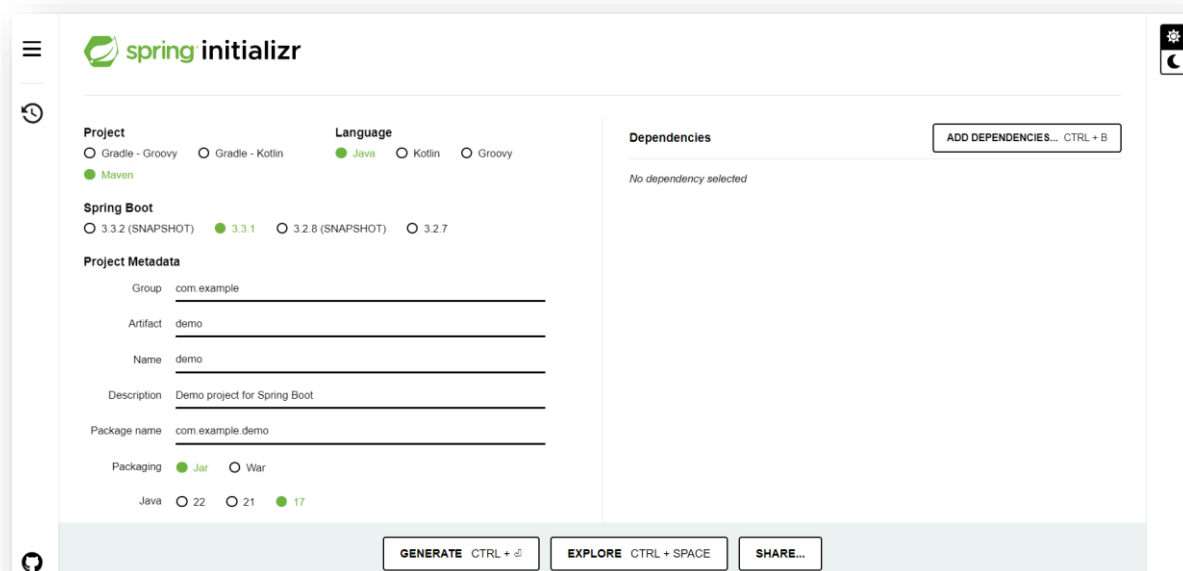


Guía práctica 01: Mi primer proyecto – Hola Mundo

En esta guía, detallaremos paso a paso el desarrollo de un proyecto en Spring llamado "Hola Mundo". A continuación, se describen los pasos para crear tu primer proyecto en Spring Boot.

Paso 1: Ir a Spring Initializr

- ❑ Abre tu navegador web y ve a la siguiente página: [Spring Initializr](https://start.spring.io/). (<https://start.spring.io/>)



The screenshot shows the Spring Initializr web application. The interface is divided into several sections:

- Project:** Radio buttons for **Gradle - Groovy**, **Gradle - Kotlin**, **Java** (selected), **Kotlin**, and **Groovy**.
- Spring Boot:** Radio buttons for **3.3.2 (SNAPSHOT)**, **3.3.1** (selected), **3.2.8 (SNAPSHOT)**, and **3.2.7**.
- Project Metadata:**
 - Group:**
 - Artifact:**
 - Name:**
 - Description:**
 - Package name:**
- Packaging:** Radio buttons for **Jar** (selected) and **War**.
- Java:** Radio buttons for **22**, **21**, and **17** (selected).
- Dependencies:** A section with the text "No dependency selected" and a button "ADD DEPENDENCIES... CTRL + B".
- Buttons:** At the bottom, there are three buttons: "GENERATE CTRL + G", "EXPLORE CTRL + SPACE", and "SHARE...".

