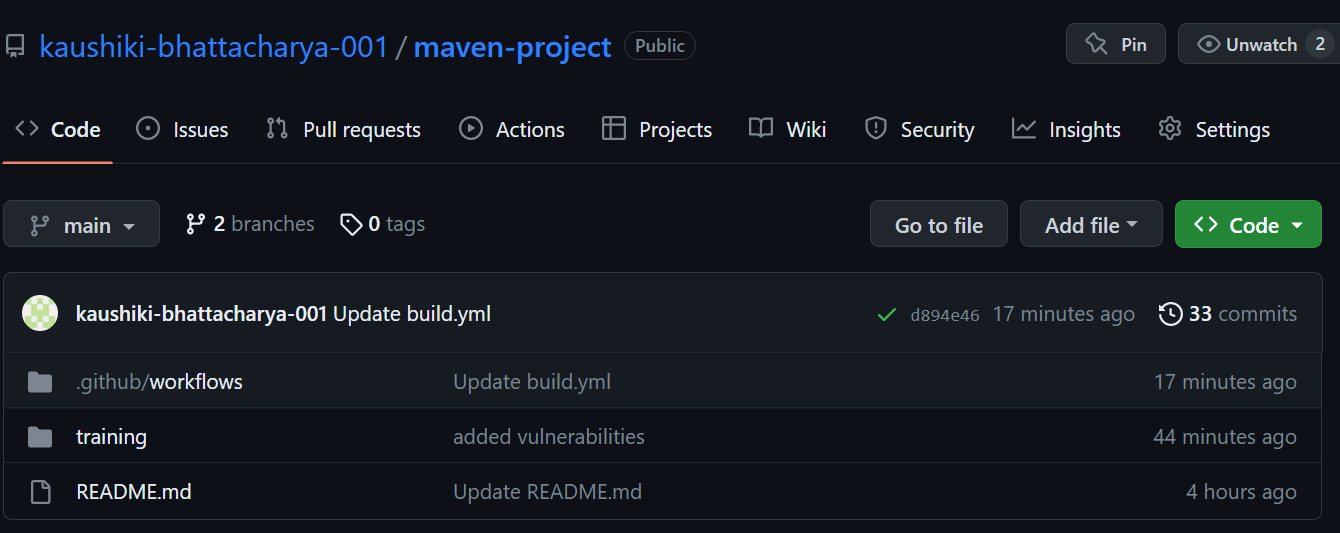
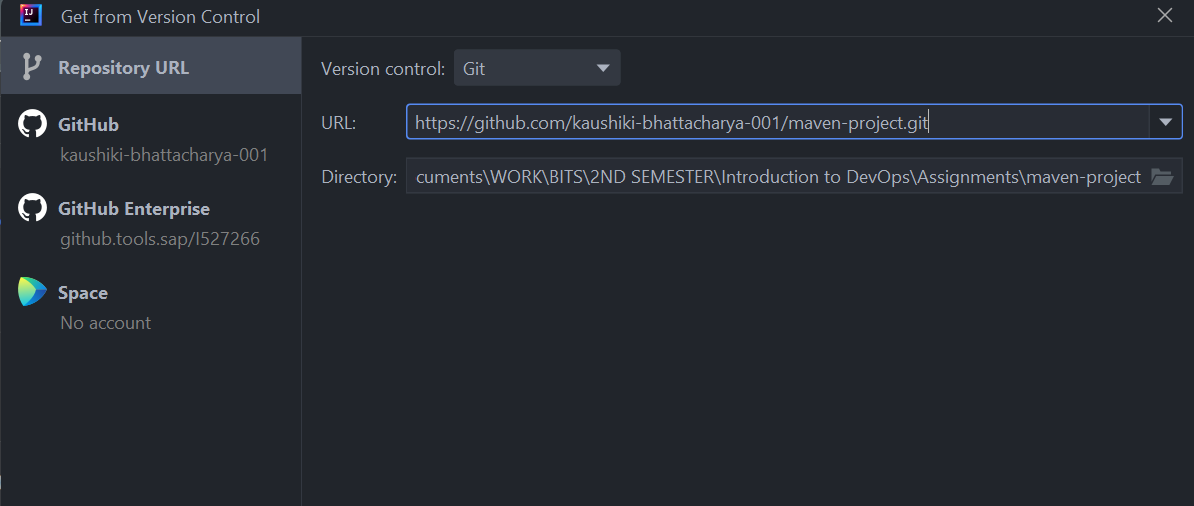
LAB ASSIGNMENTS

