

How to start with spring boot web application with maven



Gagan Jain

Follow

2 min read

Nov 6, 2023

Listen

Share

Starting a Spring Boot web application with Maven is a common and straightforward process. You can create a simple web application by following these steps:

Image Ref: <https://www.interviewbit.com/blog/spring-boot-architecture/>

Prerequisites:

- Ensure you have Java and Maven installed on your system.

Step 1: Create a Spring Boot Project

1. Open a terminal or command prompt.
2. Navigate to the directory where you want to create your Spring Boot project.
3. Use the Spring Initializer tool, which is a convenient way to generate a Spring Boot project with the necessary dependencies:

```
mvn archetype:generate -DarchetypeGroupId=org.springframework.boot -DarchetypeArtifactId=spring-boot-archetype -DarchetypeVersion=2.7.0
```

In this command:

- `-DgroupId`: Specify the group ID for your project.
 - `-DartifactId`: Specify the project's artifact ID (your project's name).
 - `-Dversion`: Set the project's version.
 - `-DinteractiveMode=false`: Disable interactive mode.
1. After running the command, a new directory named `my-web-app` (or your specified project name) will be created with the Spring Boot project structure.

Step 2: Open the Project in an IDE

You can use an Integrated Development Environment (IDE) like IntelliJ IDEA, Eclipse, or Visual Studio Code to work with your project. Import the project into your preferred IDE.

Step 3: Write a Simple Web Application

Now that you have your project set up, you can start writing your Spring Boot web application:

1. Open the `src/main/java/com/example/mywebapp` (or your package structure) directory, and you will find a file named `MyWebAppApplication.java`. This file contains the main class for your Spring Boot application.
2. Create a simple Spring MVC controller to handle HTTP requests. You can create a new Java class, for example, `HelloController.java`, in the same package or a different package. Here's an example of a simple controller:

```
package com.example.mywebapp;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;
@RestController
public class HelloController {
    @GetMapping("/hello")
    public String hello() {
        return "Hello, World!";
    }
}
```

1. Start your Spring Boot application by running the `MyWebAppApplication` class.

Step 4: Access the Web Application

Your Spring Boot web application will start, and you can access it in a web browser or using a tool like `curl` or `httpie`. By default, it runs on <http://localhost:8080>.

- Open a web browser and go to: <http://localhost:8080/hello>

You should see the “Hello, World!” message.

Step 5: Customize and Build

You can further customize your Spring Boot application by adding more controllers, services, and views. You can also configure properties, add dependencies, and package your application for deployment.

To build a deployable JAR file for your application, run the following command in your project's root directory:

```
mvn clean package
```

This will create a JAR file in the `target` directory that you can run on any Java-enabled server or cloud platform.

That's a basic guide on how to start a Spring Boot web application using Maven. You can expand your application by adding more features, views, and dependencies based on your project's requirements.