Paso 2: Configurar el Proyecto

- ❑ En la página de **Spring Initializr**, verás varios campos y opciones para configurar tu proyecto. Completa los campos de la siguiente manera:
 - **Project:** Selecciona "Maven".
 - **Language:** Selecciona "Java".
 - **Spring Boot:** Selecciona la versión más reciente estable (3.3.1)
 - **Project Metadata:** (escribe)
 - **Group:** com.holamundo.ejemplo
 - **Artifact:** holamundo
 - **Name:** holamundo
 - **Description:** Proyecto Hola Mundo en Spring Boot
 - **Package name:** (autogenera)
 - **Packaging:** Jar
 - **Java:** Selecciona la versión de Java que tengas instalada (ej. 17).

Project

☐ Gradle - Groovy
 ☐ Gradle - Kotlin
 ☒ **Java**
☐ Kotlin
 ☐ Groovy

☒ **Maven**

Spring Boot

☐ 3.3.2 (SNAPSHOT)
 ☒ **3.3.1**
☐ 3.2.8 (SNAPSHOT)
 ☐ 3.2.7

Project Metadata

Group

Artifact

Name

Description

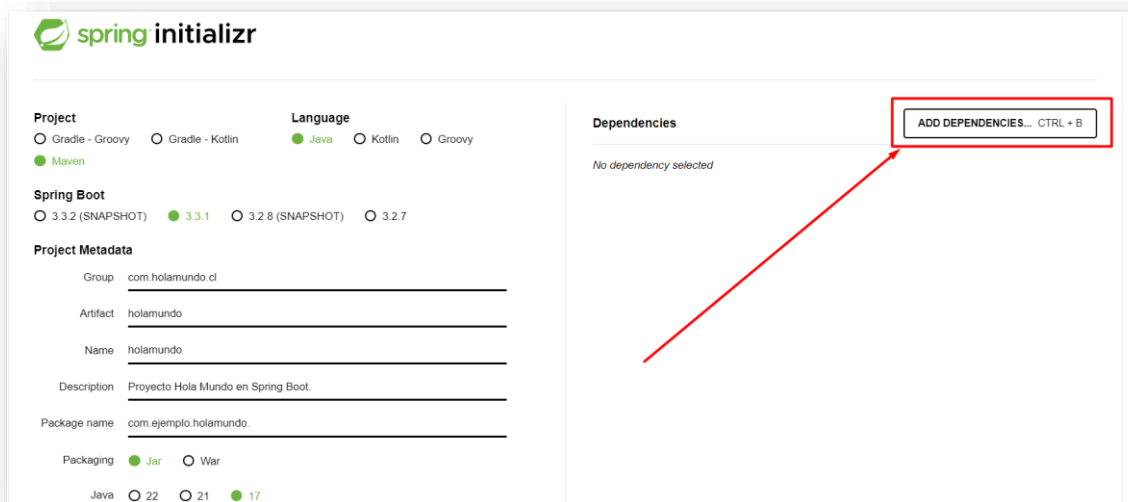
Package name

Packaging ☒ **Jar** ☐ War

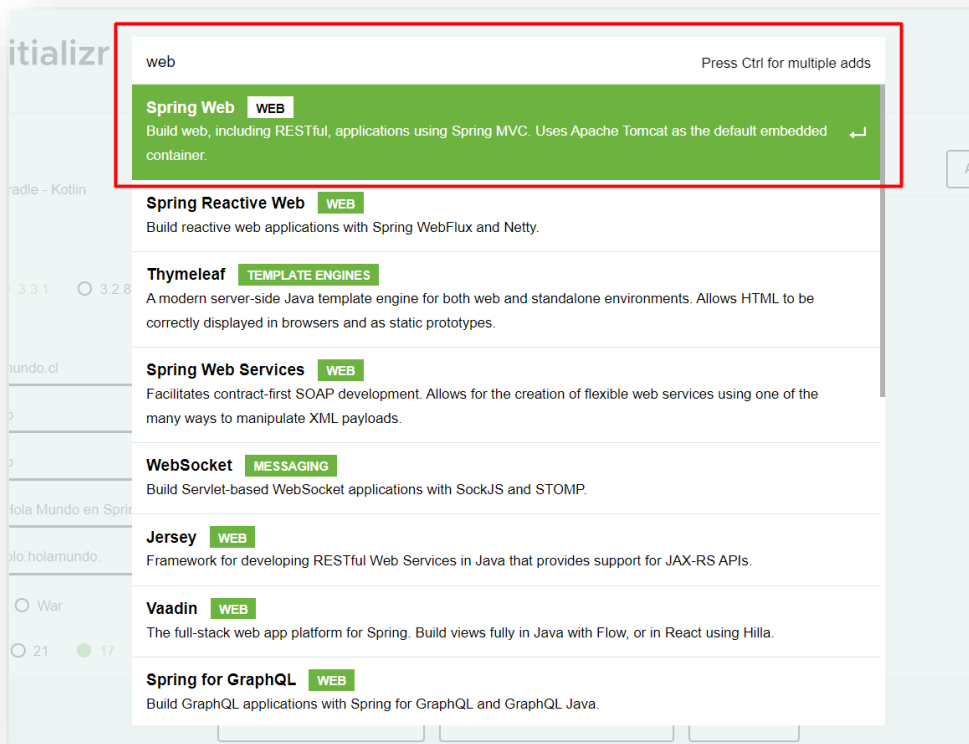
Java ☐ 22 ☐ 21 ☒ **17**

Paso 3: Añadir Dependencias

- ☐ Haz clic en el botón "Add Dependencies" y añade la siguiente dependencia:
- Spring Web:** Para crear aplicaciones web, incluyendo RESTful.

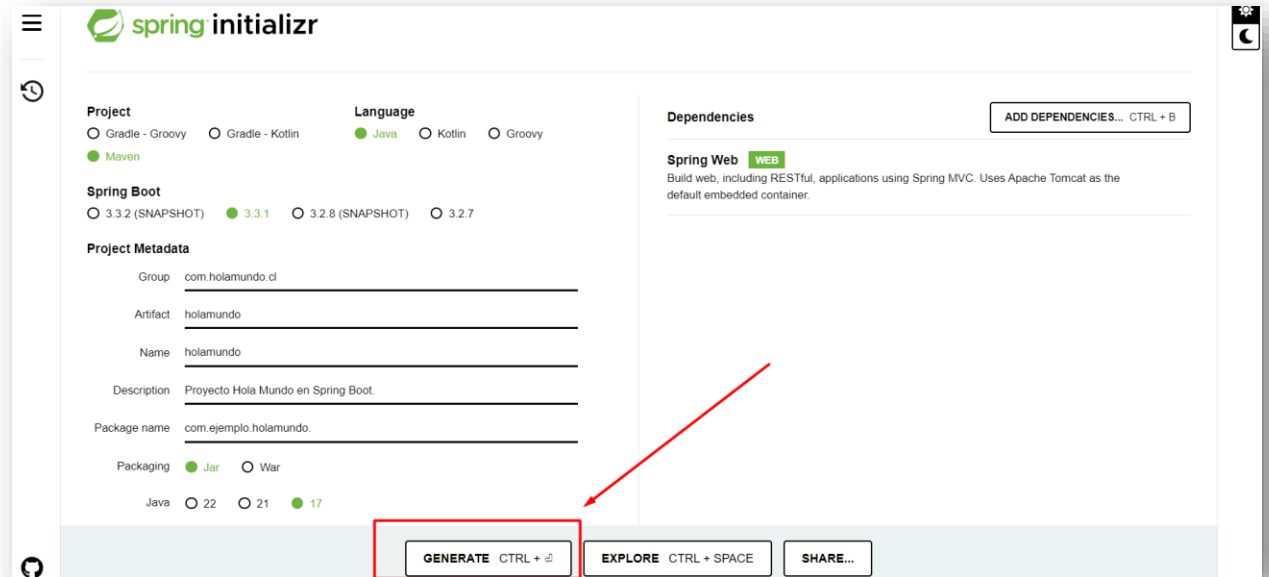


The image shows the Spring Initializr web interface. On the left, the project configuration is identical to the previous screenshot. On the right, the 'Dependencies' section is visible, showing 'No dependency selected'. A red box highlights the 'ADD DEPENDENCIES... CTRL + B' button, with a red arrow pointing to it from below.