1. GIT
2. Get your project up and running locally

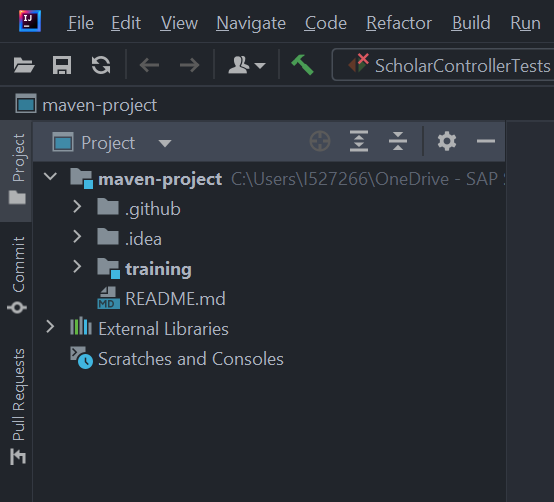
Firstly, I created a Maven project using SpringBoot, and uploaded the project in GitHub repo maven-project.



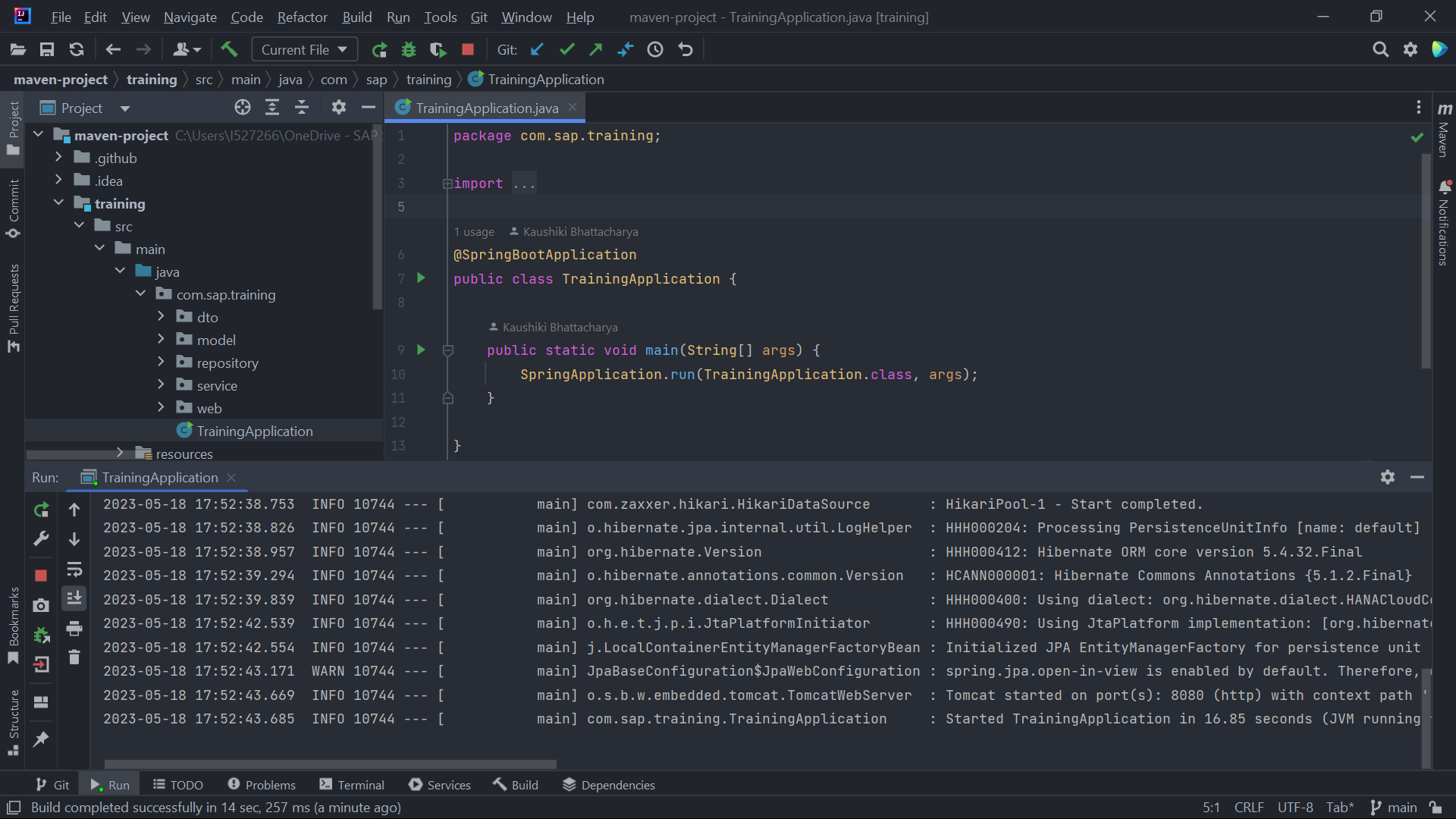
Using git clone, I have setup the repository locally.



