

# 240907 -API

HelloController

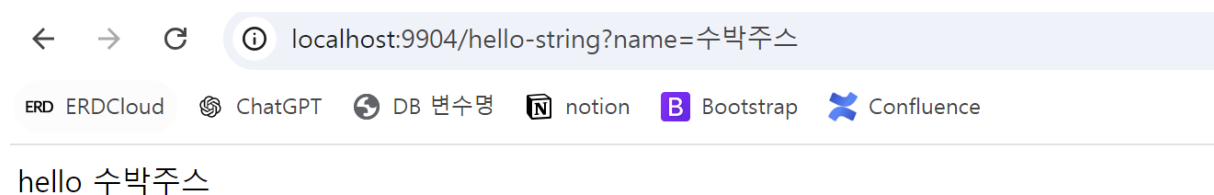
```
@GetMapping("hello-string")
@ResponseBody
public String helloString(@RequestParam("name") String name) {
    return "hello " + name; // "hello 수박주스"
}
```

## @ResponseBody

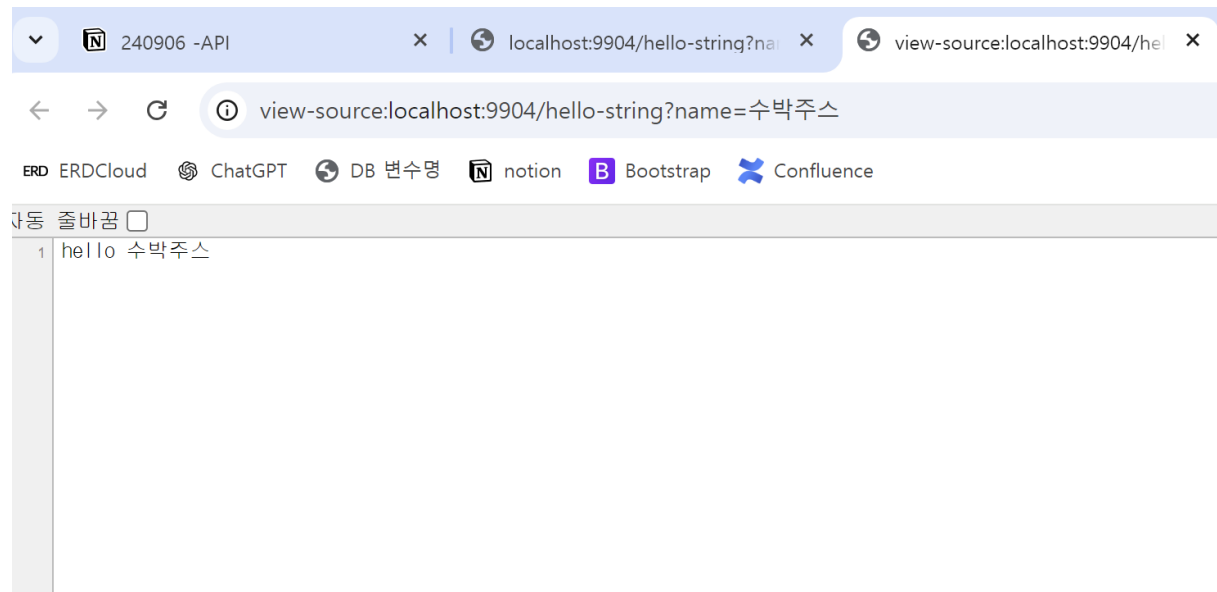
- 메서드의 반환 값을 HTTP 응답 본문(body)에 직접 담아서 반환할 때 사용하는 어노테이션
- 뷰로 전달되는 것이 아니라 HTTP 응답 본문으로 직접 출력

