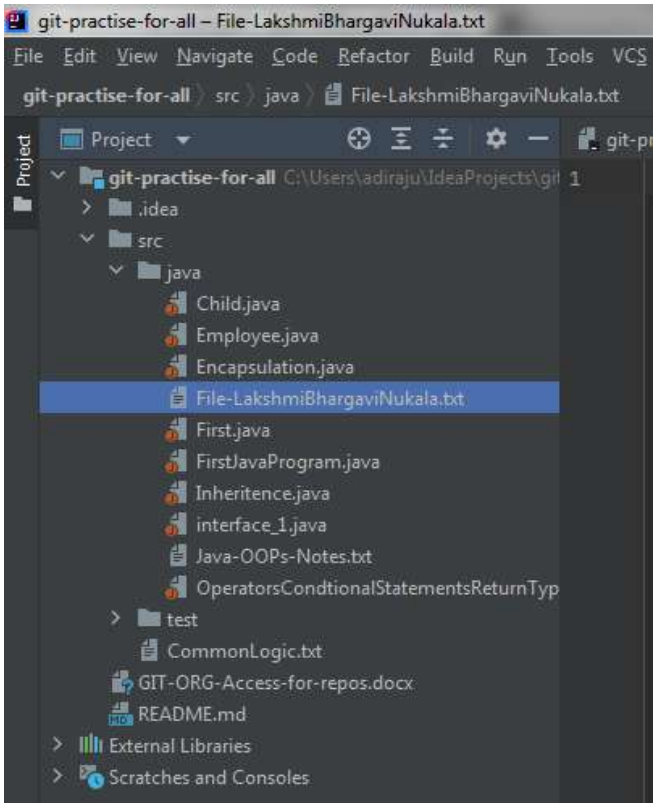
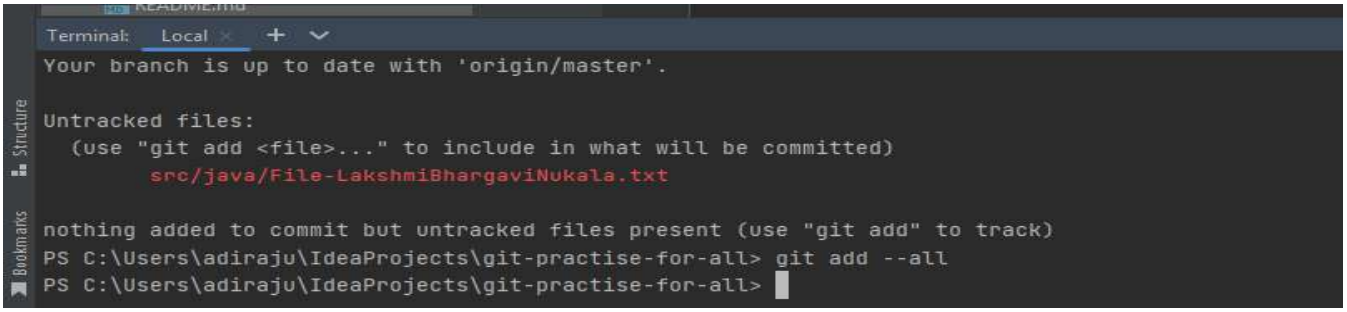
Name: Lakshmi Bhargavi Nukala
Email: bhargavi.aqua04@gmail.com
Reg No: 222037
Hall Ticket No: 22063CDO117

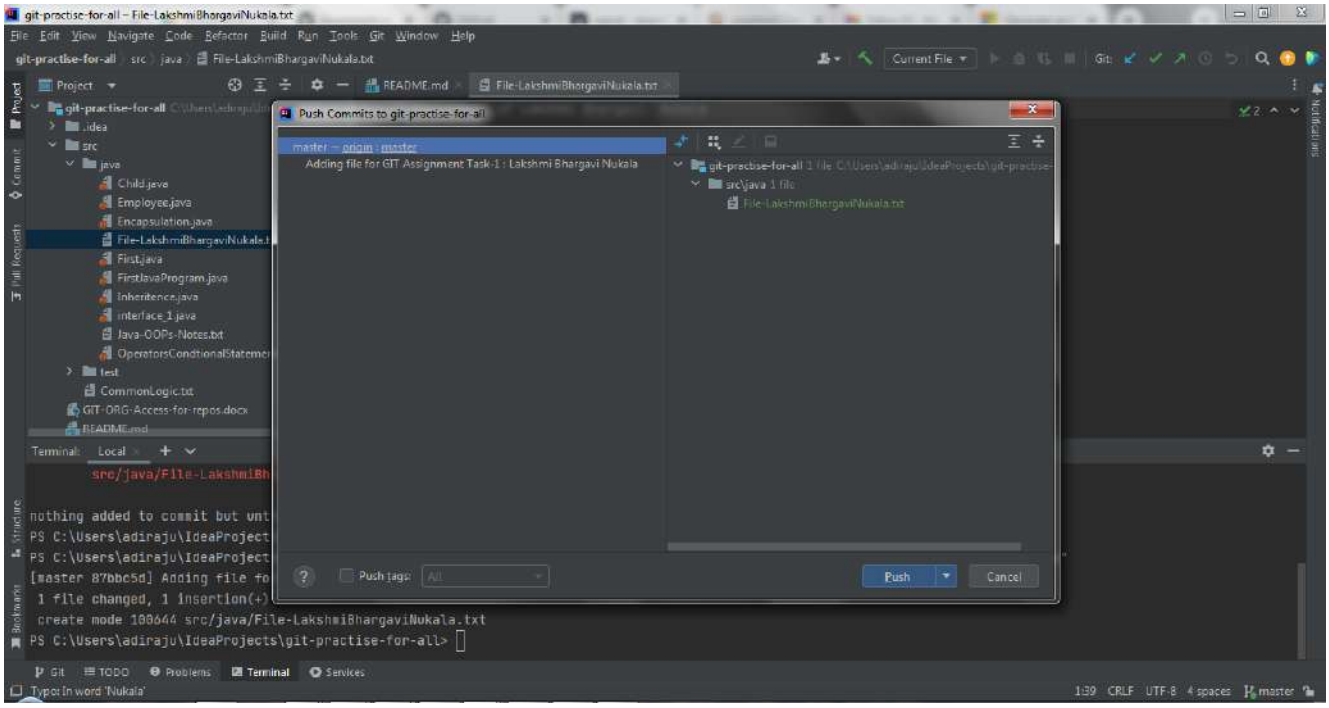
GIT:
Task-1
Add a new file with your name in IntelliJ

