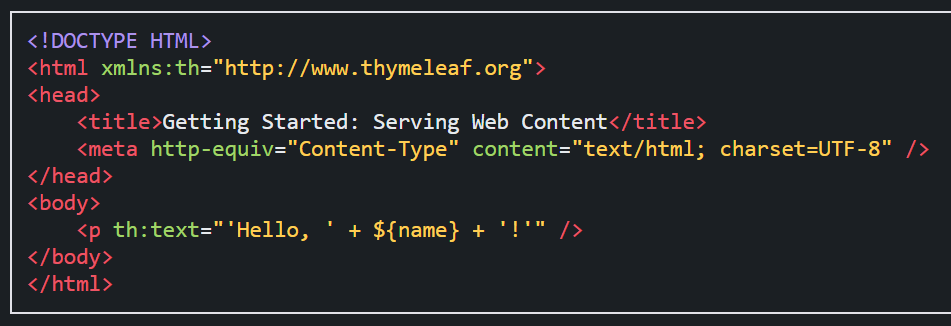
**APLICACIÓN WEB**

* Para poder crear una aplicación web, creamos nuevamente el proyecto MAVEN, esta vez dentro de dependencias incluimos la de Web y la de Thymeleaf. Para guiarnos mejor entramos a la documentación de SpringBoot: <https://spring.io/guides/gs/serving-web-content/>
* Ahora crearemos el controlador web, para esto en la documentación tenemos el código necesario, @Controller, @GetMapping



* Getmapping indica que se puede acceder a la función solamente con peticiones GET, en este caso @RequestParam indica que va a ir algún parámetro, en este caso parámetro Nombre, no es un parámetro obligatorio (required=false) y si no lo tiene, se coloca el World. String name, es donde se va a capturar ese parámetro y Model model es propio de MVC, y es donde se va a agregar las variables para luego mostrarlas en la vista. El return greeting, esta buscando la pagina que tiene ese nombre. Puede cambiarse la pagina desde el paréntesis del GetMapping. @GetMapping(“/paginabuscada”).
* Para probar esto, la documentación dice que debemos agregar una pagina en Resources/templates. Esto se hace y copiamos la pagina que sugiere la documentación, con esto pegamos la pagina que está en la documentación en el archivo html.
* 
* Como vemos, hay un párrafo “Hello” y un ${name}, esa variable es la que manejamos en el controlador, por eso se llama igual y debe hacerlo. $ Es un ExpressionLanguage EL.
* Entonces, en pocas palabras en el modelo, se tiene una variable nombre, esta misma variable se recupera o mapea en la vista.
* Si accedemos por el puerto de escucha a la pagina <http://localhost:8080/greeting>, nos va a aparecer Hello World que es el mensaje definido, pues como no se pasan parámetros lo que acompaña a Hello por default es World.
* En caso de que se quiera pasar algún parámetro, se agrega la variable name con su valor de esta manera: <http://localhost:8080/greeting?name=Jhonatan>
* Ahí realizará un Hello Jhonatan, donde Jhonatan es el parámetro que se esta pasando para la variable name.
* Es más fácil hacer el gitclone, que está en este repositorio:

git clone <https://github.com/spring-guides/gs-serving-web-content.git>