16 July – 2024

Day 6

8 Creating job to build react js project.

To create react js project node js must present

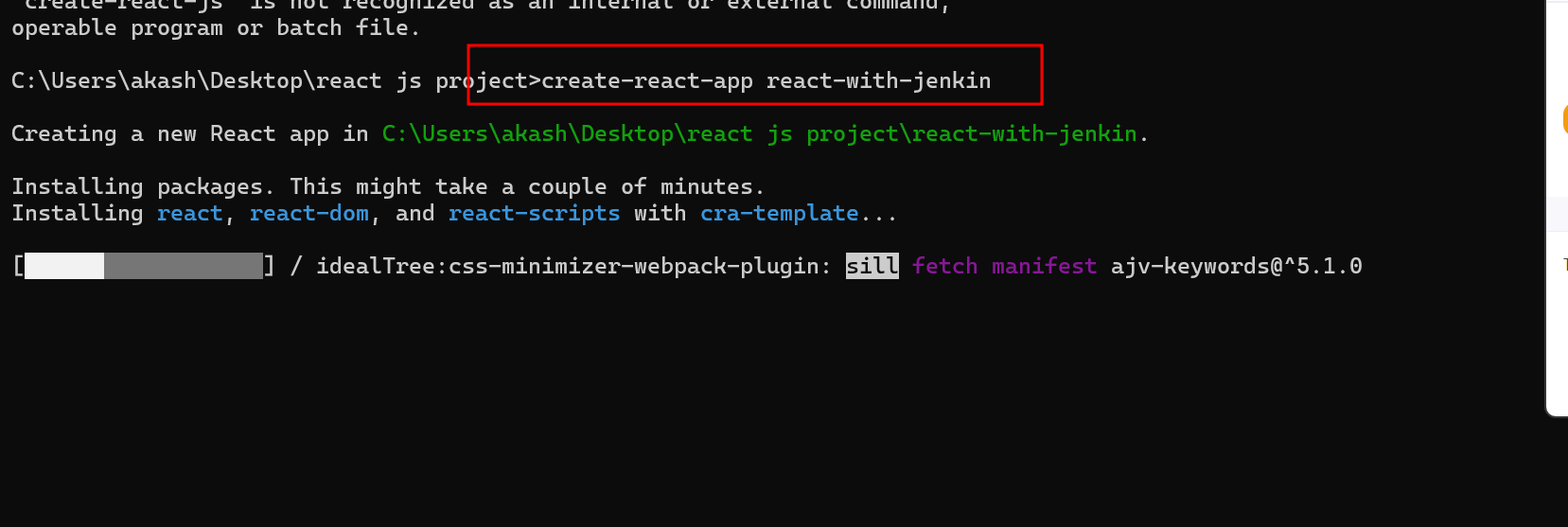
node --version

npx create-react-app react-with-jenkin

or

npm install react-react-app -g

create-react-app react-with-jenkin



Do some changes in App.js file and using npm start command please test your project.

Please push this project from local machine to your remote repository

git init

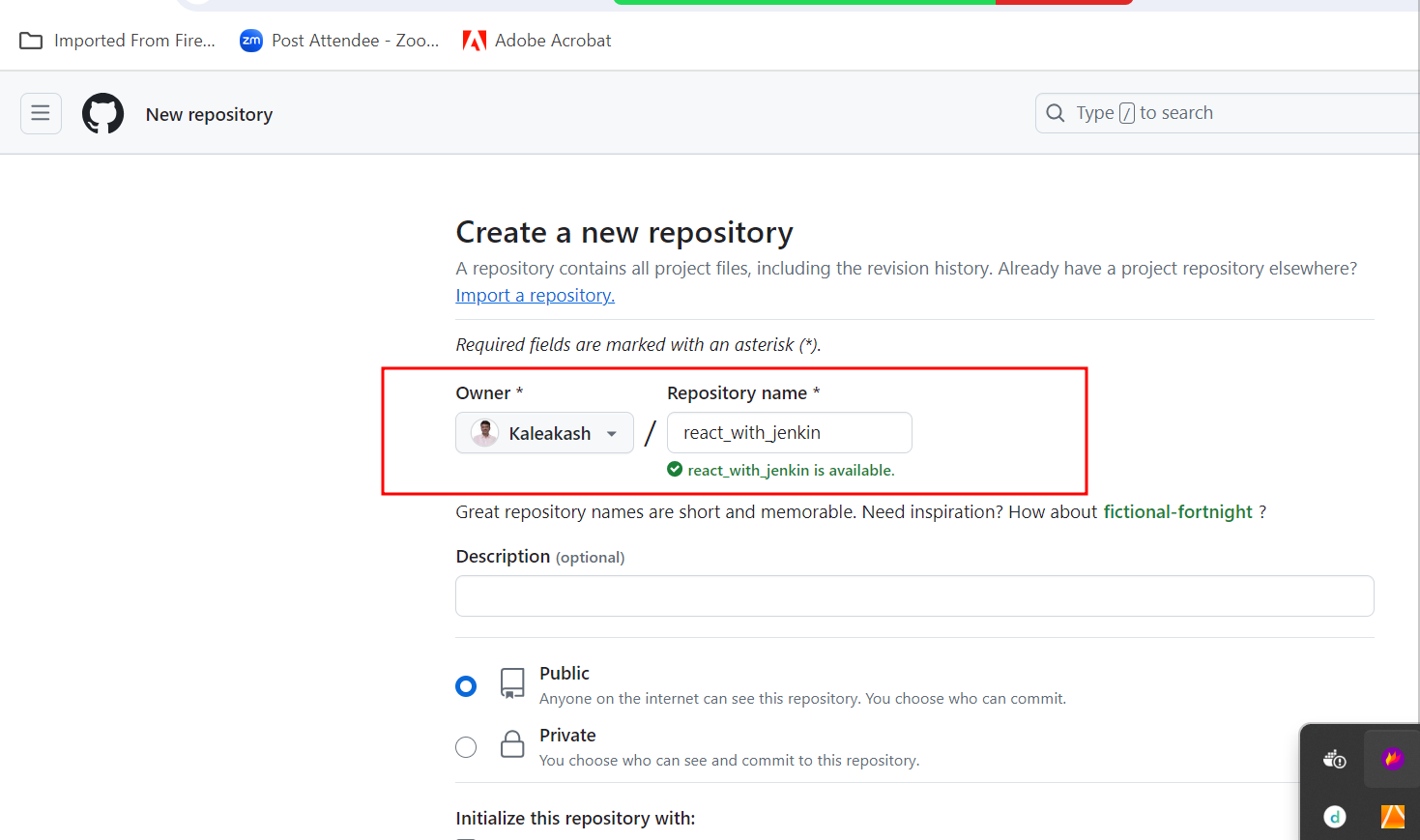
git add .

git commit -m “simple react project ready”

your code is push file os file system to local repository

please create remote repository in your git hub account and link that repository

with your local repository and push your project from local to remote repository.



**Below command connect my local repository code with remote repository**

**git remote add origin https://github.com/Kaleakash/react\_with\_jenkin.git**

**Now we can’t push this code to remote repository**

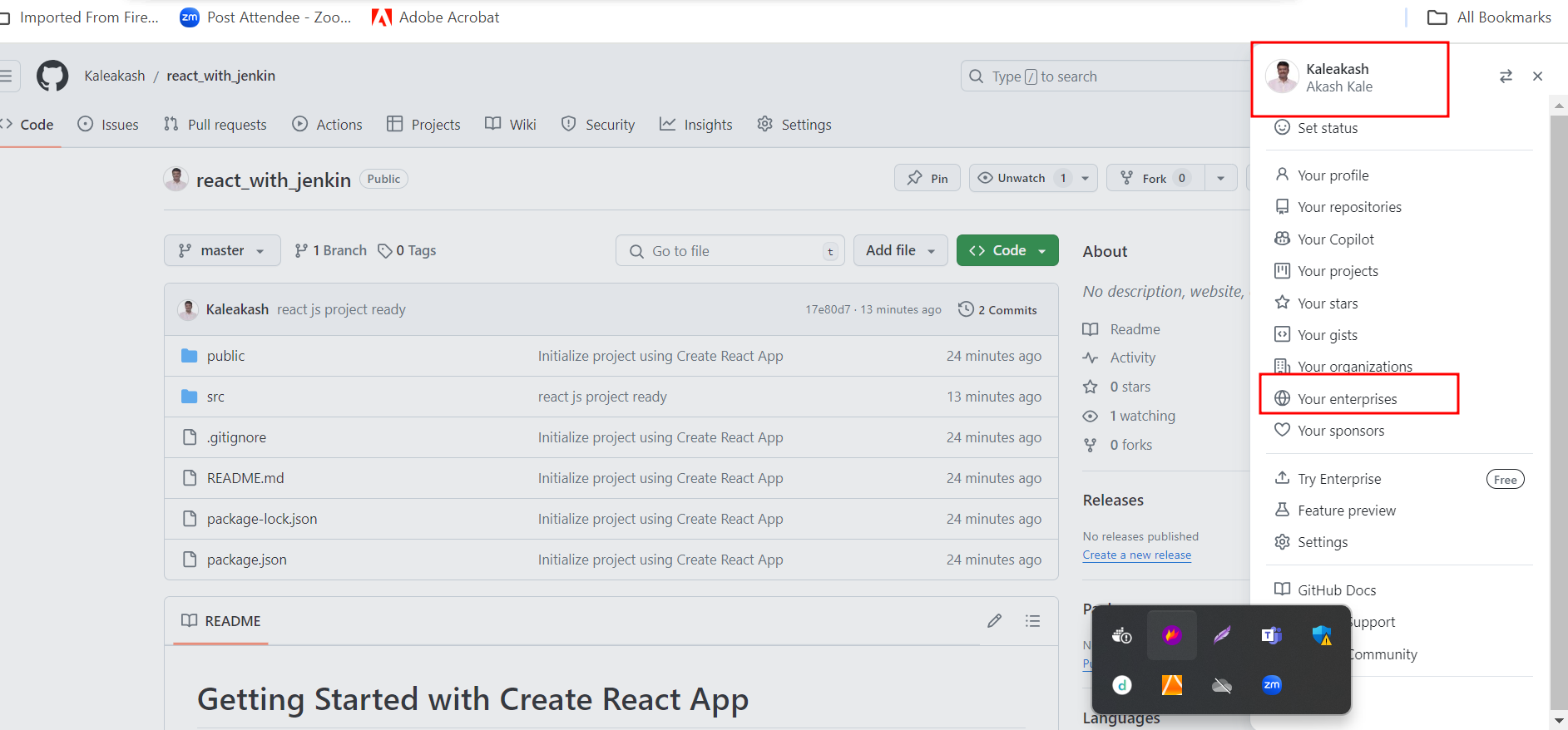
**If you are not able to push the code successfully because of some reason. We will use token to push the code**

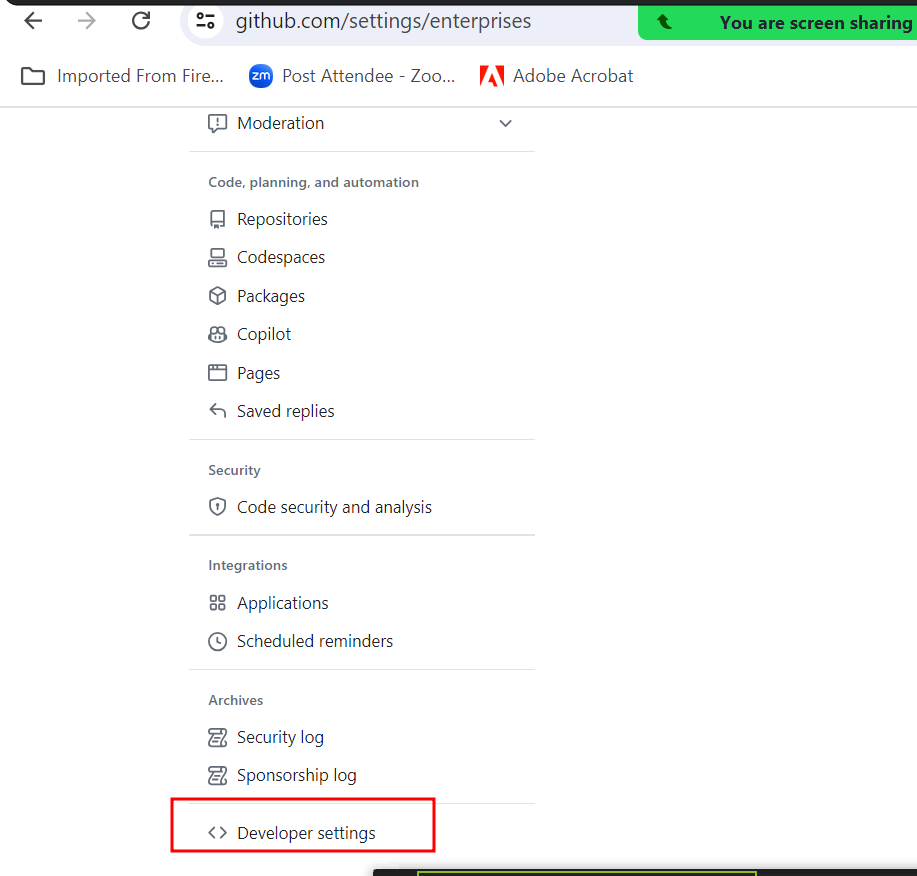
**Already local machine connected with remote machine using URL and that URL store in origin keyword.**

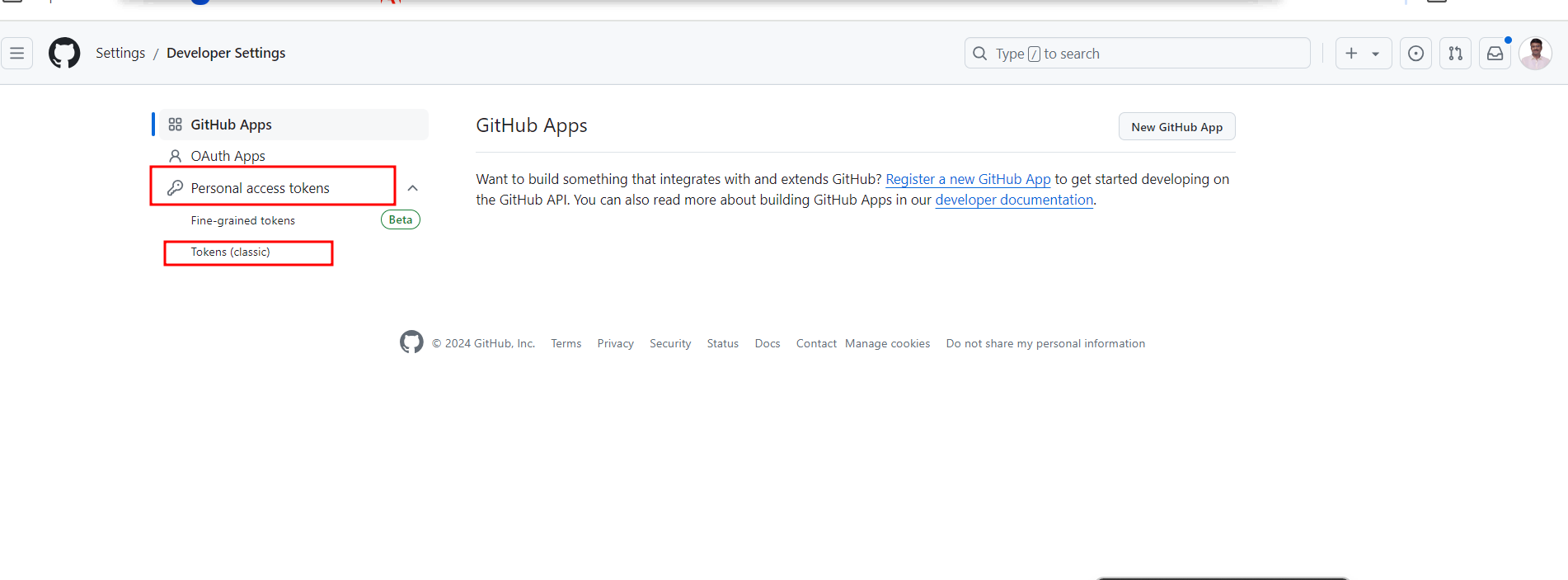
**git remote rm origin this command is use to remove origin or**