**쉽게 이해하기!** @ResponseBody : http body부에 이 데이터를 직접 넣어주겠다.

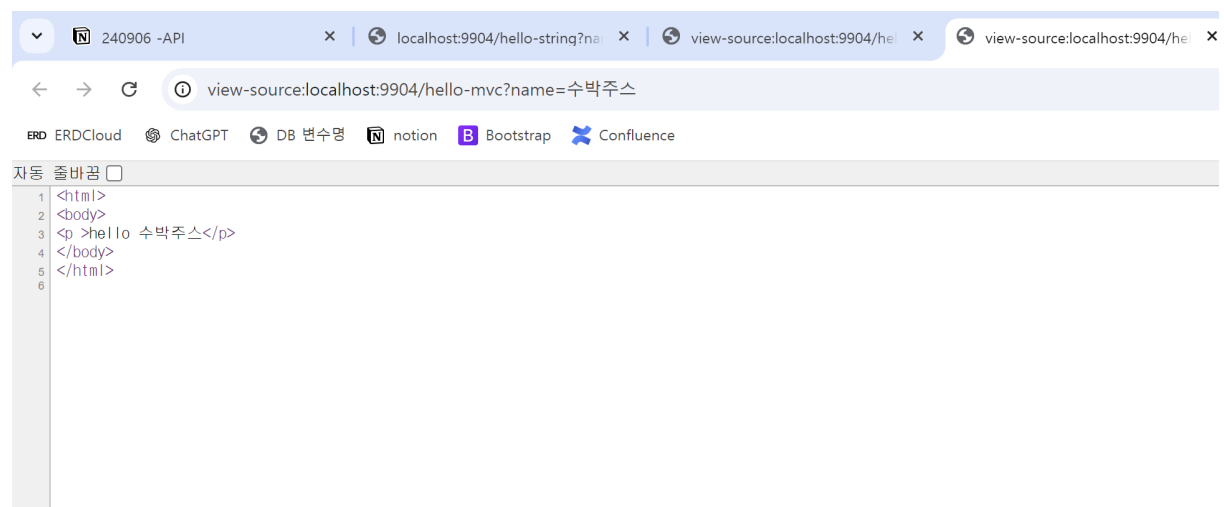
<http://localhost:9904/hello-string?name=수박주스> 로 접속하면 겉보기엔 똑같지만



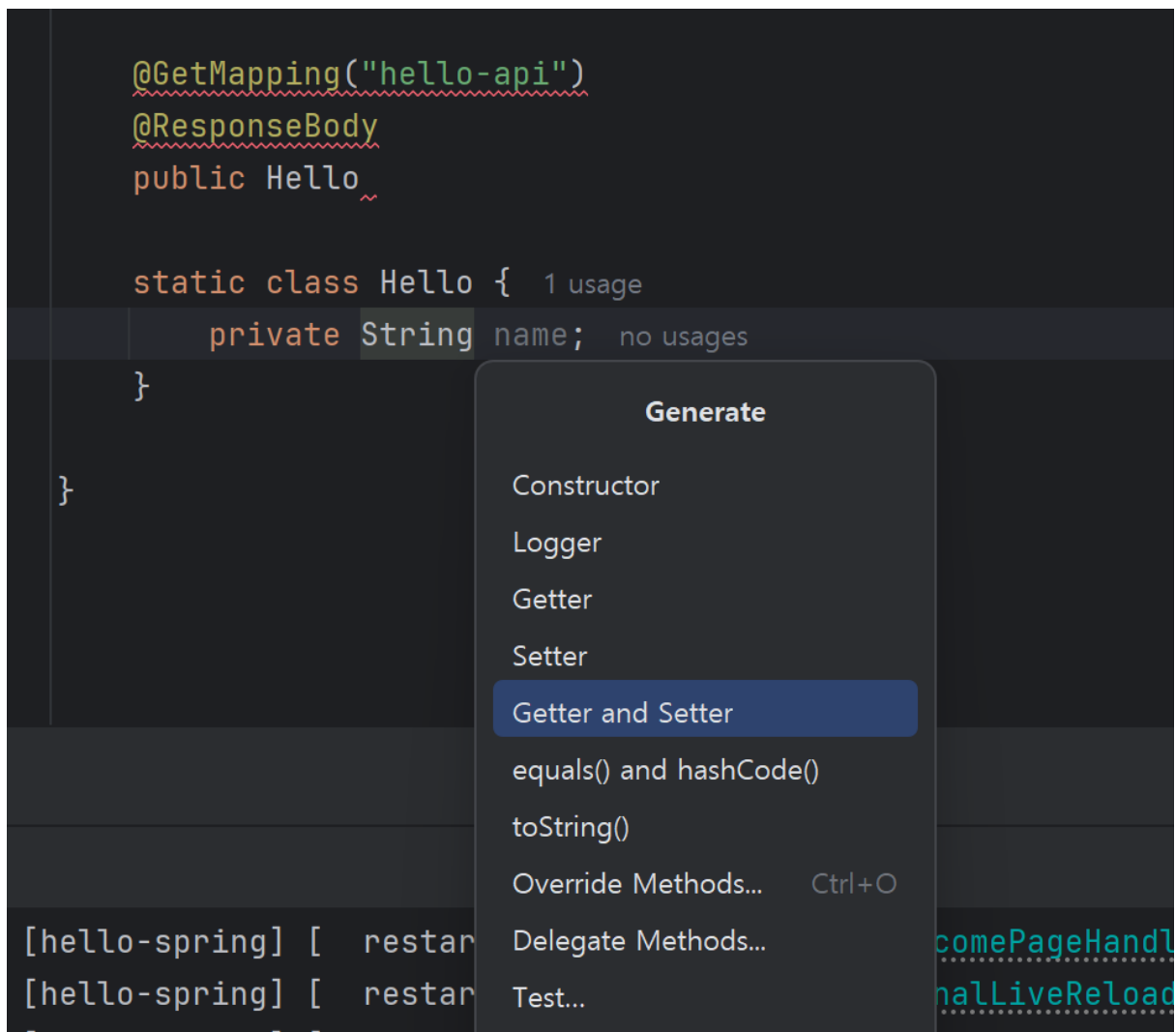
소스보기를 해보면 기존과는 다르게 아래와 같이 html코드가 보이지 않는다.