Paso 4: Generar el Proyecto

- ❑ Una vez que hayas configurado todo, haz clic en el botón **"GENERATE"** para descargar un archivo ZIP con tu proyecto Spring Boot.



The image shows the Spring Initializr web application interface. The page is titled "spring initializr" and features a sidebar with navigation icons. The main content area is divided into several sections: "Project" with radio buttons for "Gradle - Groovy", "Gradle - Kotlin", "Java" (selected), "Kotlin", and "Groovy"; "Maven" (selected); "Spring Boot" with radio buttons for "3.3.2 (SNAPSHOT)", "3.3.1" (selected), "3.2.8 (SNAPSHOT)", and "3.2.7"; "Project Metadata" with input fields for "Group" (com.holamundo.cl), "Artifact" (holamundo), "Name" (holamundo), "Description" (Proyecto Hola Mundo en Spring Boot), and "Package name" (com.ejemplo.holamundo); "Packaging" with radio buttons for "Jar" (selected) and "War"; and "Java" with radio buttons for "22", "21", and "17" (selected). The "Dependencies" section on the right has a button "ADD DEPENDENCIES... CTRL + B" and a "Spring Web" dependency with a "WEB" tag. At the bottom, there are three buttons: "GENERATE CTRL + G" (highlighted with a red box and a red arrow), "EXPLORE CTRL + SPACE", and "SHARE...".

Project
☐ Gradle - Groovy ☐ Gradle - Kotlin ☒ Java ☐ Kotlin ☐ Groovy
☒ Maven

Spring Boot
☐ 3.3.2 (SNAPSHOT) ☒ 3.3.1 ☐ 3.2.8 (SNAPSHOT) ☐ 3.2.7

Project Metadata
Group
Artifact
Name
Description
Package name

Packaging ☒ Jar ☐ War
Java ☐ 22 ☐ 21 ☒ 17

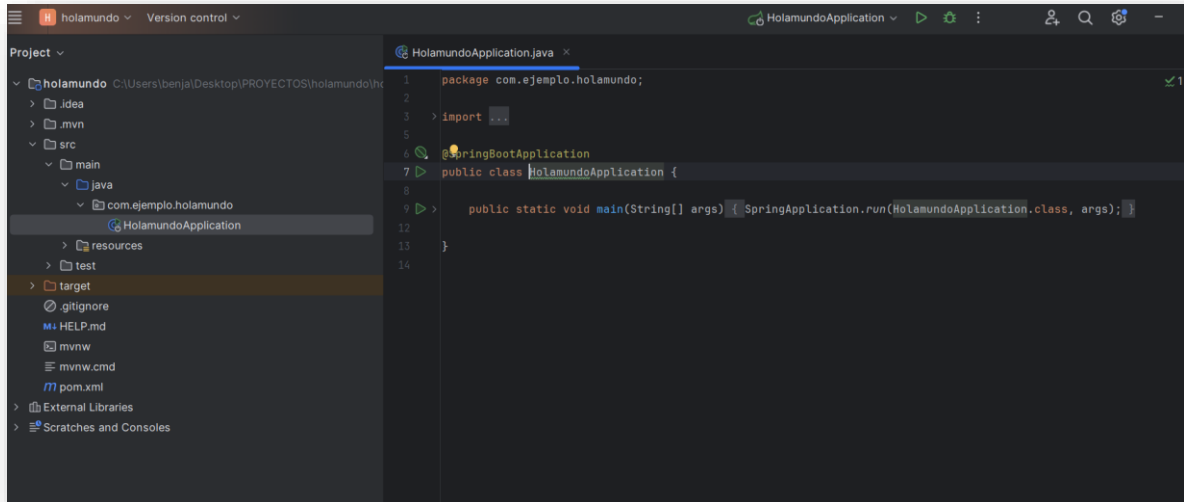
Dependencies ADD DEPENDENCIES... CTRL + B