Project in IntelliJ :



After running maven clean install, running the application:

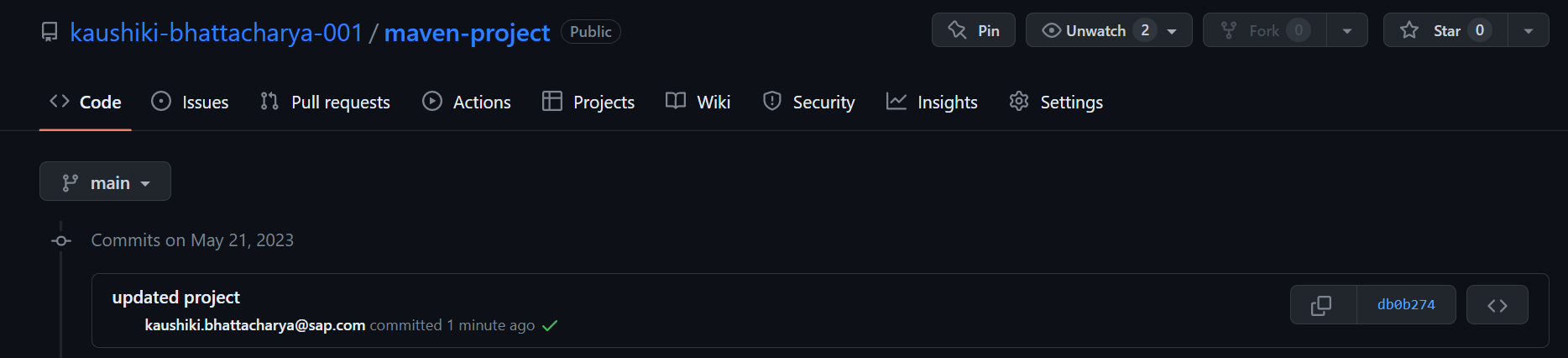


1. Build an initial CI pipeline on GitHub

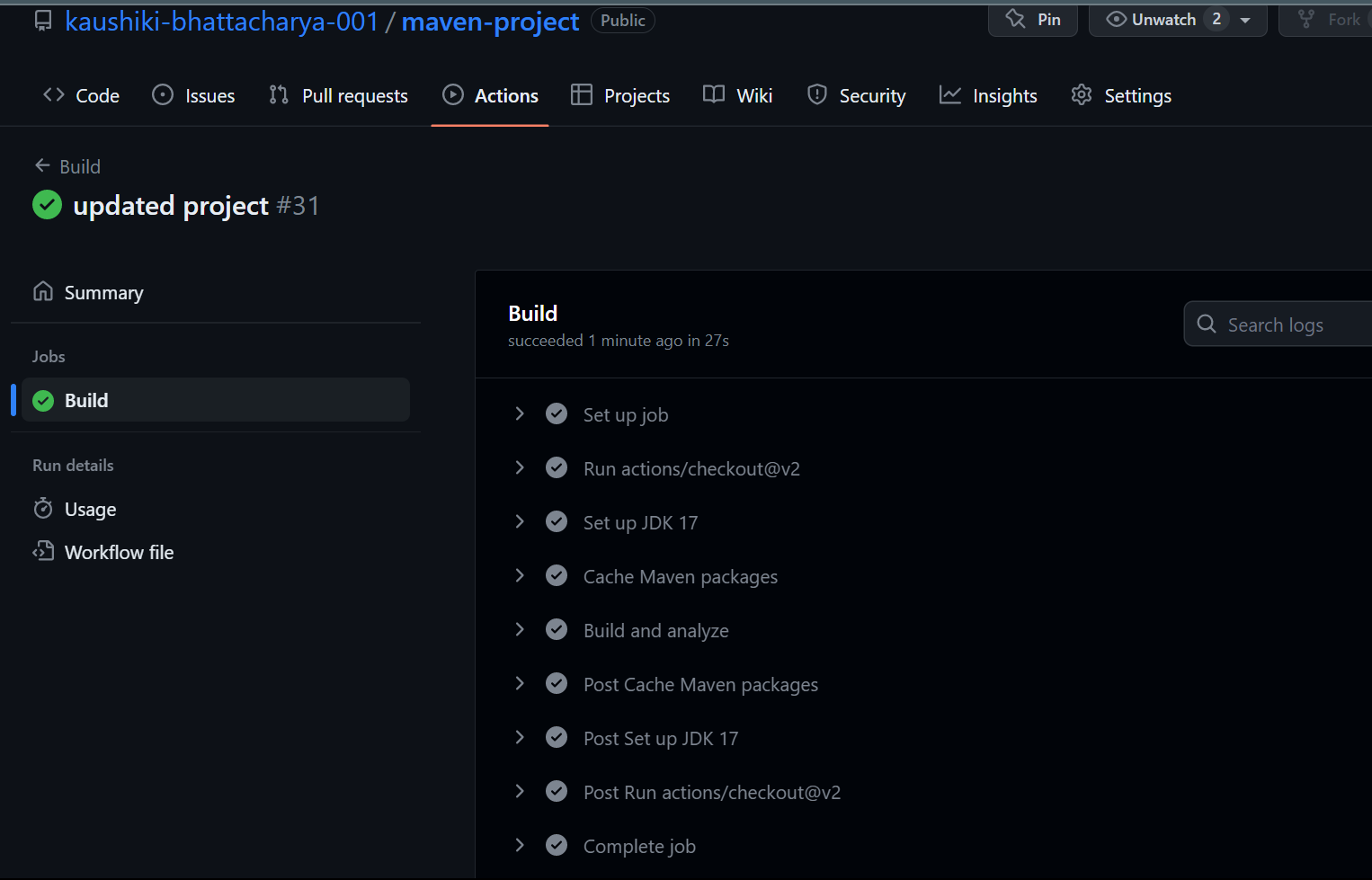
We can easily show Continuous Integration in GitHub using a build.yml file in the repository. Using this, everytime we push some code in our repository, it automatically triggers the build.yml file and generates a build report.