기존은 아래처럼



만들고자 하는 변수에 대고 alt + Insert 후 getter setter 생성



## HelloController

```
@GetMapping("hello-api")
@ResponseBody
public Hello helloApi(@RequestParam("name") String name)
    Hello hello = new Hello();
    hello.setName(name);
    return hello;
}

static class Hello {
    private String name;

    public String getName() {
```

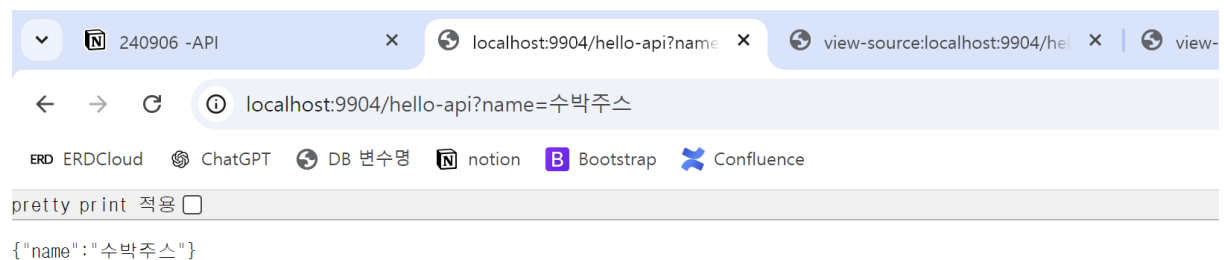
```

        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}

```

<http://localhost:9904/hello-api?name=수박주스>



- JSON방식 - 심플, 가볍다 (XML보다 유용하다)
- **@ResponseBody**
  - HTTP의 BODY에 문자 내용을 직접 반환
  - viewResolver 대신에 HttpMessageConverter가 동작
  - 기본 문자처리 : StringHttpMessageConverter
  - 기본 객체처리 : MappingJackson2HttpMessageConverter

- byte 처리 등등 기타 여러 `HttpMessageConverter`가 기본으로 등록되어 있음
- 객체를 **JSON**으로 바꿔주는 라이브러리 크게 두가지. **Jackson**, 구글에서 만든 **GSON**