

Spring Framework

**REST API**

## @RestController란?

기존의 @Controller는 return에 Model 담아서 JSP에 전달하는 방식으로 사용했었습니다

@RestController어노테이션은 return에 처리하는 데이터를 조금 다른 타입으로 처리하는 형식입니다.

즉, 객체(데이터)를 반환할 수도 있고 객체(데이터)를 받을 수도 있다는 뜻입니다



그러면 어떻게 보내고 어떻게 받을 건데요?

**바로 JSON형식 (JavaScript Object Notation) 입니다**

## @RestController란?

- 비동기 **요청 전용 컨트롤러** 라고 생각하면 됩니다
- RequestMapping으로 들어오는 요청을 받아들이는 것은 동일하지만 return의 결과는 뷰리졸버가 아니라 **요청한 화면으로 리턴**됩니다

```
@RestController
@RequestMapping("/RestControl")
public class RestBasicController {

    @RequestMapping(value = "/getText", produces = "text/plain; charset=utf-8")
    public String getText() {

        System.out.println(MediaType.TEXT_PLAIN_VALUE);

        return "안녕하세요";
    }
}
```

← → ↻ ⓘ localhost:8181/myweb/RestControl/getText

안녕하세요

```
@RestController
@RequestMapping("/RestControl")
public class RestBasicController {

    @RequestMapping(value = "/getObject")
    public TestVO getObject() {

        return new TestVO(1, "홍길동", "kkk123");
    }
}
```

← → ↻ ⓘ localhost:8181/myweb/RestControl/getObject

{"num":1,"name":"홍길동","id":"kkk123"}

객체를 보낼수도 있다, 단 자동으로 JSON형으로 변환해주는 **jackson-databind** 라이브러리가 반드시 필요합니다

## @RestController 기본 준비사항

### Ajax사용 준비사항 (json라이브러리 다운)

```
<dependency>
  <groupId>com.fasterxml.jackson.core</groupId>
  <artifactId>jackson-databind</artifactId>
  <version>2.9.8</version>
</dependency>
```

JSON(JavaScript Object Notation)은 자바스크립트 객체로 구성된 데이터입니다  
자바스크립트 객체 형태의 문자열인 셈입니다  
{ }를 이용합니다

ex)  
{키 : 값, 키 : 값, 키 : [배열]}

## @RestController 기본 어노테이션

어노테이션	기능
@RestController	Controller가 rest방식임을 명시
@ResponseBody	뷰리졸버로 전달하는게 아니라 데이터를 요청한 화면으로 전달함을 명시 <b>RestController에는 이미 포함</b>
@PathVariable	URL경로에 파라미터를 줄 수도 있으며, URL경로의 있는 값을 추출할 때 사용
@RequestBody	JSON데이터를 자동으로 바인딩처리

## @RestController의 전송방식

작업	전송방식	URI
등록	POST	/reply/등록
조회	GET	/reply/{id}
수정	PUT	/reply/{id}제이슨
삭제	DELETE	/reply/{id}

다양한 전송방식은 URL주소에 다음과 같은 형식으로 표기하는 것을 규칙으로 하지만 JSON형식으로 주고받도록 처리합니다

## @PathVariable의 사용예제

```
@RequestMapping("/getPath/{id}/{pw}")
public HashMap<String, TestVO> getPath(@PathVariable("id") String id,
                                       @PathVariable("pw") String pw) {

    HashMap<String, TestVO> map = new HashMap<>();
    map.put("info", new TestVO(10, "길순", "hhh") );

    return map;
}
```

## @RequestBody의 사용예제

```
@RestController
@RequestMapping("/RestControl")
public class RestBasicController {

    @RequestMapping("/getJson")
    public ArrayList<TestVO> getJson(@RequestBody TestVO vo) {

        ArrayList<TestVO> list = new ArrayList<TestVO>();

        return list;
    }
}
```

비동기통신으로 넘어오는 JSON데이터를 자동으로 TestVO에 맵핑

➡ JSON데이터를 전달하는 형식은  
구글확장프로그램 Rest Client에서 테스트 가능

객체반환

## REST API를 이용한 댓글 비동기 처리

- 비동기 처리를 조금더 편하게 사용할 수 있는 제이쿼리를 사용하도록 합니다.
- 자바스크립트에 대한 이해가 있다면 제이쿼리는 5분만에 적용할 수 있습니다.

## 댓글 테이블

```
create table freereply (  
    rno number(10,0),  
    bno number(10,0),  
    reply varchar2(1000),  
    replyId varchar2(50),  
    replyPw varchar2(50),  
    replydate date default sysdate,  
    updatedate date default sysdate  
);  
create SEQUENCE freereply_seq;  
alter table freereply add CONSTRAINT freereply_pk PRIMARY KEY (rno);
```

### 전체적인구도

1. ajax로 댓글등록 -> 목록메서드호출
2. ajax로 댓글목록 처리 (가지온데이터를 innerHTML로 화면에 처리)  
-메서드의 매개변수(페이지 번호, 페이지리셋여부)
3. ajax로 삭제 수정 클릭시 이벤트 동작처리  
-이벤트 등록시점이 댓글목록 전이기 때문에 이벤트 위임함수처리
4. ajax로 삭제처리 -> 목록메서드호출
5. ajax로 수정처리 -> 목록메서드호출
6. 더보기 처리 -> 목록메서드에서 조건처리

