Download the project from spring intitializr. Visit <https://start.spring.io/>

**Select the following**

Project: Maven

Language: Java

Spring Boot: 2.7.5

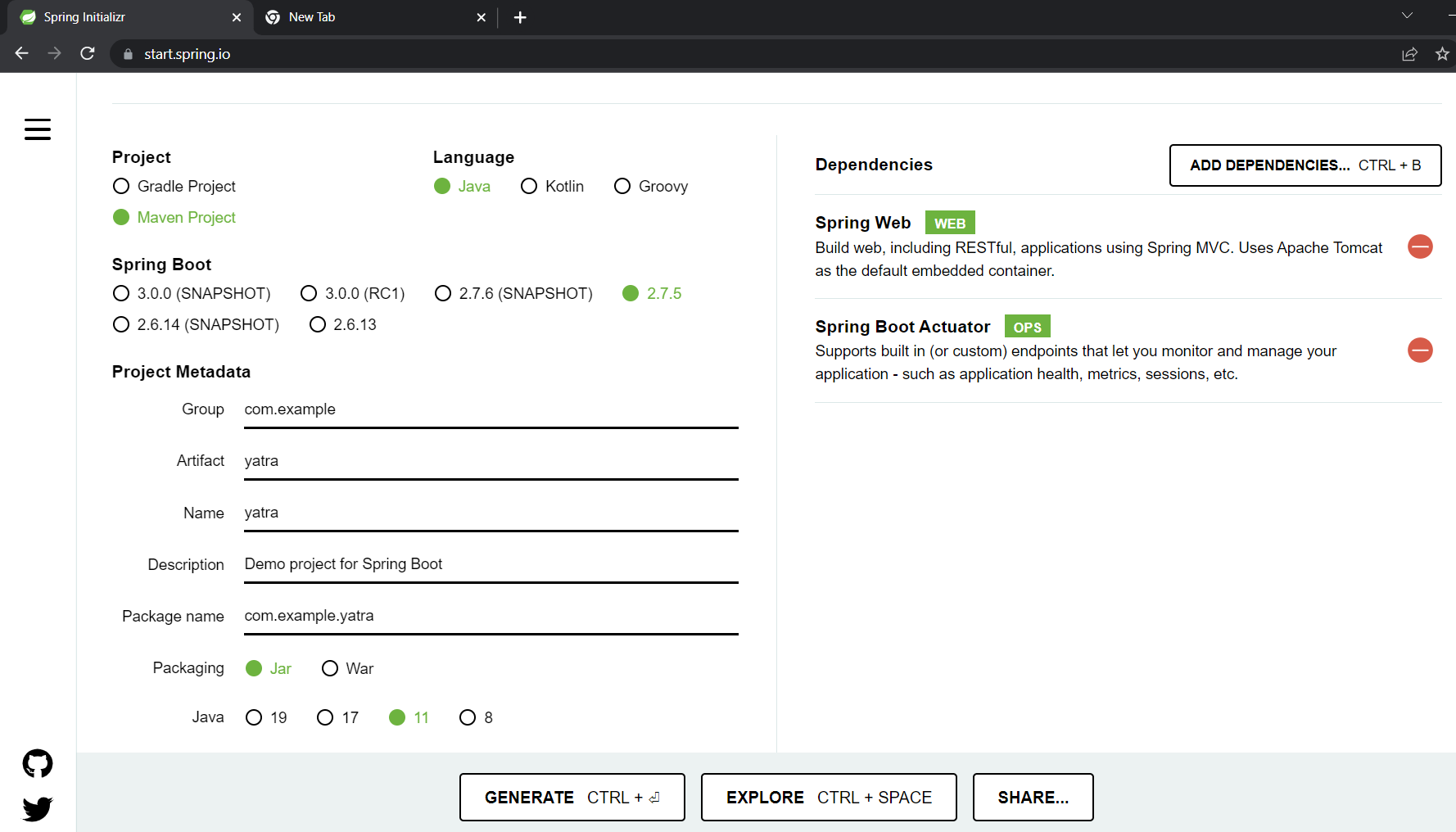
Rename the Artifact in Project Meta data (Example: Yatra)

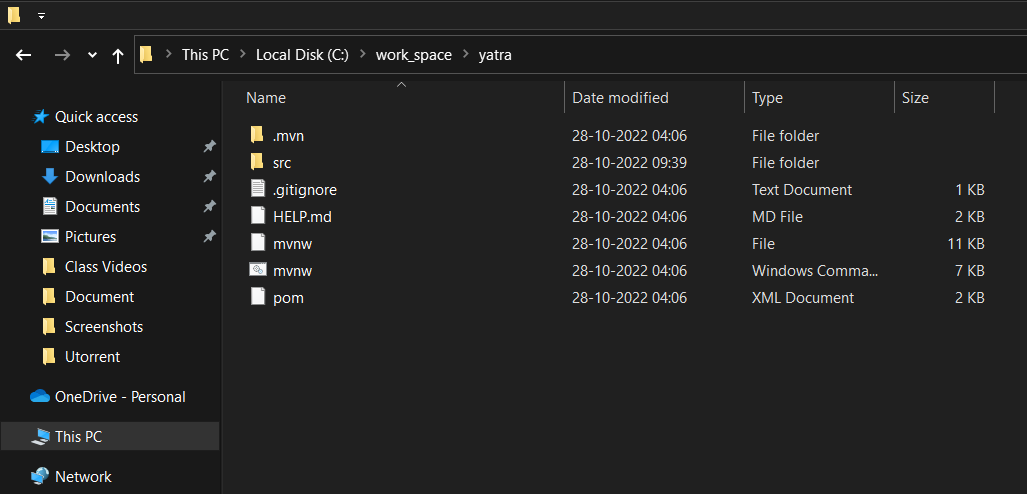
Packaging: Jar

Java Version: 11

Add Dependencies: **Spring web** and **Spring boot actuator**

Then Click on **Generate** or press **Ctrl + Enter**



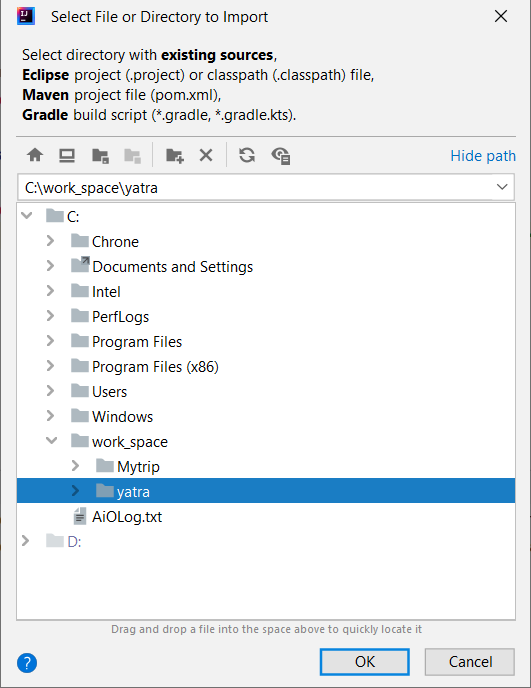
It will Generate a Zip file(yatra.zip). Extract it and copy the extracted folder to your workspace (C:\work\_space)