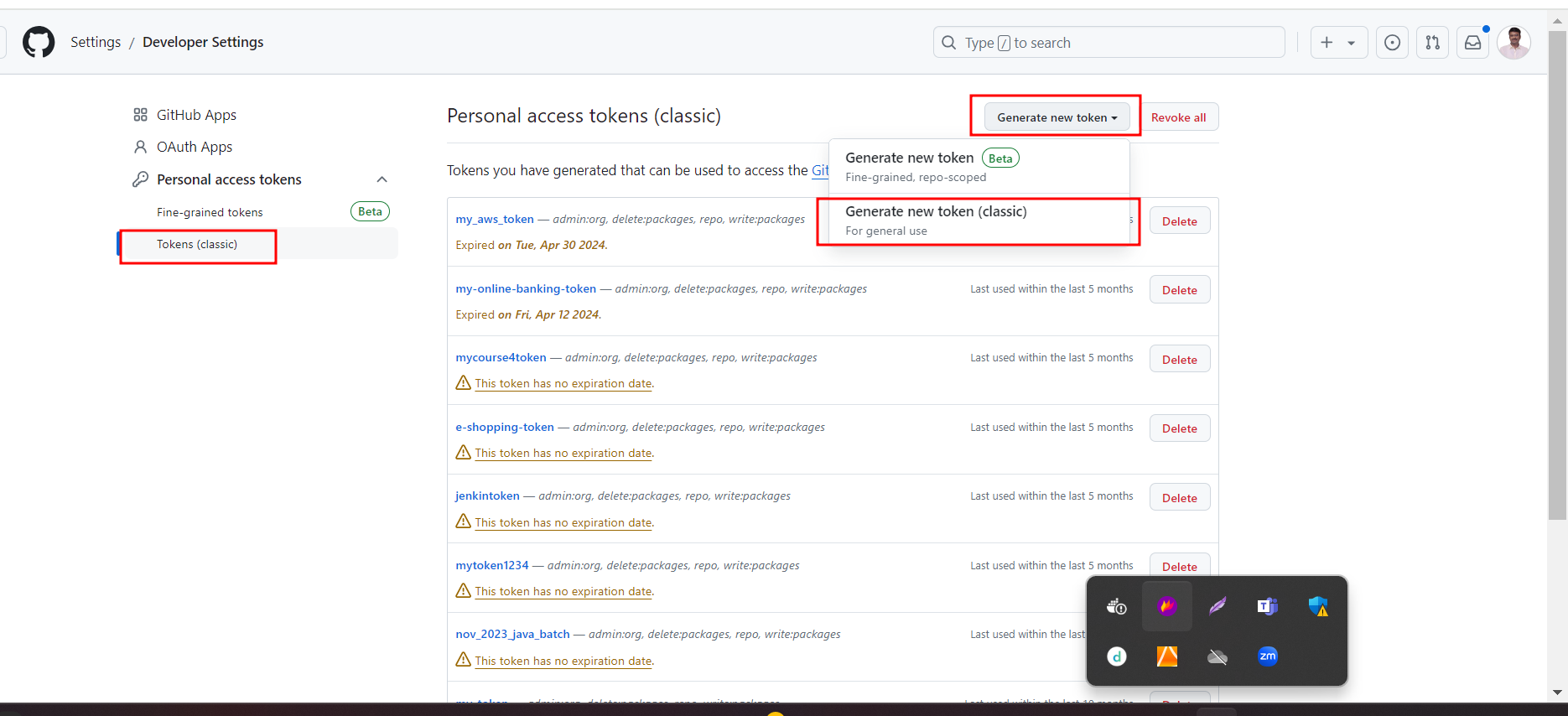
**link with remote repository**

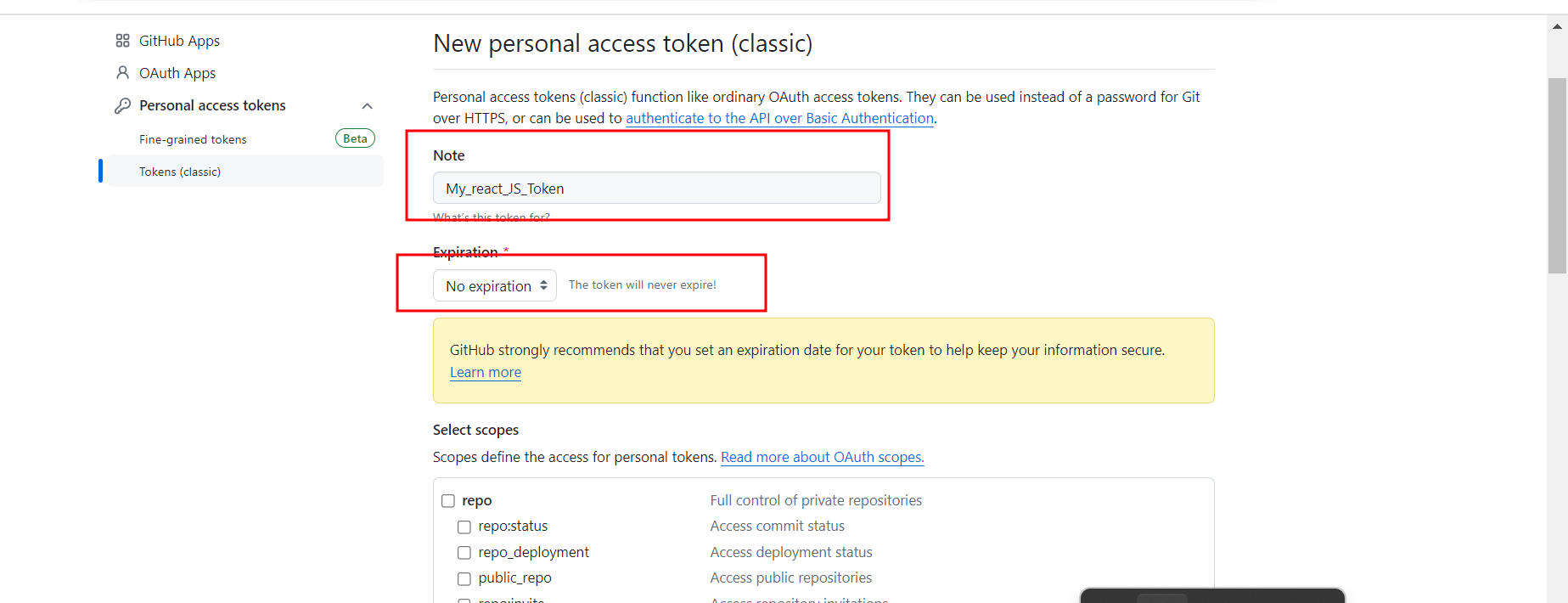
**Now we will generate token.**

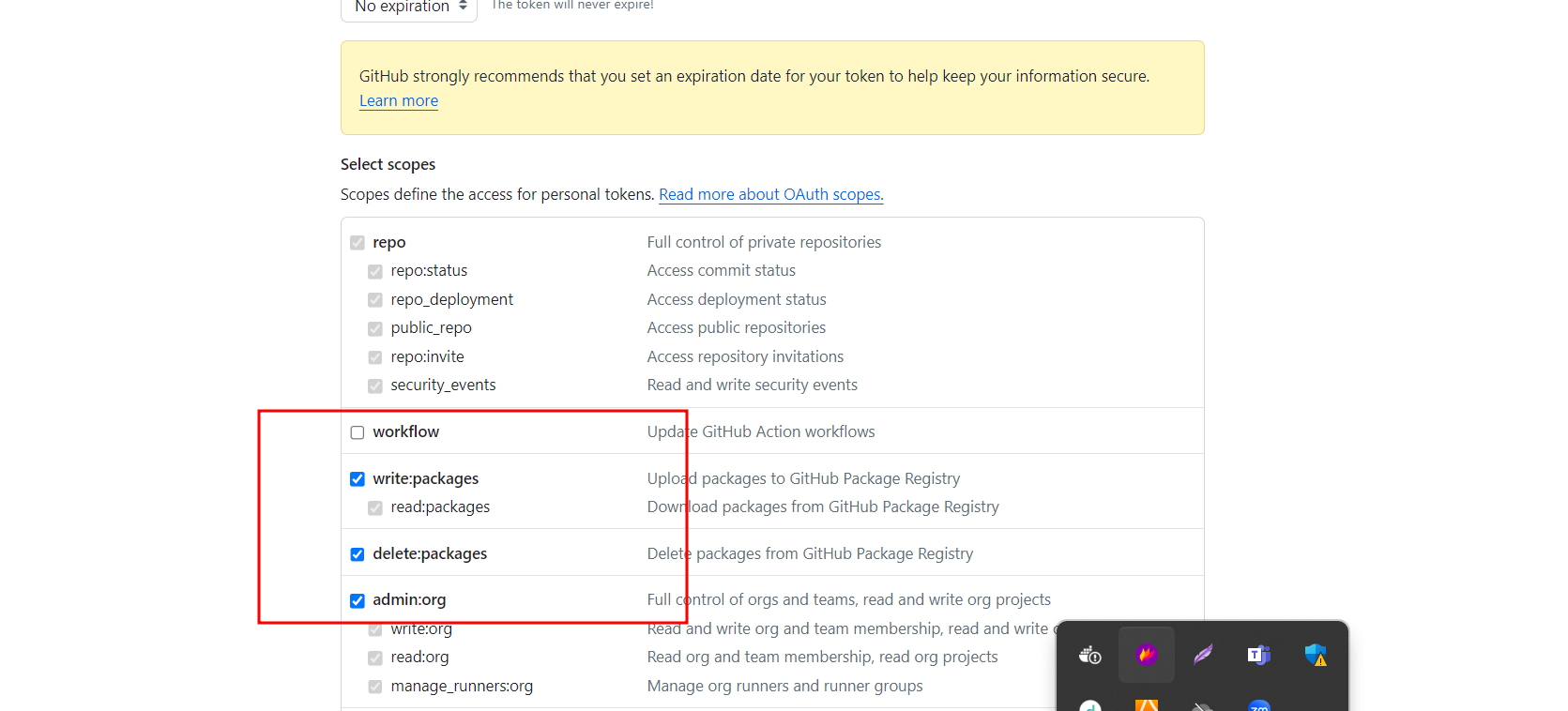


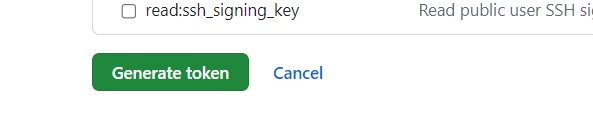




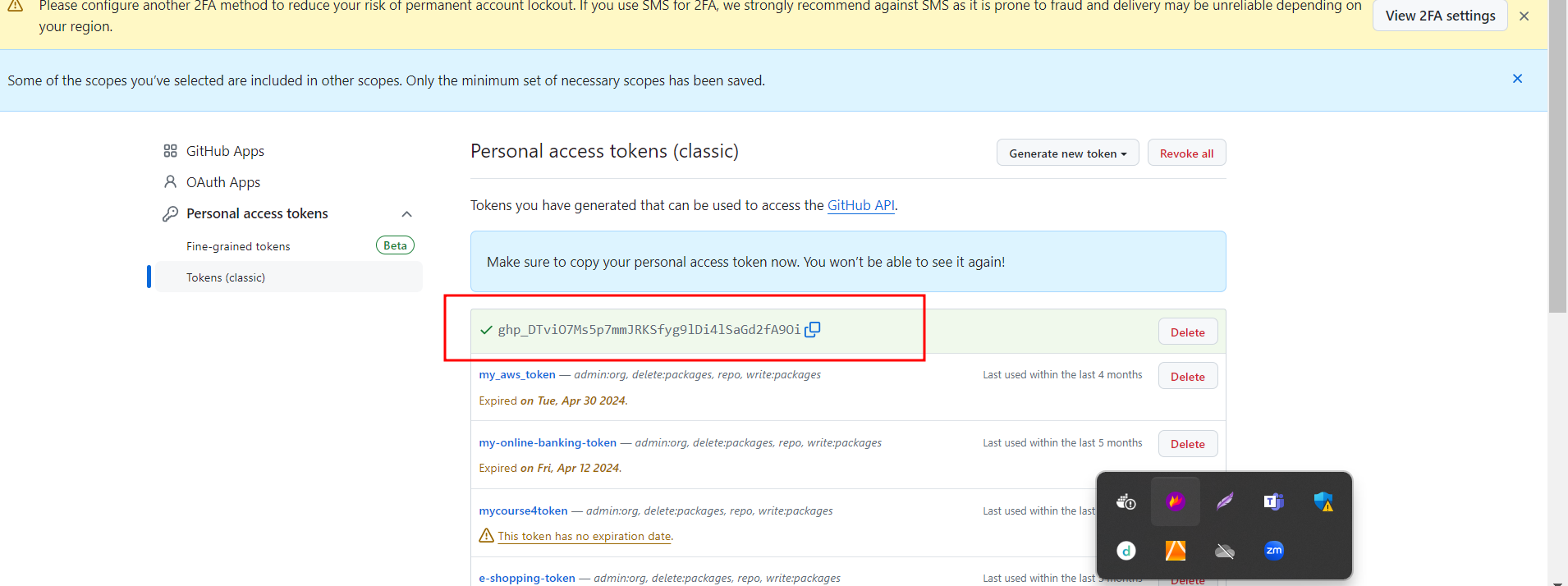








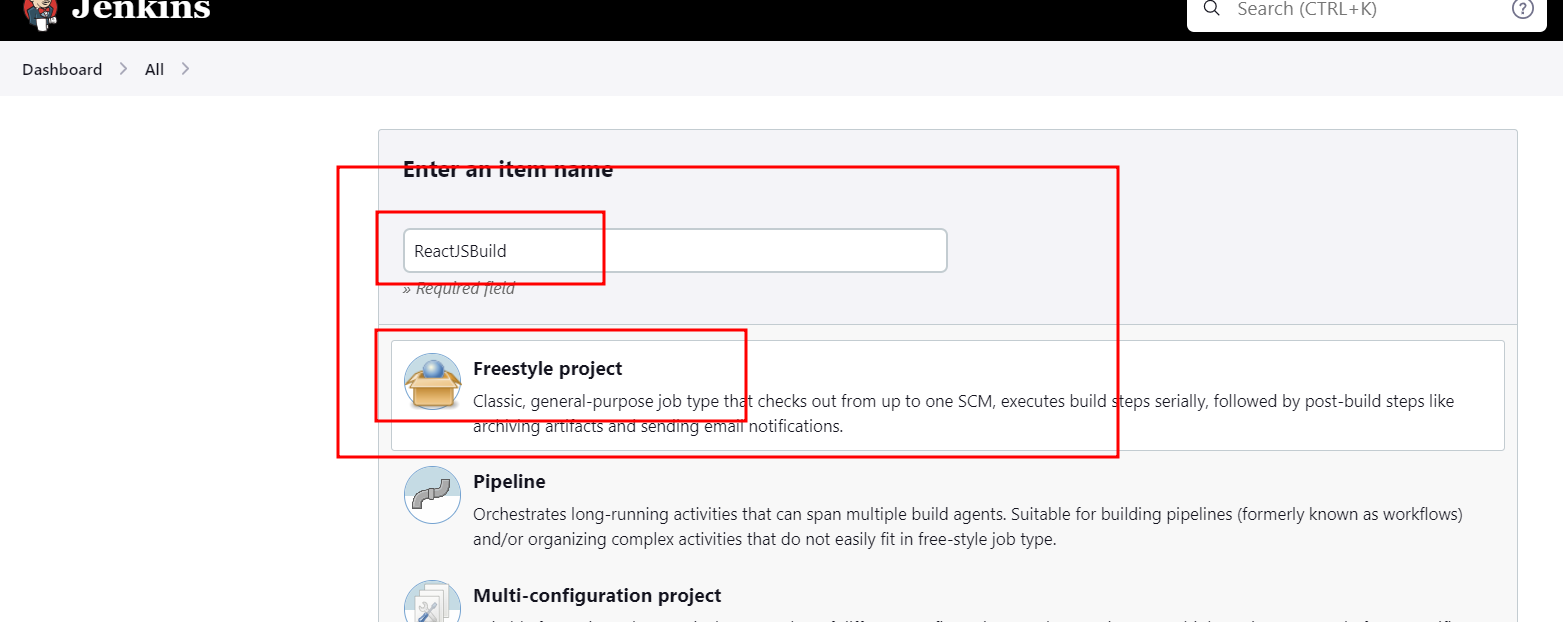