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

GENERATE CTRL + G **EXPLORE** CTRL + SPACE **SHARE...**

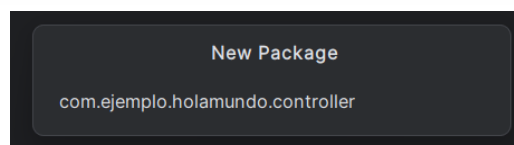
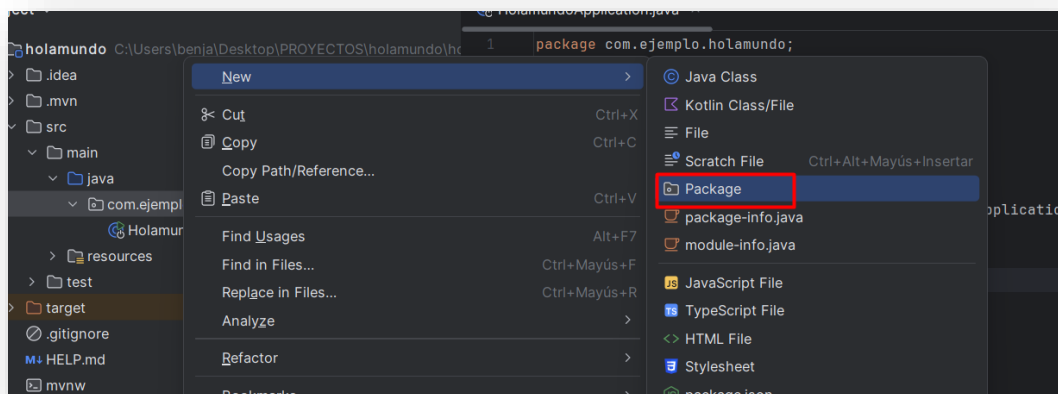
Paso 5: Importar el Proyecto en tu IDE

- ❑ Descomprime el archivo ZIP descargado y abre tu IDE preferido (VS Code o IntelliJ IDEA).
 - Navega hasta la carpeta descomprimida y selecciona el proyecto para importarlo.