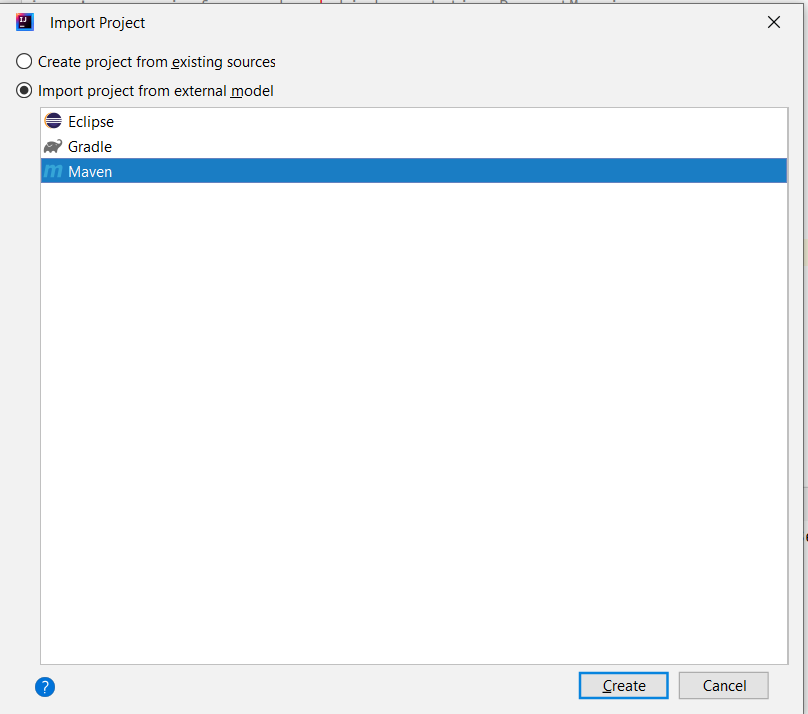
Open Intellij Idea

File->New->Project from existing sources

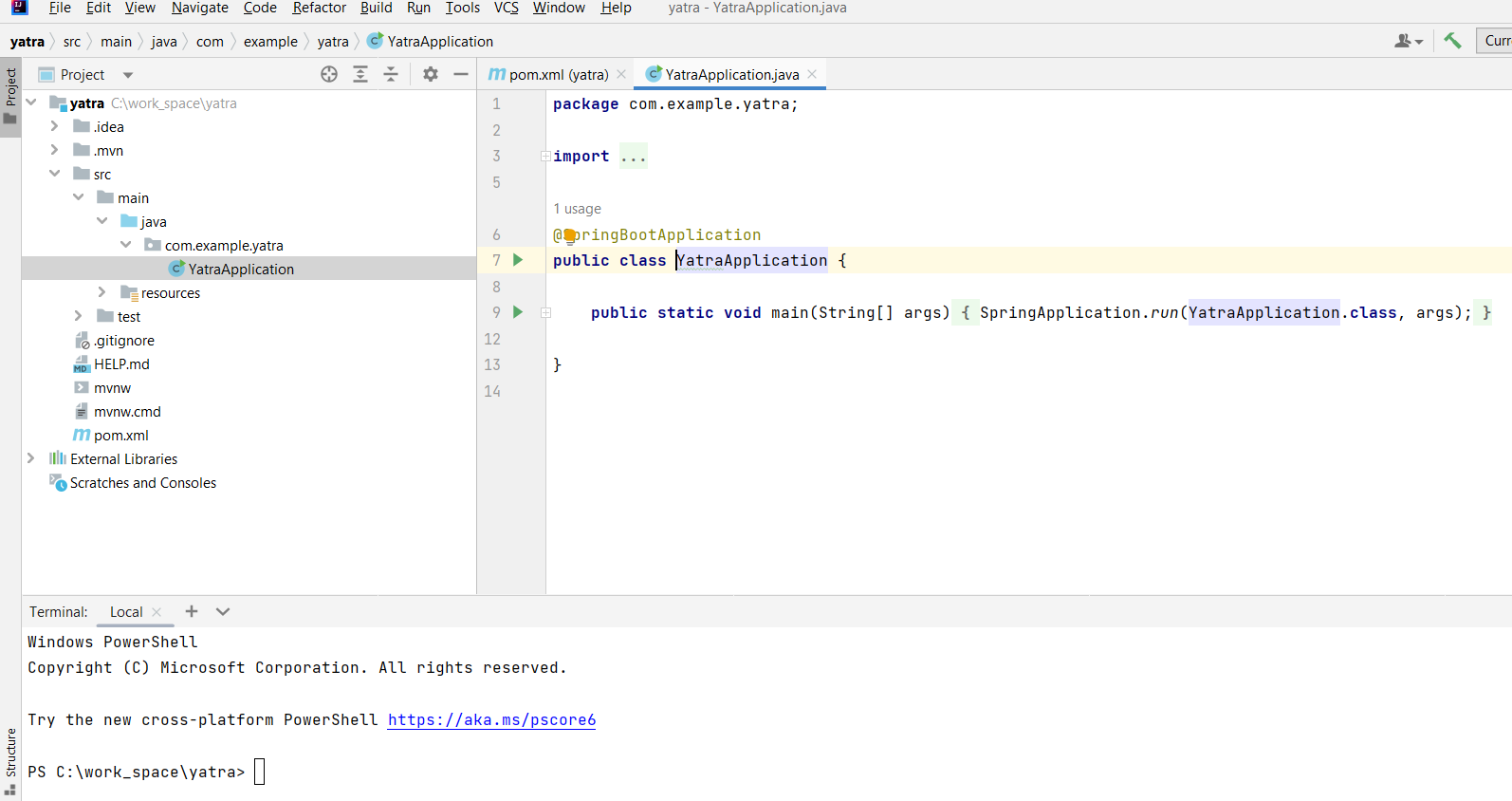
Then select your project which is copied to work\_Space.

