



# Múltiples parámetros-JSON

## Múltiples parámetros

- En un solo método de nuestra API podemos recibir tantos parámetros como lo necesitemos. Para demostrarlo vamos a crear una calculadora de edades que va a recibir una fecha de nacimiento y va a retornar la edad asociada.

```
@GetMapping("/{day}/{month}/{year}")  
public String getAge(@PathVariable Integer day, @PathVariable Integer month, @PathVariable Integer year)
```

### @PathVariable:

- Nos permite extraer información que es parte de la estructura de la URI pero que no se trata como un par nombre=valor, a diferencia del @RequestParam que representa un valor que se envía en un pedido.

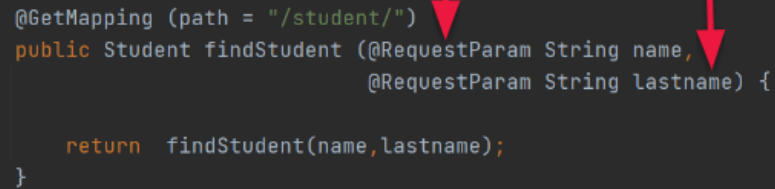
```
@GetMapping("/user/{userId}")  
public User getUser(@PathVariable("userId") String userId) {  
    ...  
}
```

### @RequestParam:

Es una anotación que permite recibir parámetros desde una ruta mediante el método GET, para trabajar con ellos e incluso poder emitir una respuesta que dependa de los parámetros que sean obtenidos.

Cada uno de los parámetros generalmente se ubican en la URL después de un signo de pregunta “?” y están anidados por un “&”

Por ejemplo: <http://localhost:8080/student?name=Horacio&lastname=Quiroga>



```
@GetMapping (path = "/student/")  
public Student findStudent (@RequestParam String name,  
                             @RequestParam String lastname) {  
  
    return findStudent(name,lastname);  
}
```

## JSON

## JAVA SCRIPT OBJECT NOTATION

**JSON** es un **formato de texto sencillo** utilizado para el **intercambio de datos** entre distintos sistemas.

**JSON** tiene dos elementos centrales, los objetos y sus valores intrínsecos:

Objetos

- Están representados mediante llaves, su apertura { indica el comienzo de la estructura de un objeto y su cierre el final } .

Valores Intrínsecos

Los objetos tienen valores / estado y los mismos se representan mediante **claves** (o keys) y **valores** (o value).

- Las Claves deben ser cadenas de caracteres y representan el “nombre” del valor.
- Los valores son tipos de datos soportados por JSON.

Las claves y valores se separan entre sí mediante “:”

En caso de necesitar agregar más claves y valores, se separan entre sí mediante “,”