**Copy the token**

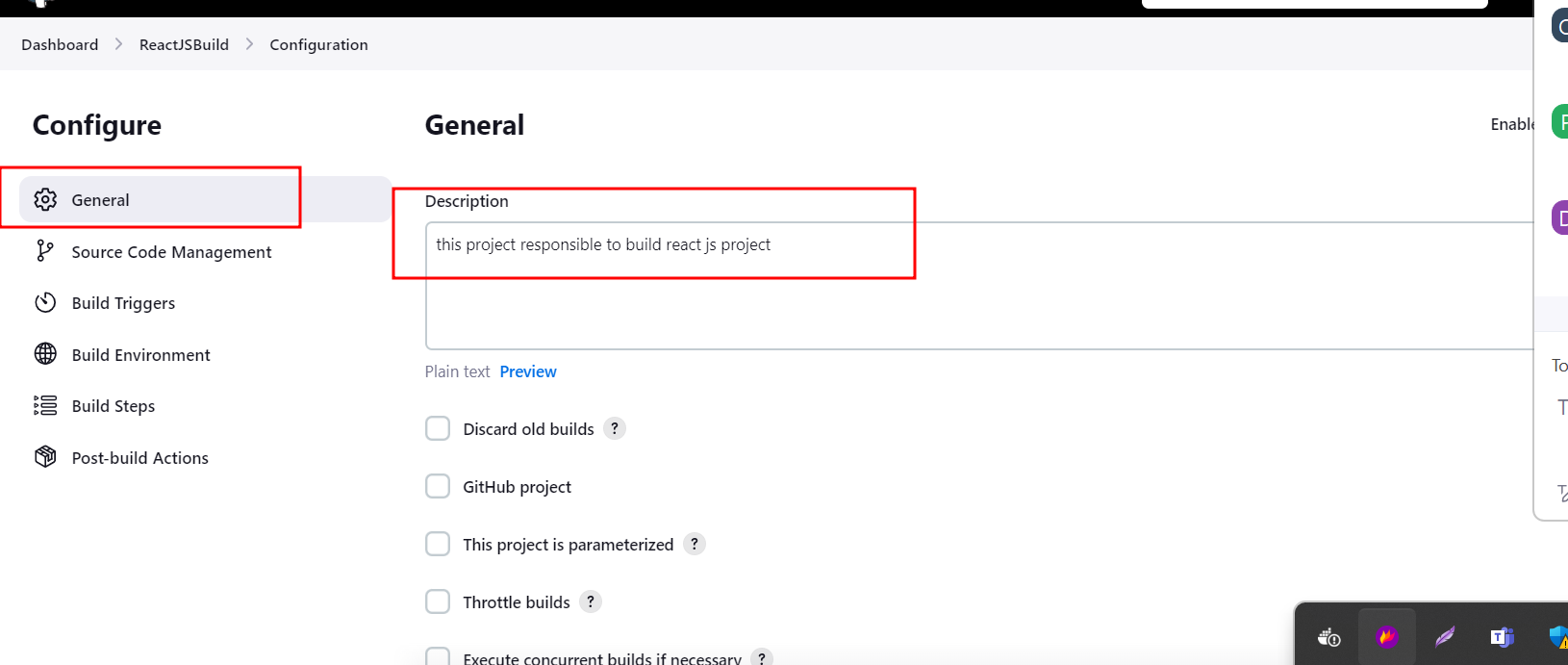


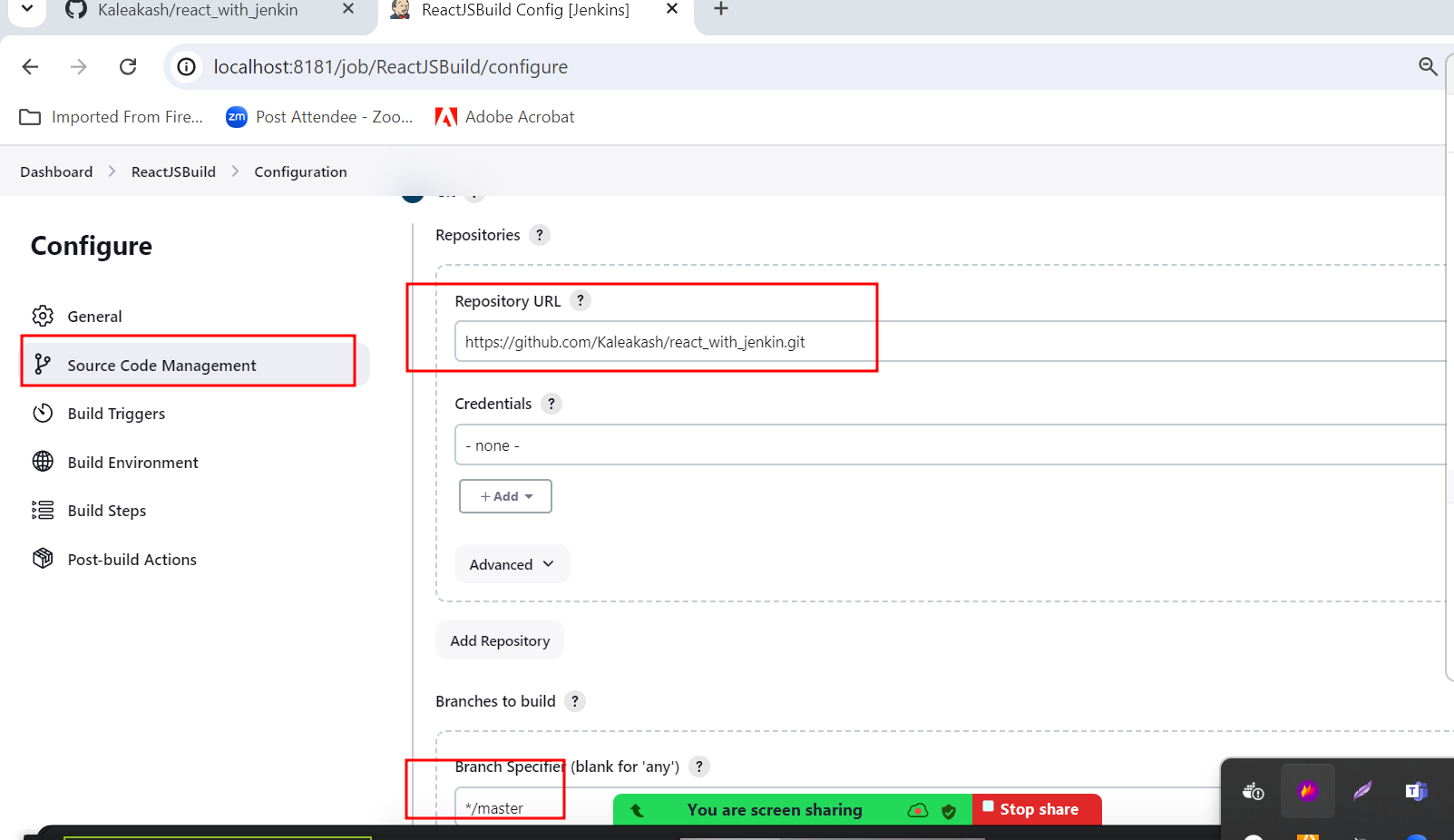
**git remote add origin https://ghp\_DTviO7Ms5p7mmJRKSfyg9lDi4lSaGd2fA9Oi@github.com/Kaleakash/react\_with\_jenkin.git**

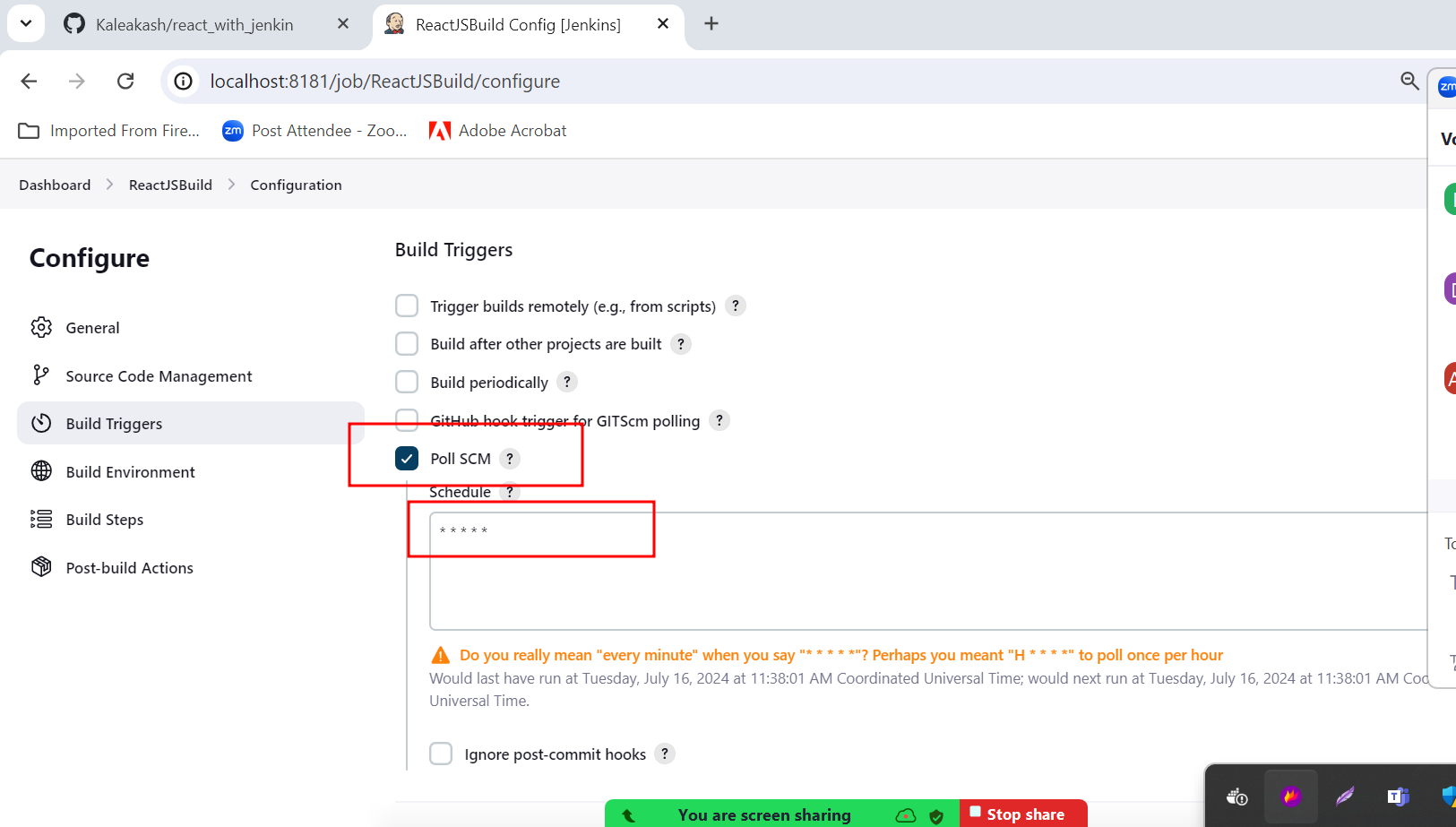
**git push -u origin HEAD**

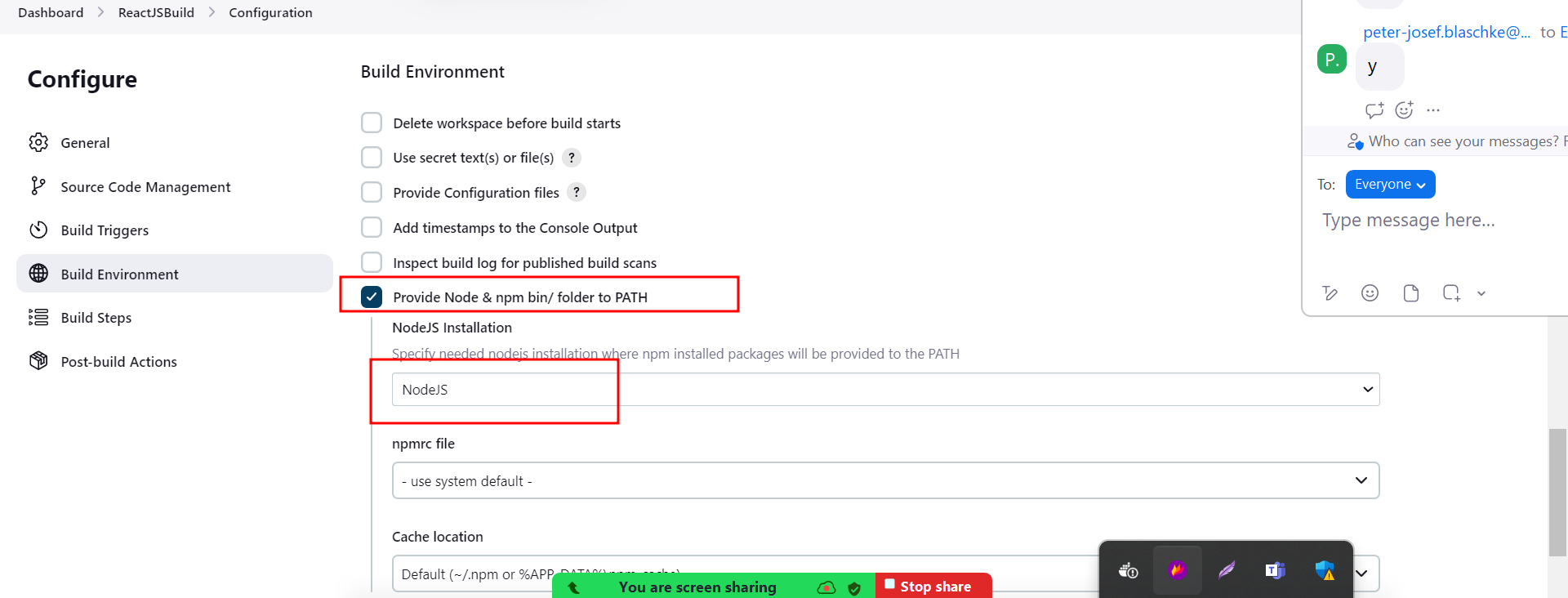
Creating Jenkin job to build react js project.