Select Yatra -> Click OK

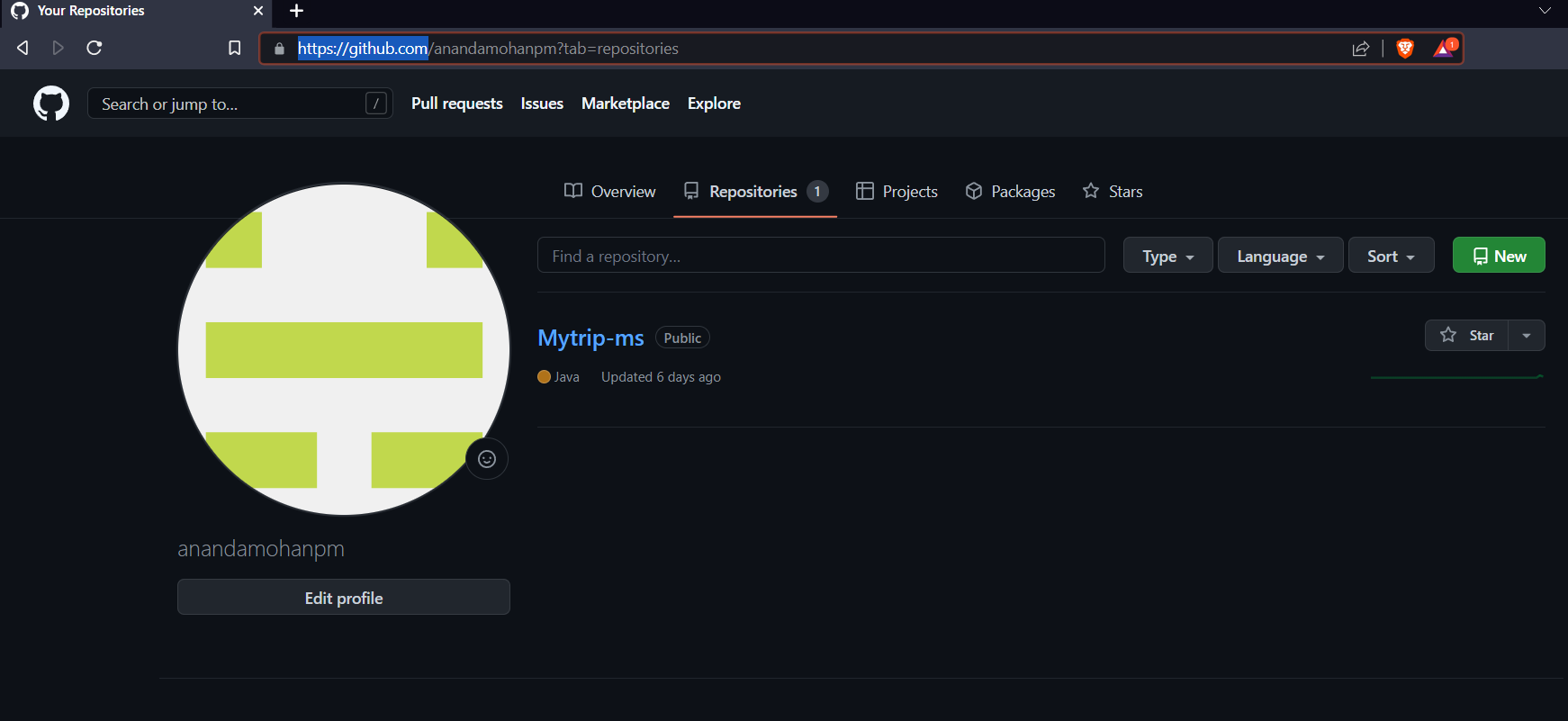




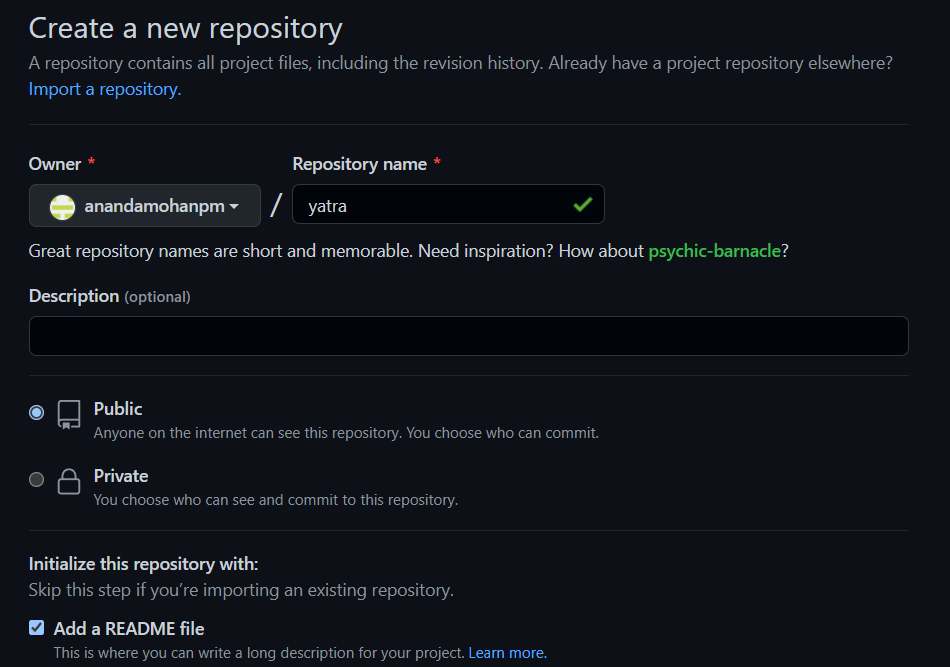
Click create then open in new window



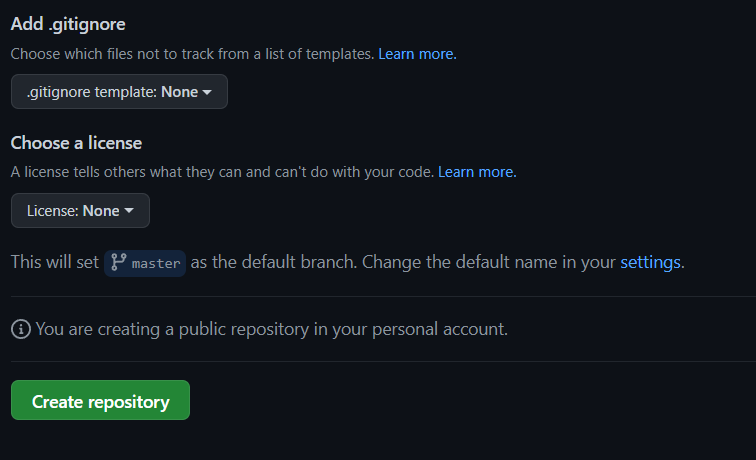
Create a repository in https://github.com/. Login to your account and select repositories and click on new



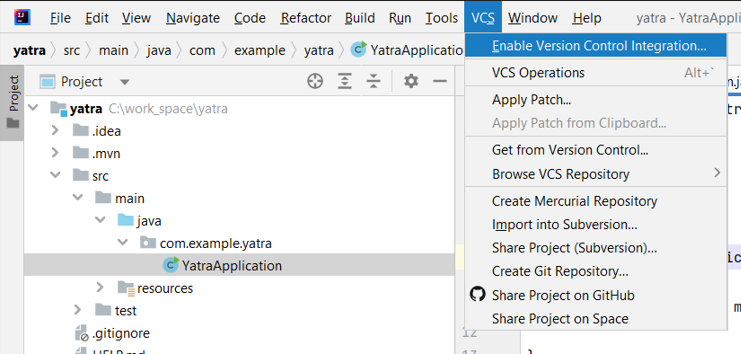
Create a repository named yatra. Select Public to access for everyone. Select Add a README file



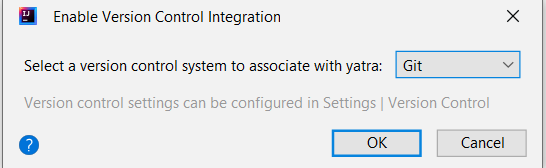
Check the branch whether it is main or master. Click on Create repository.



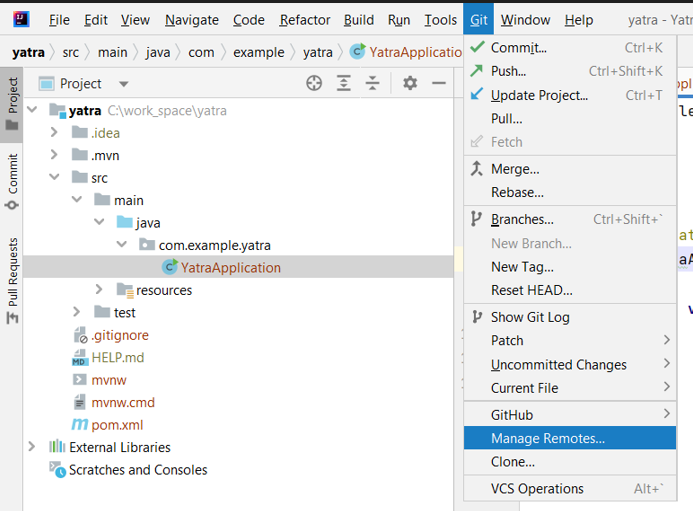
Open Intellij IDEA for linking the local repo with Git hub repo. Click on VCS -> Enable version control Integration



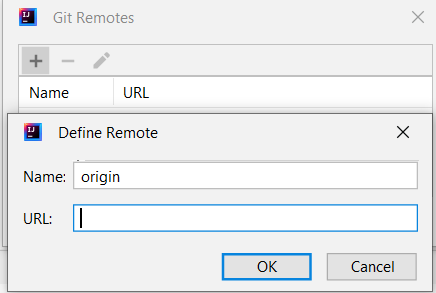
Select Git and click ok



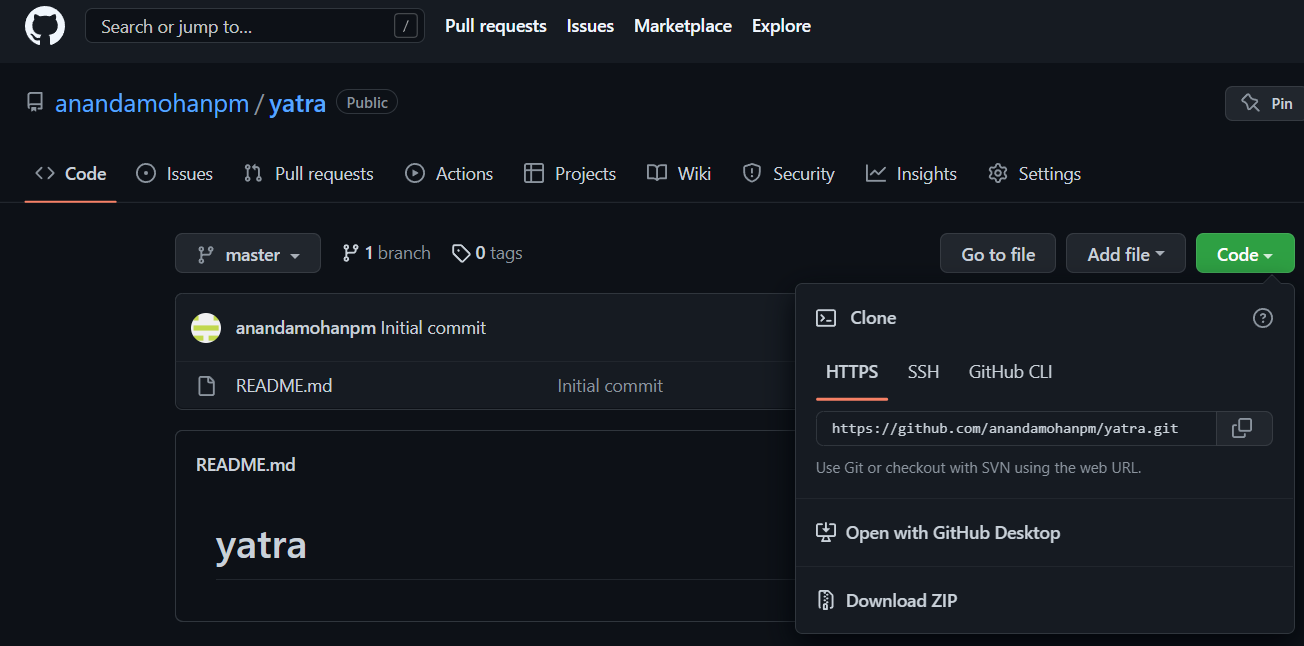
Click on Git -> Manage Remotes



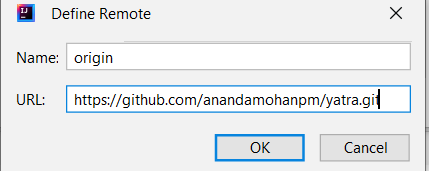
Click on + sign to add the remote repo link



Go to git hub for getting the Git hub link. Click on code then copy the link (HTTPS)



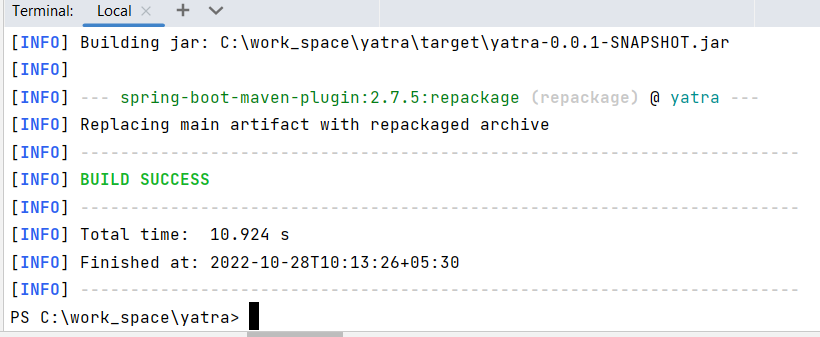
Paste the link to Git Remotes.



