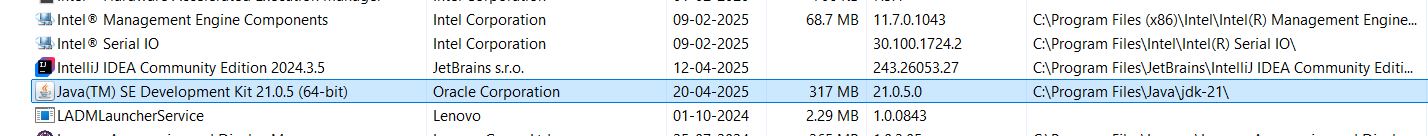
**Day1: Download, Install and configure**

Download -> Install -> configure path in system environment variables

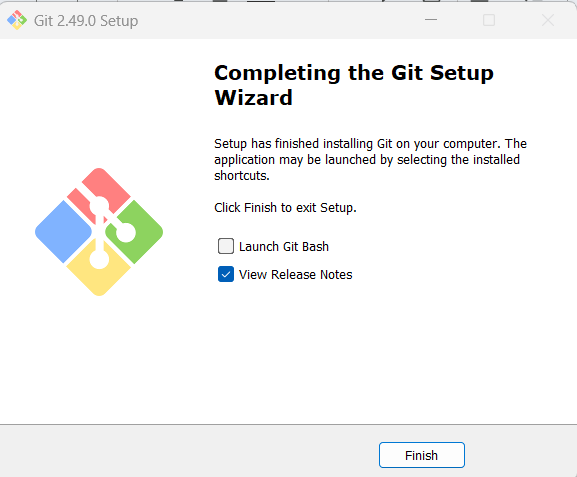
* Java

<https://download.oracle.com/java/21/archive/jdk-21.0.5_windows-x64_bin.exe>



* Git

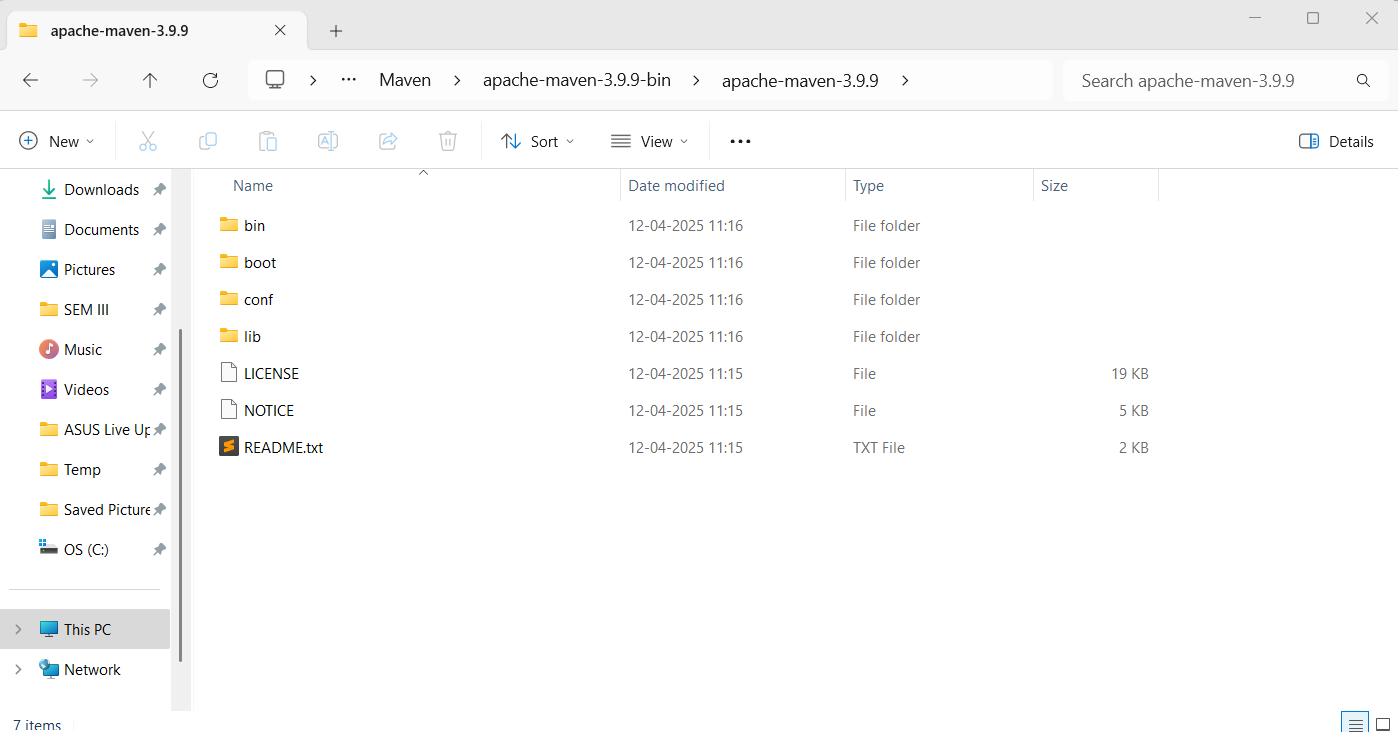
<https://github.com/git-for-windows/git/releases/download/v2.49.0.windows.1/Git-2.49.0-64-bit.exe>