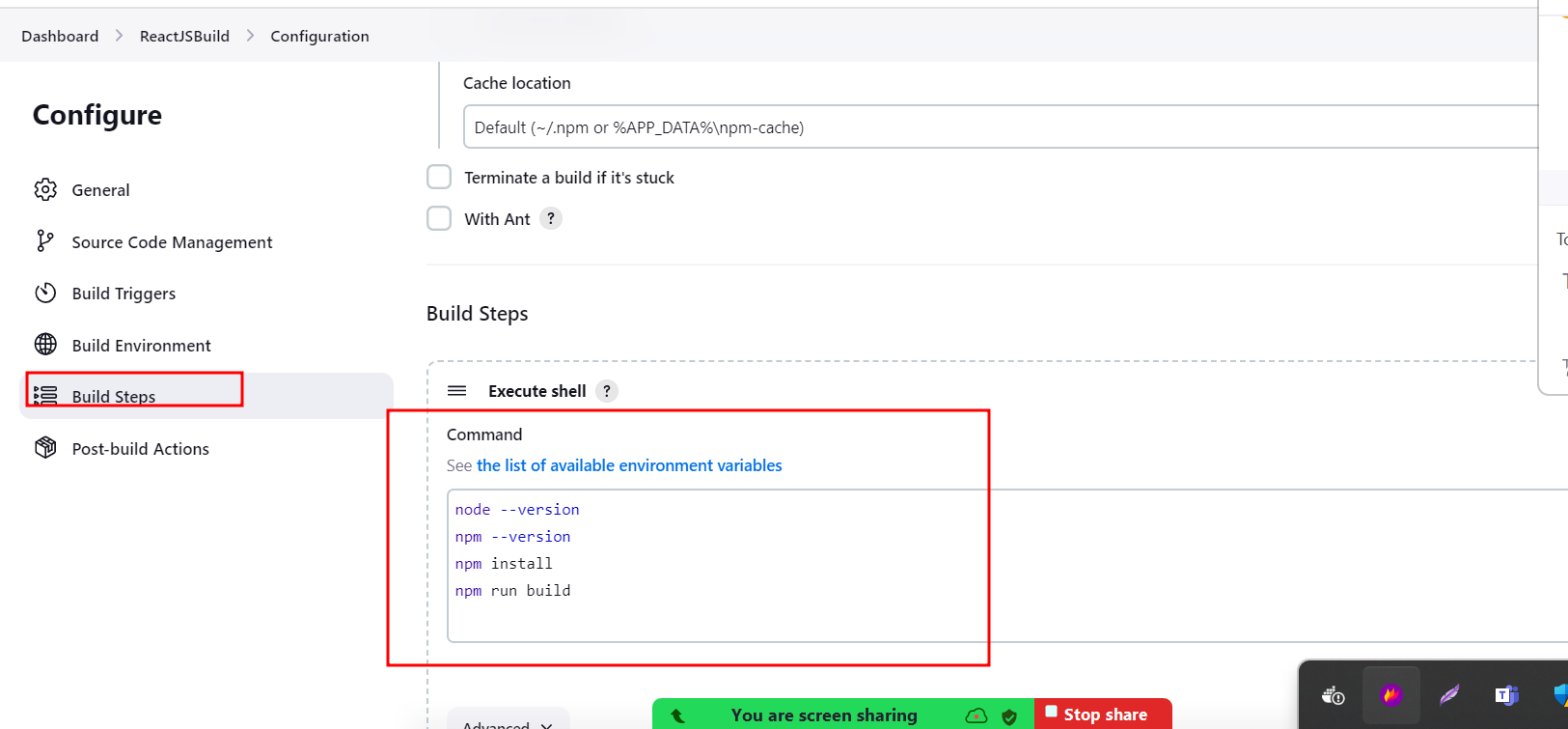






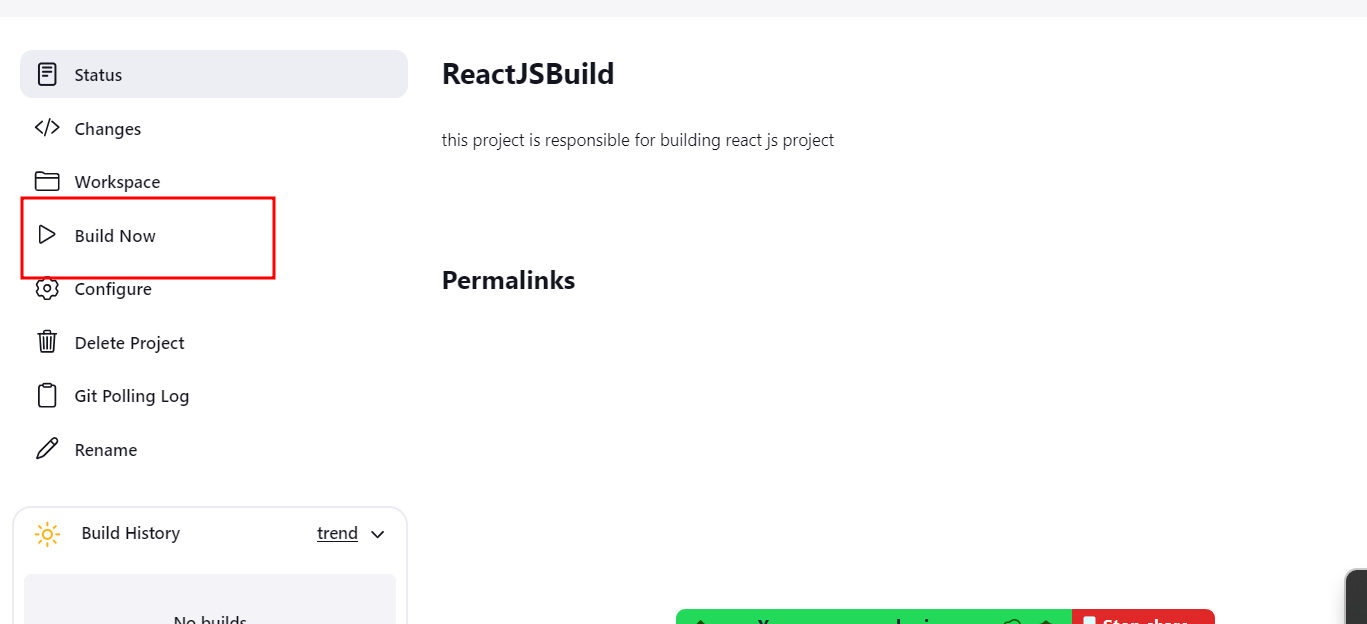


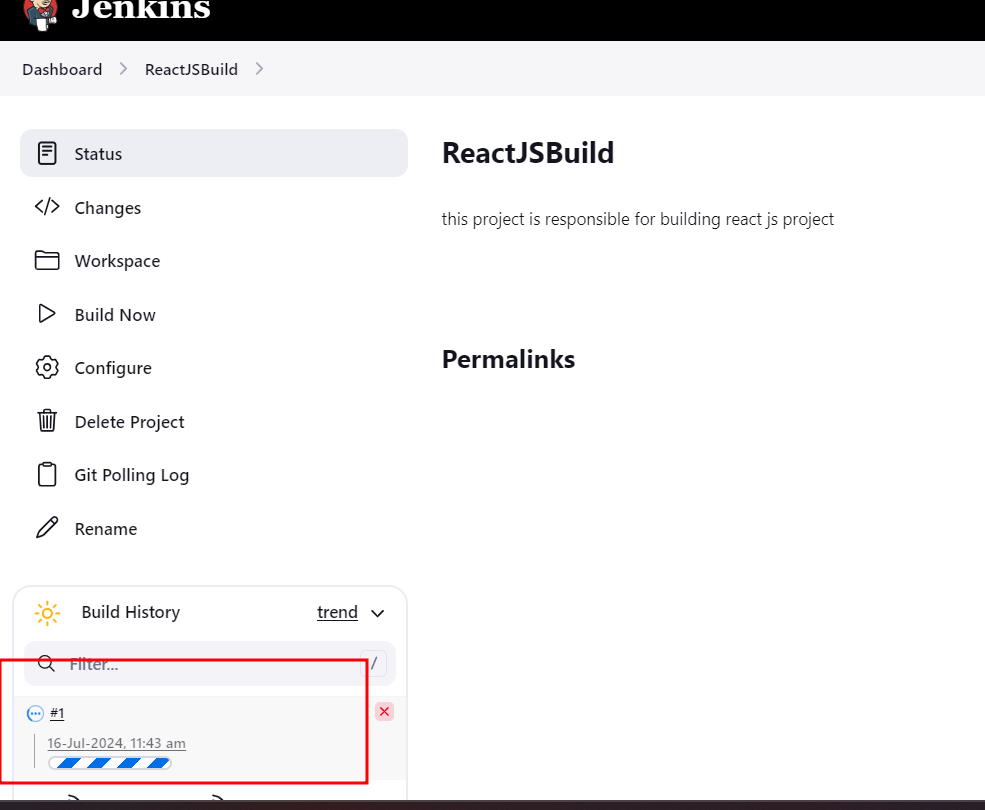




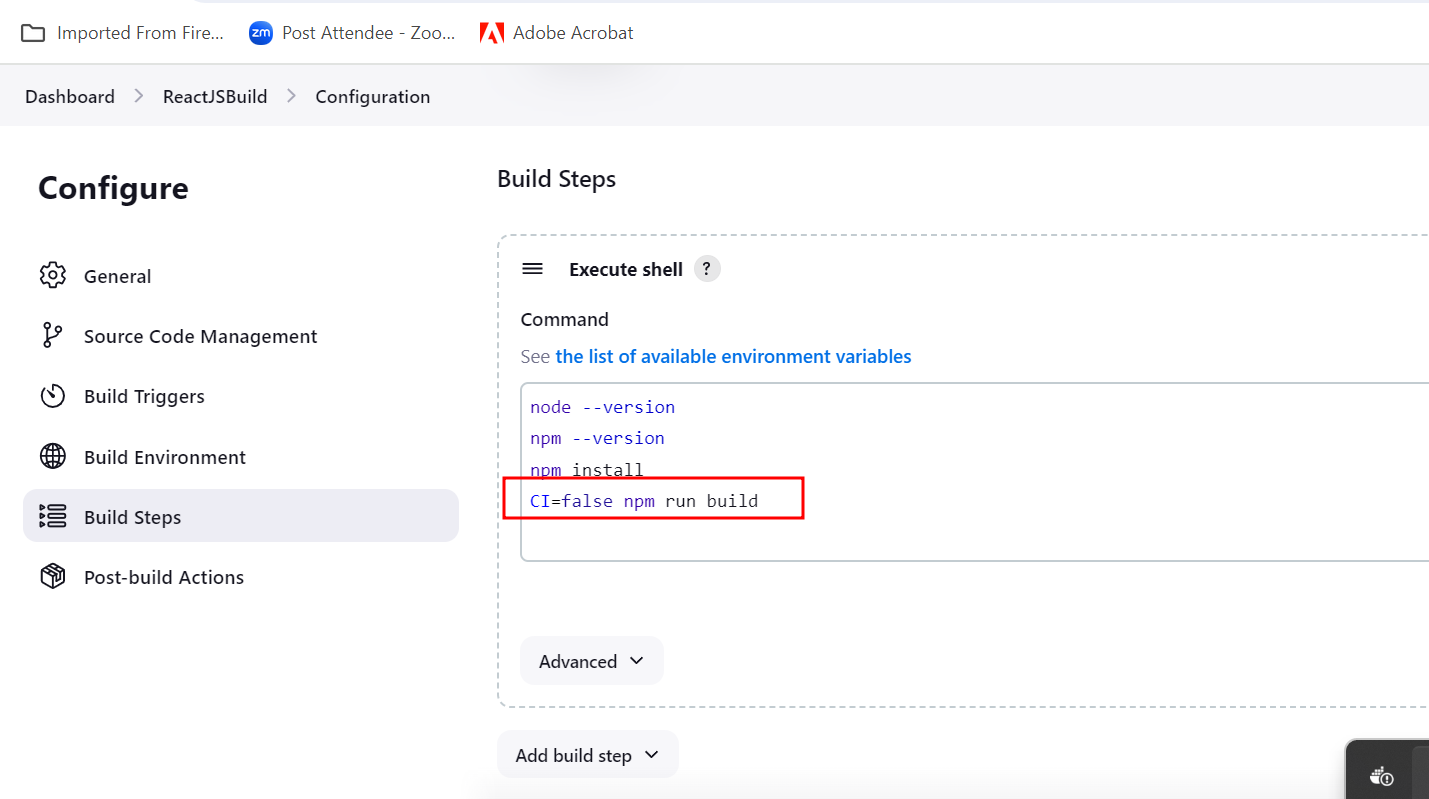
Click on apply and save

Then build the project



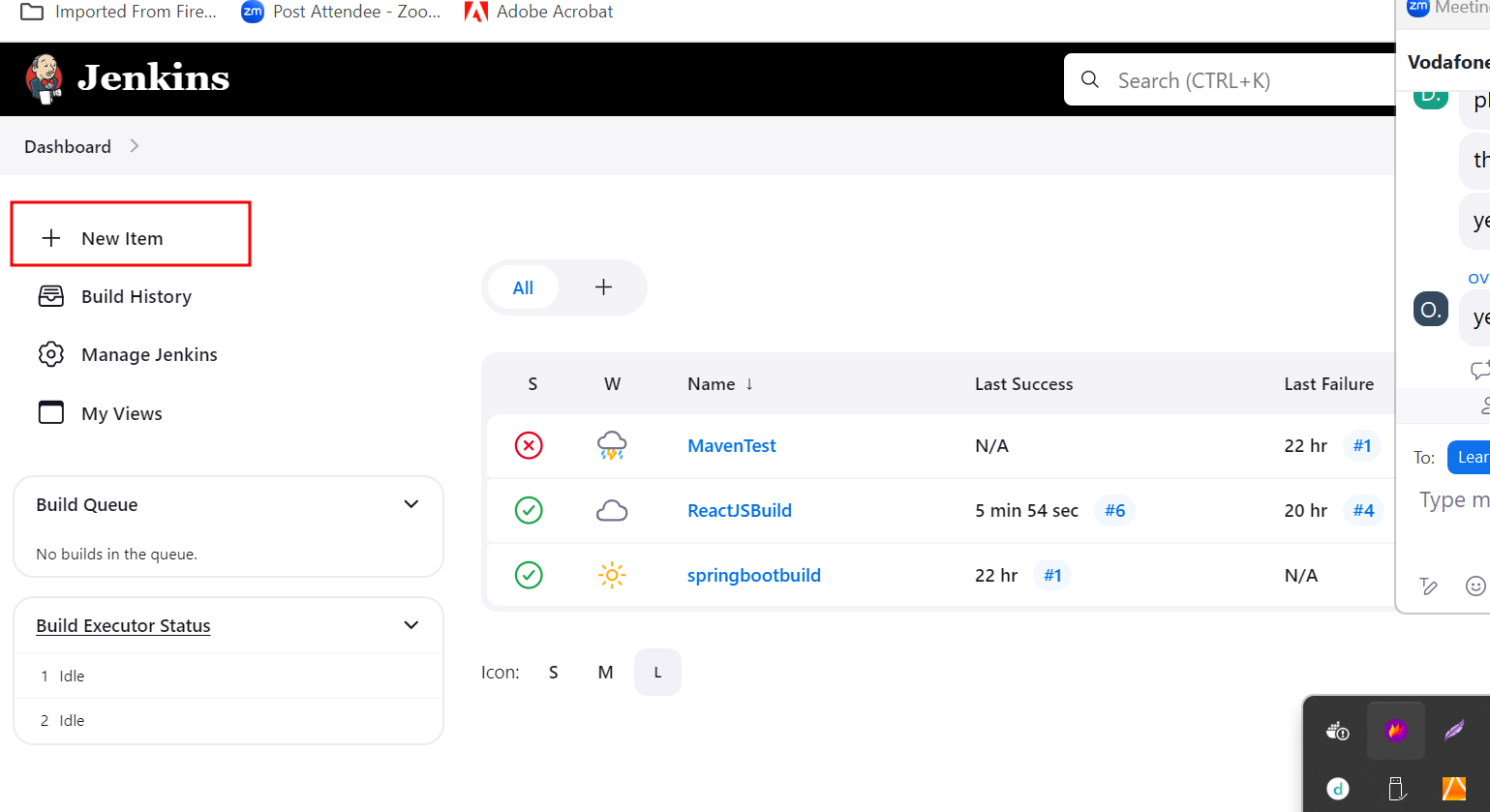


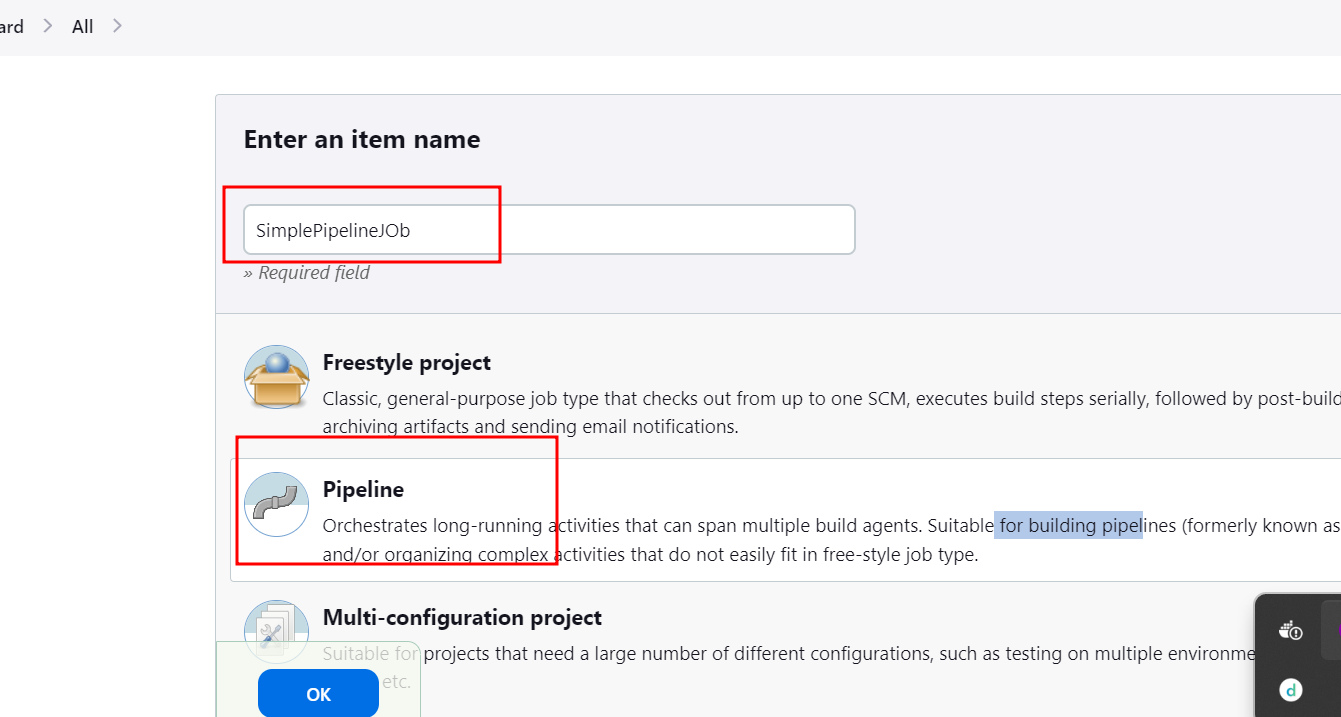
