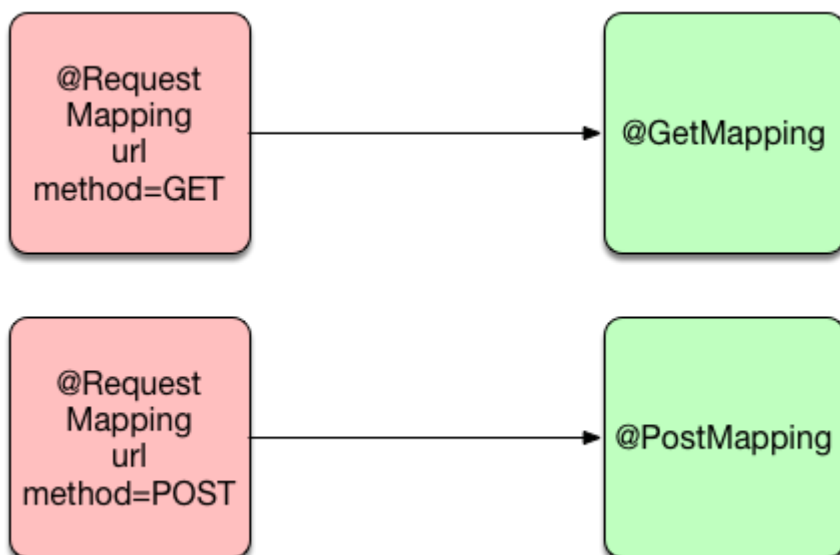


CURSO SPRING FRAMEWORK

-75% BLACK FRIDAY

APUNTATE!!

Spring GetMapping es una anotación de Spring Framework que aparece con Spring 4.3 y nos permite simplificar el manejo de los diferentes métodos de Spring MVC y los @RequestMapping que a veces se hacen un poco pesados.



A partir de ahora nosotros podemos usar estas anotaciones y simplificar el código. Para ello vamos a construir un ejemplo utilizando Spring Boot y con las dependencias de MVC y Thymeleaf.

```
<?xml version="1.0"
encoding="UTF-8"?>
<project
```

[illegible]

```

&lt;parent&gt;
&lt;groupIdorg.springframework.boot&lt;/groupId&
&lt;artifactId&lt;
&lt;spring-boot-starter-
parent&lt;/artifactId&
&lt;version&lt;
&lt;1.5.8.RELEASE&lt;/version&
&lt;relativePath/&
&lt;!-- lookup parent from
repository --&lt;/parent&
;

&lt;properties&
&lt;project.build.sourceEncoding&
&lt;UTF-8&lt;/project.bu
ild.sourceEncoding&
&lt;project.reporting.outputEncoding&
&lt;UTF-8&lt;/projec
t.reporting.outputEncoding&
&lt;java.version&
&lt;1.8&lt;/java.version&
&lt;/properties&

```

```
&lt;dependencies&gt;
&lt;dependency&gt;
&lt;groupId&gt;org.springframework.boot&lt;/groupId&gt;
&lt;/dependency&gt;
&lt;artifactId&gt;spring-boot-starter-web&lt;/artifactId&gt;
&lt;/dependencies&gt;
```

```
&lt;dependency&gt;
&lt;groupId&gt;org.springframework.boot&lt;/groupId&gt;
&lt;artifactId&gt;spring-boot-starter-thymeleaf&lt;/artifactId&gt;
&lt;/dependency&gt;
&lt;dependency&gt;
&lt;groupId&gt;org.springframework.boot&lt;/groupId&gt;
&lt;artifactId&gt;
```

```
;gt;spring-boot-starter-  
test&amp;&amp;&amp;&amp;&amp;&amp;&amp;</artifactId&amp;&amp;&amp;&amp;&amp;  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</scope&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
est&amp;&amp;&amp;&amp;&amp;&amp;&amp;</scope&amp;&amp;&amp;&amp;&amp;&amp;&amp;  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</dependency&amp;&amp;&amp;&amp;&amp;&amp;&amp;  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</dependencies&amp;&amp;&amp;&amp;&amp;&amp;&amp;  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</build&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</plugins&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</plugin&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</groupId&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</org.springframework.boot&amp;&amp;&amp;&amp;&amp;&amp;&amp;</groupId&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</artifactId&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</spring-boot-maven-  
plugin&amp;&amp;&amp;&amp;&amp;&amp;&amp;</artifactId&amp;&amp;&amp;&amp;&amp;&amp;&amp;  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</plugin&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</plugins&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</build&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t  
&amp;&amp;&amp;&amp;&amp;&amp;&amp;</project&amp;&amp;&amp;&amp;&amp;&amp;&amp;>t
```

El primer paso es ver el código de la aplicación principal:

```
package com.arquitecturajava.spring;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import
org.springframework.web.servlet.config.annotation.WebMvcConfigurerAdapter;

@SpringBootApplication
public class Application extends WebMvcConfigurerAdapter {

    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }
}
```

Una vez tenemos esto construido el siguiente paso es dar de alta un controlador:

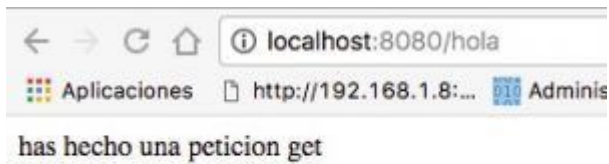
```
package com.arquitecturajava.spring;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.ResponseBody;

@Controller
```

```
public class HolaController {  
  
    @GetMapping("/hola")  
    @ResponseBody  
    public String hola() {  
  
        return "has hecho una peticion get";  
  
    }  
  
    @GetMapping("/formulario")  
    public String hola2() {  
  
        return "formulario";  
  
    }  
  
    @PostMapping("/holapost")  
    @ResponseBody  
    public String holaPost() {  
  
        return "has hecho una peticion post";  
  
    }  
}
```

Este controlador hace uso de las nuevas anotaciones de spring @GetMapping ,@PostMapping etc. Como se puede ver si solicitamos la url “/hola” nos devolverá “has hecho una peticion get” . Vamos a arrancar la aplicación de SpringBoot y probarlo.



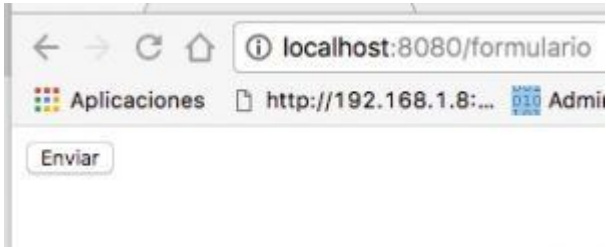
Funciona perfectamente . Una vez que hemos realizado la primera petición solicitamos la carga del formulario con “/formulario”.

```
&lt;html&gt;
&lt;body&gt;

&lt;form action="holapost"
method="post"&gt;
&lt;input
type="submit"/&gt;
&lt;/form&gt;

&lt;/body&gt;
&lt;/html&gt;
```

Esto nos cargará una página html con un botón que realiza una petición post a la url “holapost”.



Pulsamos el botón y nos mostrará el resultado



Acabamos de usar spring GetMapping y PostMapping para trabajar con Spring MVC y eliminar el uso de @RequestMapping.

CURSO SPRING REST
-75% BLACK FRIDAY
APUNTATE!!

Otros artículos relacionados

1. [Spring MVC Flash Attributes](#)
2. [Utilizando Spring MVC Bean Validation](#)
3. [Spring MVC Configuración \(I\)](#)
4. [Spring MVC](#)