Download --> configure path in system environment variables

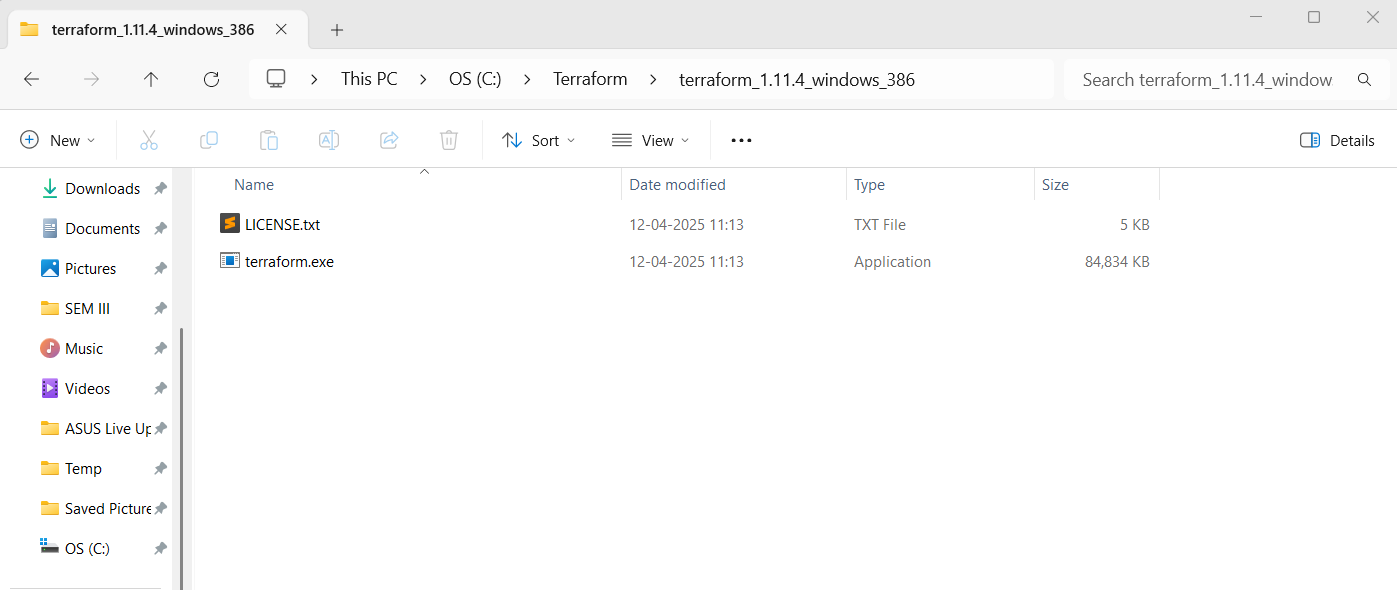
* Maven

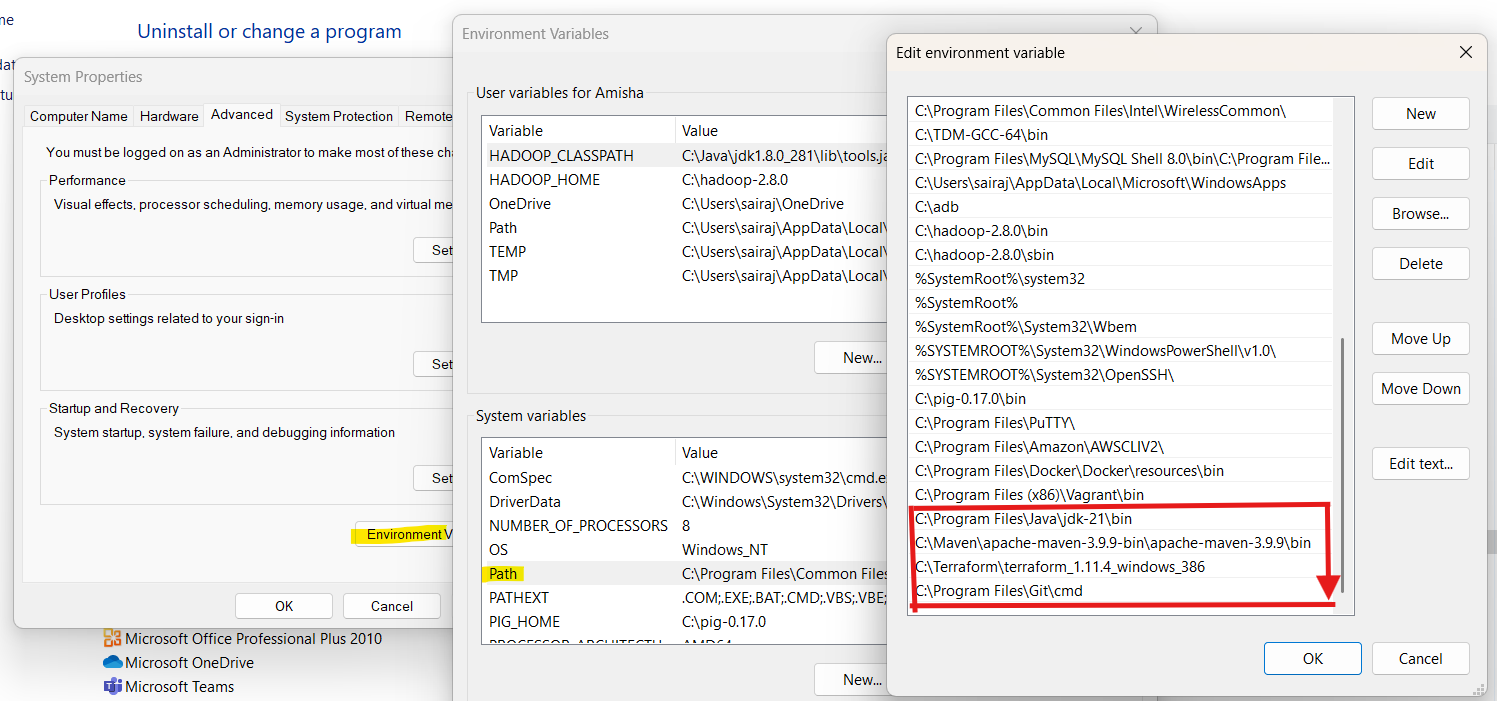
<https://dlcdn.apache.org/maven/maven-3/3.9.9/binaries/apache-maven-3.9.9-bin.zip>



* Terraform

<https://releases.hashicorp.com/terraform/1.11.4/terraform_1.11.4_windows_386.zip>





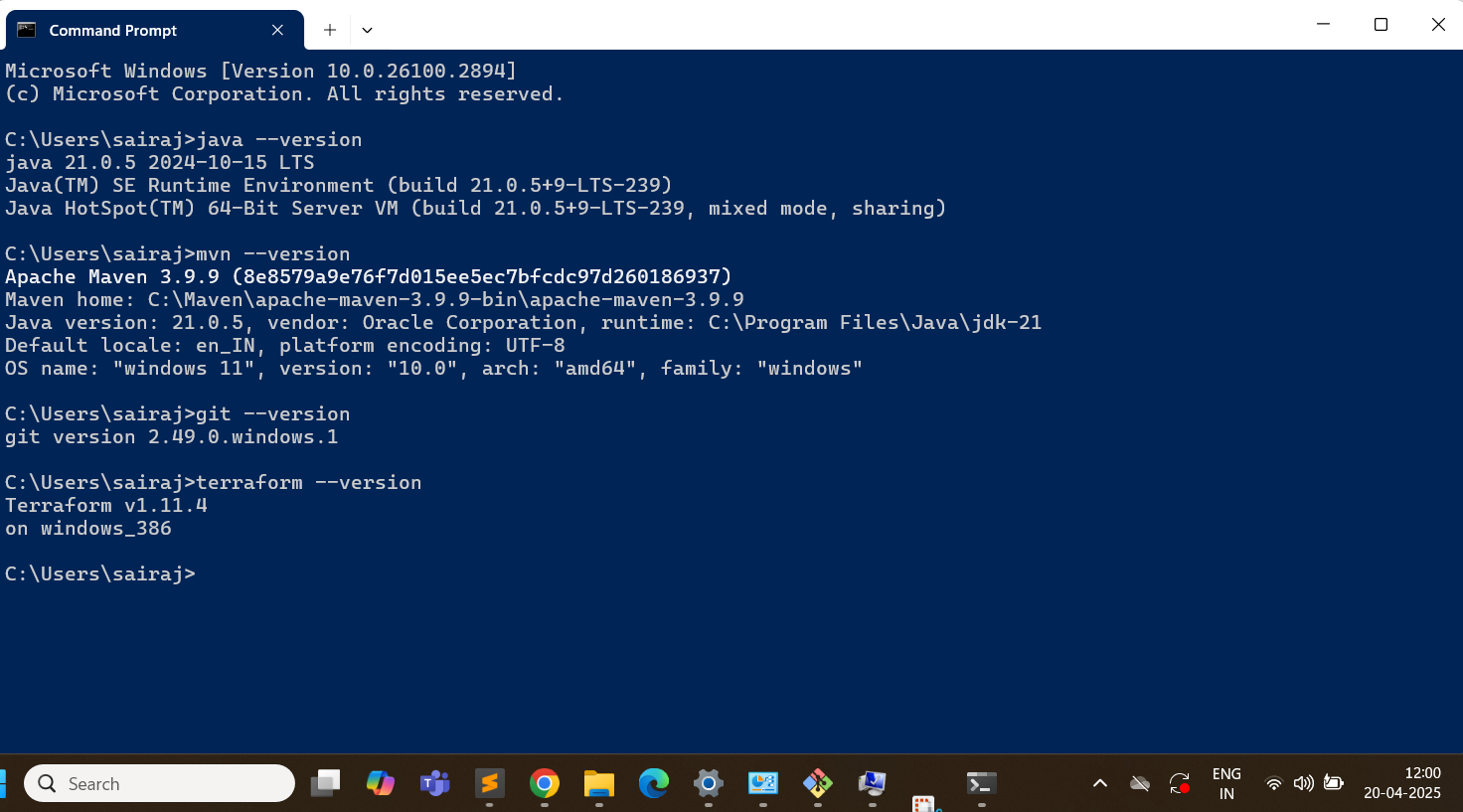
**Validate the installation:**

>java --version

>mvn --version

>git --version

>terraform --version



**Install IntelliJ IDEA & VSCode:**

IntelliJ IDEA Community Edition

<https://www.jetbrains.com/idea/download/?section=windows>

VSCode

<https://code.visualstudio.com/download>

**Create accounts:**

* AWS account
* Github Account
* Docker hub account

**Day3: Creating our first microservices**

Go to <https://start.spring.io/>

Select:

Project: Maven

Language: Java,

Springboot version: 3.310

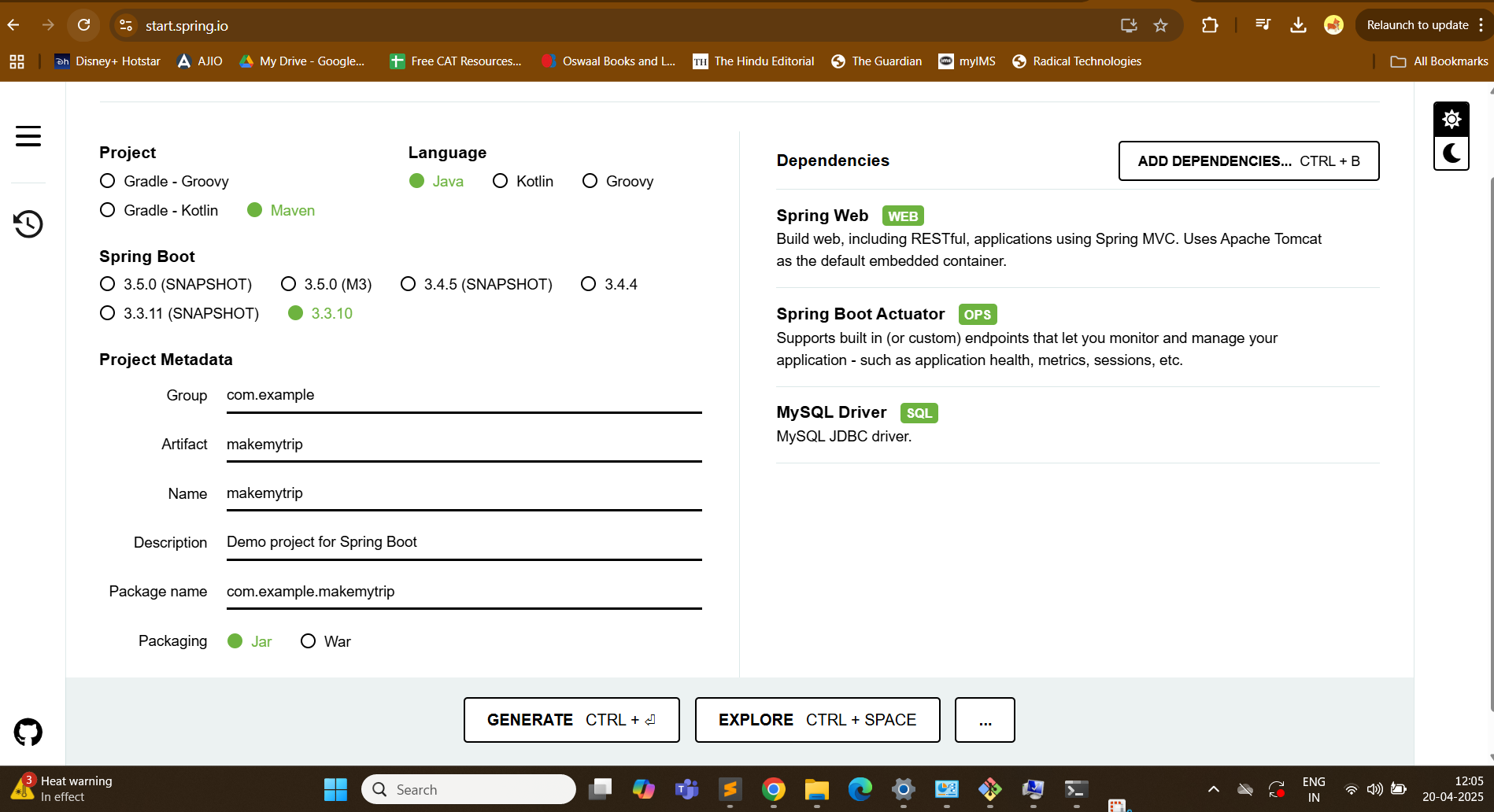
Artifact name: makemytrip

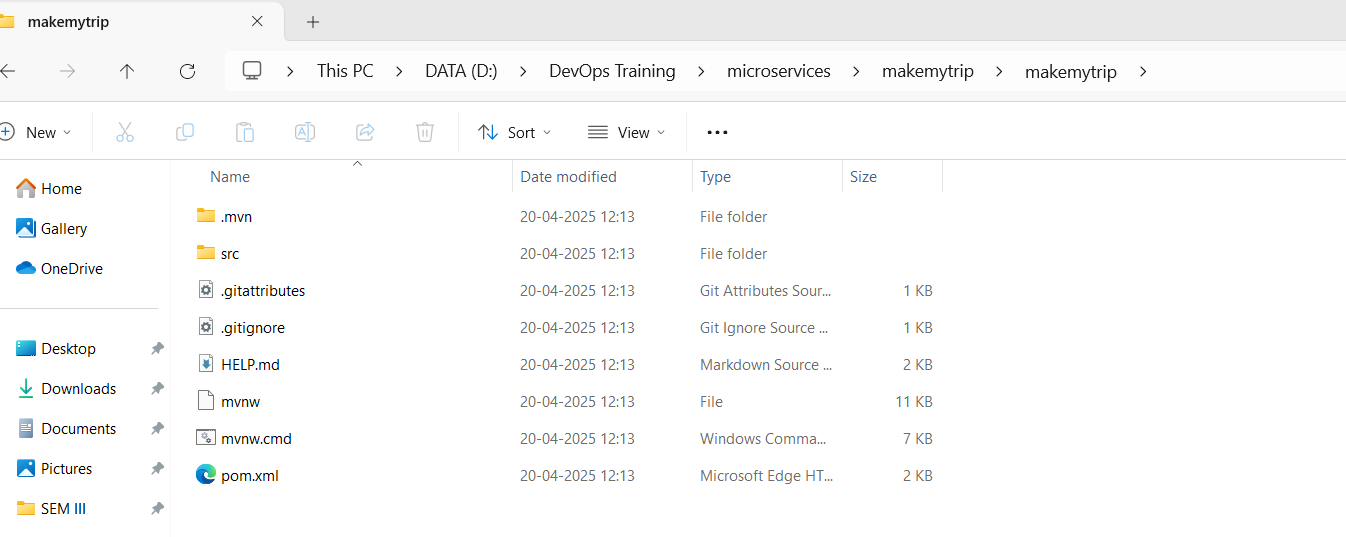
Packaging: Jar

Add the dependencies: Spring Web, String Boot Actuator

Generate the zip file

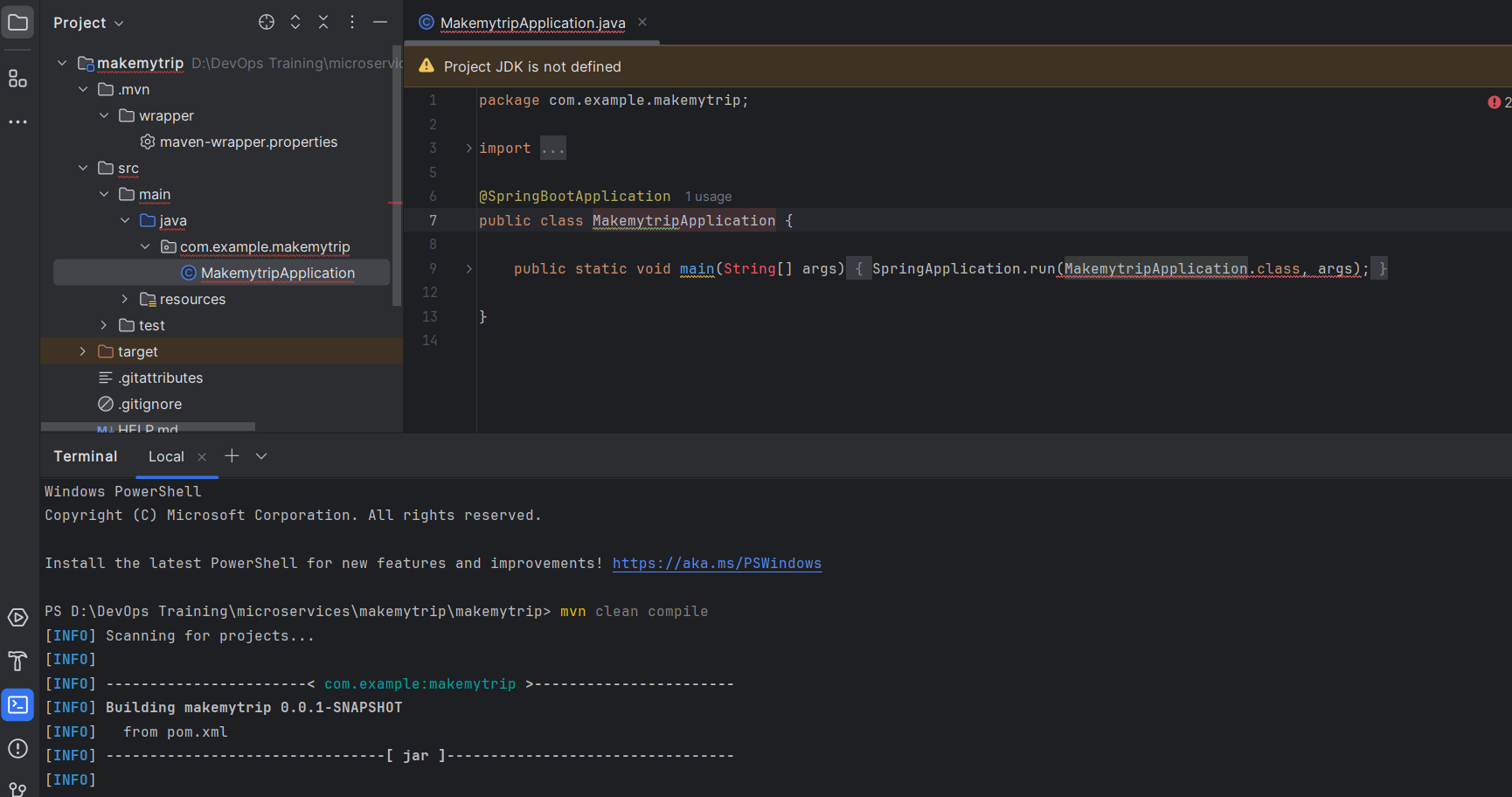
Extract the zip file



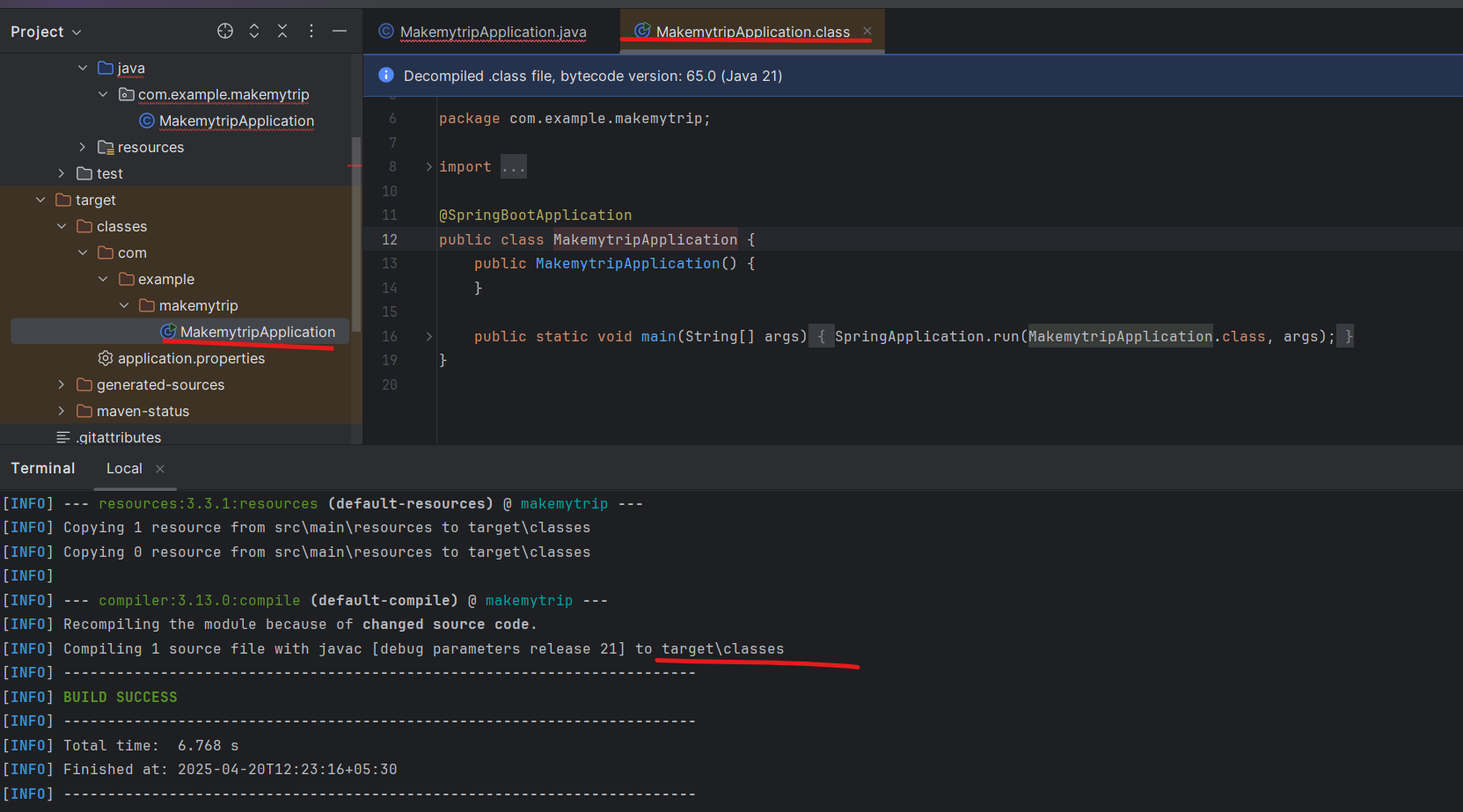


> mvn clean compile

(Creates .class file)