If fail and if you are running CI and CD tool using Docker images please write below

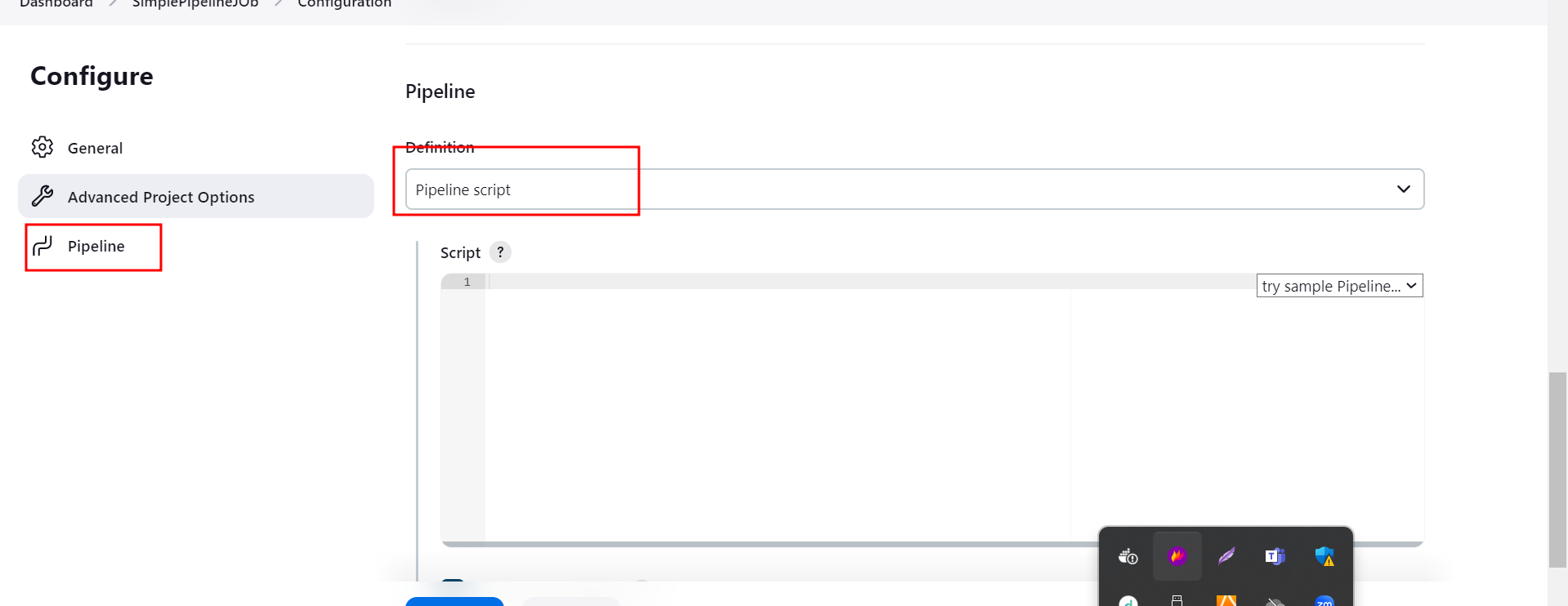


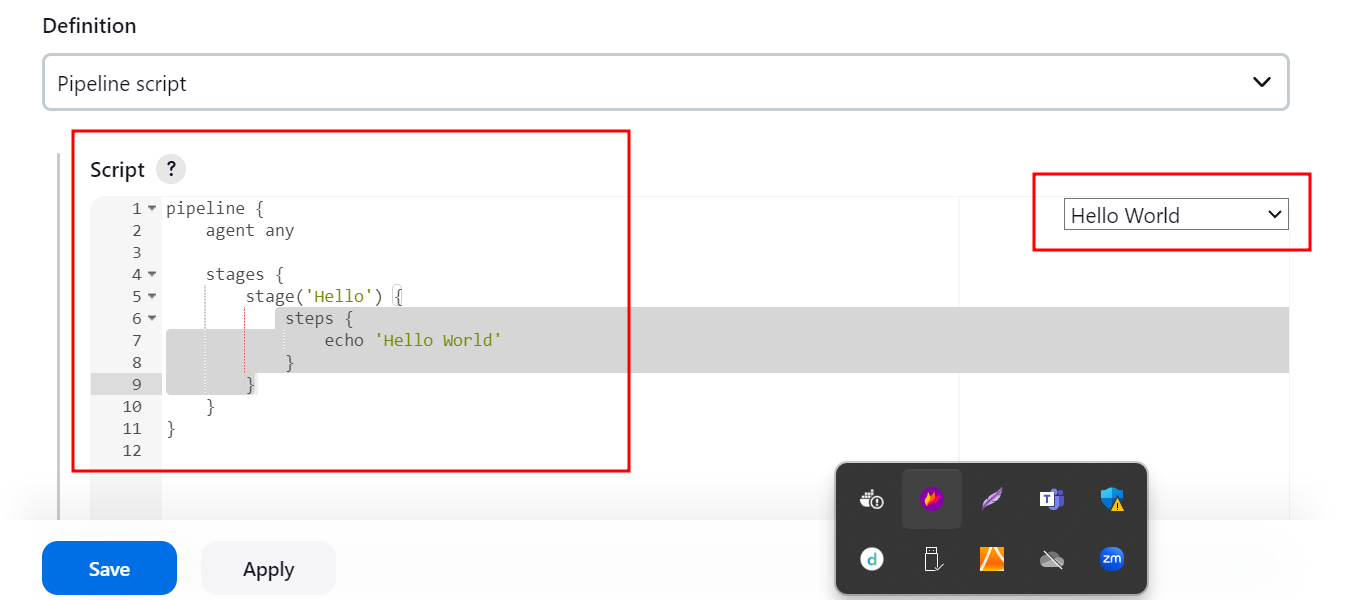
Jenkin Pipeline Or CI/CD pipeline : CI and CD pipe line which is responsible to execute set of event or jobs which interconnected with each others.

Creating simple CI and CD pipe line jobs.