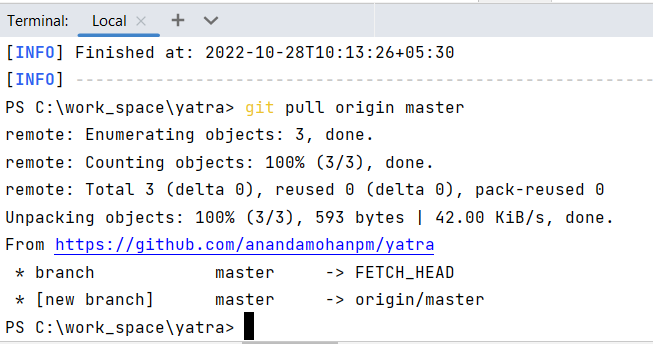
Click ok->ok

Compile the code in Intellij idea and push the code to Git hub repo.

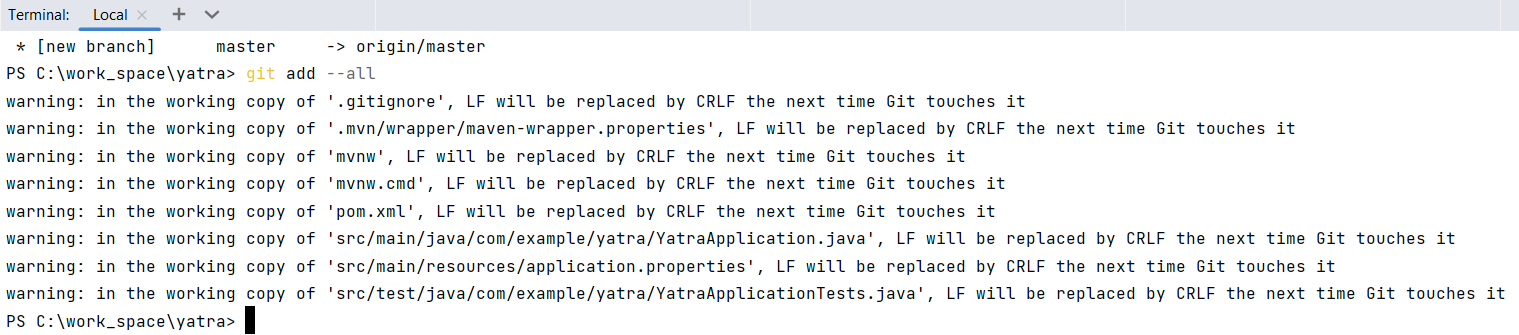
**$mvn clean package**



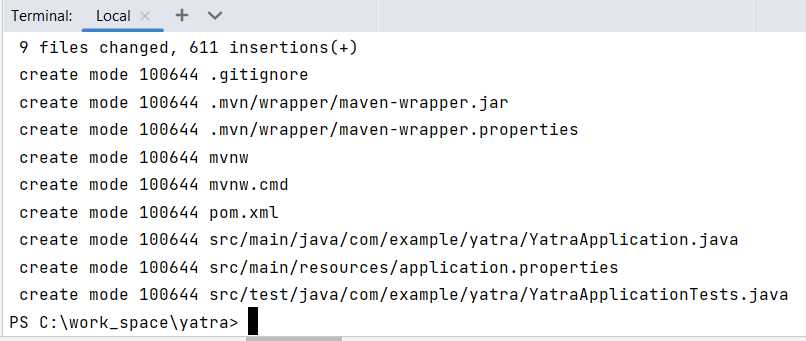
**$git pull origin master**



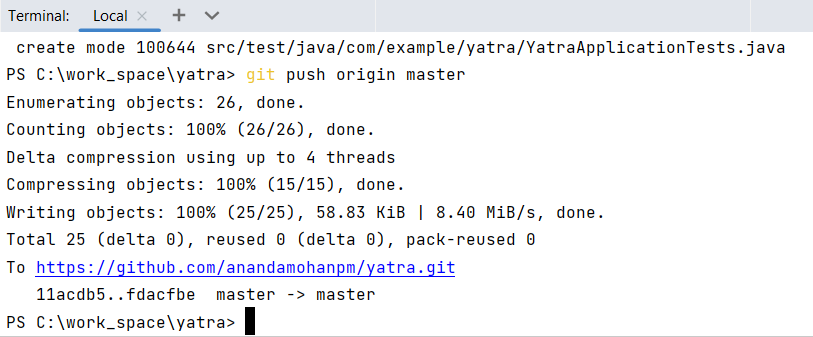
**$git add --all**



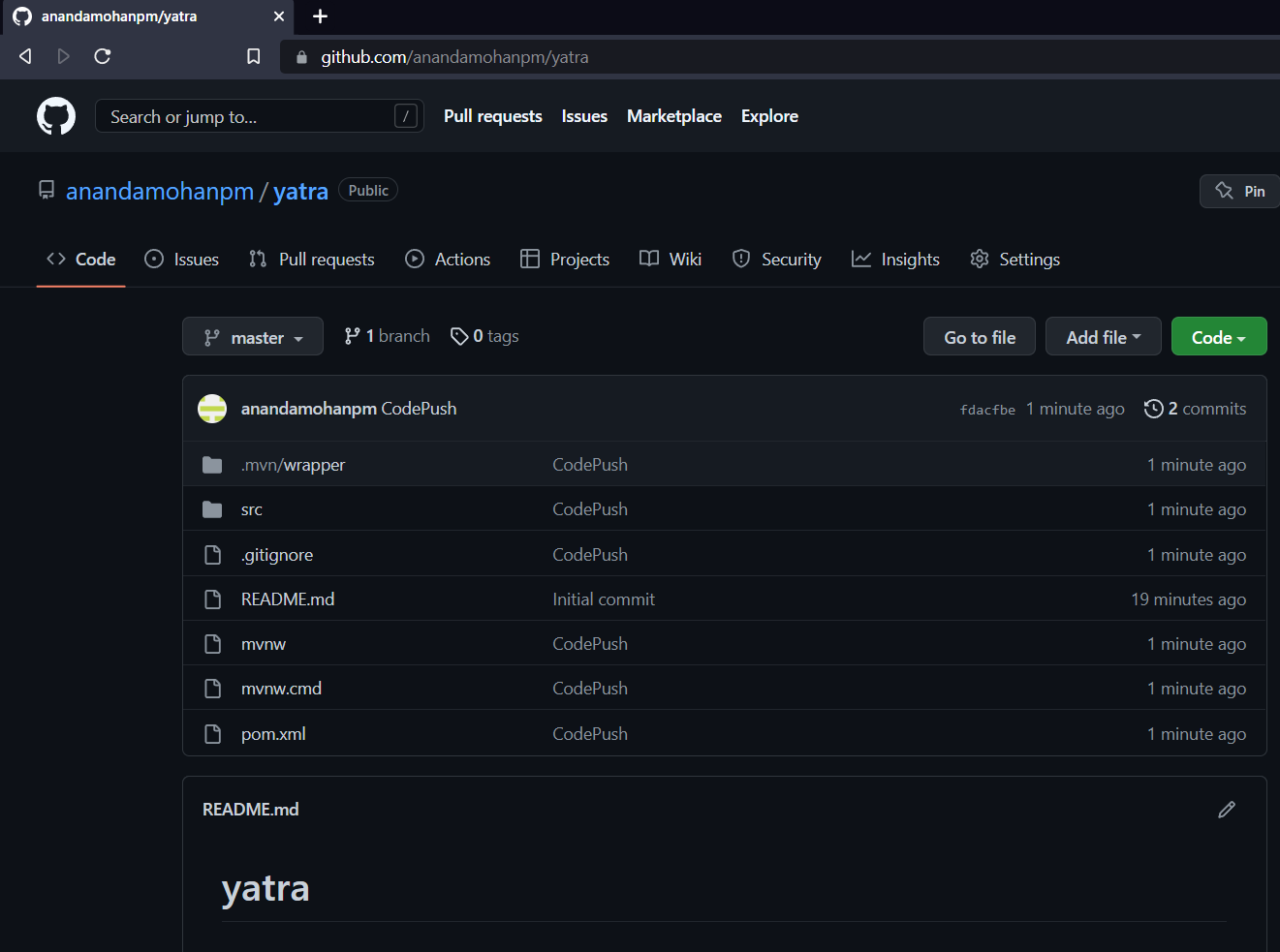
**$git commit -m “CodePush”**



**$git push origin master**

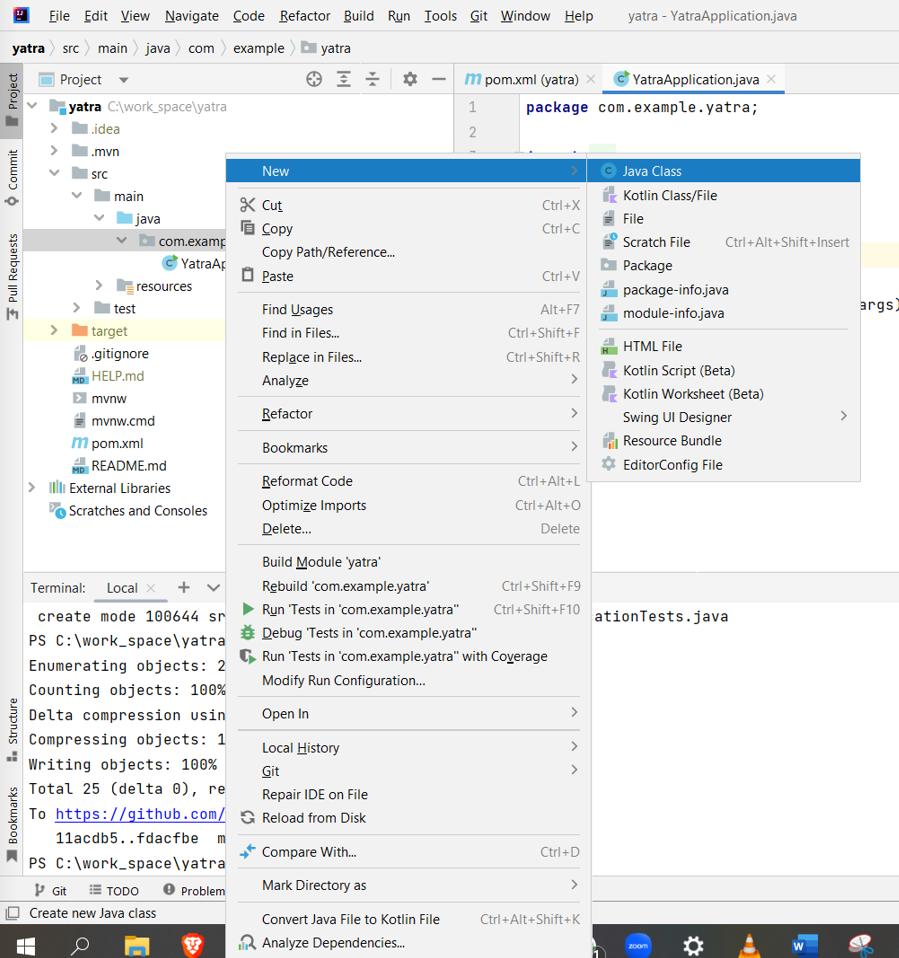