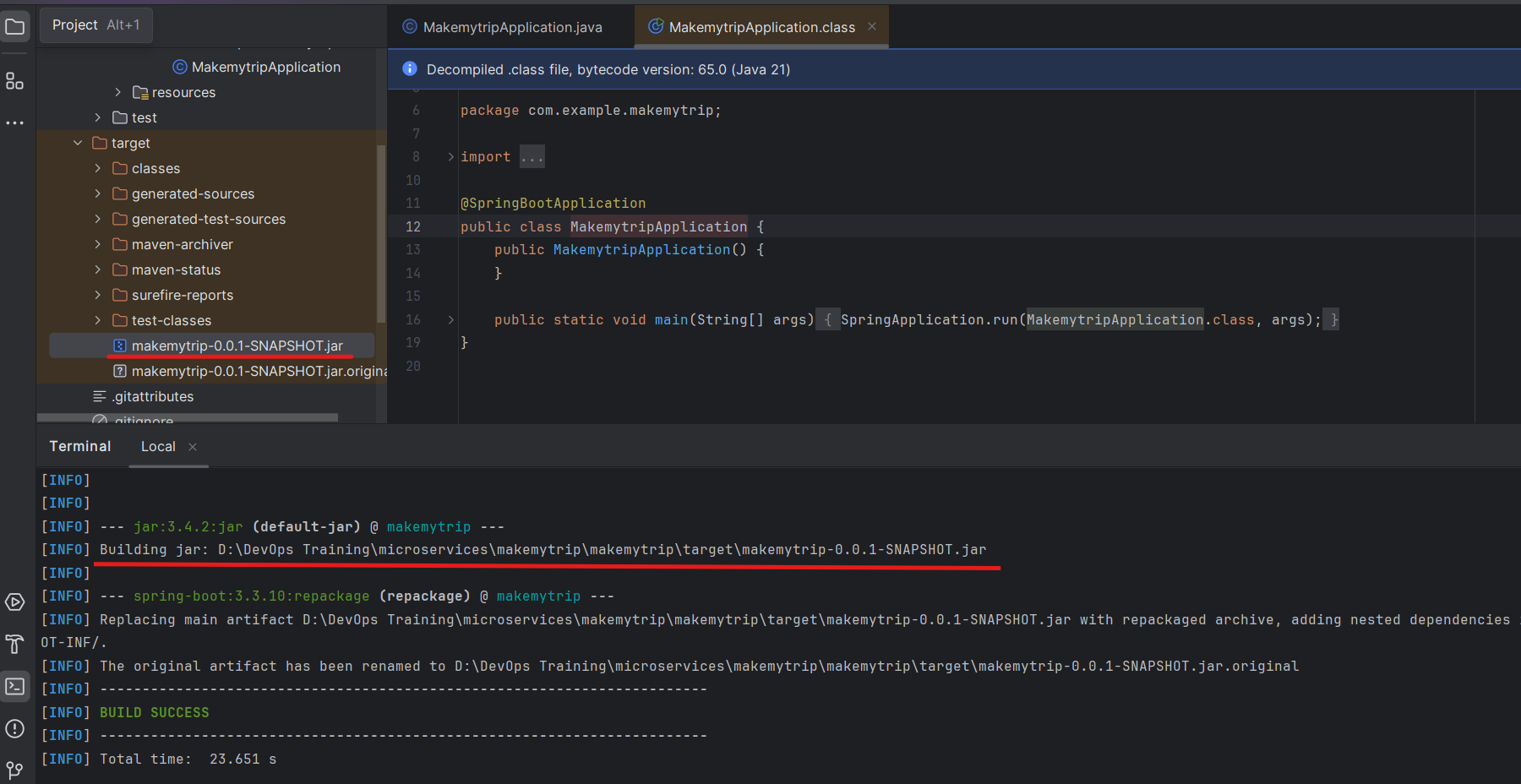
Location of .class file



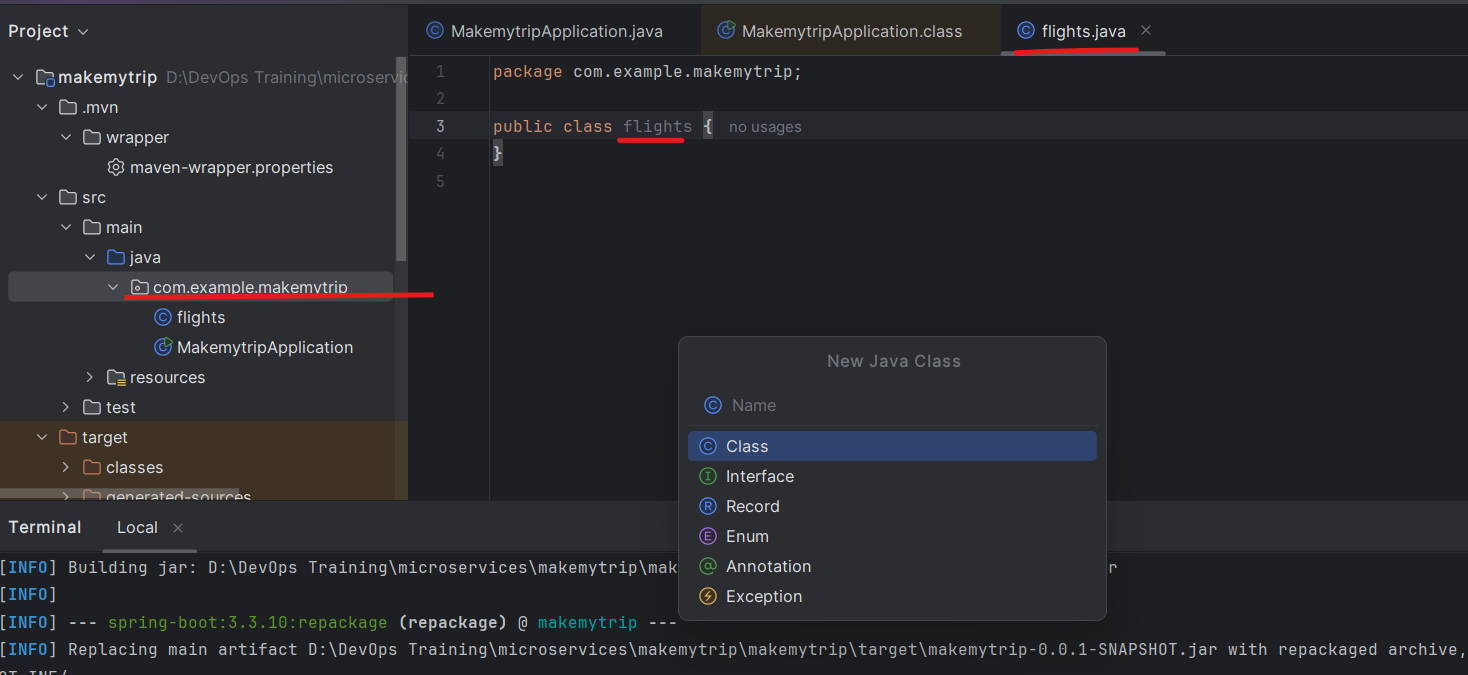
Now package:

> mvn clean package

Default package is .jar



Now add a java fileRight click on java skeleton > new > java class > flights



Add following to to this file flights.java:

package com.example.makemytrip;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class flights {