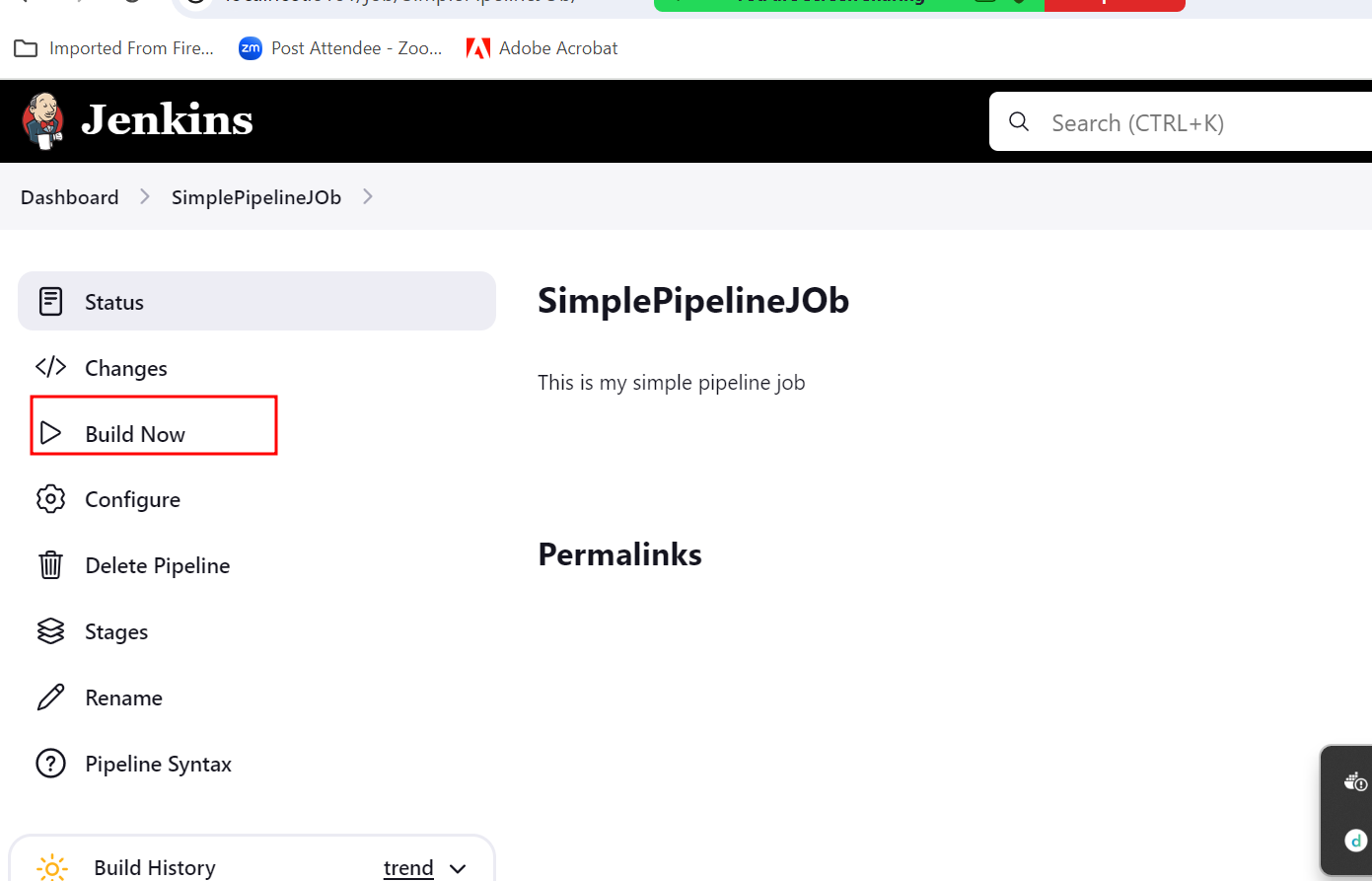


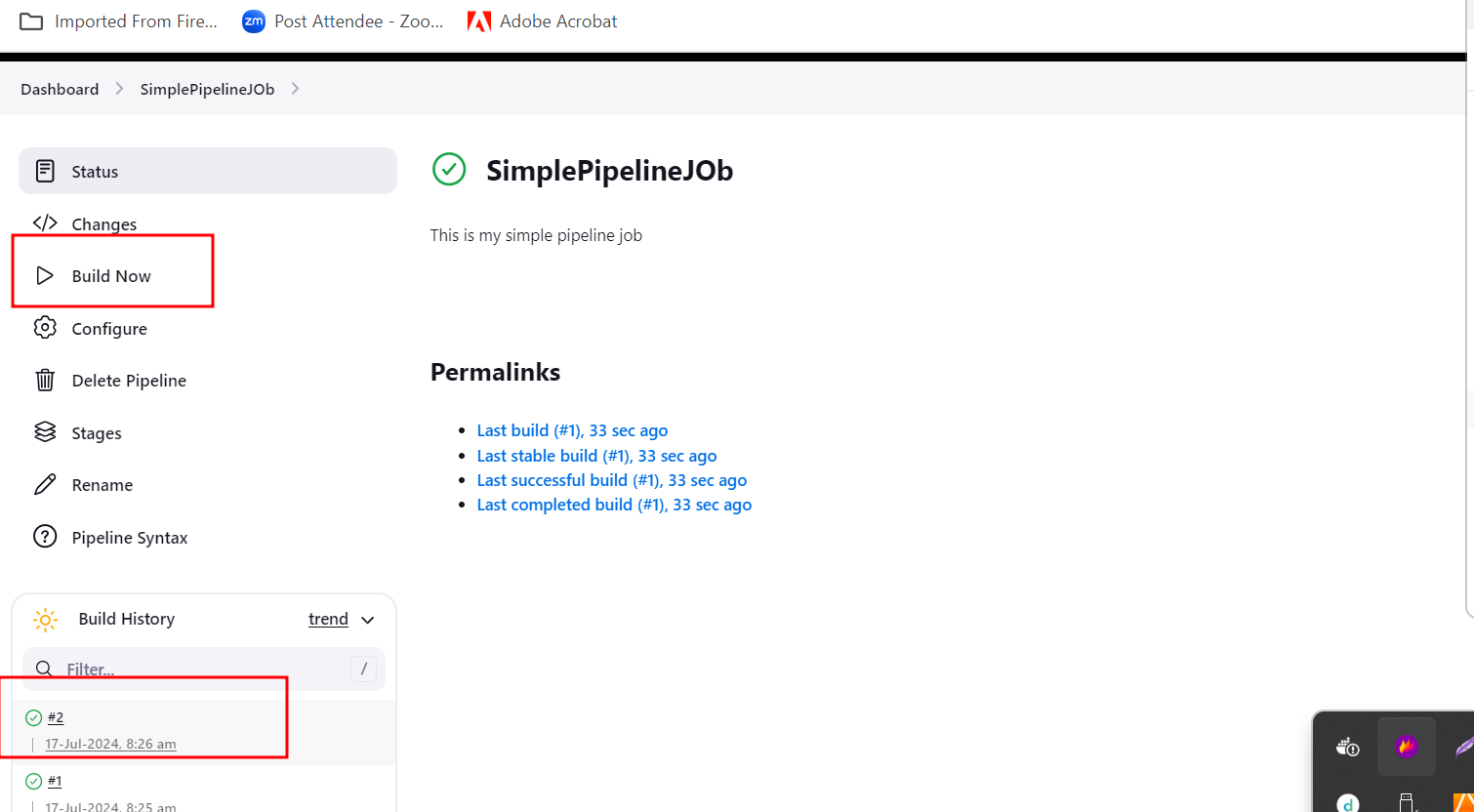


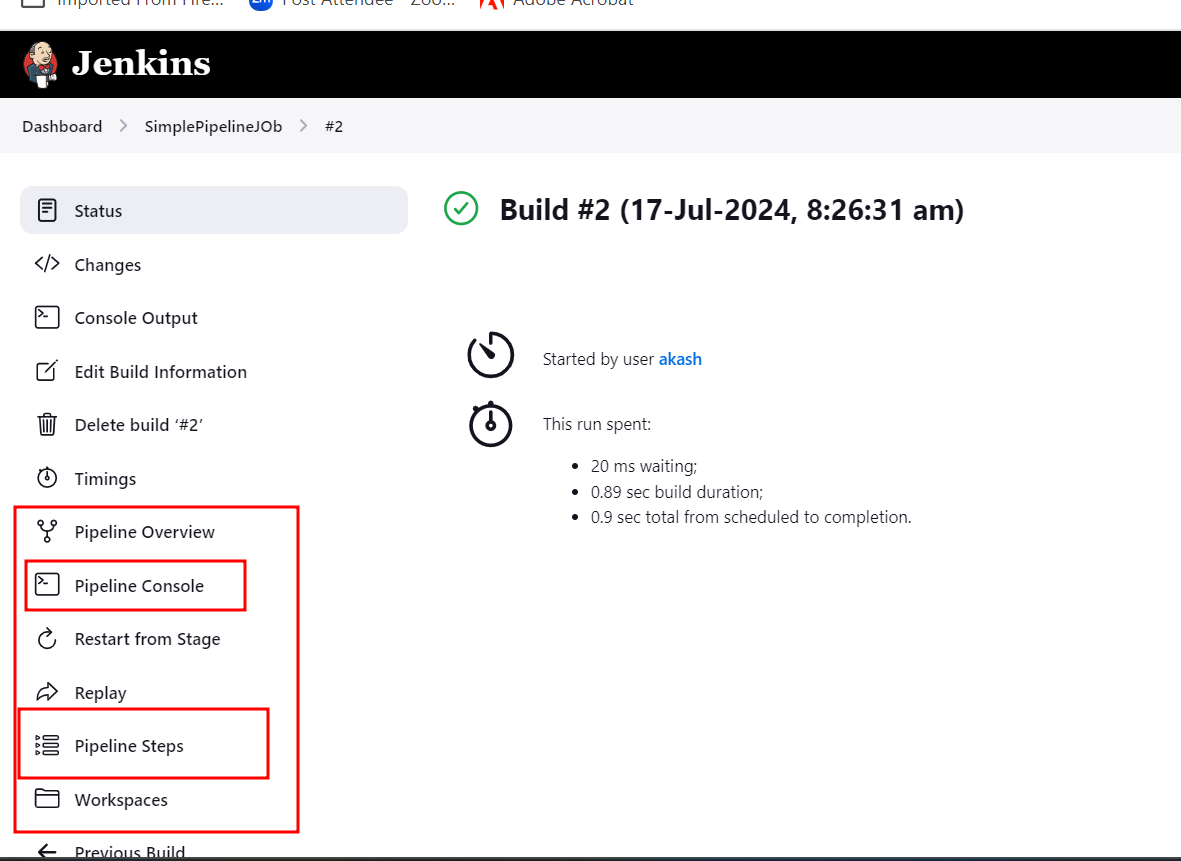


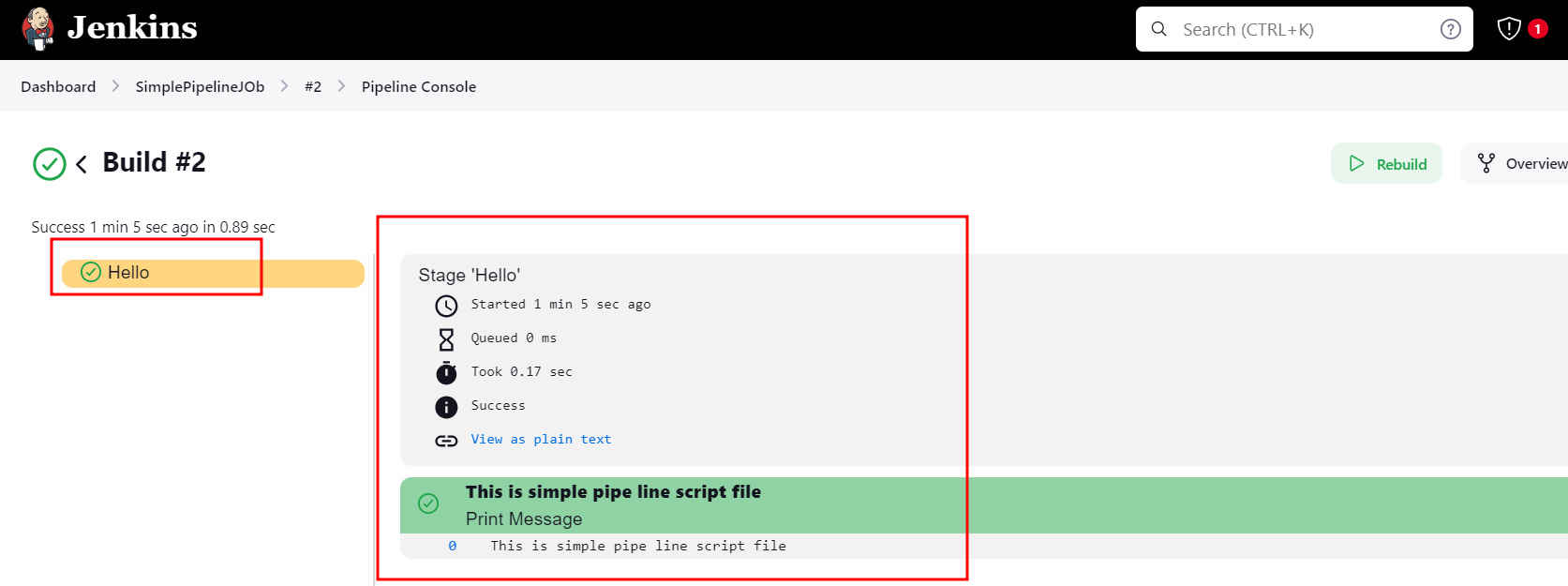


Apply and save then build it.



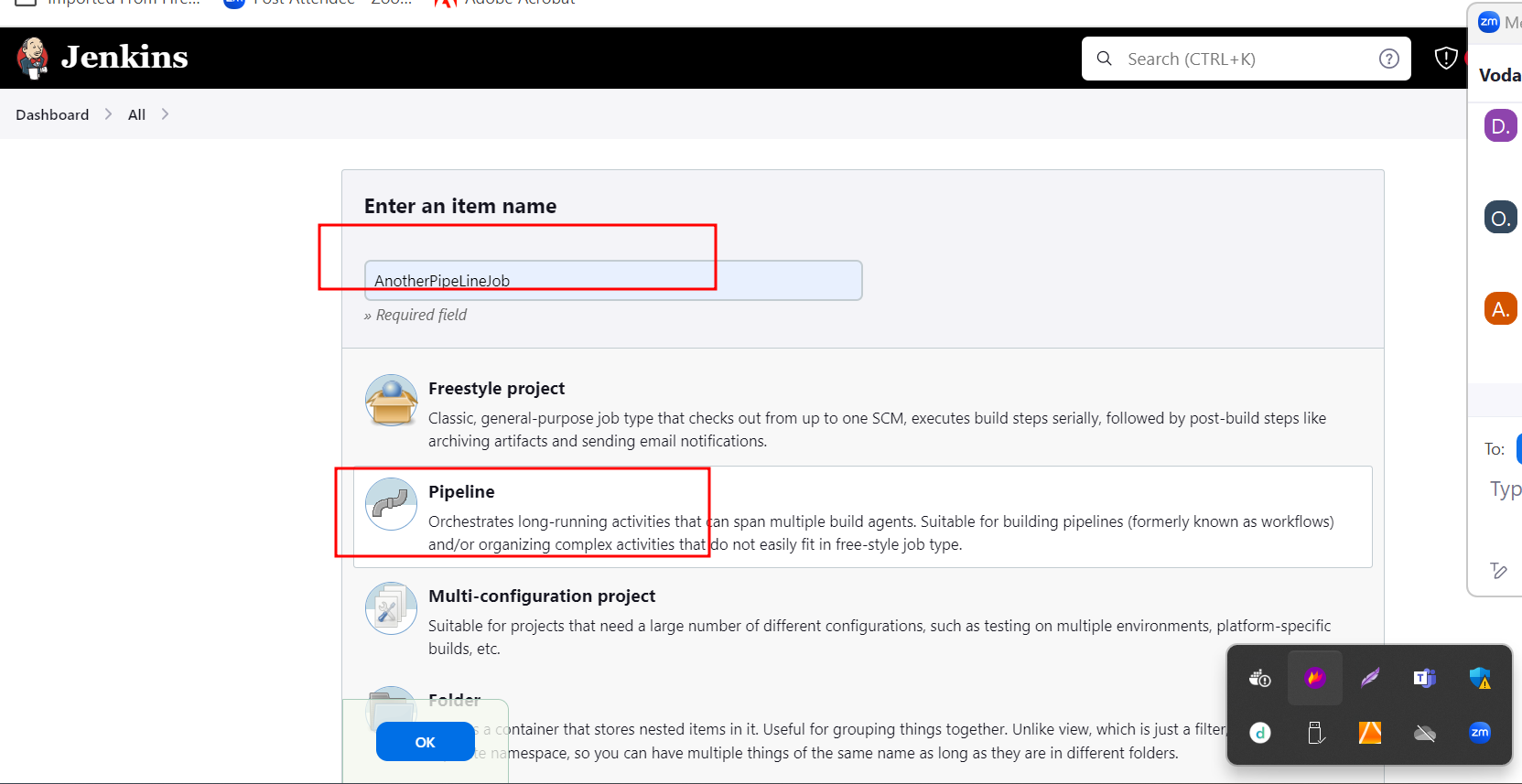








Another pipe line job with customer script file.



Sample script file

pipeline {

agent any

stages {

stage('Checking the version of all software') {

steps {

echo 'Check all version of software'

sh 'java --version'

}

}

stage('Compile all program') {

steps {

echo 'All program compiled'

}

}

stage('Creating jar or war or build file') {

steps {

echo 'creating jar or war files'

}

}

stage('Test the project') {

steps {

echo 'Testing the project'

