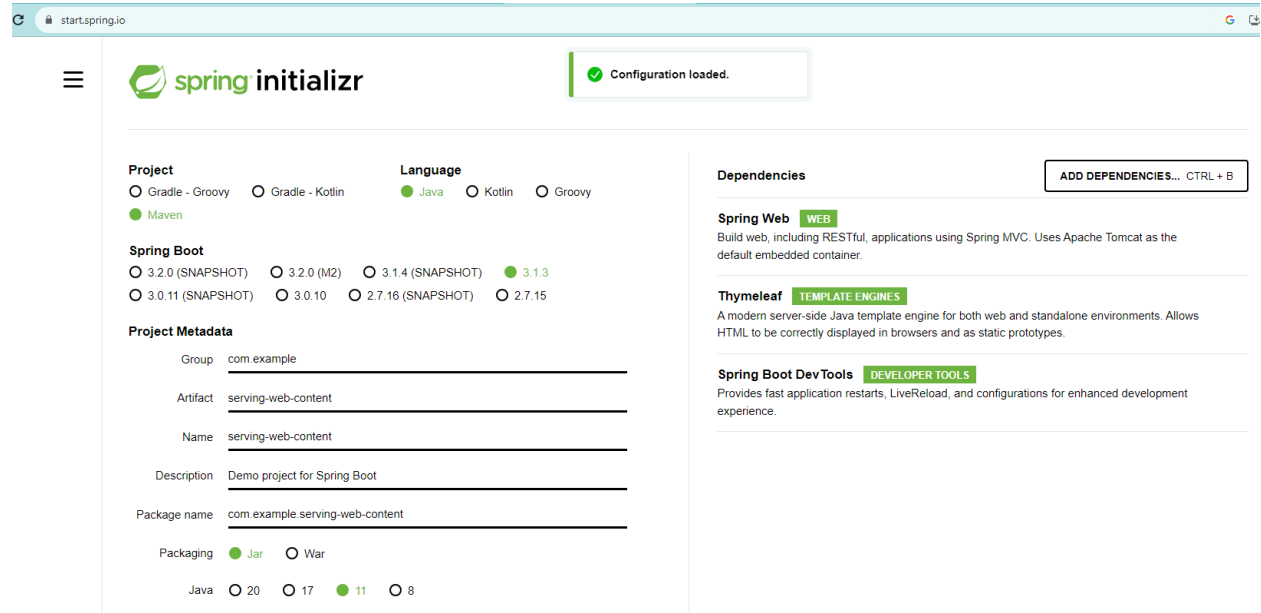


# How to Display html files in spring boot?

Go to start.spring.io and add following things

Step 1:-



The screenshot shows the Spring Initializr configuration page. At the top, there is a navigation menu with a hamburger icon and the 'spring initializr' logo. A notification box indicates 'Configuration loaded.' The main content area is divided into two columns. The left column contains the 'Project' configuration section, which includes options for 'Language' (Java is selected), 'Maven', 'Spring Boot' version (3.1.3 is selected), and 'Project Metadata' fields for Group, Artifact, Name, Description, and Package name. The right column contains the 'Dependencies' section, which lists 'Spring Web' (WEB), 'Thymeleaf' (TEMPLATE ENGINES), and 'Spring Boot Dev Tools' (DEVELOPER TOOLS). A button labeled 'ADD DEPENDENCIES... CTRL + B' is located at the top right of the dependencies section.

Note add following dependencies

## Dependencies

ADD DEPENDENCIES... CTRL + B

### Spring Web

WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.



### Thymeleaf

TEMPLATE ENGINES

A modern server-side Java template engine for both web and standalone environments. Allows HTML to be correctly displayed in browsers and as static prototypes.