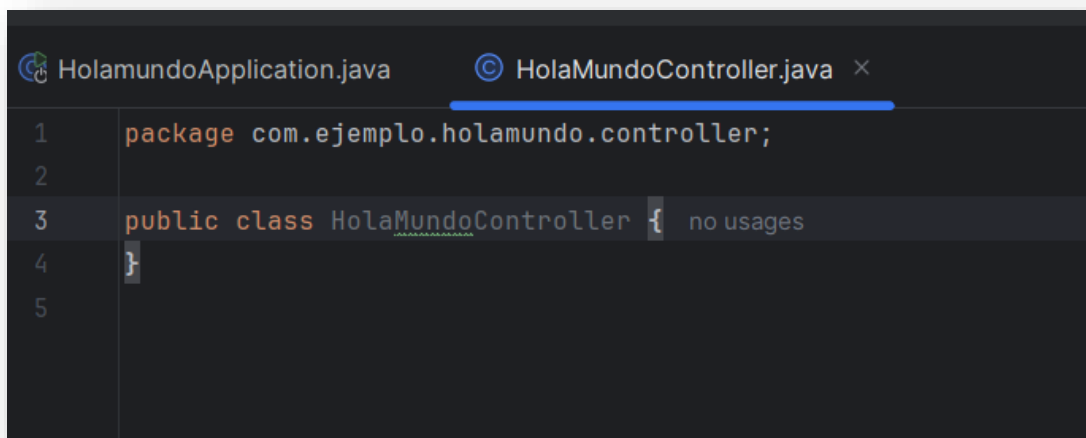
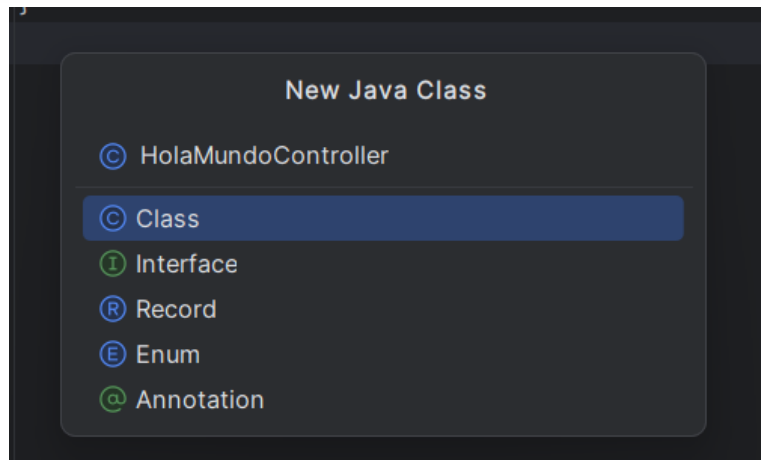
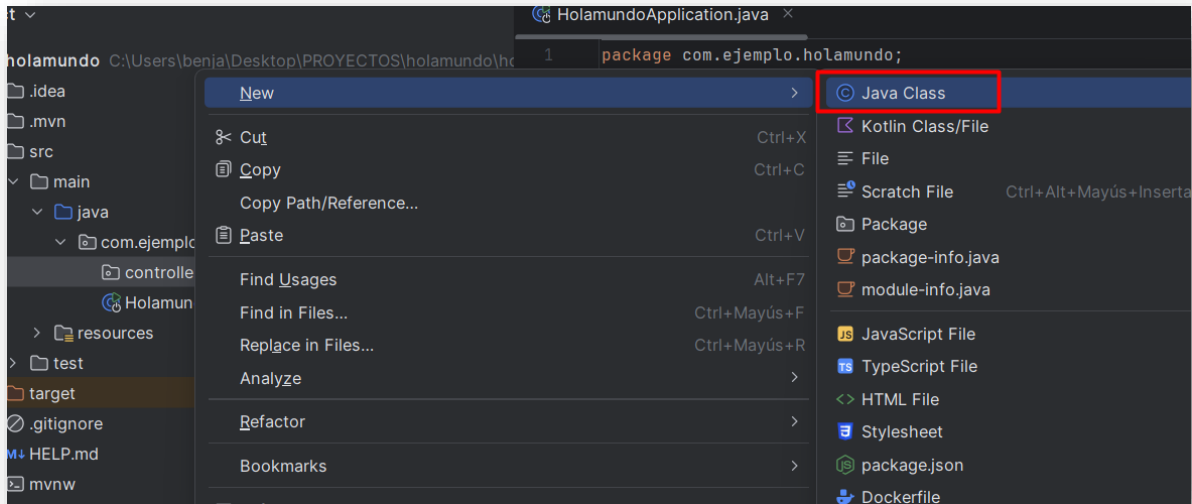


Paso 6: Crear un Controlador

- ❑ Dentro de tu proyecto en el IDE, navega a *src/main/java/com/ejemplo/holamundo/*
 - Crea un nuevo paquete(carpetita) llamado **controller**.