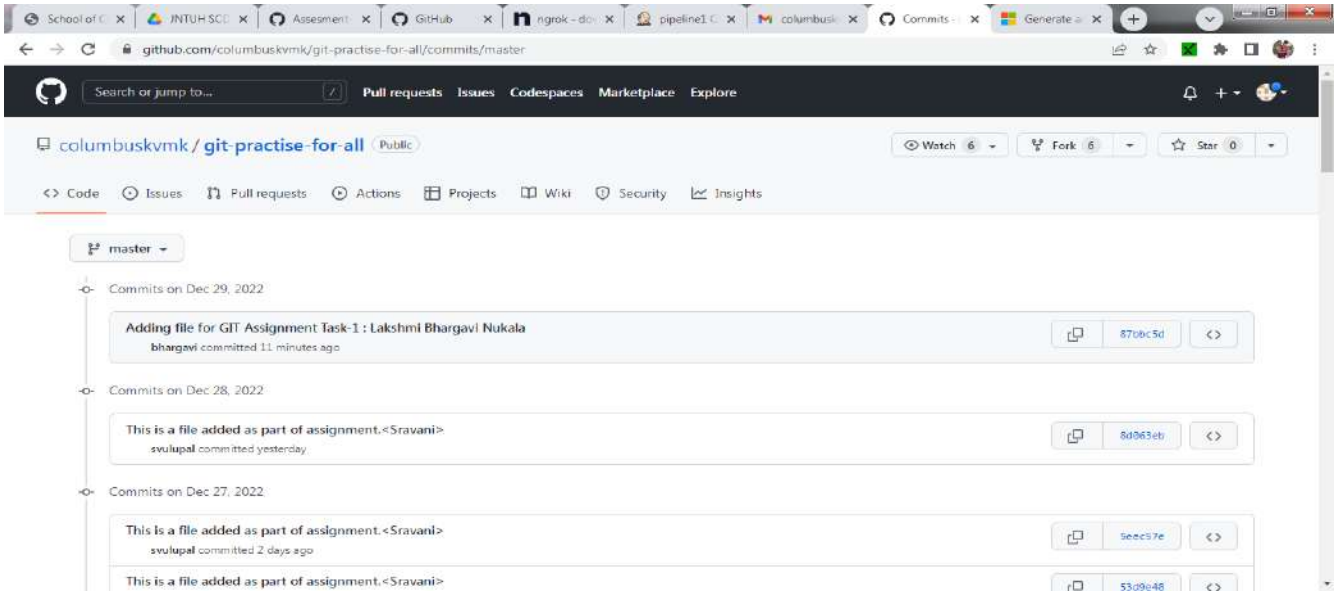


Add and Commit it to master branch



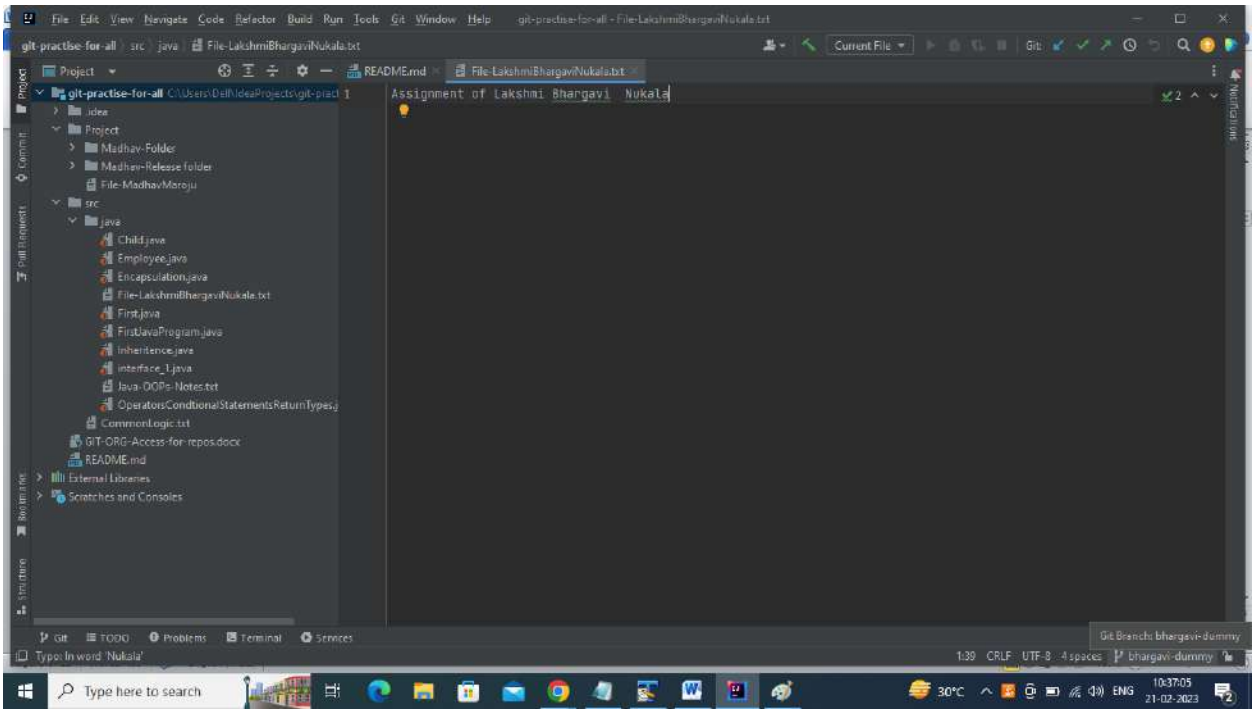
```
Terminal:  Local
        src/java/File-LakshmiBhargaviNukala.txt

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\adiraju\IdeaProjects\git-practise-for-all> git add --all
PS C:\Users\adiraju\IdeaProjects\git-practise-for-all> git commit -m "Adding file for GIT Assignment Task-1 : Lakshmi Bhargavi Nukala"
[master 87bbc5d] Adding file for GIT Assignment Task-1 : Lakshmi Bhargavi Nukala
 1 file changed, 1 insertion(+)
 create mode 100644 src/java/File-LakshmiBhargaviNukala.txt
PS C:\Users\adiraju\IdeaProjects\git-practise-for-all>
```

