

Report Lab1- Fintech Lab

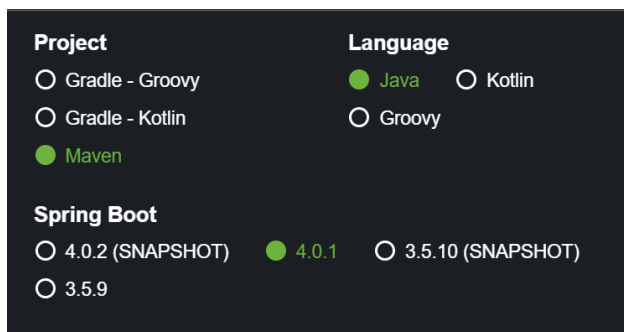
Name: Tanvika Shetty

Regno: 240958001

Spring Initializer:

Q1) Setup the spring boot application and print the hello world message.

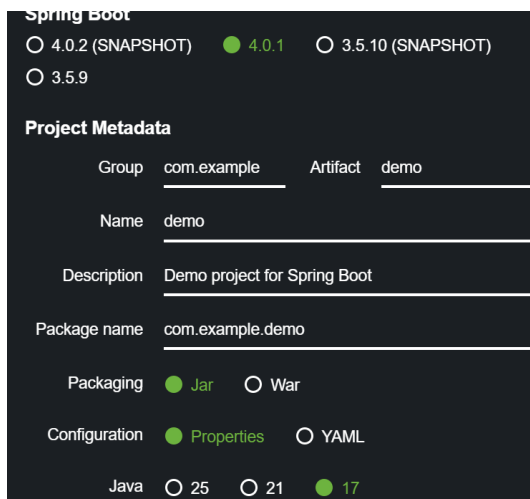
- Created a new Spring Boot project using Spring Initialize.
- Selected Maven Project, Java, and Spring Boot version.
- Added the Spring Web dependency.
- Generated the project and imported it into the IDE.
- Created a REST controller using `@RestController`.
- Defined a GET mapping to return a Hello World message.
- Ran the application using the main Spring Boot class.



This screenshot shows the initial configuration options for a Spring Boot project. Under the 'Project' section, 'Maven' is selected. Under the 'Language' section, 'Java' is selected. Under the 'Spring Boot' section, version '4.0.1' is selected.

| Project | Language |
|--|--|
| <input type="radio"/> Gradle - Groovy | <input checked="" type="radio"/> Java <input type="radio"/> Kotlin |
| <input type="radio"/> Gradle - Kotlin | <input type="radio"/> Groovy |
| <input checked="" type="radio"/> Maven | |

| Spring Boot |
|---|
| <input type="radio"/> 4.0.2 (SNAPSHOT) <input checked="" type="radio"/> 4.0.1 <input type="radio"/> 3.5.10 (SNAPSHOT) |
| <input type="radio"/> 3.5.9 |

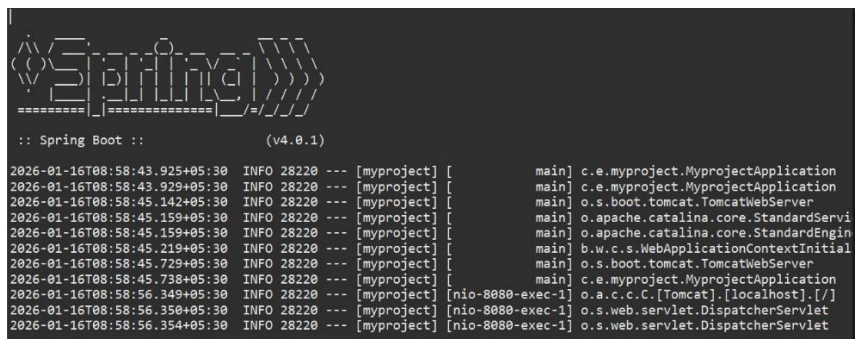


This screenshot shows the 'Project Metadata' configuration for the Spring Boot project. Fields include Group (com.example), Artifact (demo), Name (demo), Description (Demo project for Spring Boot), Package name (com.example.demo), Packaging (Jar), Configuration (Properties), and Java version (17).

| Project Metadata | |
|------------------|---|
| Group | com.example |
| Artifact | demo |
| Name | demo |
| Description | Demo project for Spring Boot |
| Package name | com.example.demo |
| Packaging | <input checked="" type="radio"/> Jar <input type="radio"/> War |
| Configuration | <input checked="" type="radio"/> Properties <input type="radio"/> YAML |
| Java | <input type="radio"/> 25 <input type="radio"/> 21 <input checked="" type="radio"/> 17 |

Q2) Setup the spring boot application with hello world message in separate package named Controller

- A Spring Boot project was created using Spring Initializr with Java 17 and Maven configuration.
- The project metadata was configured with package name `com.example.demo`.
- The Spring Web dependency was added to enable RESTful web services.
- A new package named **controller** was created inside the base package.
- A controller class was created and annotated with `@RestController`.
- A GET mapping was defined to return a Hello World message.
- The application was executed successfully using the embedded Tomcat server.



```

Spring
=====
:: Spring Boot :: (v4.0.1)

2026-01-16T08:58:43.925+05:30 INFO 28220 --- [myproject] [main] c.e.myproject.MyprojectApplication
2026-01-16T08:58:43.929+05:30 INFO 28220 --- [myproject] [main] c.e.myproject.MyprojectApplication
2026-01-16T08:58:45.142+05:30 INFO 28220 --- [myproject] [main] o.s.boot.tomcat.TomcatWebServer
2026-01-16T08:58:45.159+05:30 INFO 28220 --- [myproject] [main] o.apache.catalina.core.StandardServi
2026-01-16T08:58:45.159+05:30 INFO 28220 --- [myproject] [main] o.apache.catalina.core.StandardEngin
2026-01-16T08:58:45.219+05:30 INFO 28220 --- [myproject] [main] b.w.c.s.WebApplicationContextInitial
2026-01-16T08:58:45.729+05:30 INFO 28220 --- [myproject] [main] o.s.boot.tomcat.TomcatWebServer
2026-01-16T08:58:45.738+05:30 INFO 28220 --- [myproject] [main] c.e.myproject.MyprojectApplication
2026-01-16T08:58:56.349+05:30 INFO 28220 --- [myproject] [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/]
2026-01-16T08:58:56.350+05:30 INFO 28220 --- [myproject] [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet
2026-01-16T08:58:56.354+05:30 INFO 28220 --- [myproject] [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet

```

Q3) Test the above application in POSTMAN.

- The Spring Boot application was started successfully.
- POSTMAN was opened to test the REST API.
- A new request was created in POSTMAN.
- The HTTP method was set to GET.