Now you can view the codes are pushed to Git hub repo

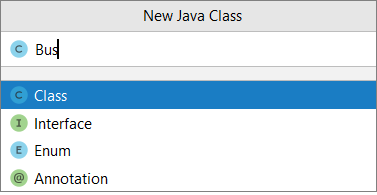


**Adding new java class**

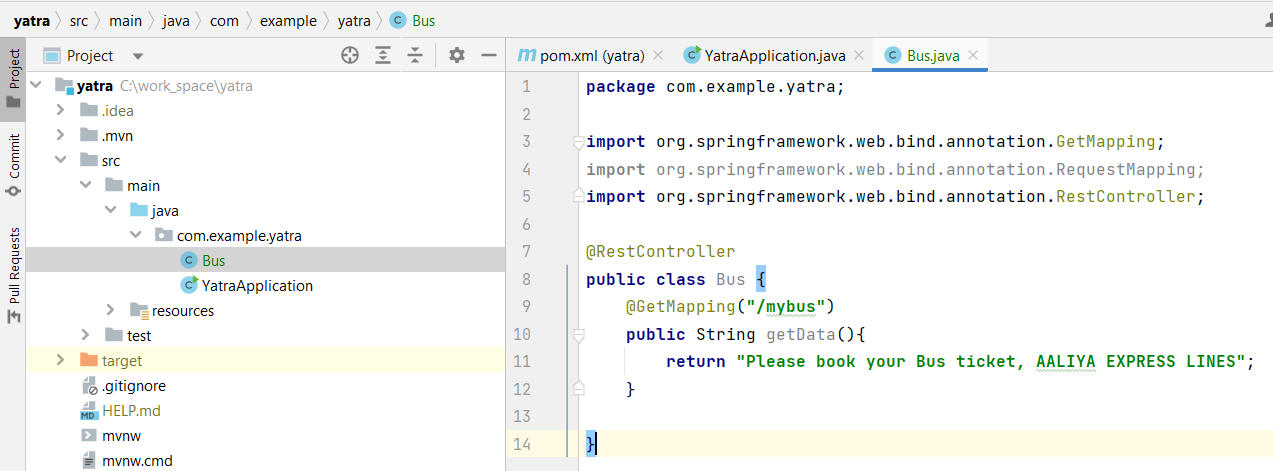
Right click on com.example.yatra. click on New-> Java class



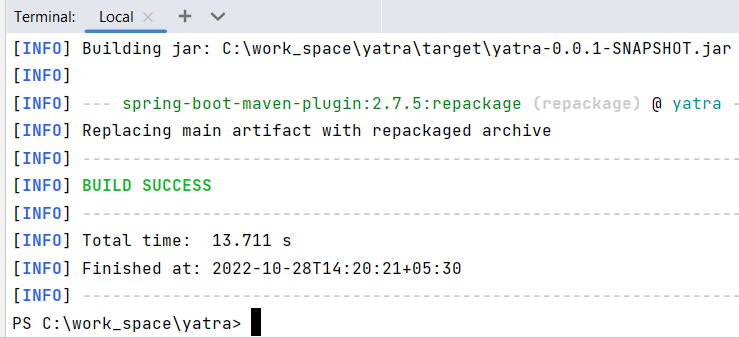
Type the class name and press enter key.



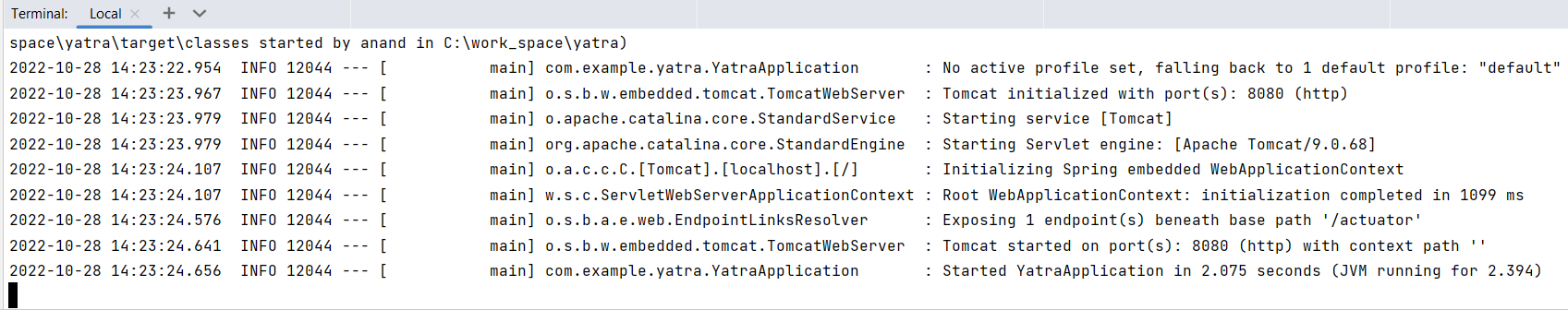
Bus Class is added and you can write your program.