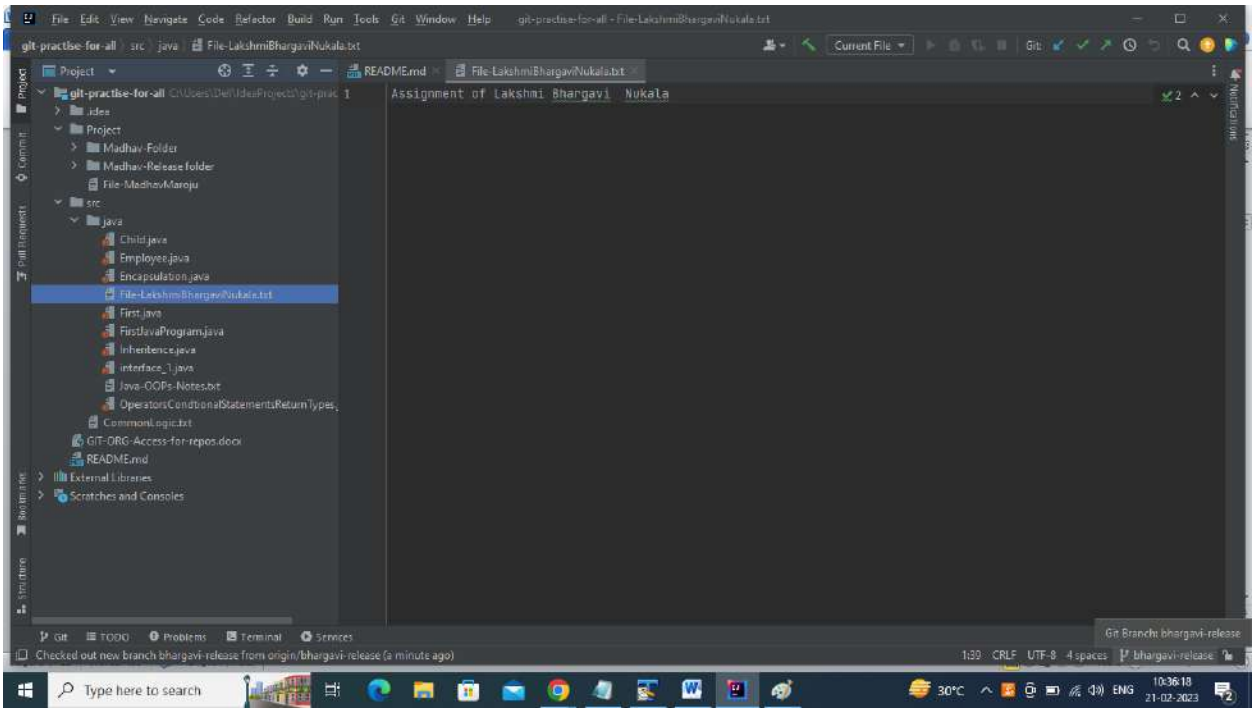




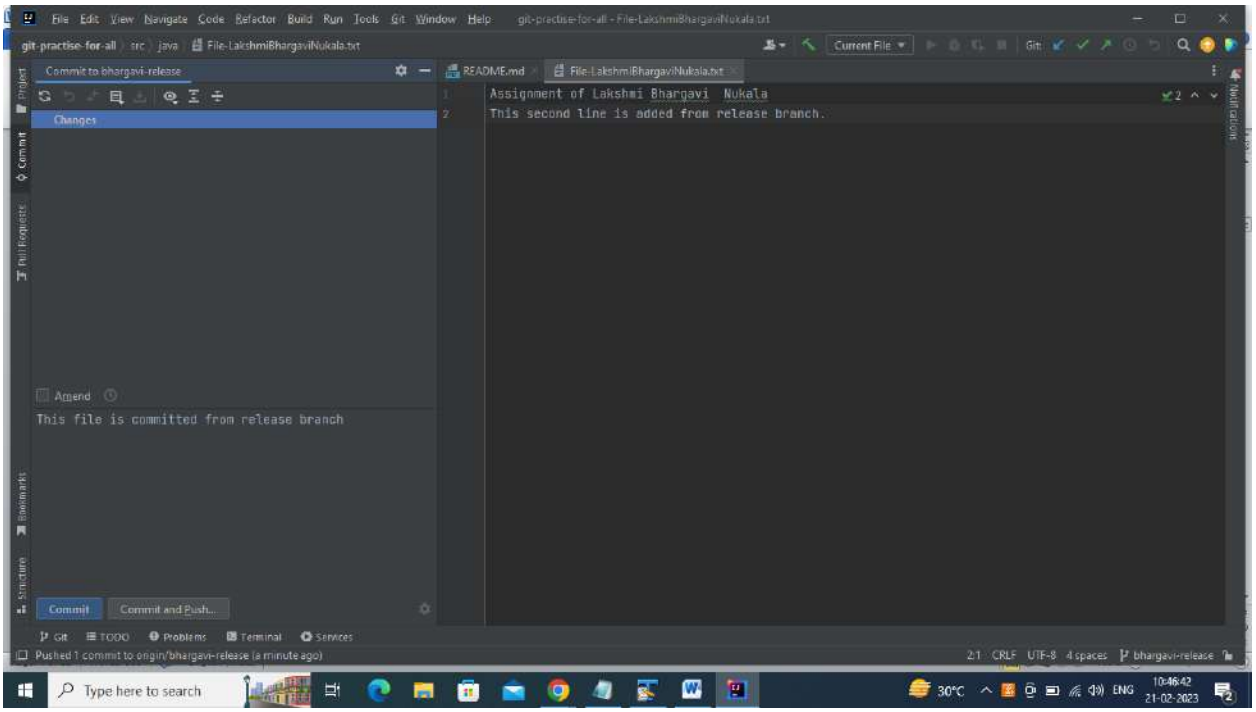
GIT Task-2:
Created two branches in GIT
bhargavi-dummy
bhargavi-release

In IntelliJ, check out the code from the branches using GIT Fetch and GIT Pull options.