### Spring Boot DevTools

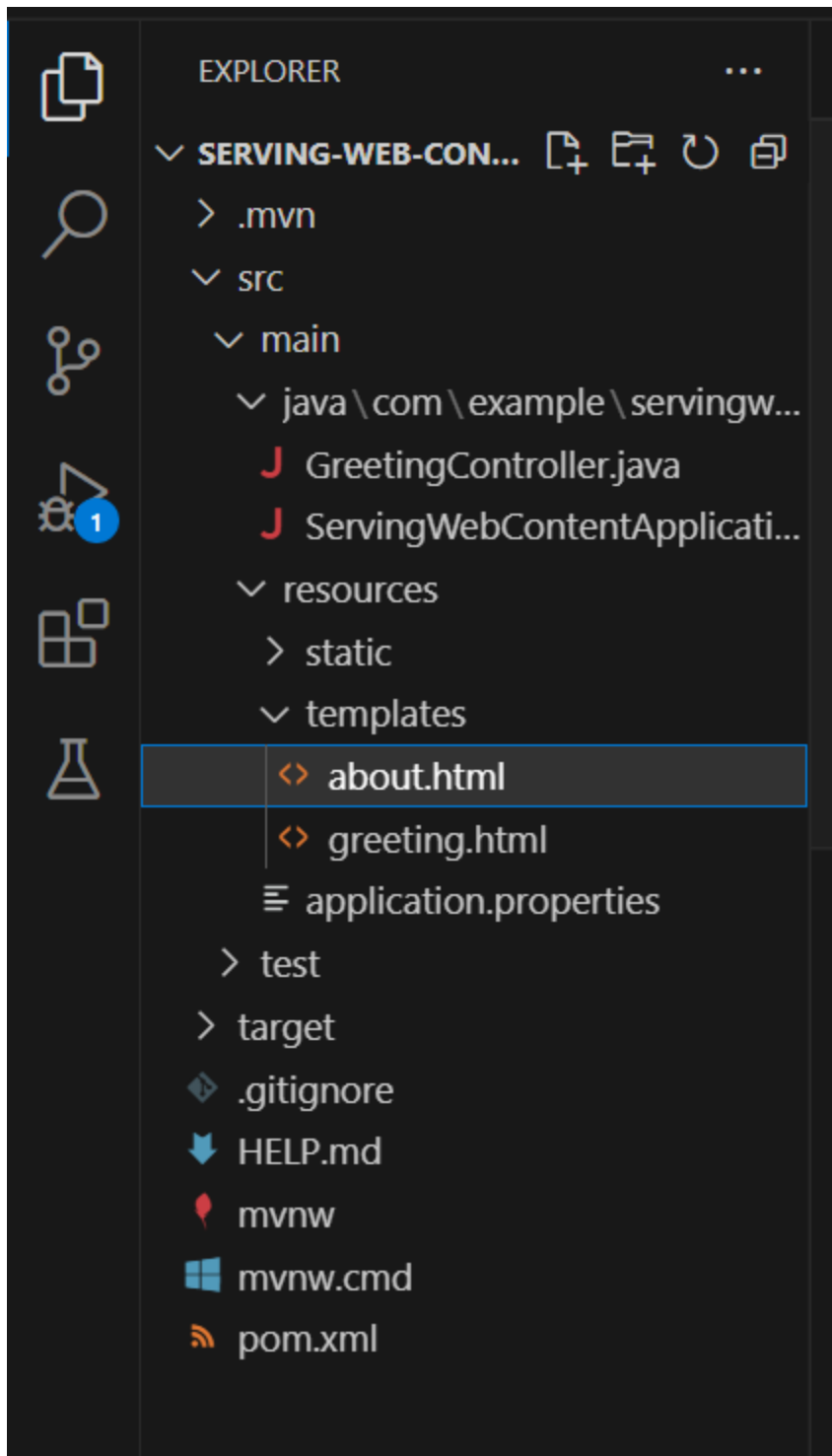
DEVELOPER TOOLS

Provides fast application restarts, LiveReload, and configurations for enhanced development experience.



And click on generate you will get zip and extract it and open with visual code





Now here you will create two html files under templates folder

**greeting.html** file under templates folder

```

<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <title>Getting Started: Serving Web Content</title>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
  <h1> Hello Om sir</h1>
  <p th:text="'Hello, ' + ${name} + '!'" />
</body>
</html>

```

**about.html** file code under templates folder :-

```

<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <title>Getting Started: Serving Web Content</title>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
  <h1> about Om sir</h1>
</body>
</html>

```

After it

Create GreetingController.java file under main package folder **servingwebcontent** :-

```

package com.example.servingwebcontent;

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

@Controller
public class GreetingController {

    @GetMapping("/greeting")
    public String greeting(@RequestParam(name="name", required=false,
defaultValue="World") String name, Model model) {
        model.addAttribute("name", name);
    }
}

```

```
        return "greeting";
    }

    @GetMapping("/about")
    public String about() {

        return "about";
    }
}
```

Your application file code will be like this

[ServingWebContentApplication.java](#) file

```
package com.example.servingwebcontent;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class ServingWebContentApplication {

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

}
```

Output:-



← → ↻ ⓘ localhost:8080/greeting

# Hello Om sir

Hello, World!

← → ↻ ⓘ localhost:8080/greeting?name=omsir

# Hello Om sir

Hello, omsir!