080/sample/ex02List?ids=111&ids=222&ids=333

INFO : org.scoula.ex03.controller.SampleController - ids: [111, 222, 333]

## 3 Controller의 파라미터 수집

### SampleController.java

```
...
public class SampleController {
    ...

    @GetMapping("/ex02List")
    public String ex02List(@RequestParam("ids") String[] ids) {
        log.info("array ids: " + Arrays.toString(ids));

        return "ex02List";
    }
}
```

## 3 Controller의 파라미터 수집

### ✓ 객체 리스트

- 객체 리스트를 쿼리 파라미터로 받기

### 3 Controller의 파라미터 수집

#### SampleDTOList.java

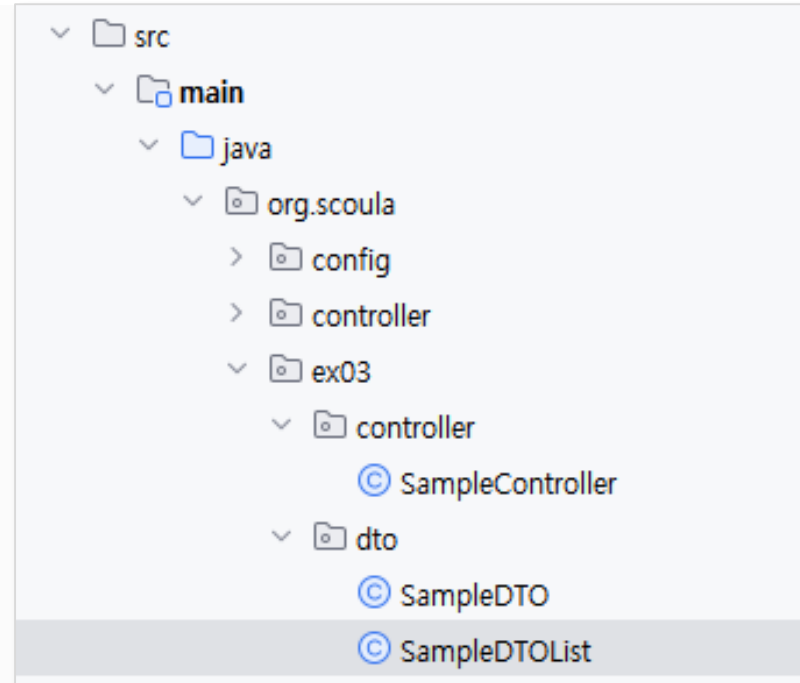
```
package org.scoula.ex03.dto;

import lombok.Data;

import java.util.ArrayList;
import java.util.List;

@Data
public class SampleDTOList {
    private List<SampleDTO> list;

    public SampleDTOList() {
        list = new ArrayList<>();
    }
}
```



### 3 Controller의 파라미터 수집

#### SampleController.java

```
...
public class SampleController {
    ...

    @GetMapping("/ex02Bean")
    public String ex02Bean(SampleDTOList list) {
        log.info("list dtos: " + list);

        return "ex02Bean";
    }
}
```

http://localhost:8080/sample/ex02Bean?list[0].name=aaa&list[2].name=bbb  
[, ]문자 때문에 에러 발생

http://localhost:8080/sample/ex02Bean?list%5B0%5D.name=aaa&list%5B2%5D.name=bbb

INFO : org.scoula.ex03.controller.SampleController - list dtos: SampleDTOList(list=[SampleDTO(name=aaa, age=0), SampleDTO(name=null, age=0), SampleDTO(name=bbb, age=0)])

## 3 Controller의 파라미터 수집

### ✓ @DateTimeFormat

- 커스텀 바인딩이 가장 많이 사용되는 타입 → Date
- DTO 객체의 Date 타입 필드에 @DateTimeFormat을 지정

```
@DateTimeFormat(pattern="yyyy-MM-dd")  
private Date reg_date;
```

### 3 Controller의 파라미터 수집

#### TodoDTO.java

```
package org.scoula.ex03.dto;

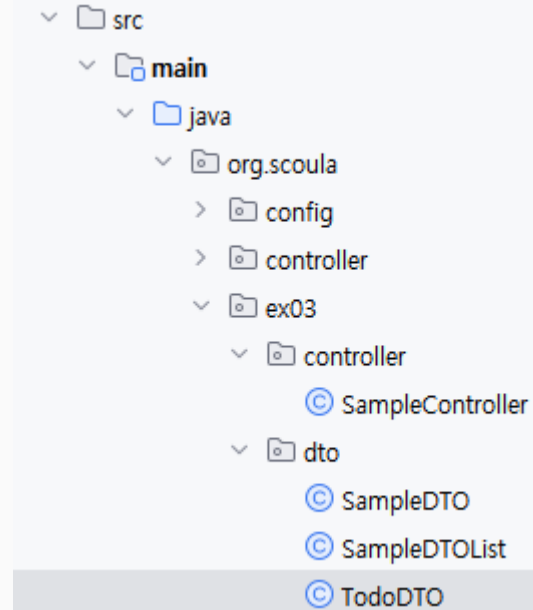
import java.util.Date;

import org.springframework.format.annotation.DateTimeFormat;

import lombok.Data;

@Data
public class TodoDTO {
    private String title;

    @DateTimeFormat(pattern="yyyy/MM/dd")
    private Date dueDate;
}
```



### 3 Controller의 파라미터 수집

#### SampleController.java

```
...
public class SampleController {
    ...

    @GetMapping("/ex03")
    public String ex03(TodoDTO todo) {
        log.info("todo: " + todo);
        return "ex03";
    }
}
```

<http://localhost:8080/sample/ex04?title=test&dueDate=2023/01/01>

INFO : org.scoula.ex03.controller.SampleController - todo: TodoDTO(title=test, dueDate=Sun Jan 01 00:00:00 KST 2023)