@GetMapping("/flights")

public String getData() {return "Book your FLIGHTS from Kolkata to Pune!";}

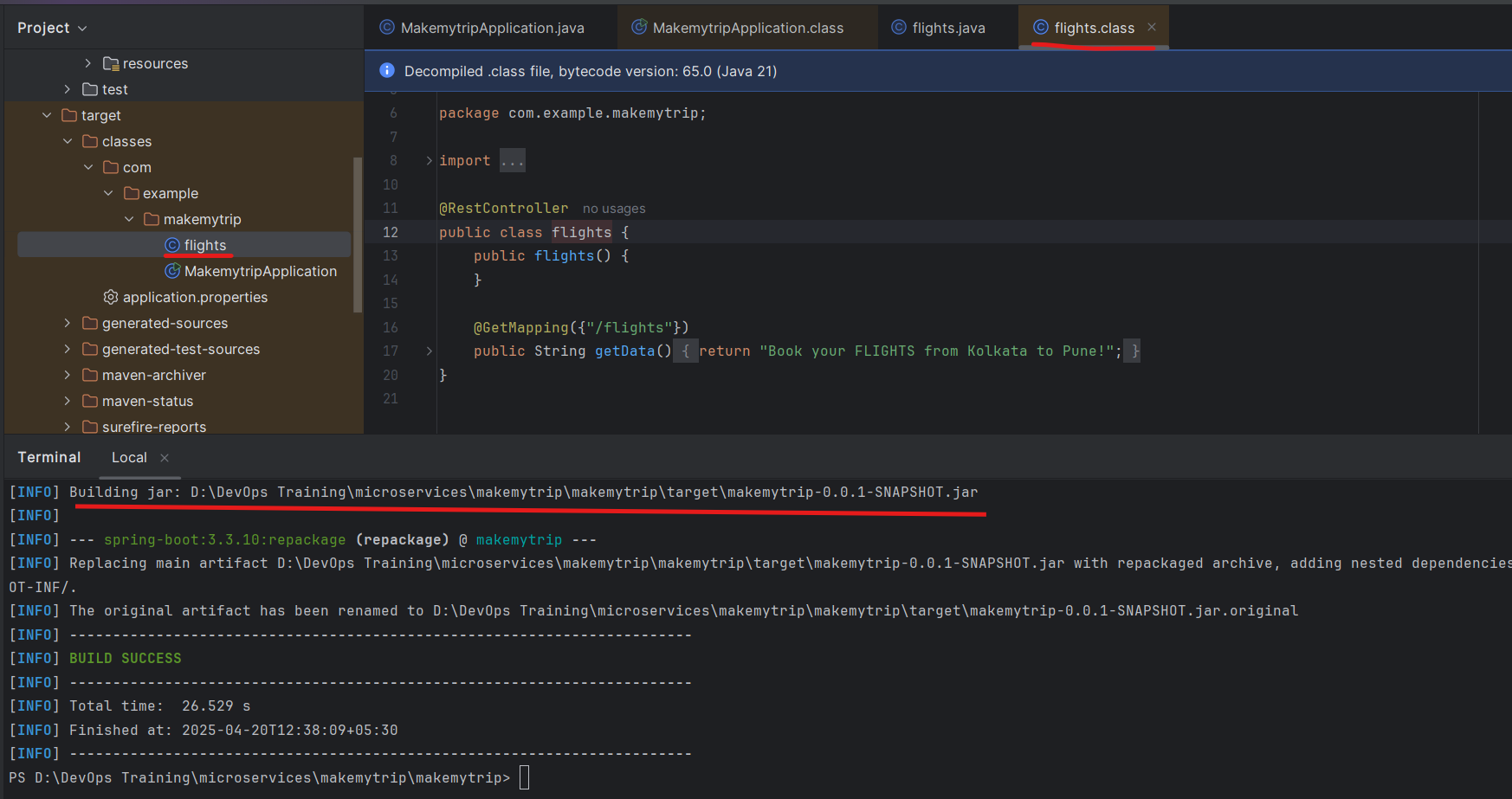
}

After ading new code - you need to compile and package it again.

> mvn clean compile

> mvn clean package

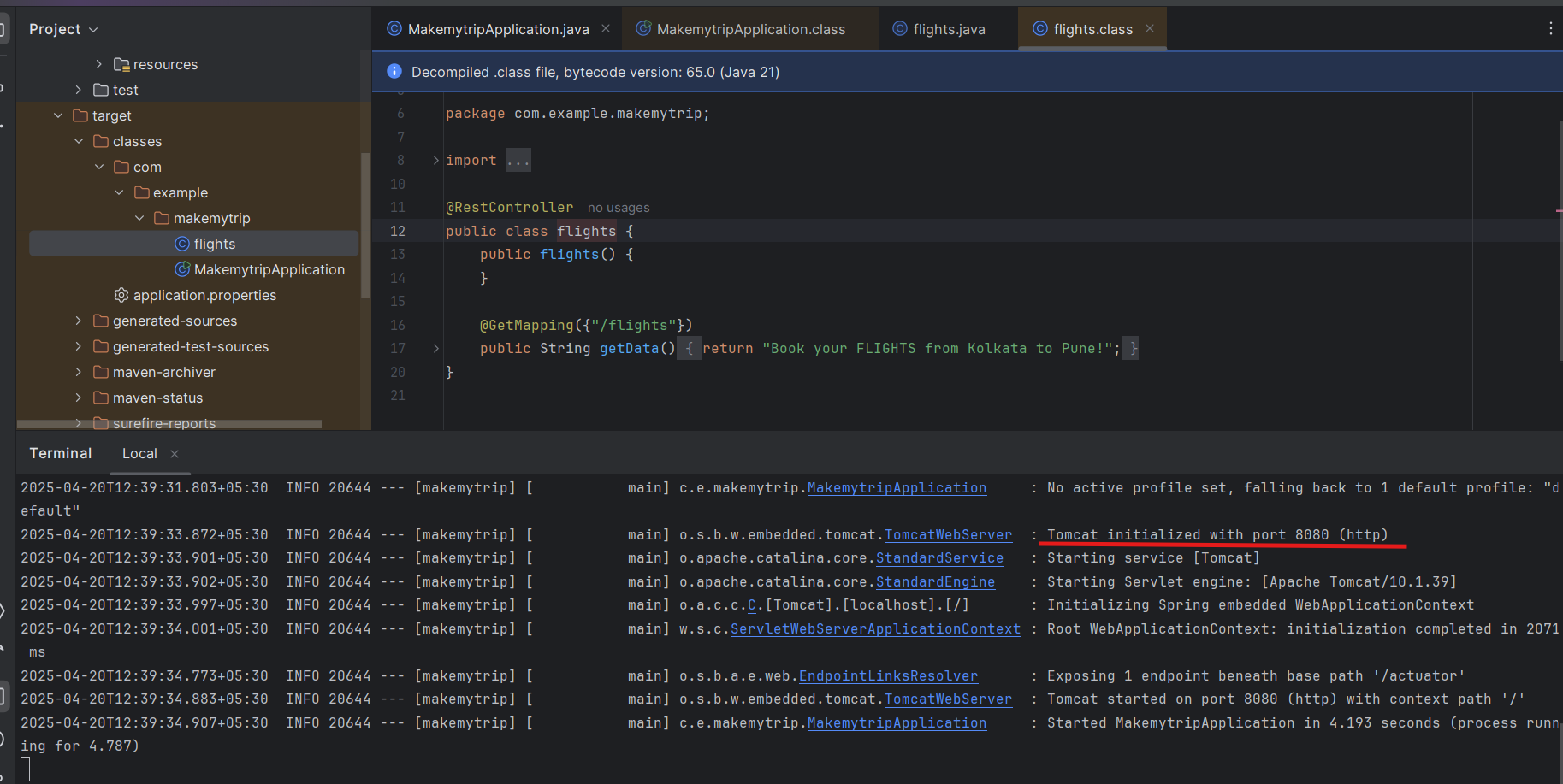
.class and .jar file is generated



> mvn spring-boot:run

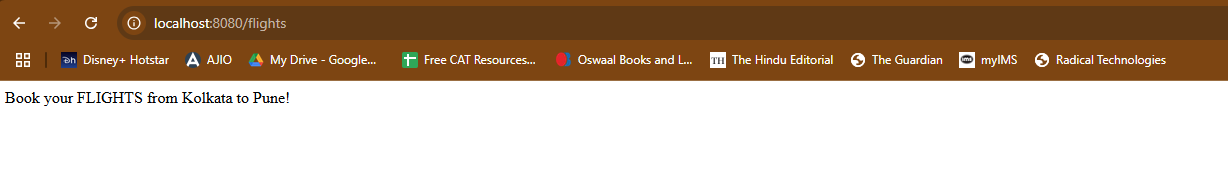
With help of this we will run the application on local system.