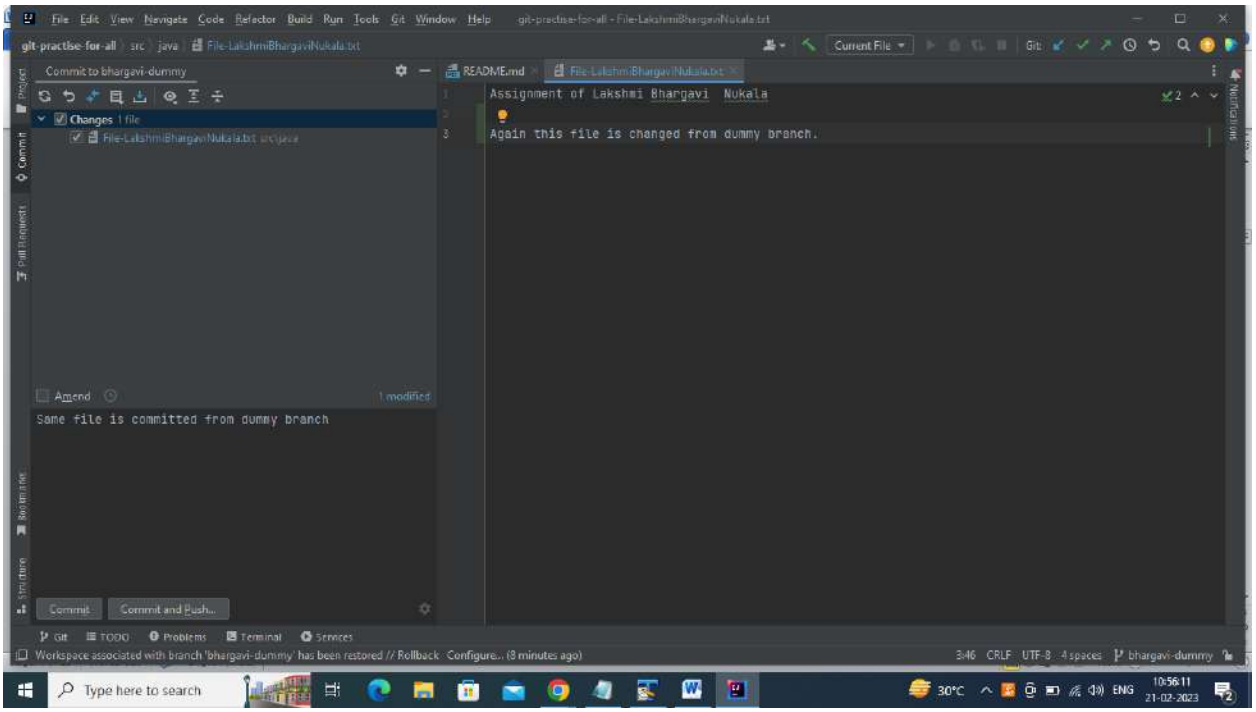




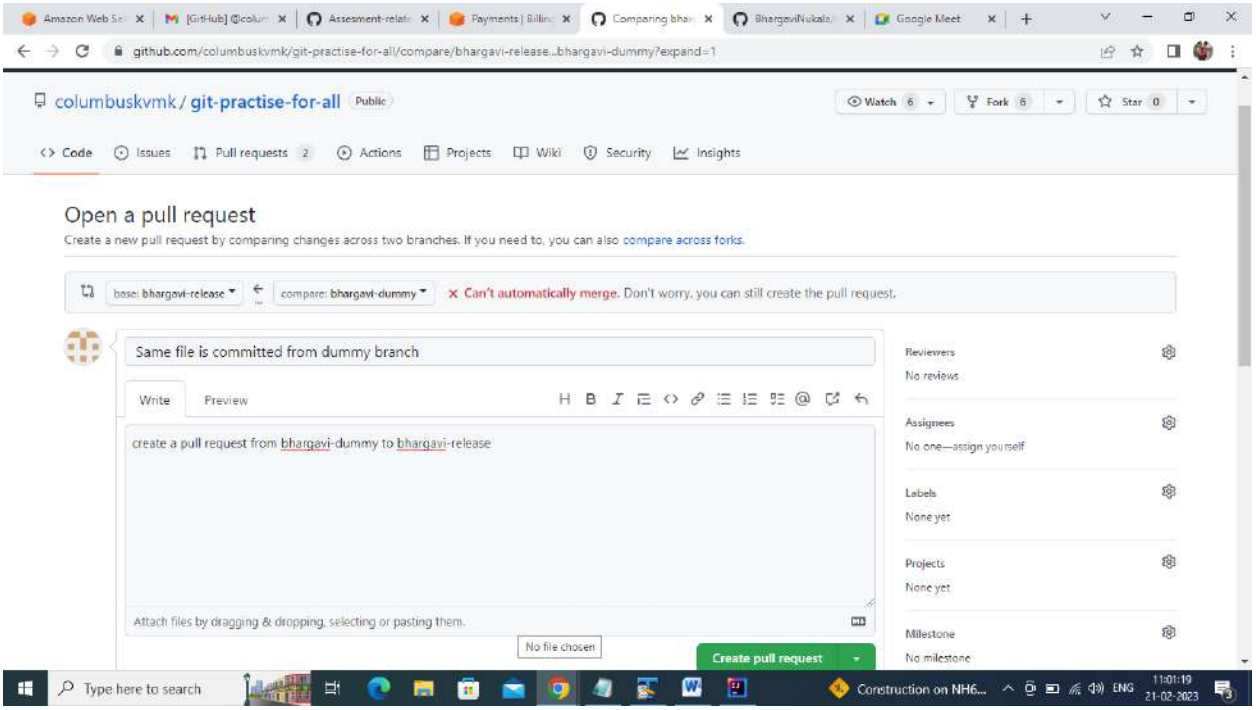
Change the file File-LakshmiBhargaviNukala.txt in bhargavi-release branch, commit and push.



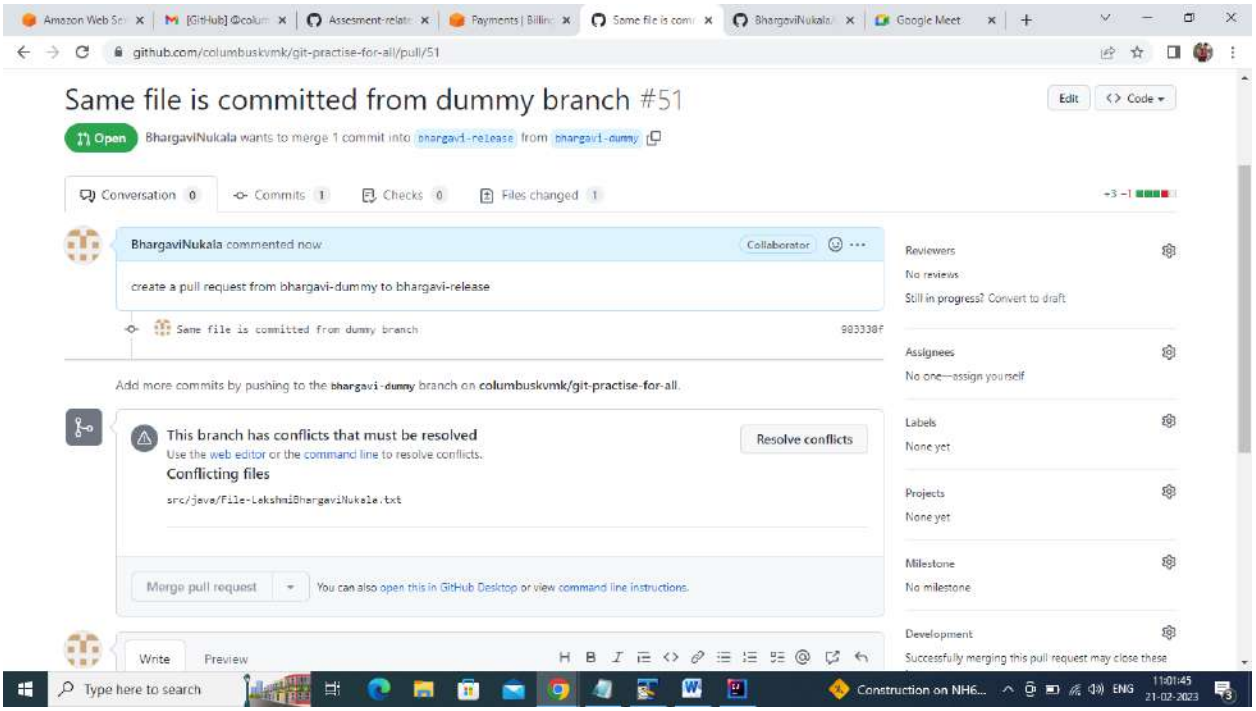
Now switch the branch to bhargavi-dummy and without pulling the code, change the same file and commit and push.