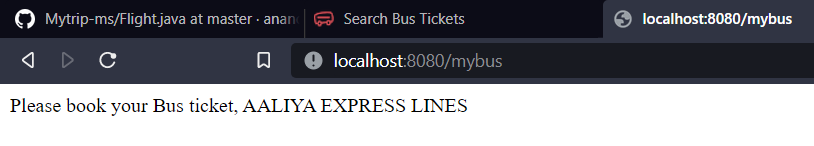
Compile the program using the code **$mvn clean package**



Compilation completed without any error. It takes 13.711 s to complete. Now you can run the program using the code **$mvn spring-boot:run**



Now you can see the output of our program in browser. Type in address bar: localhost:8080/mybus



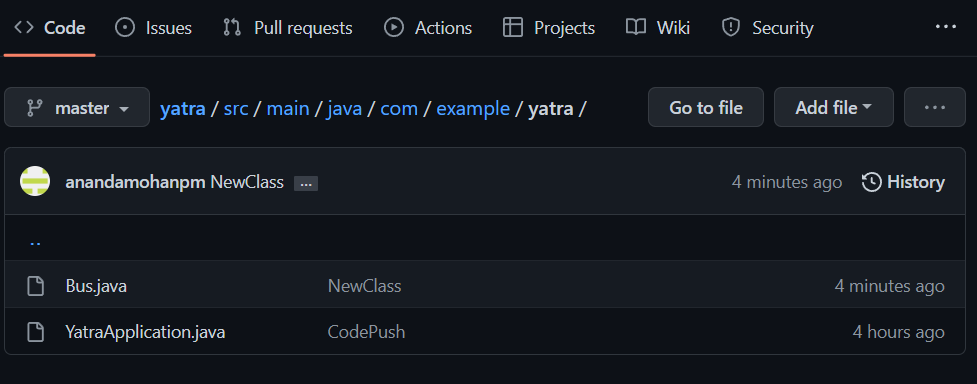
Commit the new class to Git hub repo by using the code

#git add –all

#git commit -m “NewClass”

#git push origin master

Now you can see the “Bus” class is added to Git hub repository.



**Create following Branches in Git hub.**

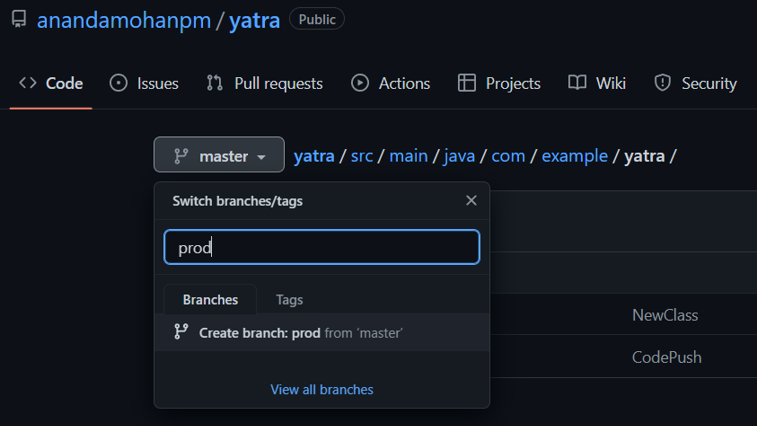
**master**

**prod**

**preprod**

**dev**

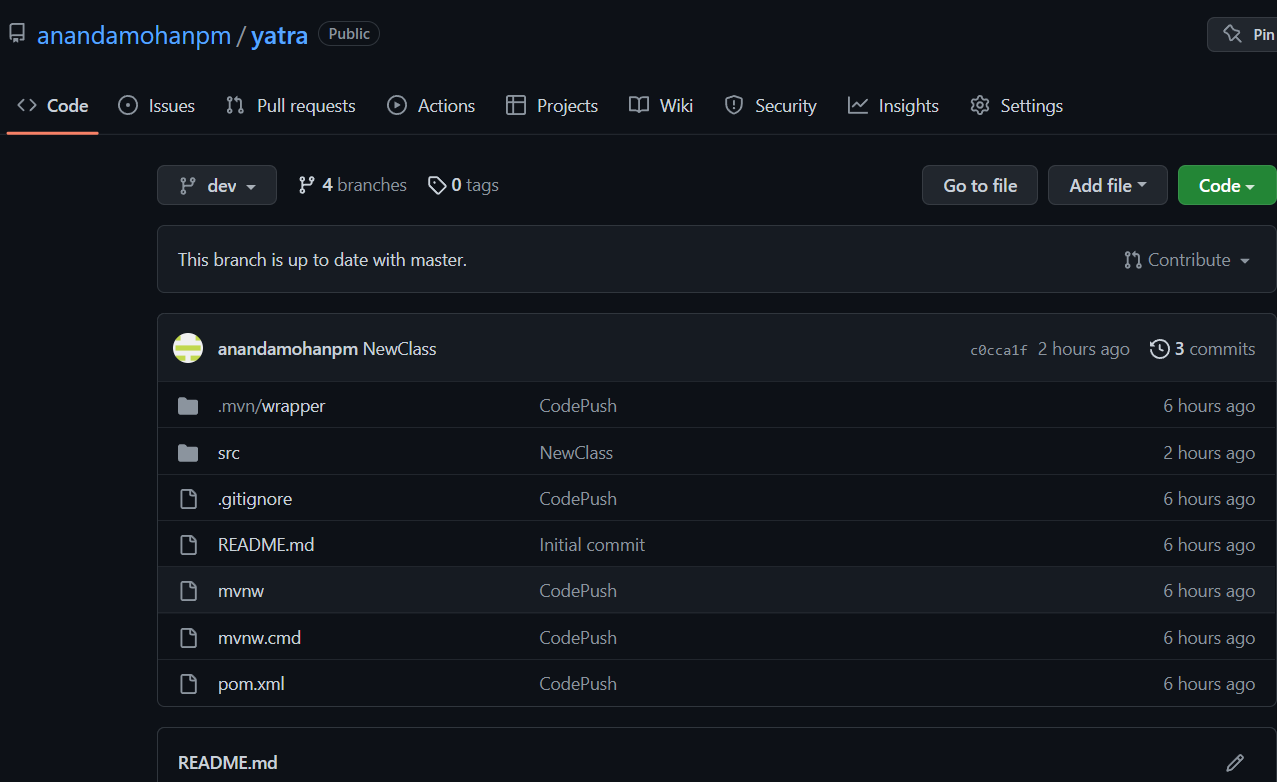
Go to git hub. Open repo. Click on master then type “prod” then click on **Create branch: prod from ‘master’**



Next Click on prod then type “preprod” then click on **Create branch: prod from ‘prod’**

Next Click on preprod then type “dev” then click on **Create branch: dev from ‘preprod’**

Now it will show we have 4 branches and all the codes were copied from upper branches.



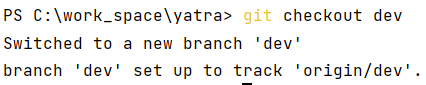
Download dev branch to local repo using the code

**#git fetch origin dev**



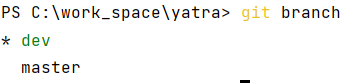
Checkout to the branch dev

**#git checkout dev**



To check which branch we are using

**#git branch**



To pull codes from the remote branch dev to local branch dev

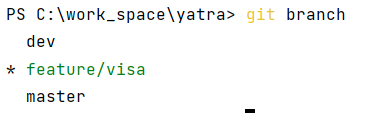
**#git pull origin dev**



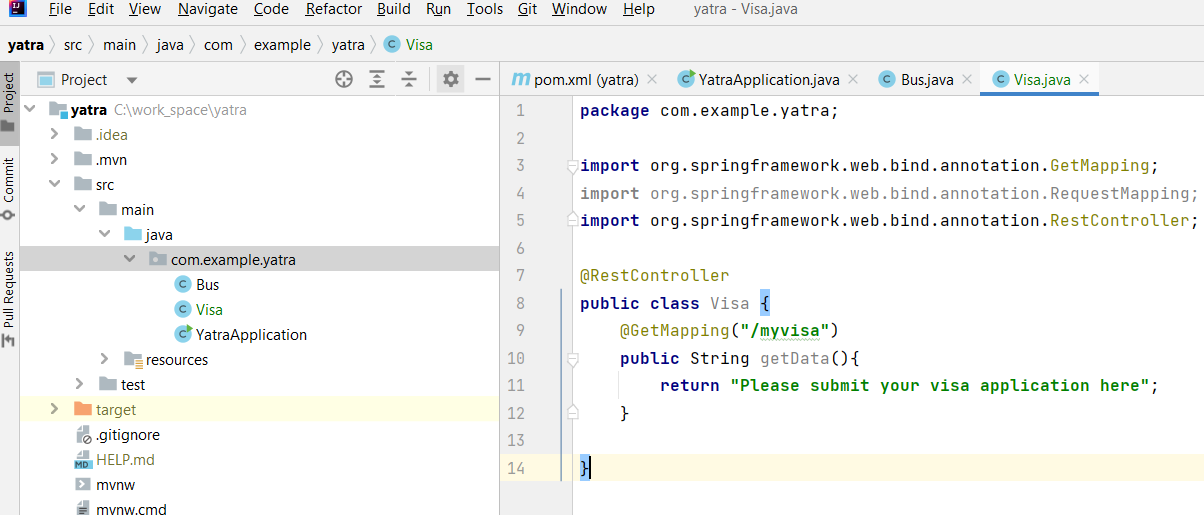
To Create a branch named **feature/visa** in local repo under **dev** branch. Verify we are in dev branch. (#git branch)

**#git checkout -b feature/visa**





Currently we are in feature/visa branch. So create a class named **visa** and push the changes to git hub.