We can see the application is running on port 8080

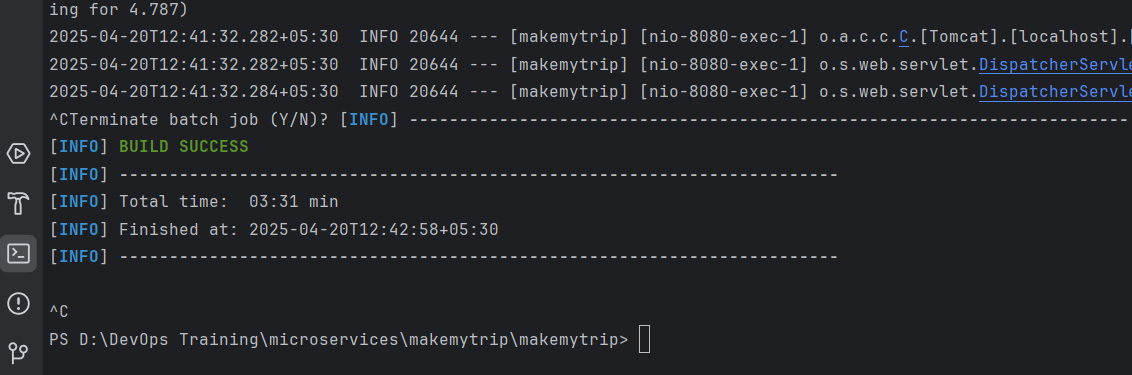


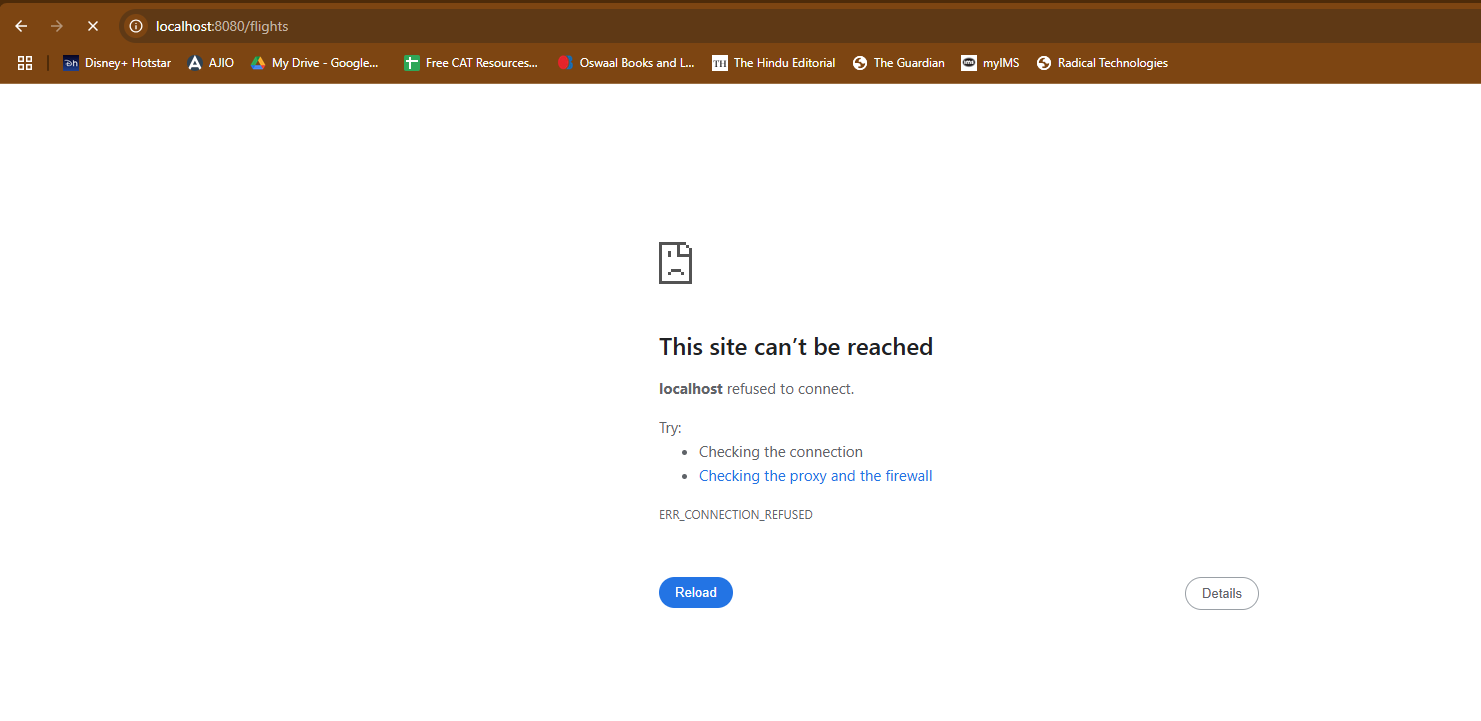
Now run the flights microservice on localhost

<http://localhost:8080/flights>



Press control C on terminal and the application will stop. If you check the link again it wont work anymore





Add new feature now: hotels.java

package com.example.makemytrip;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class hotels {

public hotels() {

}

@GetMapping({"/hotels"})

public String getData() {

return "Book your HOTELS for Pune at 50% OFF!";

}

}

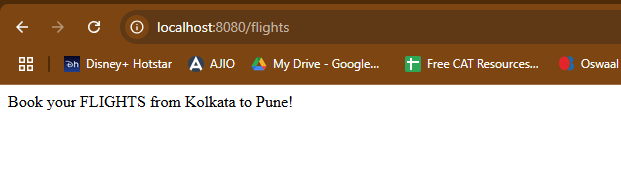
Now clean compile and package to make sure the new code is not spoiling the old working code.

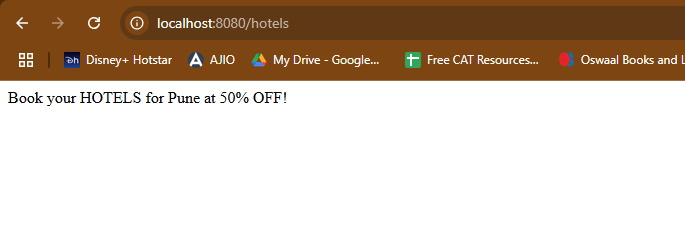
> mvn clean compile

> mvn clean package

And then verify both the applications: hotels and flights

> mvn spring-boot:run

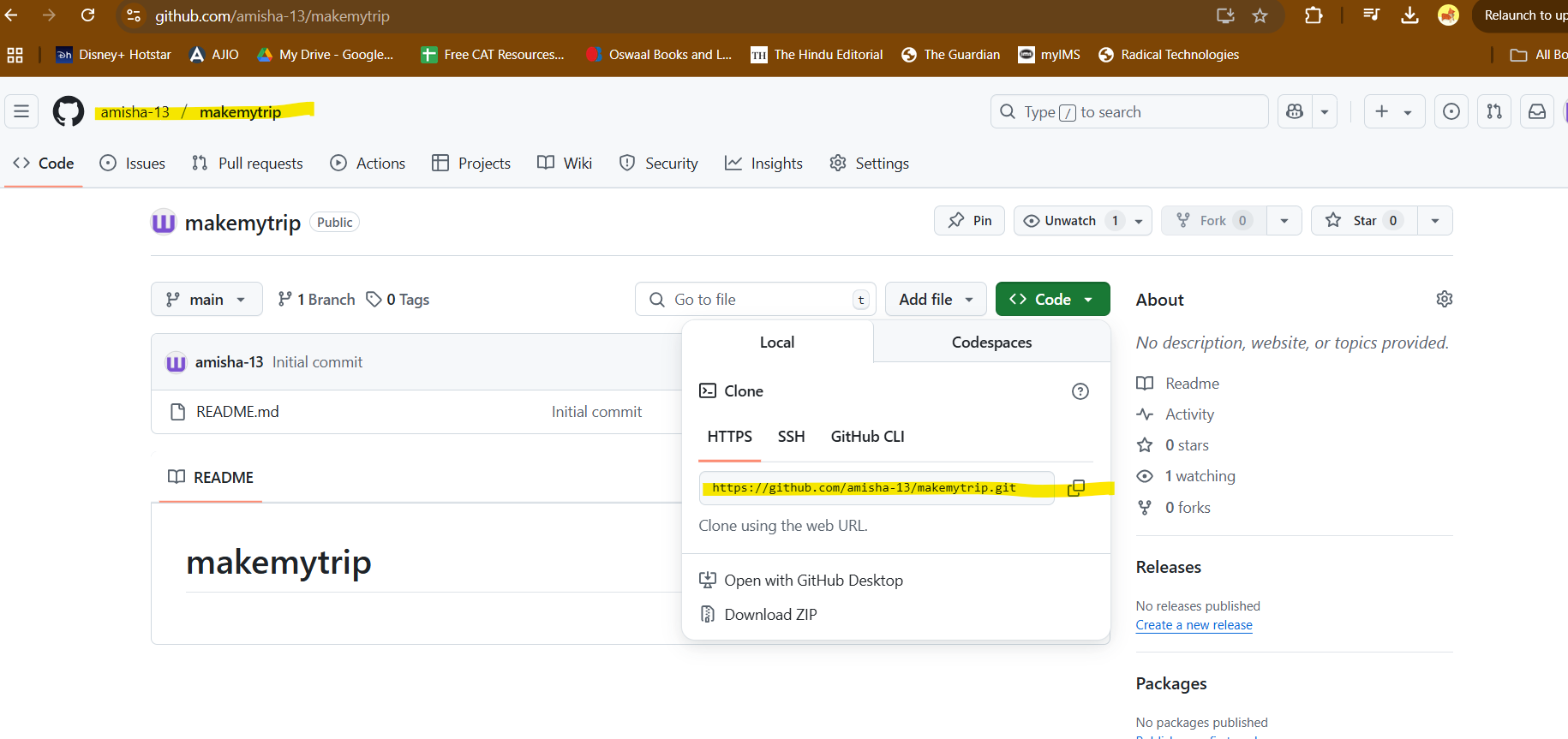




So we can see both features are working on local system.

Once the functionalities are verified and code is working - the code is supposed to be pushed on **github - shared remote code repository**.