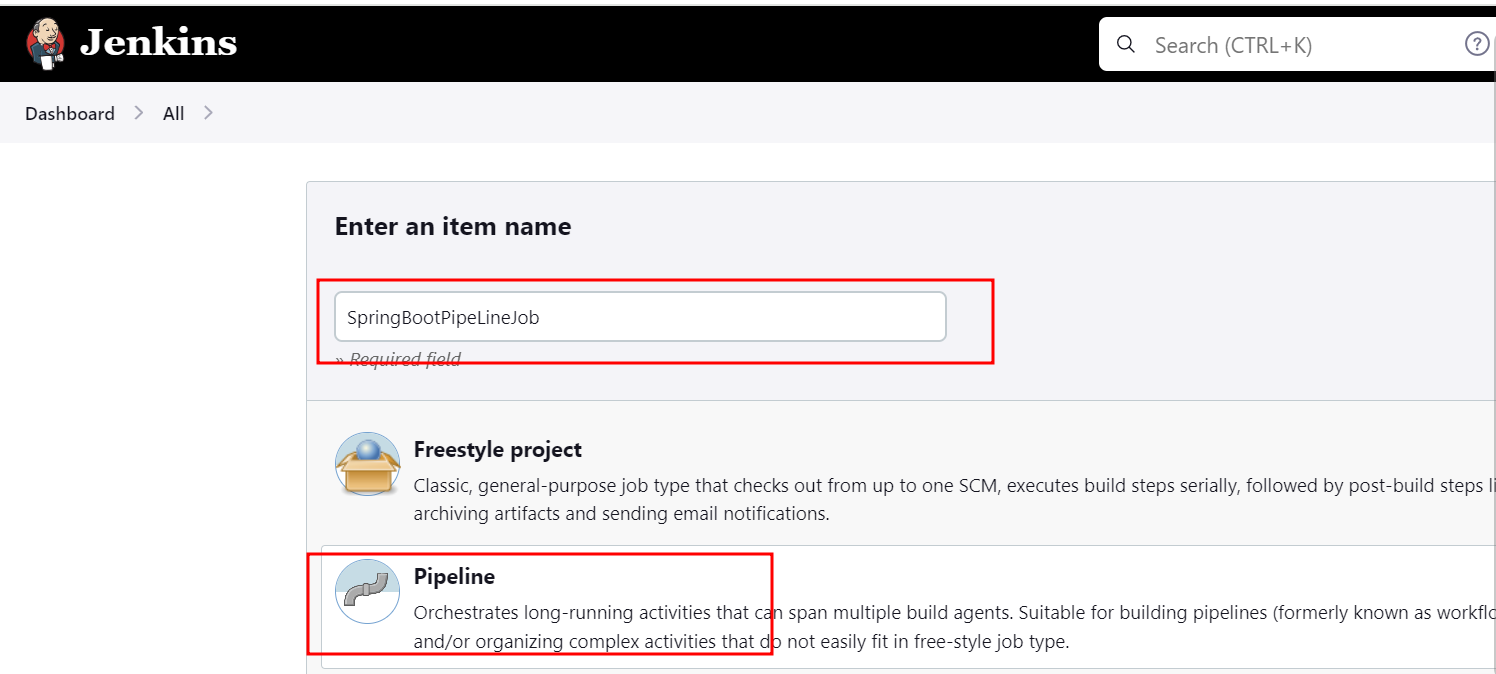
}

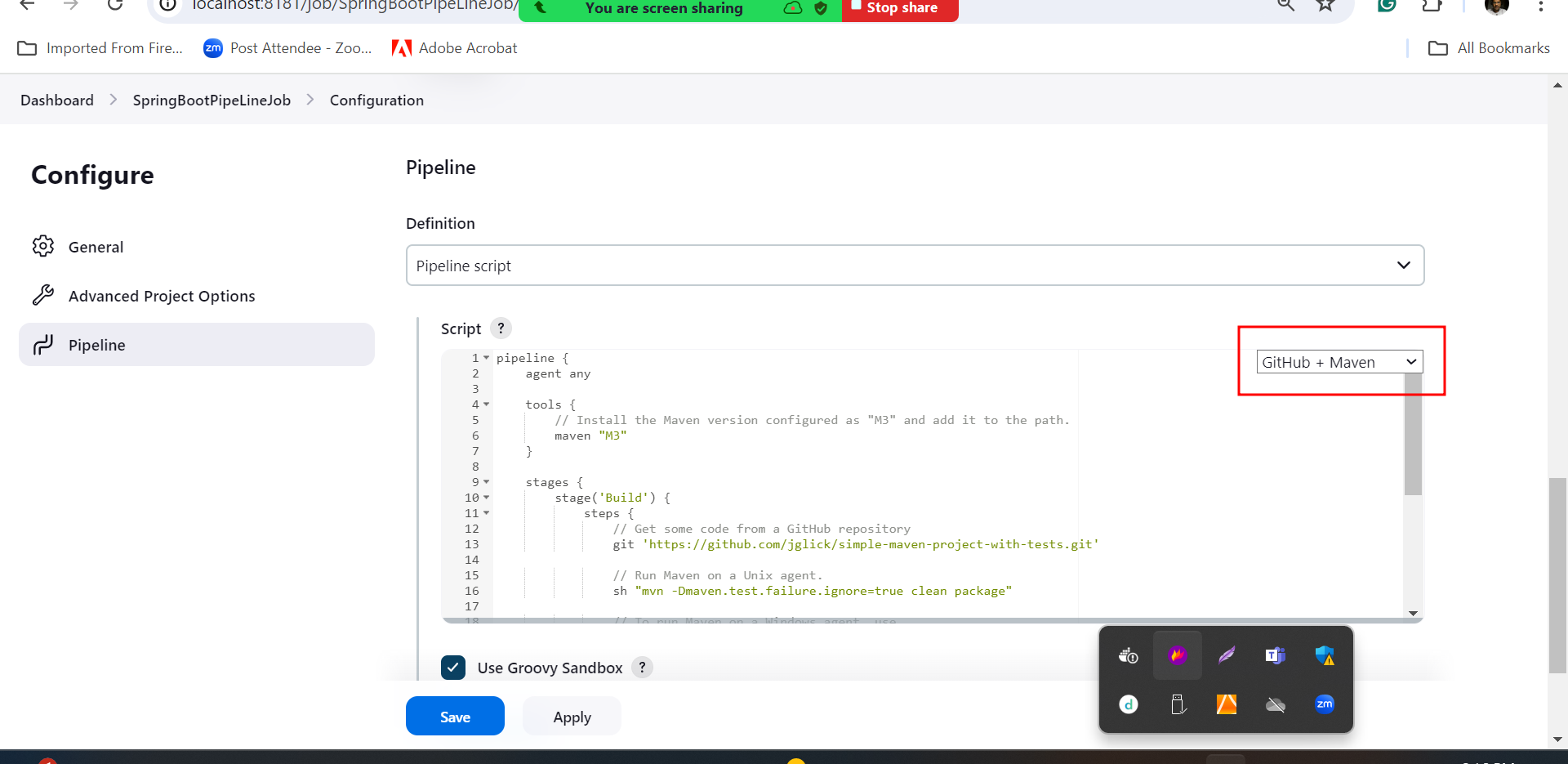
}

}

}

Creating Jenkin pipe line job responsible to run Maven project(Spring boot projects)





Sample script file responsible to build maven project if you want your custom maven spring boot project you can provide your project github URL.

pipeline {

agent any

tools {

// Install the Maven version configured as "M3" and add it to the path.

maven "M3"

}

stages {

stage('Build') {

steps {

// Get some code from a GitHub repository

git 'https://github.com/jglick/simple-maven-project-with-tests.git'

// Run Maven on a Unix agent.

sh "mvn -Dmaven.test.failure.ignore=true clean package"

// To run Maven on a Windows agent, use

//bat "mvn -Dmaven.test.failure.ignore=true clean package"

}

post {

// If Maven was able to run the tests, even if some of the test

// failed, record the test results and archive the jar file.

success {

junit '\*\*/target/surefire-reports/TEST-\*.xml'

archiveArtifacts 'target/\*.jar'

}

}

}

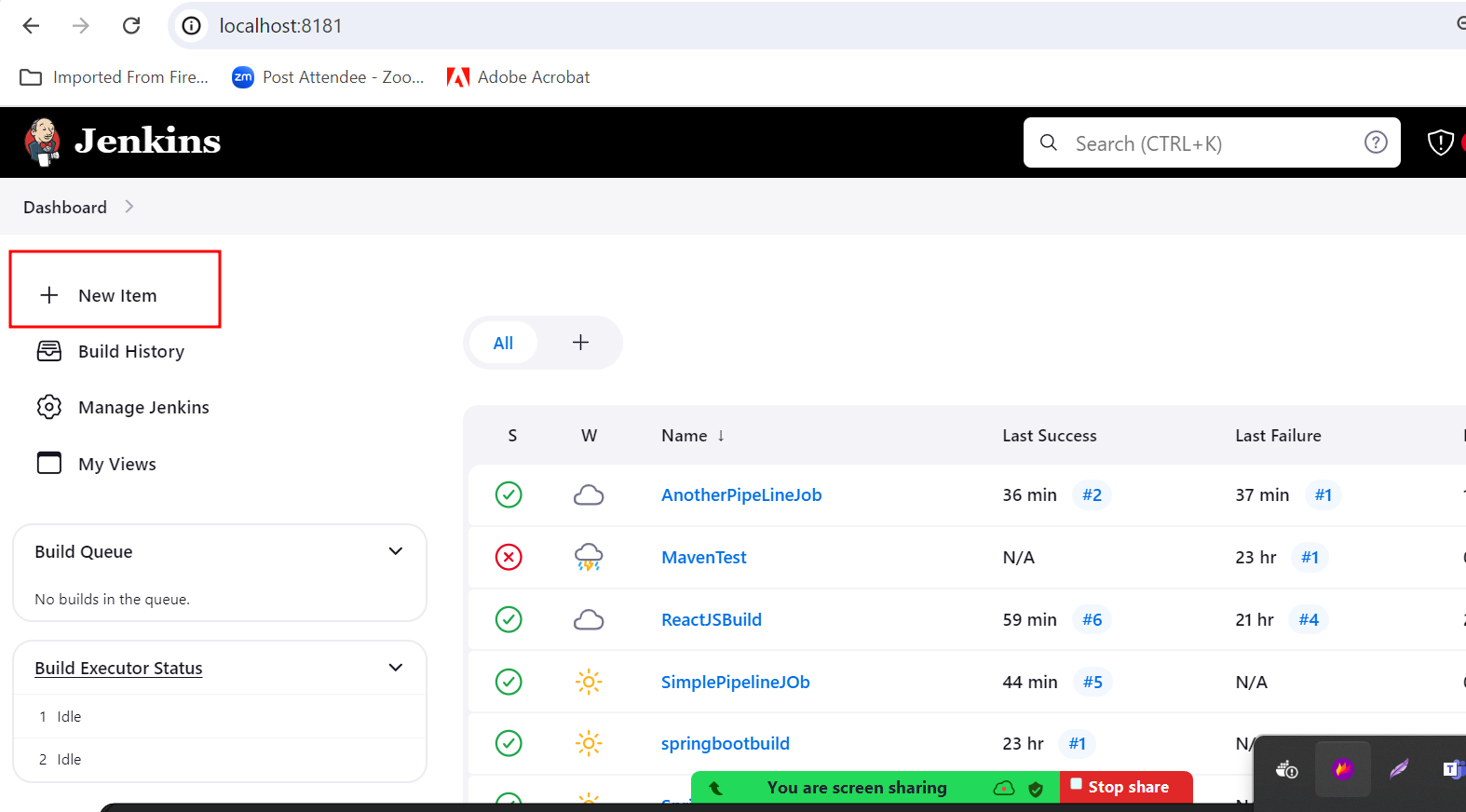
}

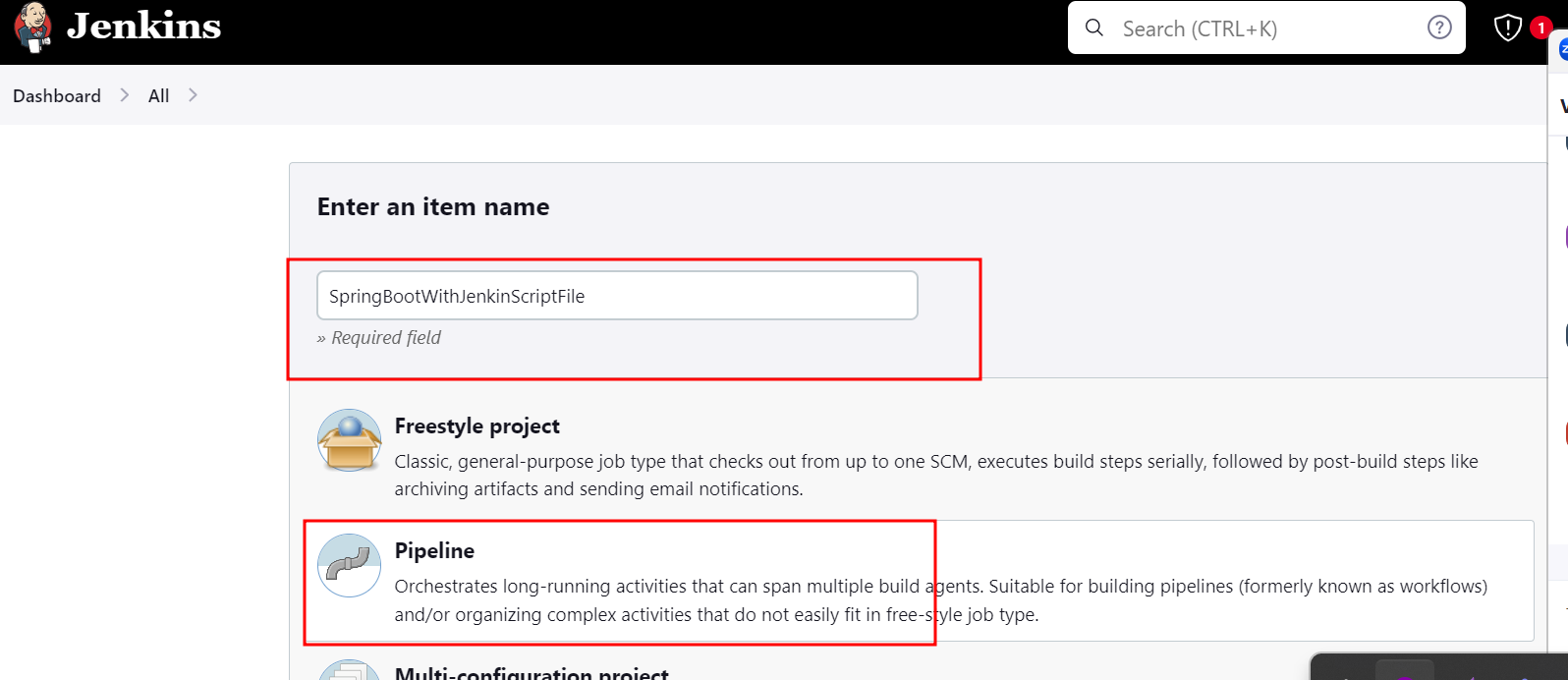
}

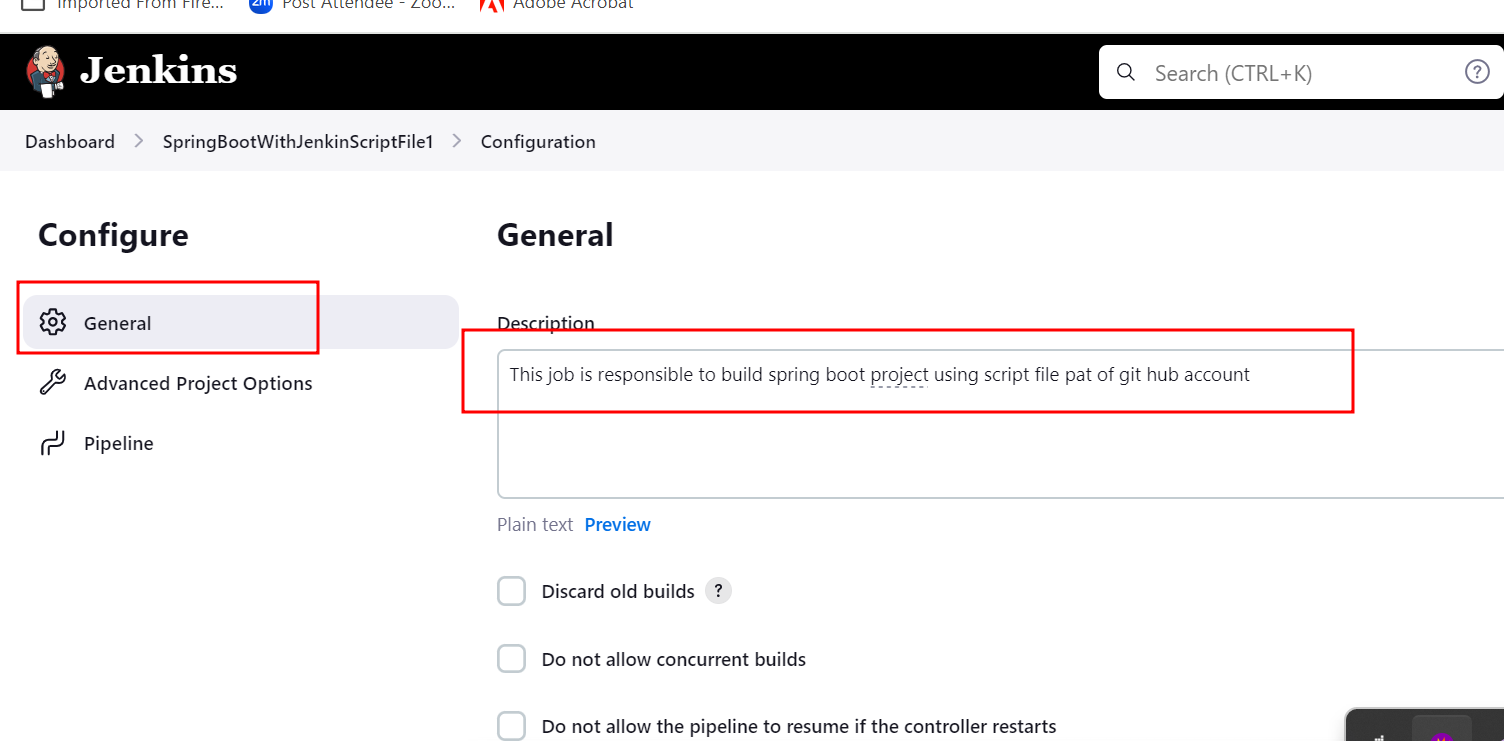
If we want to write complex script file then we need to create Jenkinsfile and write the script code.