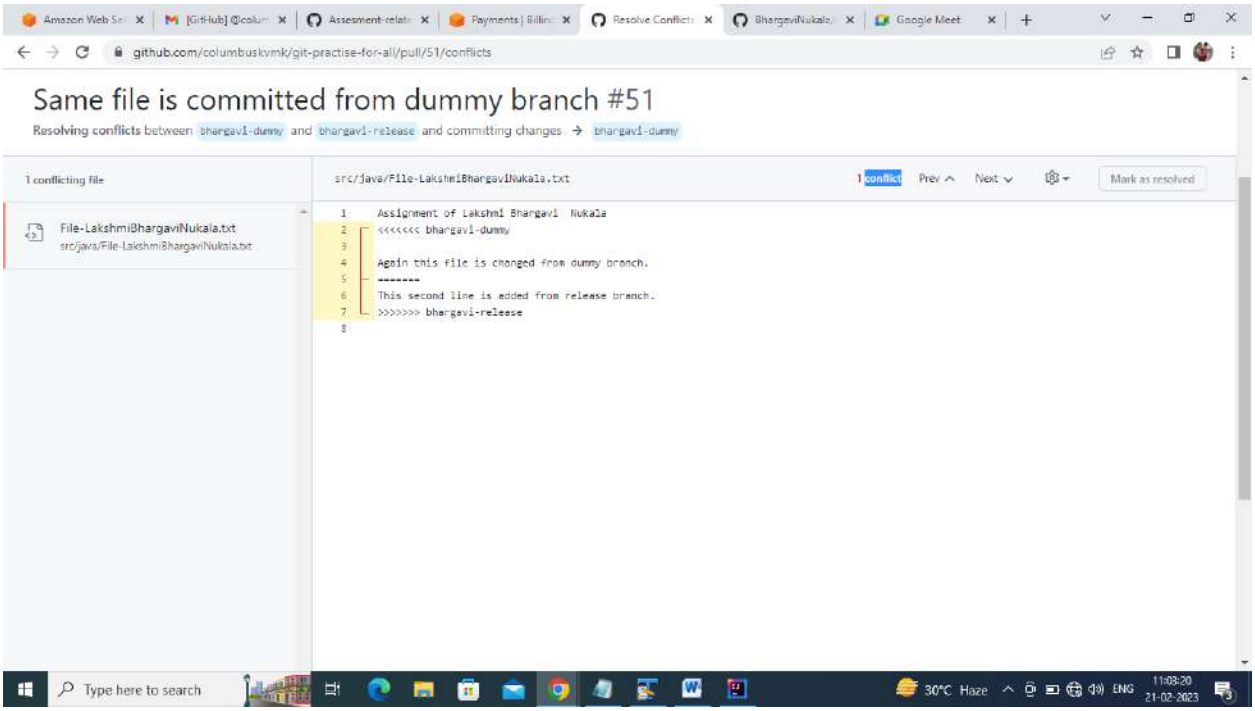


Create a Pull Request (PR) from bhargavi-dummy branch to bhargavi-release branch.

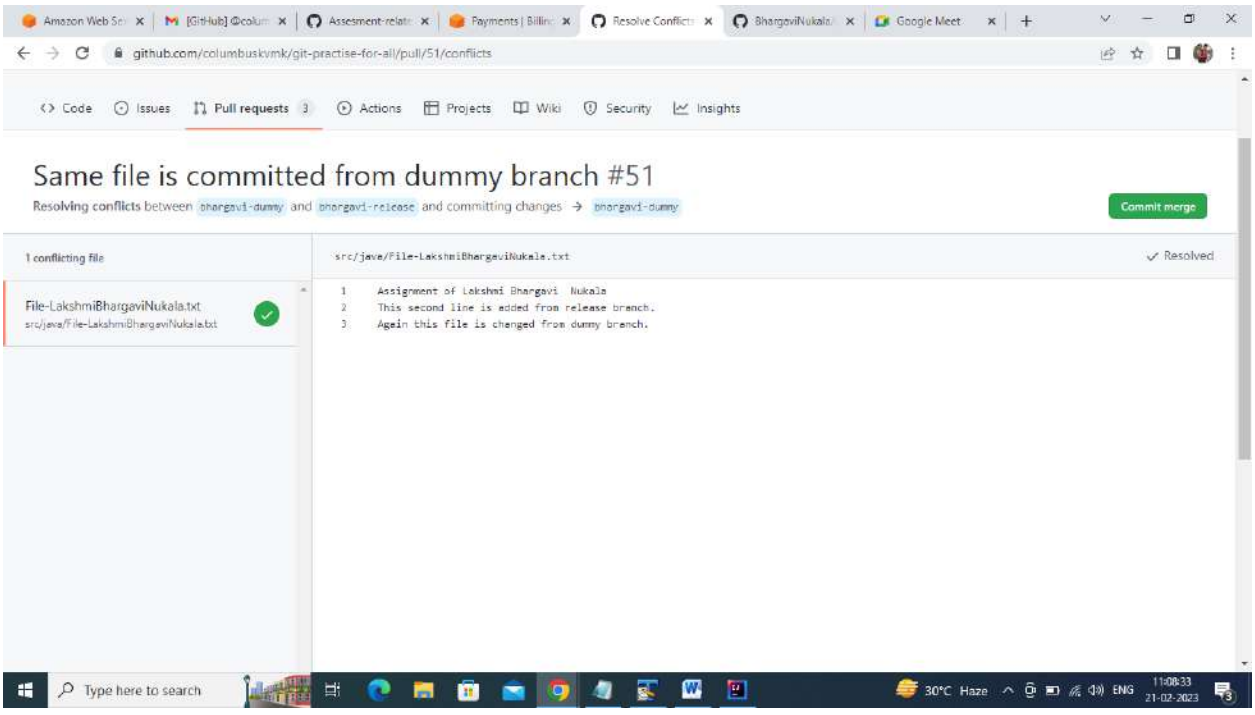


A merge request to reslove conflicts is displayed.