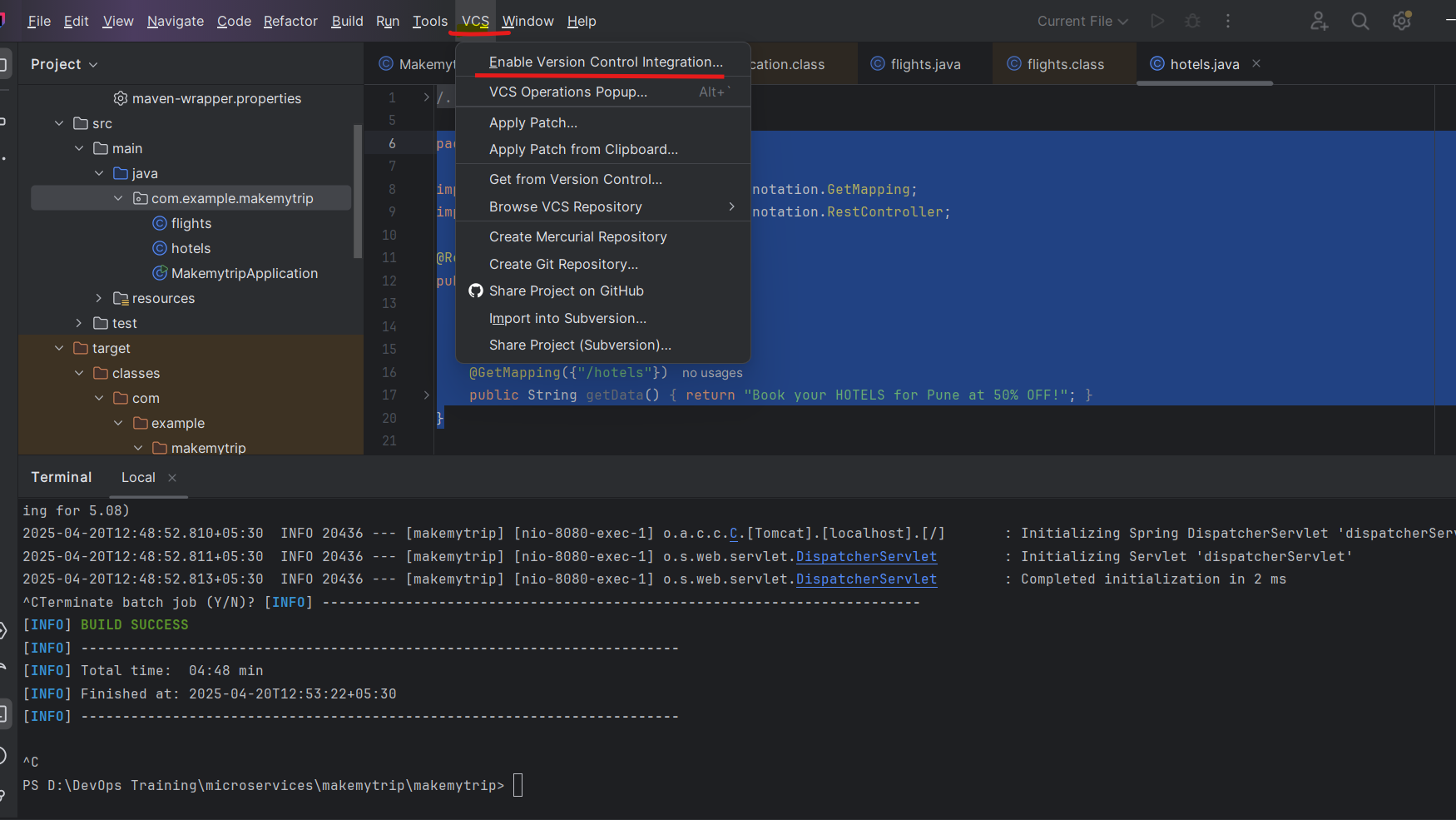
Create new repository on Github: makemytrip



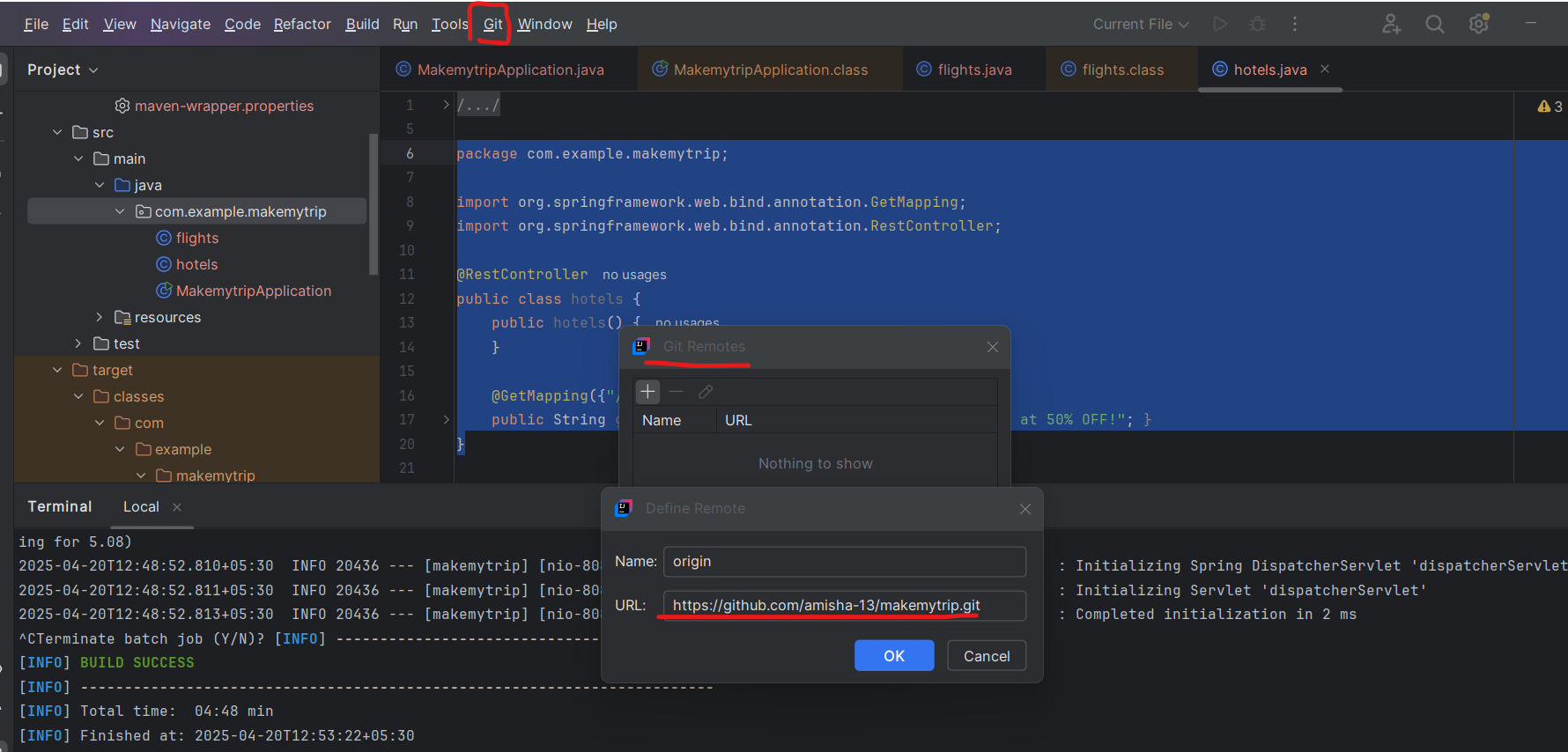
Push the code from local to remote repository

In IntelliJ IDEA enable version control integration:

VCS > Enable version control integration

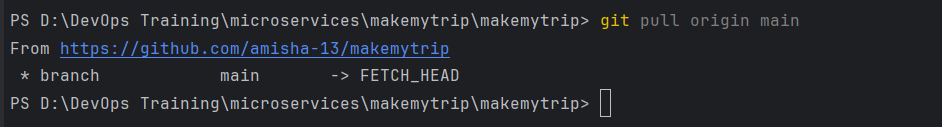


VCS > Manage remote > Enter URL of remote repository just copied



In my case - my default branch name is main and not master

> git pull origin main



1st time you run this:

git pull

git pull origin master

git add --all

git commit -m "CodeRefactor"