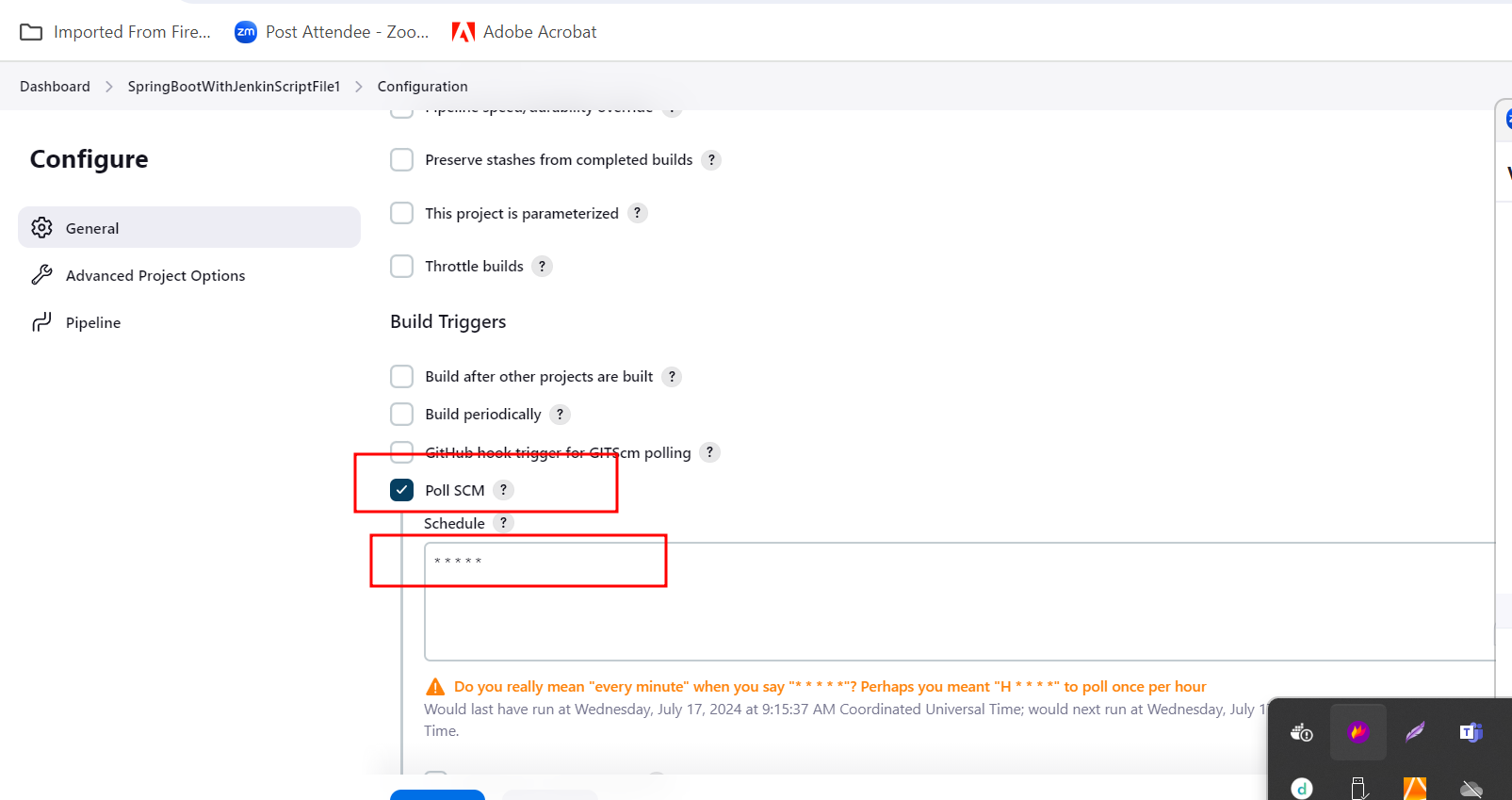
Creating another Jenkin pipeline job responsible to run Jenkinsfile script file part of git hub account.

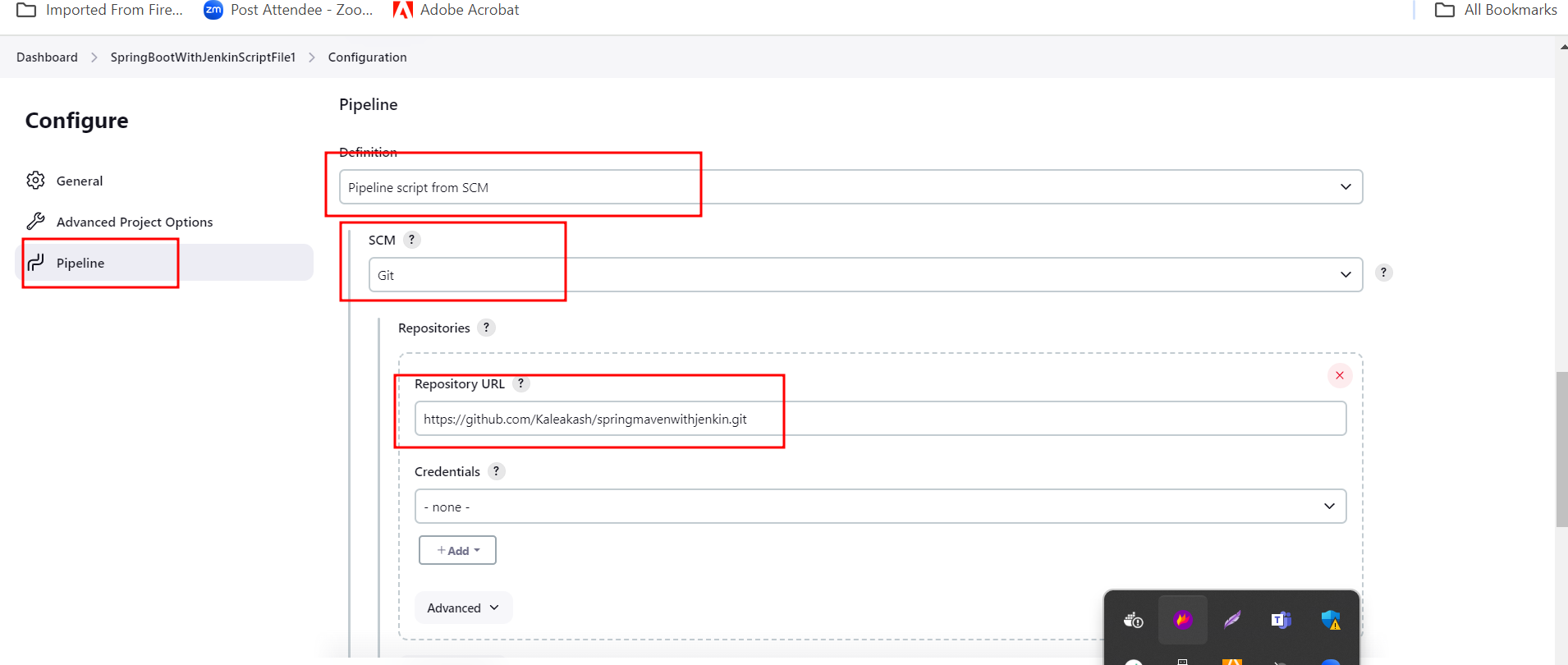
<https://github.com/Kaleakash/springmavenwithjenkin.git>

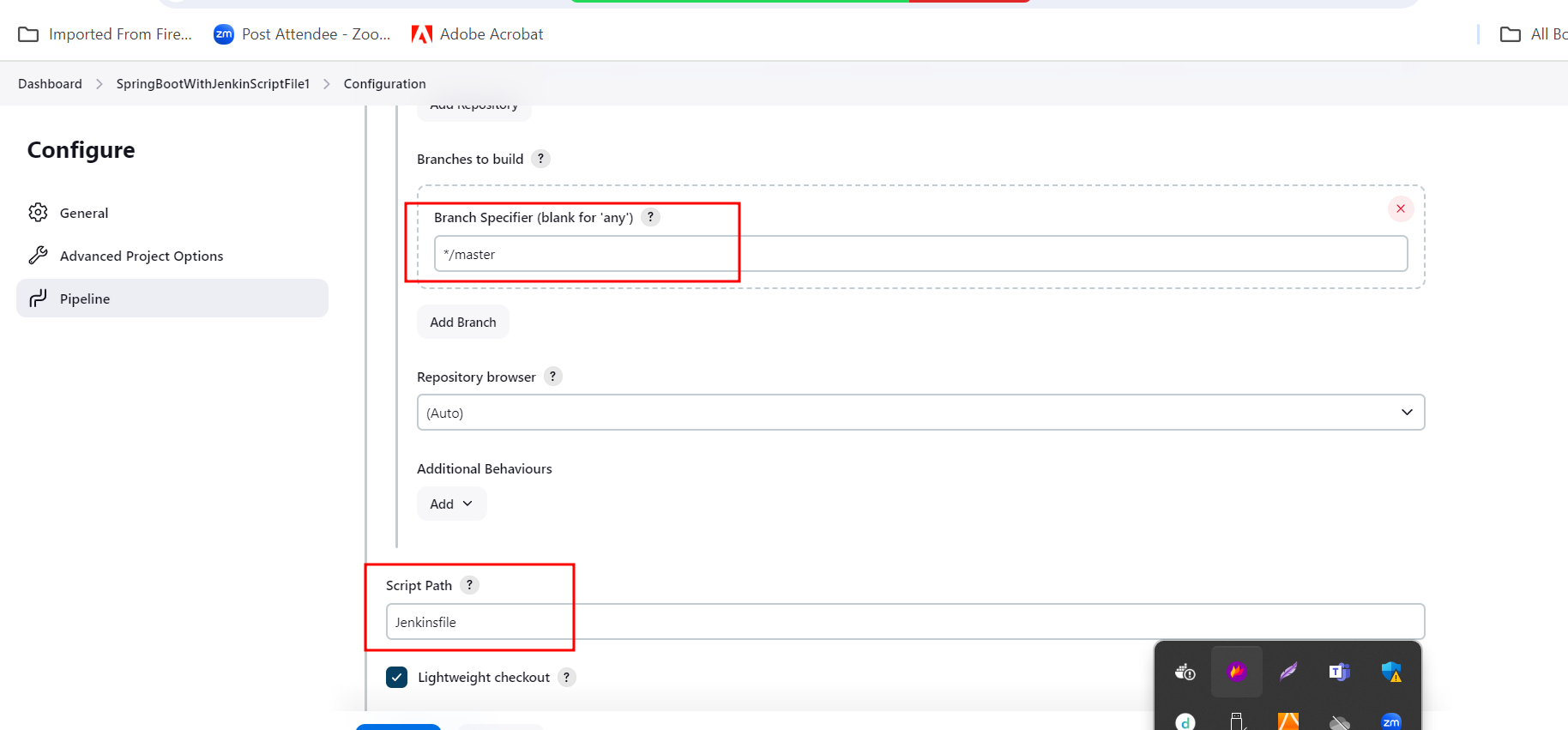




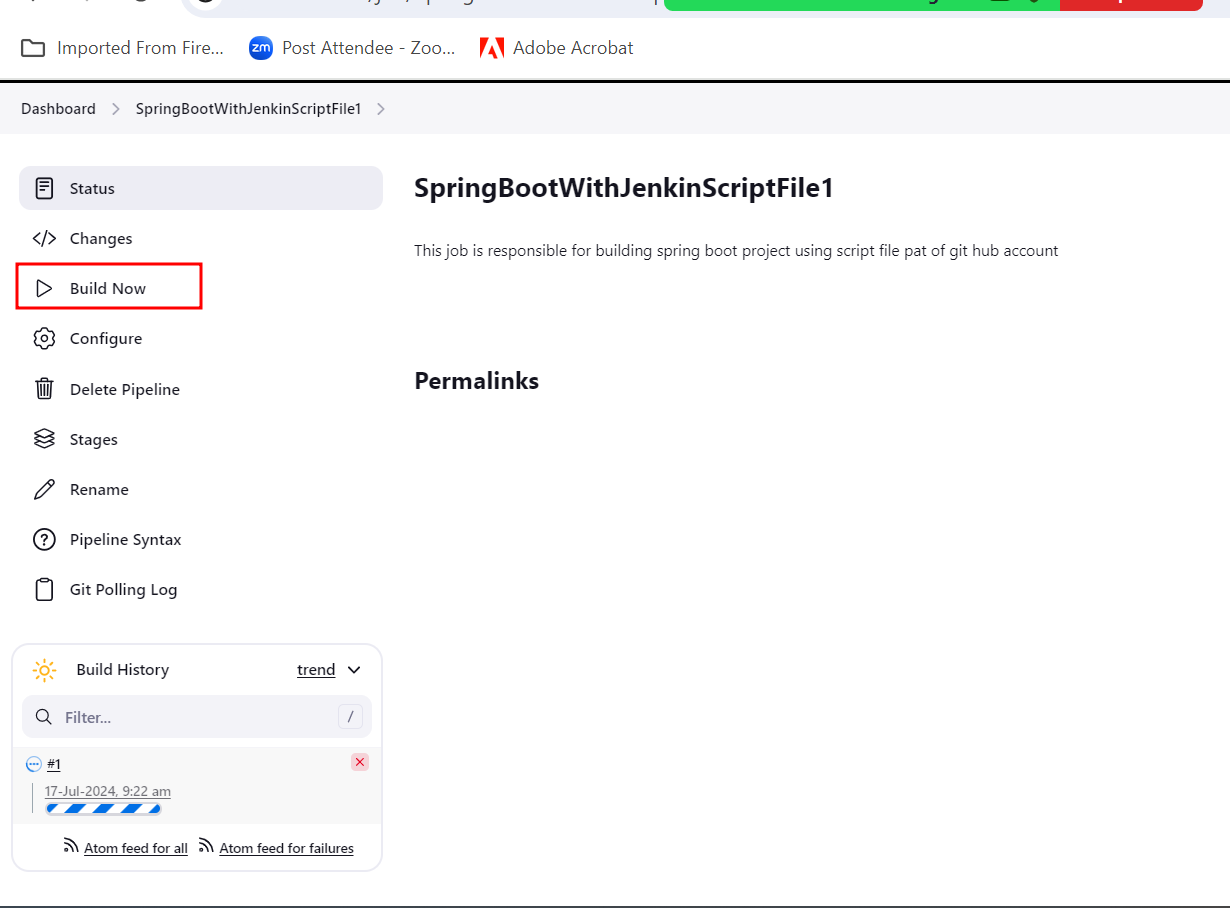








Apply and save



Cloud computing

Cloud refer to network. In place of installing all required software, tool, server, database there log of cloud provider available in marker they provide all those service base upon our requirement we need to use it and pay base upon our usages.

Types of Cloud

Public cloud : any type of user can create the account and they can use the resource provided that cloud provider. It may be free or paid resources. AWS or Azure etc.

Private cloud : this cloud maintain by organization. Only people part of that organization they can use these all services.

Hybrid cloud : combination of public and provide cloud modules.

Community cloud : this type cloud maintain by more than one organization. Generally open source technologies.

Node JS modules

Python module

Base upon type of service cloud provided they divided into three category.

1. IaaS : infrastructure as a Service

Hardware as well as Software part we need to customize.

1. PaaS : Platform as a Service

We need platform to deploy the application. Those platform can be open source and paid. We are going to deploy java, react, angular, python etc.

1. SaaS : Software as a Service

We need application those application develop using any language.

Cloud provider

AWS

Azure

Google cloud

Oracle cloud

AWS Overview :

Amazon Web Service

Working on Capstone Project

Backend project ie login-app using spring boot with mysql container running using Docker compose.

Open the terminal in the place where docker-compose file present and run the command as

docker-compose up --build -d

we create admin account using mysql docker container

first open the container using container id or container name

mysql -u root -p

password 🡪 root

it open mysql terminal

connect to db

use mdb;

**insert into login values(‘admin@gmail.com’,’admin@123’,’admin’);**

Now we create front-end application

This code please do inside frontend folder

npx create-react-app frontend-app

Or

npm install create-react-app

create-react-app frontend-app

cd frontend-app

npm install react-router-dom

npm install axios