Compile,Run, Check the output and push the codes to Git hub.

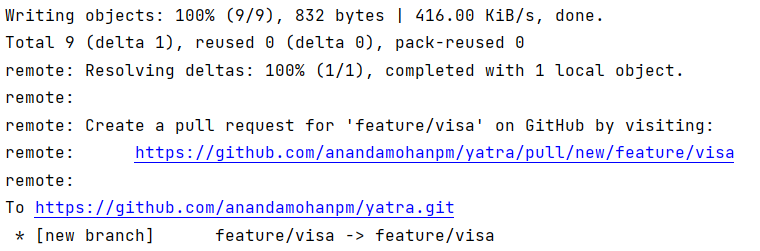
**#mvn clean package**

**#mvn spring-boot:run**

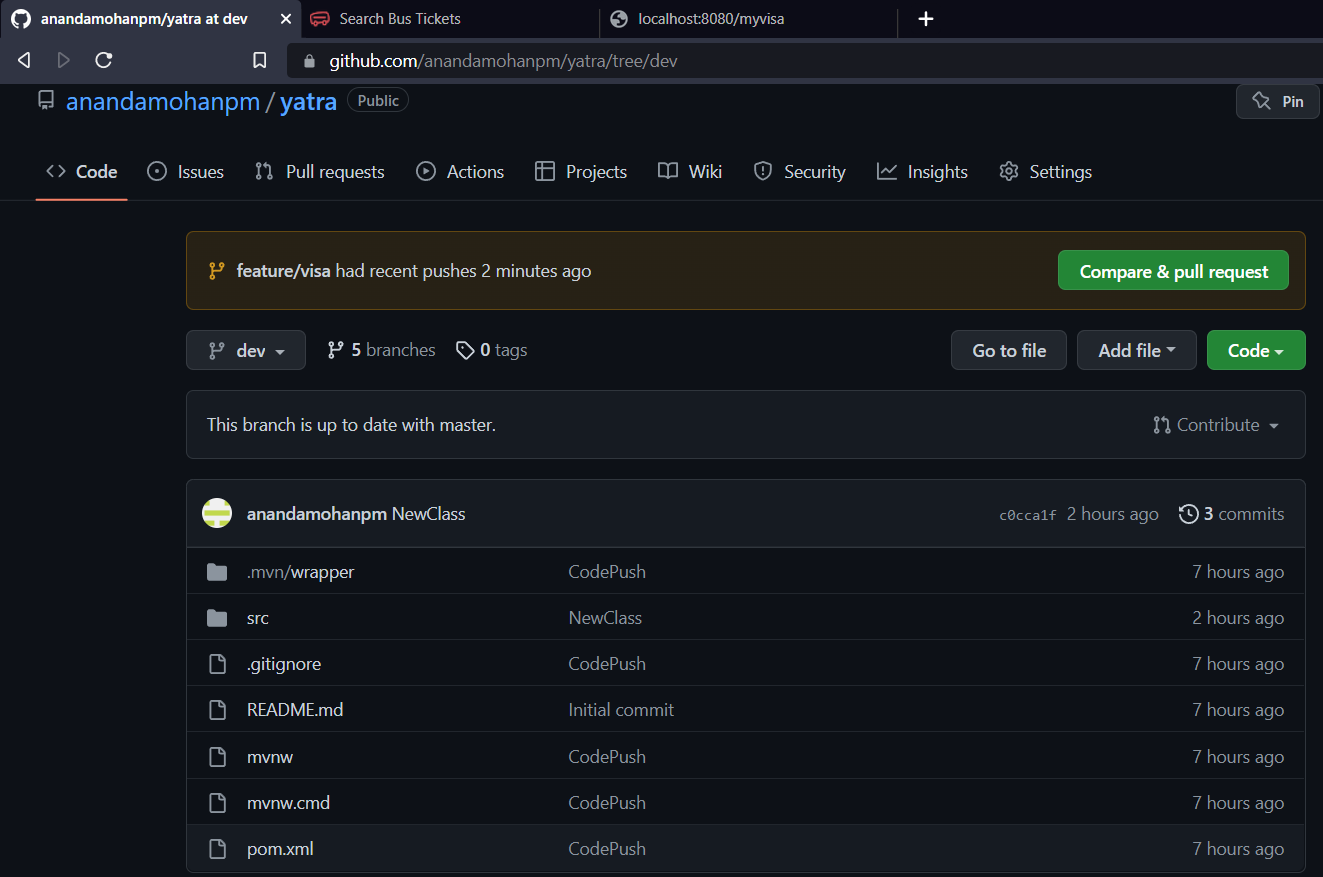
**#git add --all**

**#git commit -m "NewBranchPrgrm"**

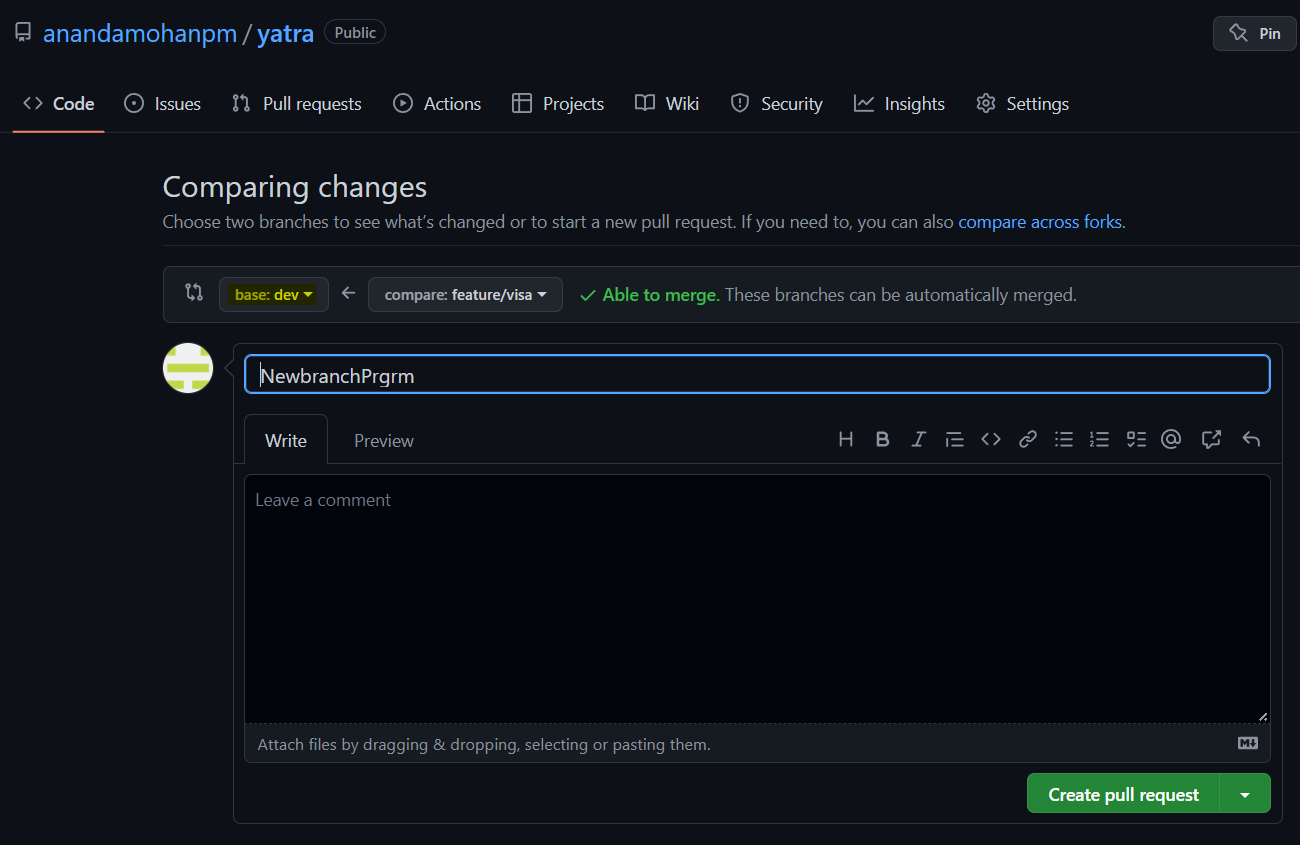
**#git push origin feature/visa**



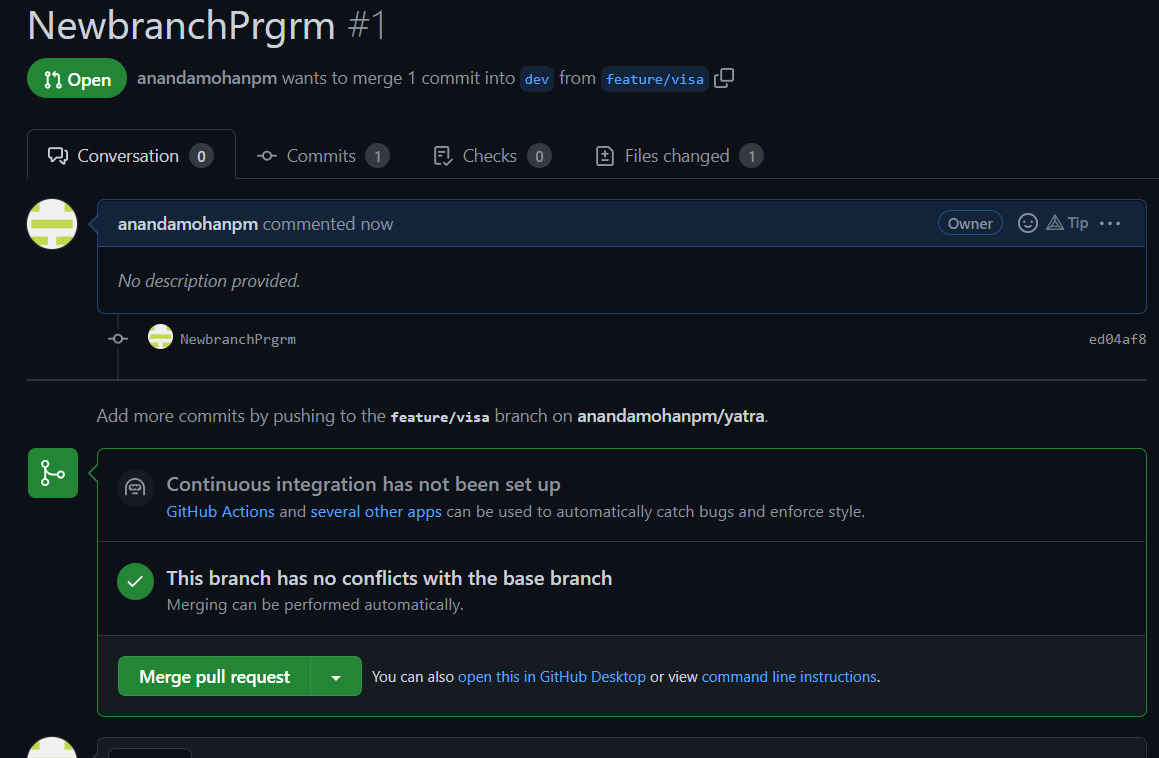
Go to git hub. Click on **compare & pull request**