Inside the build.yml file, we can configure stages to run during the build, like, install JDK or Maven, run maven install and so on.

Commited new push



Build ran successfully



Build.yml file

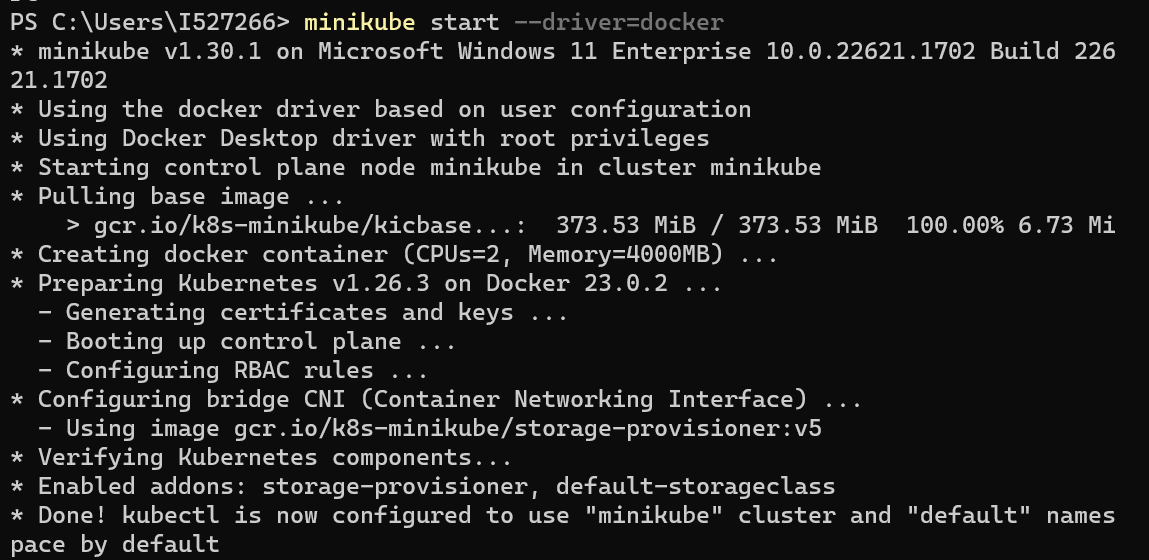


KUBERNETES:

First install docker desktop

Install exe file

Open powershell -> minikube start --driver=docker



Docker build : docker build -t scholar-project.jar .

Image load: minikube image load scholar-project.jar:latest

Build Pod: kubectl apply -f deployment.yaml

Check Pod and Deployment: kubectl get pods/deployments

Build Service: kubectl apply -f service.yaml

Check Service: kubectl get service

Check Nodes: kubectl get nodes -o wide

Start Service: minikube service scholar-project-svc