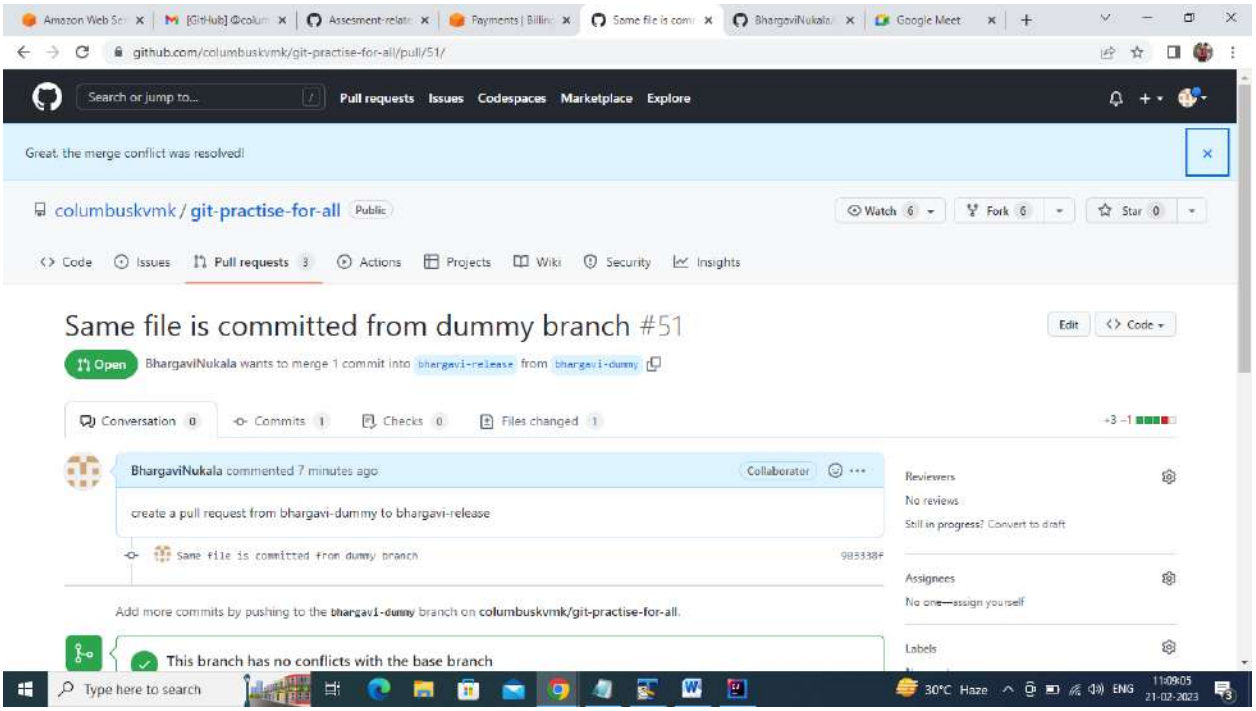




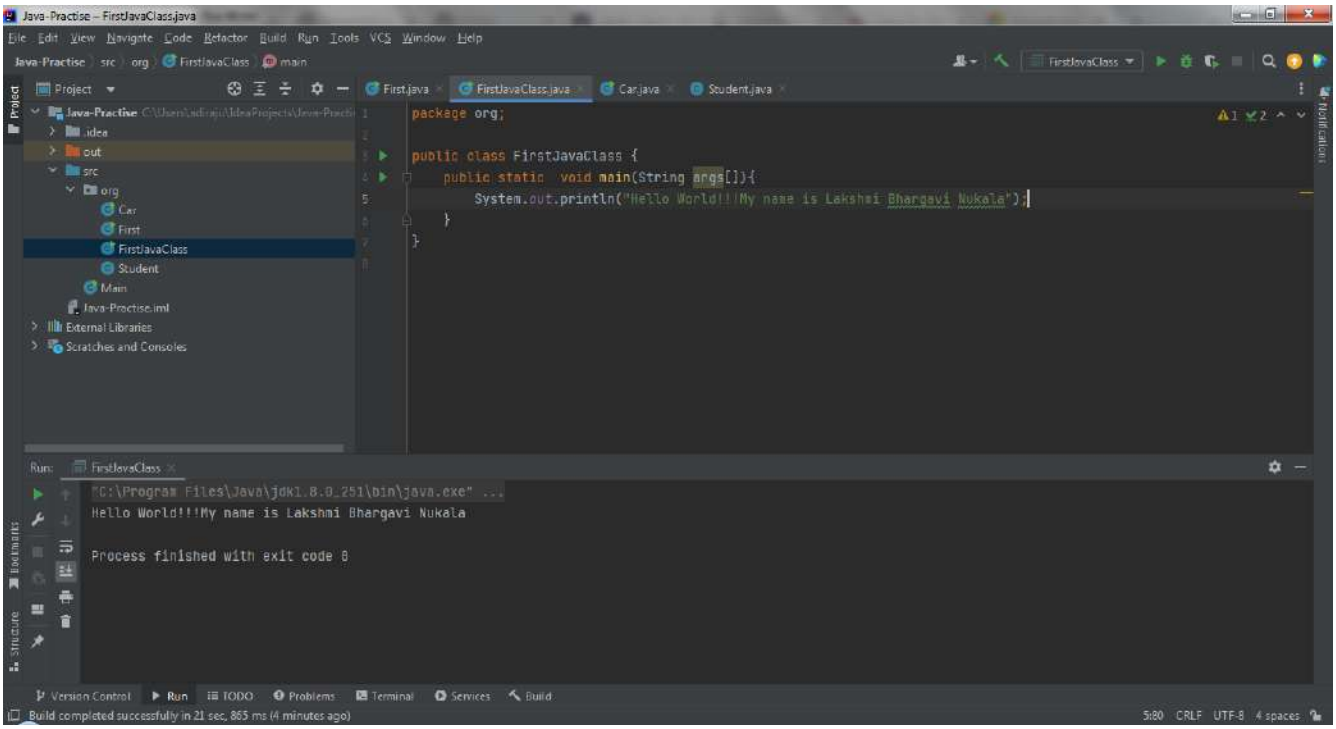
Resolve conflicts and commit merge



Merge Conflicts are resolved.



