

Spring Boot Thymeleaf es una de los starters de Spring Boot más típicos cuando construimos una aplicación web. El uso de este motor de plantillas se esta generalizando mientras que el mundo de JSP se reduce paulatinamente . Recordemos que JSP tiene ya unos 20 años y es una tecnología que aunque ha aguantado bien se le nota el paso del tiempo. Vamos a construir el ejemplo de holamundo con este starter . Para ello como siempre deberemos ir a la web de [spring boot initializer](#) .

Spring boot Thymeleaf

En este web configuraremos Spring boot para que use los starters de web y Thymeleaf ya que es el motor de plantillas que utilizaremos ,vamos a ver el pom.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>2.1.4.RELEASE</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>
    <groupId>com.arquitecturajava</groupId>
    <artifactId>plantillas</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>plantillas</name>
    <description>Demo project for Spring Boot</description>
```

```

<properties>
    <java.version>1.8</java.version>
</properties>

<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-
thymeleaf</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-
web</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-
test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
<groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-maven-
plugin</artifactId>
        </plugin>
    </plugins>
</build>

```

```
        </plugins>
    </build>
```

```
</project>
```

Una vez que tenemos ambos starters configurados el siguiente paso es construir un controlador que nos permite pasar datos a una plantilla de spring boot thymeleaf.

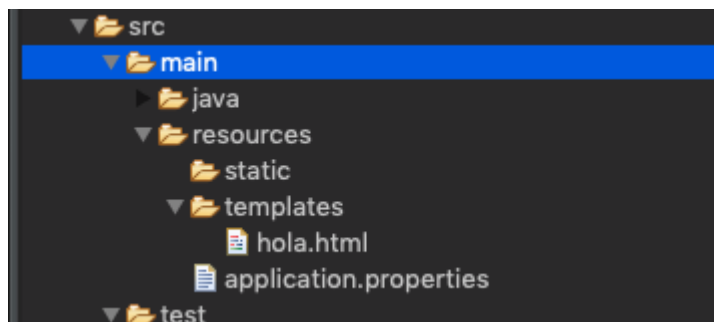
```
package com.arquitecturajava.plantillas;

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

@Controller
public class ControladorHola {

    @RequestMapping("/hola")
    public String hola(Model modelo) {
        modelo.addAttribute("mensaje","hola desde thymeleaf");
        return "hola";
    }
}
```

En nuestro caso se trata del controlador más básico que nos devuelve un mensaje de “hola desde thymeleaf” . Nos queda ver como Spring boot a través de su starter nos configura una carpeta para ubicar las plantillas html y que el controlador pueda acceder a ellas.

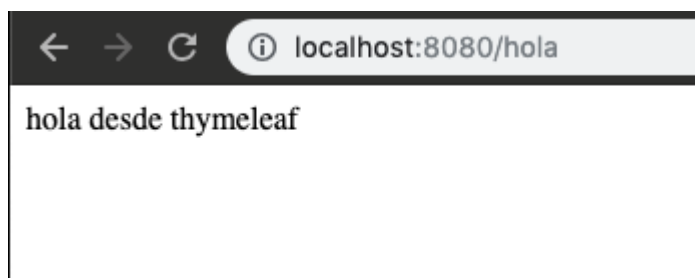


Spring Boot folders

Nos queda simplemente mostrar el contenido del fichero html que contendrá el código de Thymeleaf

```
<html xmlns:th="http://www.thymeleaf.org">
<body>
<p th:text=${mensaje}></p>
</body>
</html>
```

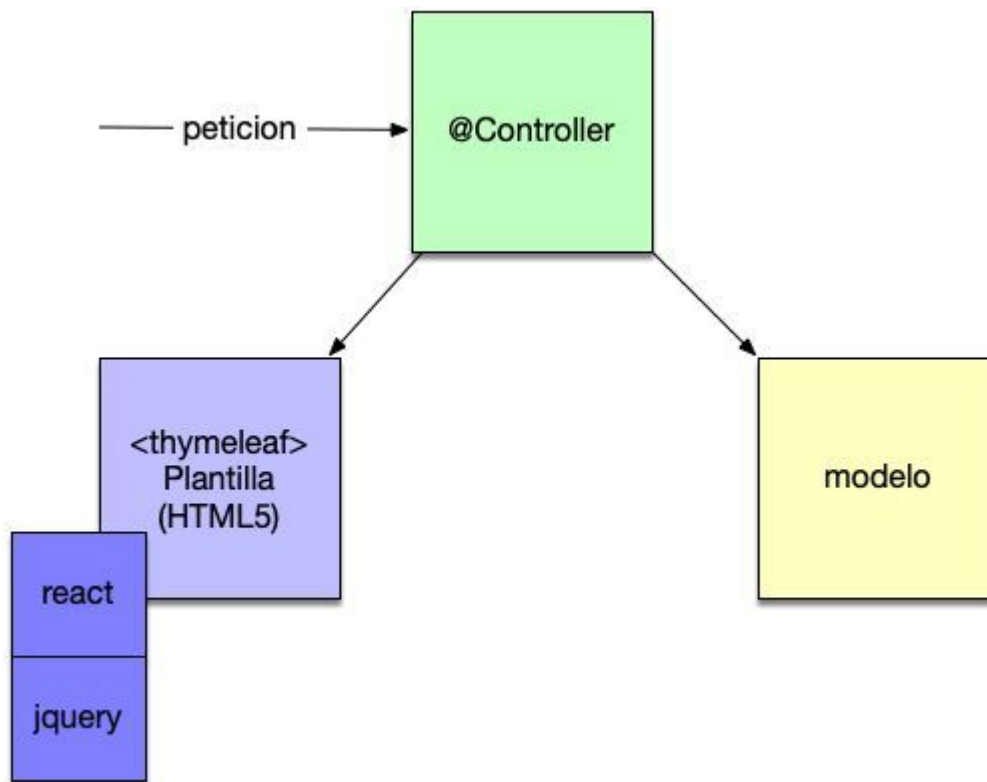
Si ejecutamos el proyecto de Spring boot veremos salir por pantalla el mensaje.



spring boot thymeleaf resultado

Acabamos de configurar de forma correcta el motor de plantillas usando Spring Boot. Recordemos que los motores de plantillas han existido desde los principios del mundo Java en cuanto el patrón MVC aparecido como referente pero no siempre han sido las tecnologías más usadas . La competencia con JSP y JSTL siempre ha existido y existirá . Sin embargo hoy en día hay que valorar sobre manera que un motor de plantillas como Thymeleaf nos

permite trabajar con ficheros html de una forma muy pura.



Simplificando la integración con otras tecnologías como React o JQuery de forma natural.

Otros artículos relacionados

- [JSP JavaScript Template String](#)
- [spring thymeleaf](#)
- [Curso Spring Boot](#)