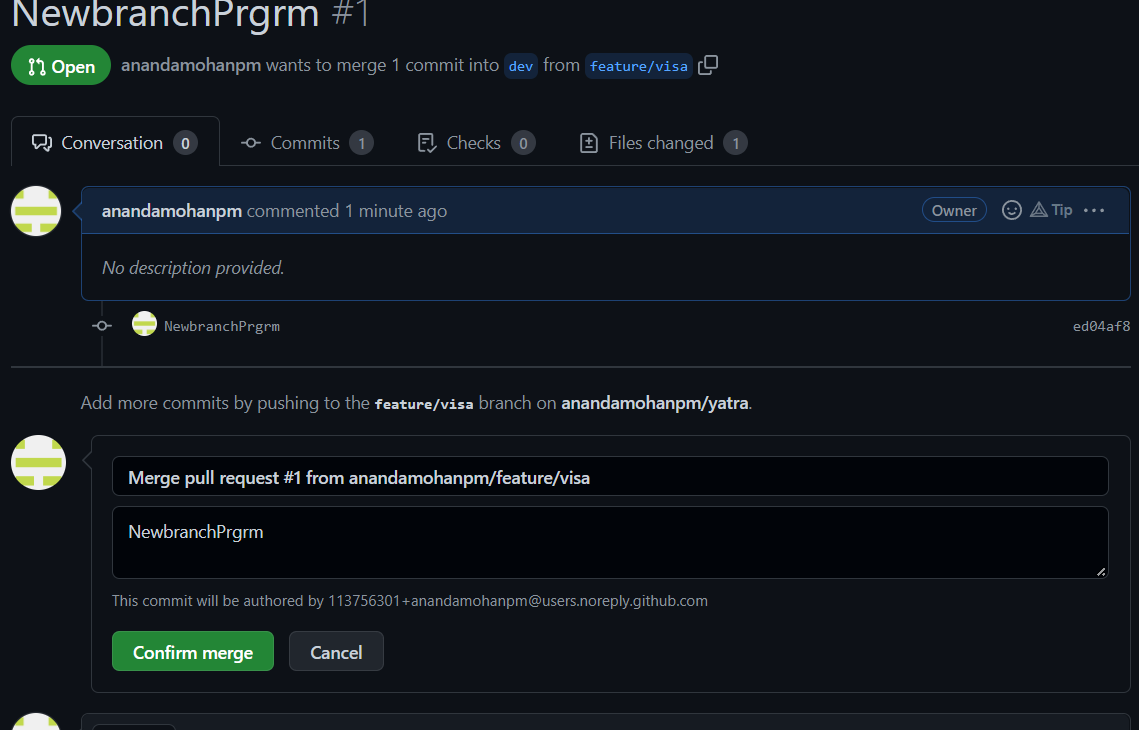
Now the **visa.java** class is inside **feature/visa** branch only. We can nowpull to **dev** branch. Change the **base branch** to **dev.** Then click on **pull request.**

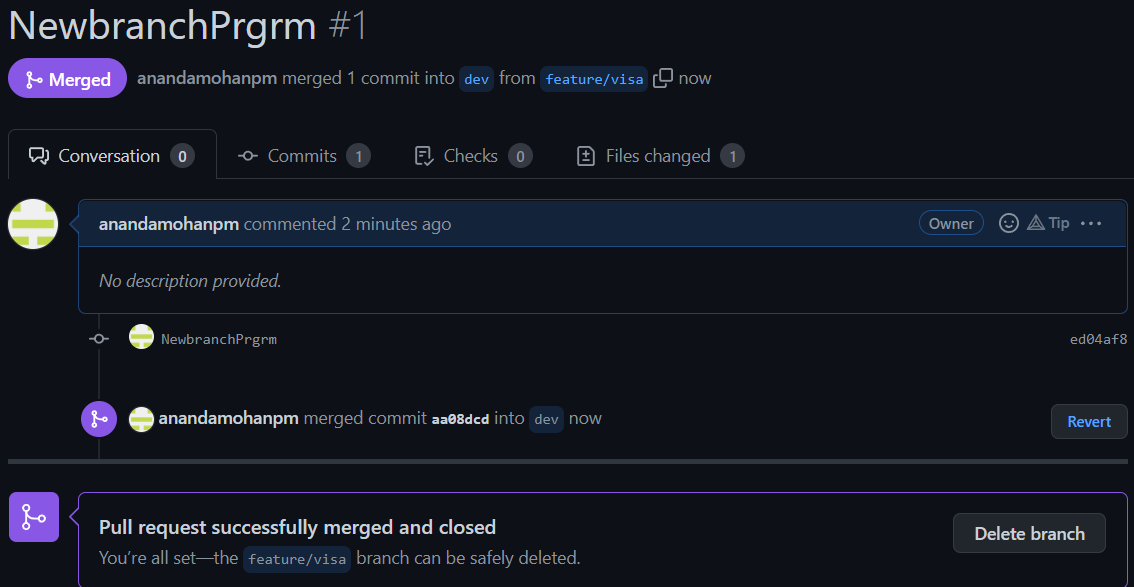


Click on **merge pull request.**



Click on **confirm merge**





You can follow the same procedure what we done here and merge to other branches. **Dev to preprod, preprod to prod, prod to master.**