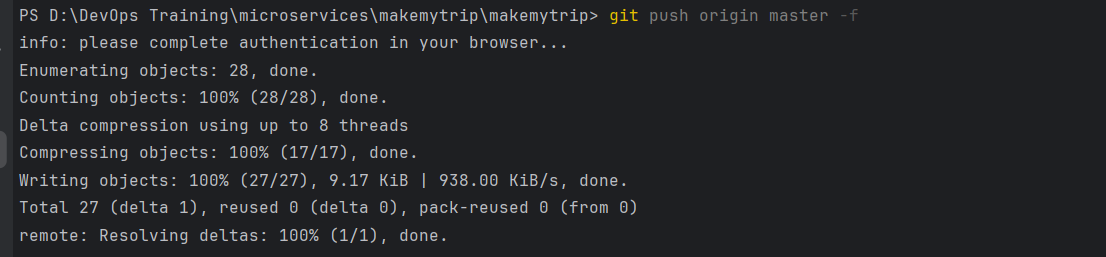
git push origin master -f

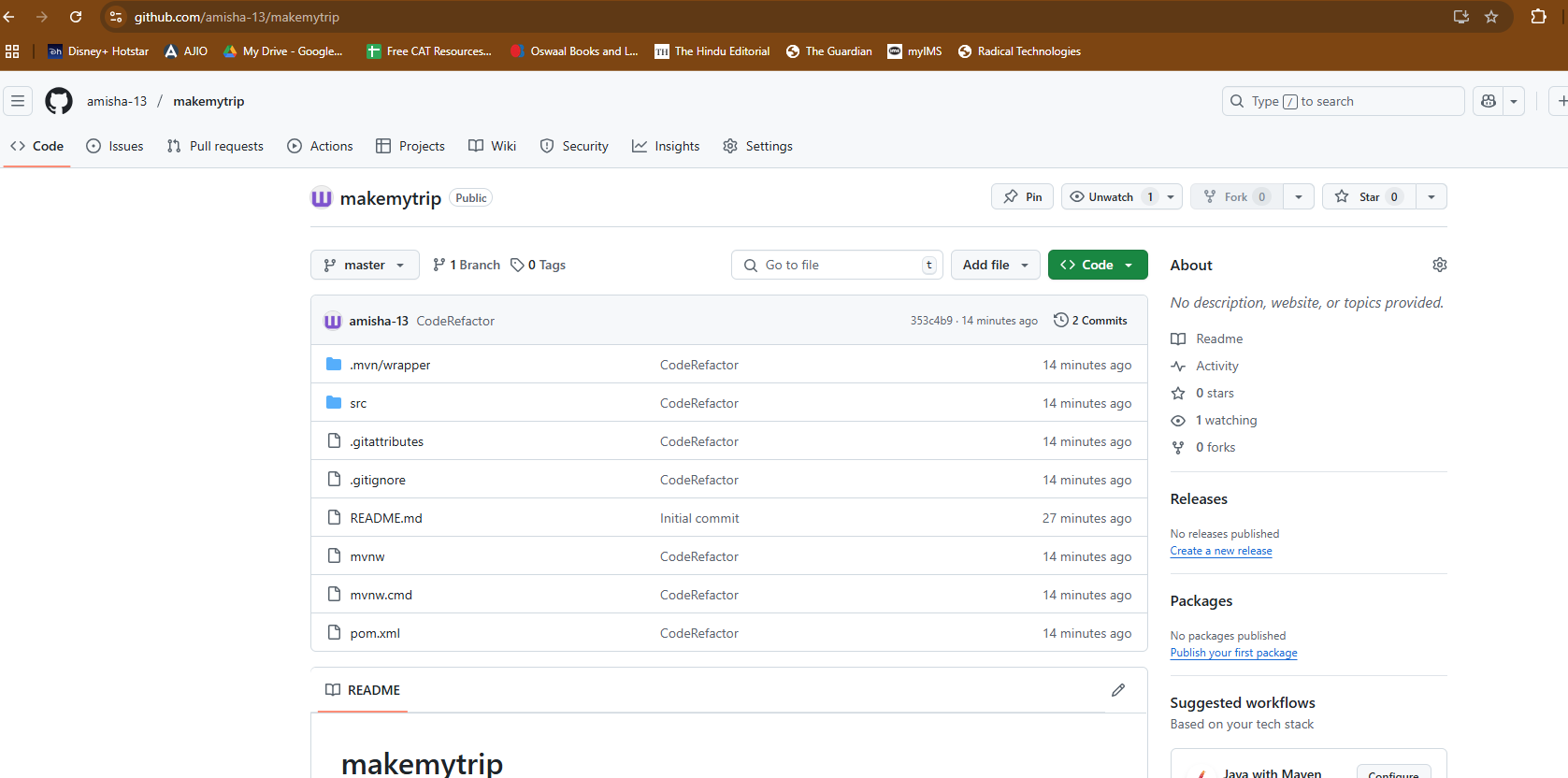
2nd and all times

git add --all

git commit -m "message"

git push origin master





Add new class - bus.java

package com.example.makemytrip;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

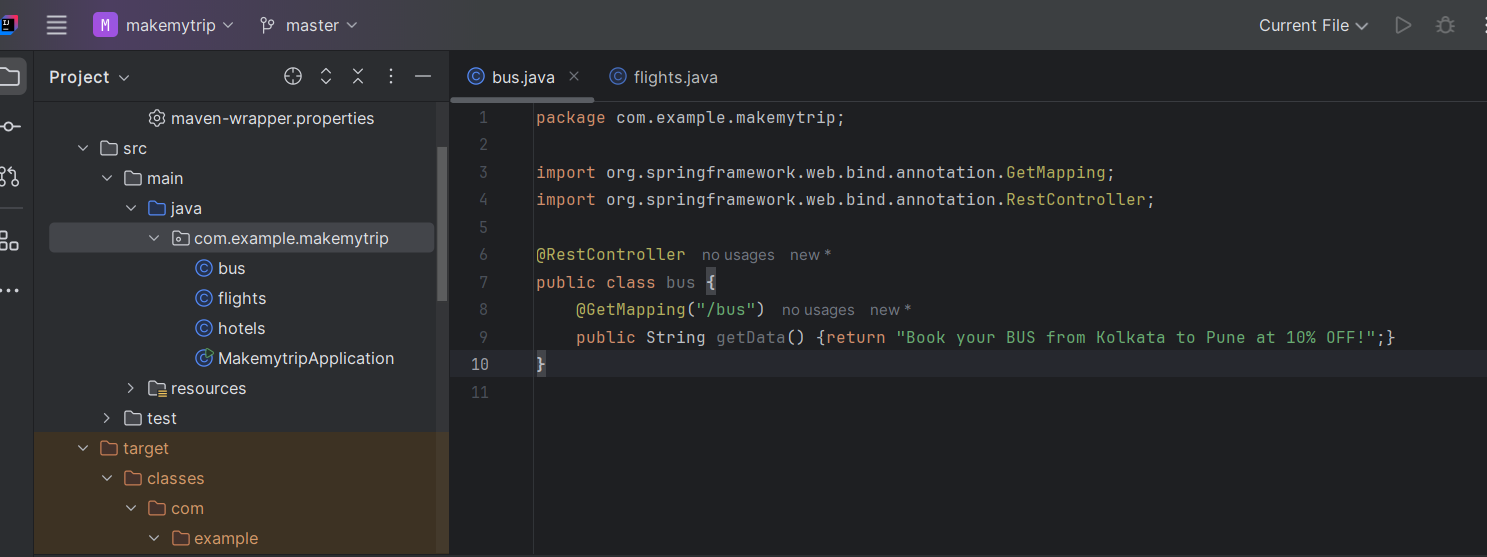
@RestController

public class bus {

@GetMapping("/bus")

public String getData() {return "Book your BUS from Kolkata to Pune at 10% OFF!";}

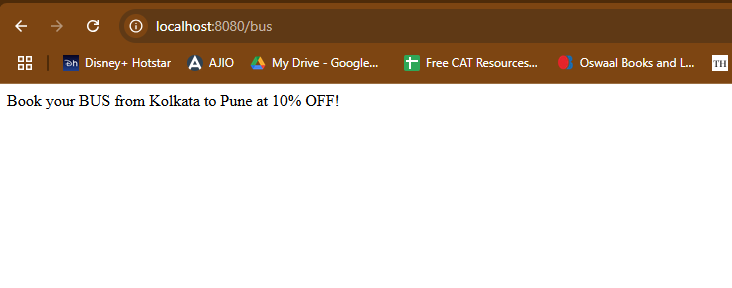
}



> mvn clean compile

> mvn clean package

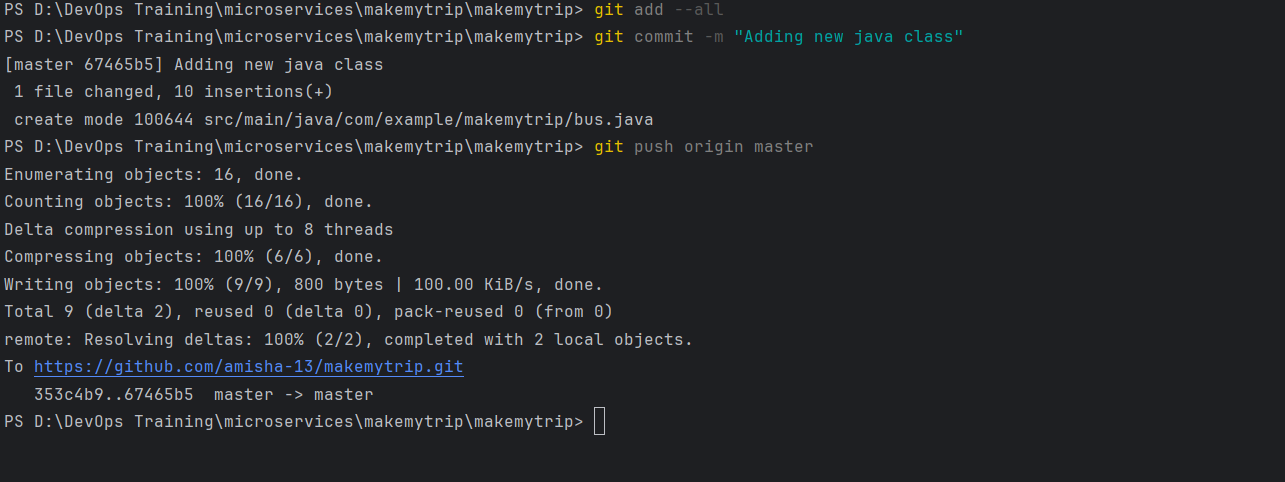
> mvn spring-boot:run



> git add --all

> git commit -m "Adding new java class"

> git push origin master



Code is now uploaded