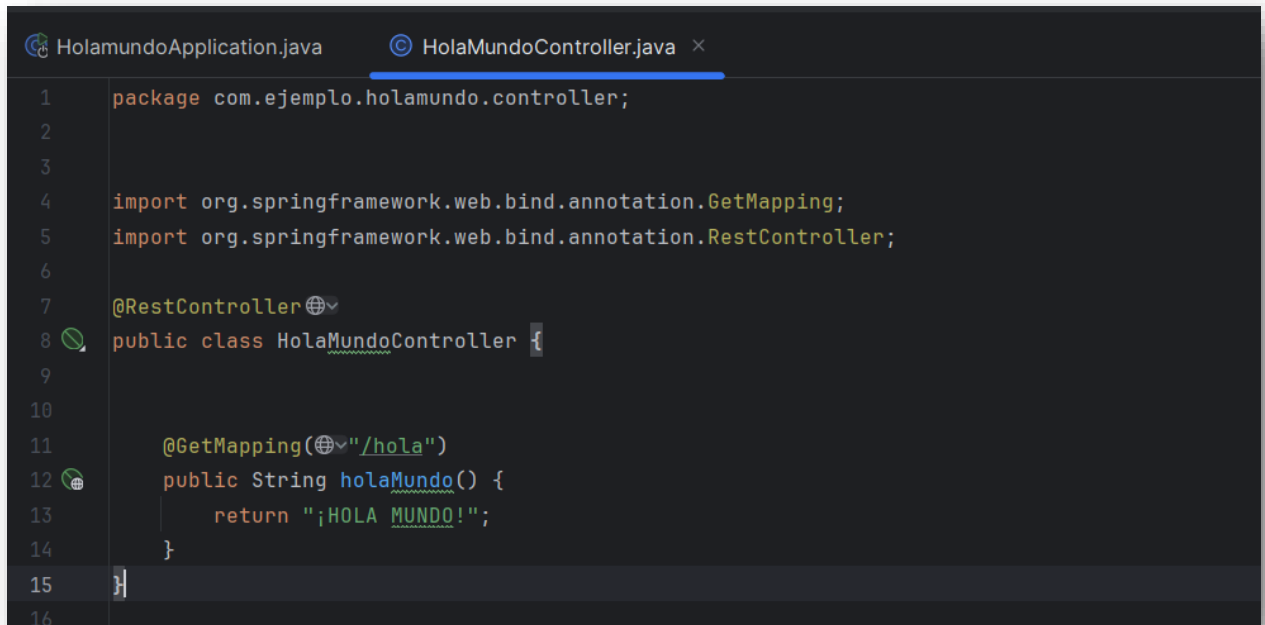


- Dentro del paquete **controller**, crea una nueva clase llamada **HolaMundoController**.



- Escribe el siguiente código en HolaMundoController.java:
 - Añadir los imports

- Crear la clase String `holaMundo()`, que retorne “¡Hola mundo!”



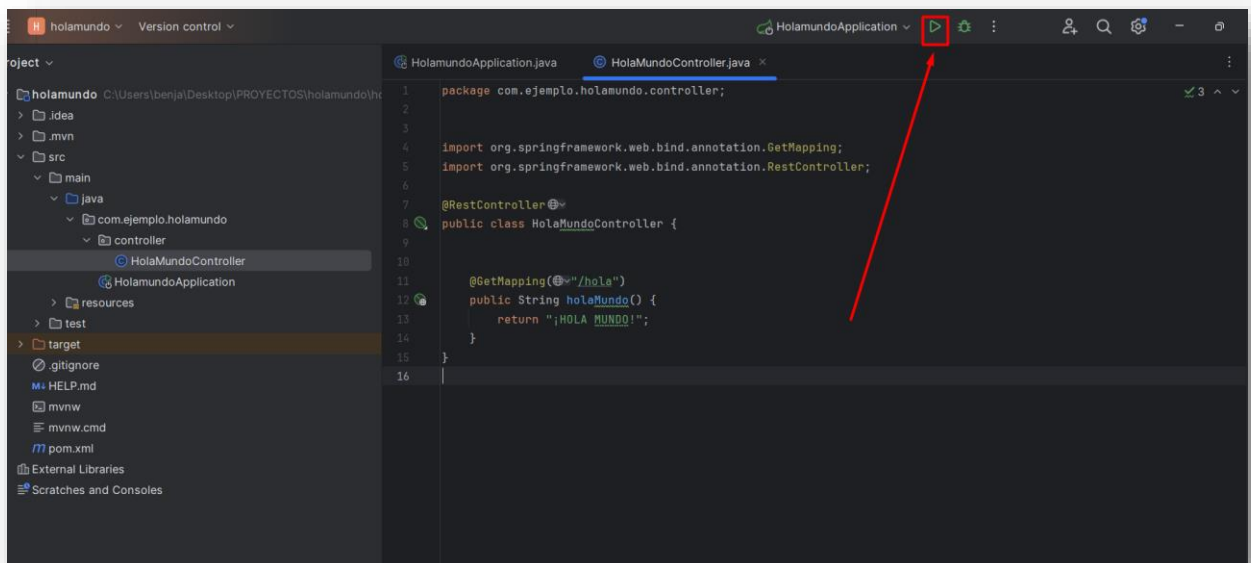
```

1 package com.ejemplo.holamundo.controller;
2
3
4 import org.springframework.web.bind.annotation.GetMapping;
5 import org.springframework.web.bind.annotation.RestController;
6
7 @RestController
8 public class HolaMundoController {
9
10
11     @GetMapping("/hola")
12     public String holaMundo() {
13         return "¡HOLA MUNDO!";
14     }
15
16