Java Task 1:
Run a simple Java application that prints our name

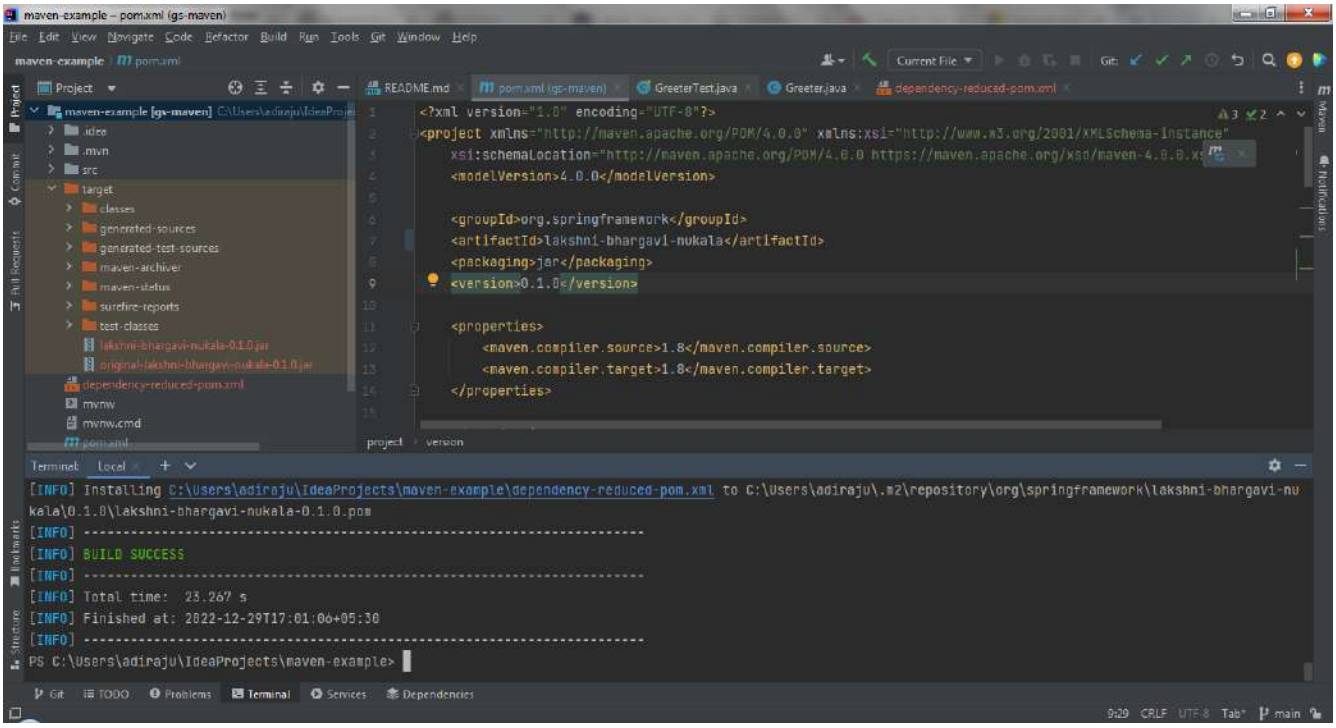


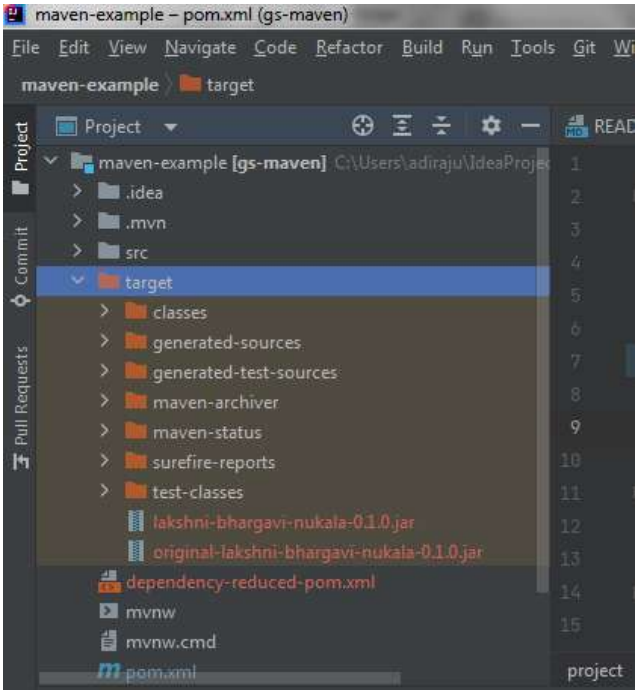
Task-2:
Byte code is in the .class file created and captured it

```
//
// Source code recreated from a .class file by IntelliJ IDEA
// (powered by FernFlower decompiler)
//
package org;
public class FirstJavaClass {
        public FirstJavaClass() {
        }
        public static void main(String[] args) {
                System.out.println("Hello World!!!My name is Lakshmi Bhargavi Nukala");
        }
}
```

Maven:
Create a maven project with artifact id as your name and generate jar/war file
In pom.xml file, change the artifact-id to your name and run the command
mvn clean install

```
<groupId>org.springframework</groupId>
<artifactId>lakshni-bhargavi-nukala</artifactId>
<packaging>jar</packaging>
<version>0.1.0</version>
```



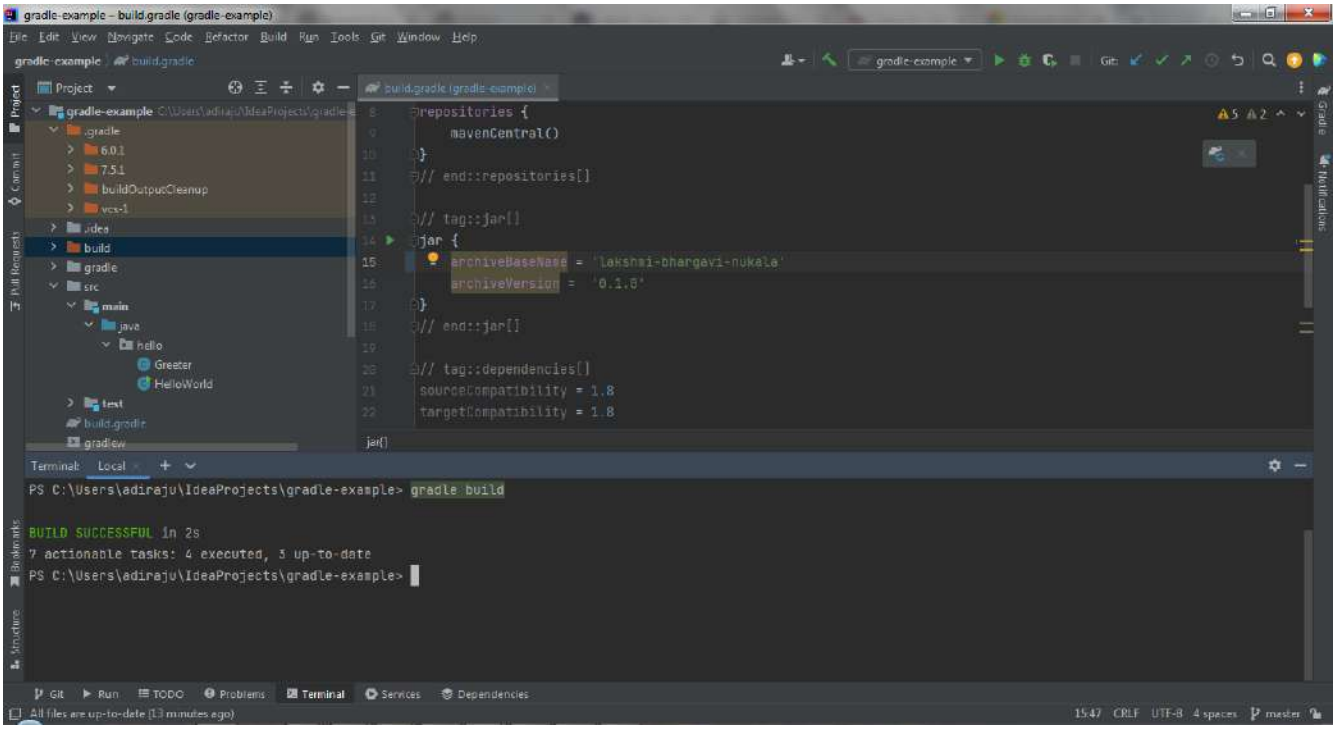


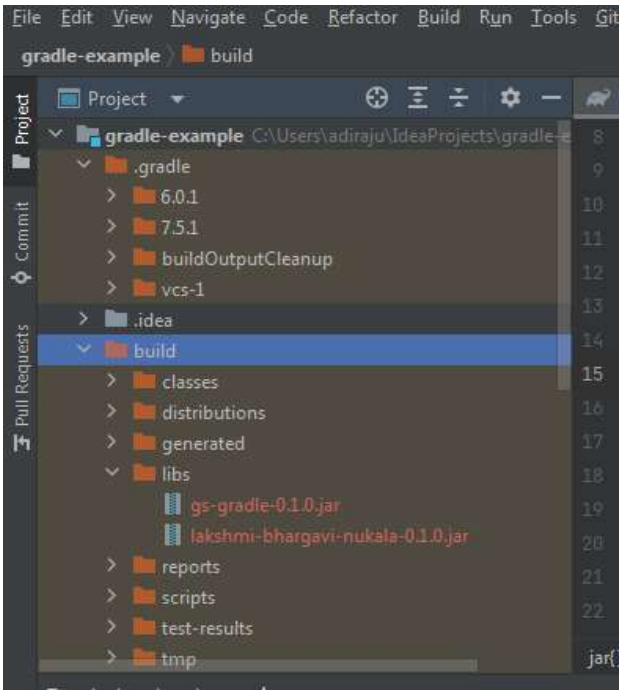
Gradle:

Create a gradle project with artifact-id as your name and generate jar/war artifact.
In build.gradle file, change the artifact-id name

```
// tag::jar[]
jar {
        archiveBaseName = 'lakshmi-bhargavi-nukala'
        archiveVersion =  '0.1.0'
}
// end::jar[]
```