```

Paso 7: Ejecutar la Aplicación

- ❑ Dentro del IDE, busca la clase **HolamundoApplication** en el paquete principal y ejecútala como una aplicación Java.



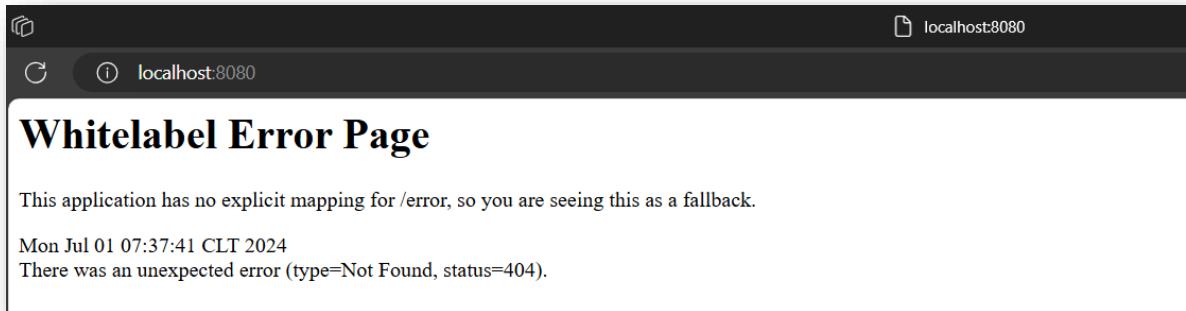
- ❑ Consola con resultados correctos

The screenshot displays an IDE with the following components:

- Source Explorer:** Shows the project structure with a package `com.ejemplo.holamundo` containing a `HolaMundoController` class.
- Code Editor:** Contains the Java code for `HolaMundoController`, which is a REST controller with a `GetMapping` annotation on the `hola` endpoint.
- Run Configuration:** A configuration for `HolamundoApplication` is selected.
- Console:** Displays the output of the application startup, including the Spring Boot banner and logs indicating that the application is running on port 8080.

Paso 8: Probar la Aplicación

- ❑ Ingresar a <http://localhost:8080/> verificar ruta
 - Dara un mensaje de error porque no tenemos nada asignado en el root



- ❑ Abre tu navegador web y ve a <http://localhost:8080/hola>
 - Deberías ver el mensaje **¡Hola, Mundo!**.



iii 🎉 🎉 🐾 FELICIDADES 🐾 🎉 🎉 !!!

Has creado tu primer proyecto en Spring y has implementado un sencillo **endpoint** que devuelve "¡Hola, Mundo!". Ahora ya conoces los pasos básicos para configurar, desarrollar y ejecutar un proyecto Spring Boot.

Estás listo para explorar más funcionalidades y construir aplicaciones en Spring.