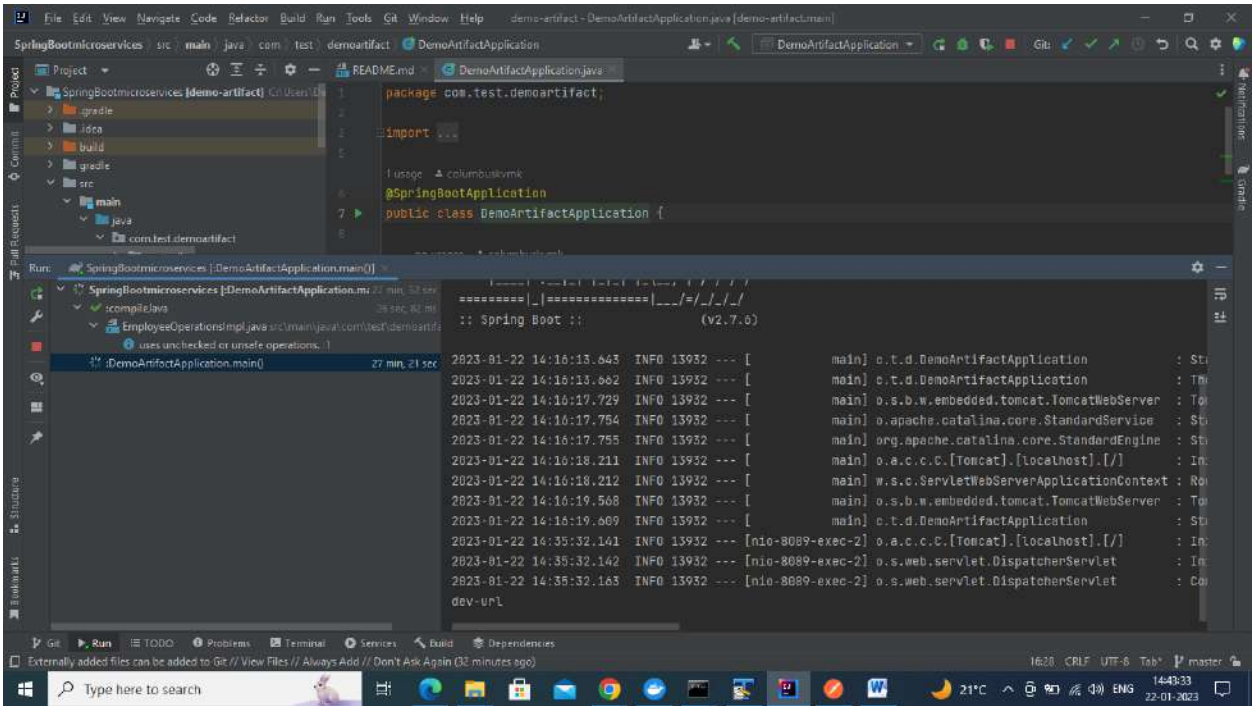
Build gradle project using the command
gradle build


gradle-example – build.gradle (gradle-example)
File Edit View Navigate Code Refactor Build Run Tools Git Window Help
gradle-example
build.gradle
Project
gradle-example C:\Users\adiraju\IdeaProjects\gradle-e
.gradle
6.0.1
7.5.1
buildOutputCleanup
vcs-1
.idea
build
gradle
src
main
java
hello
Greeter
HelloWorld
test
build.gradle
gradlew
build.gradle (gradle-example)
repositories {
mavenCentral()
}
// end::repositories[]
// tag::jar[]
jar {
archiveBaseName = 'lakshmi-bhargavi-nukala'
archiveVersion = '0.1.0'
}
// end::jar[]
// tag::dependencies[]
sourceCompatibility = 1.8
targetCompatibility = 1.8
jar{}
Terminal: Local
PS C:\Users\adiraju\IdeaProjects\gradle-example> gradle build
BUILD SUCCESSFUL in 2s
7 actionable tasks: 4 executed, 3 up-to-date
PS C:\Users\adiraju\IdeaProjects\gradle-example>
Git Run TODO Problems Terminal Services Dependencies
All files are up-to-date (13 minutes ago)
15:47 CRLF UTF-8 4 spaces master


File Edit View Navigate Code Refactor Build Run Tools Git
gradle-example
build
Project
gradle-example C:\Users\adiraju\IdeaProjects\gradle-e
.gradle
6.0.1
7.5.1
buildOutputCleanup
vcs-1
.idea
build
classes
distributions
generated
libs
gs-gradle-0.1.0.jar
lakshmi-bhargavi-nukala-0.1.0.jar
reports
scripts
test-results
tmp
Commit
Pull Requests

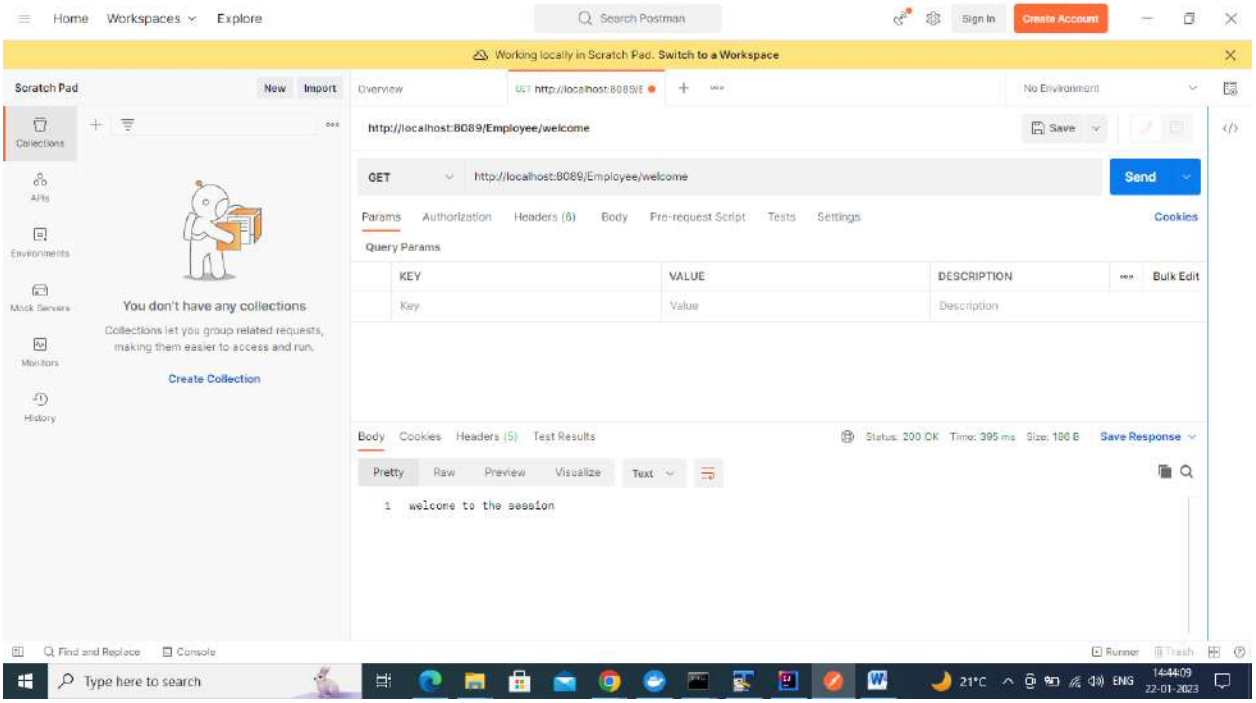
SpringBoot microservices:

Task 1: Run the SpringBoot in the repo provided

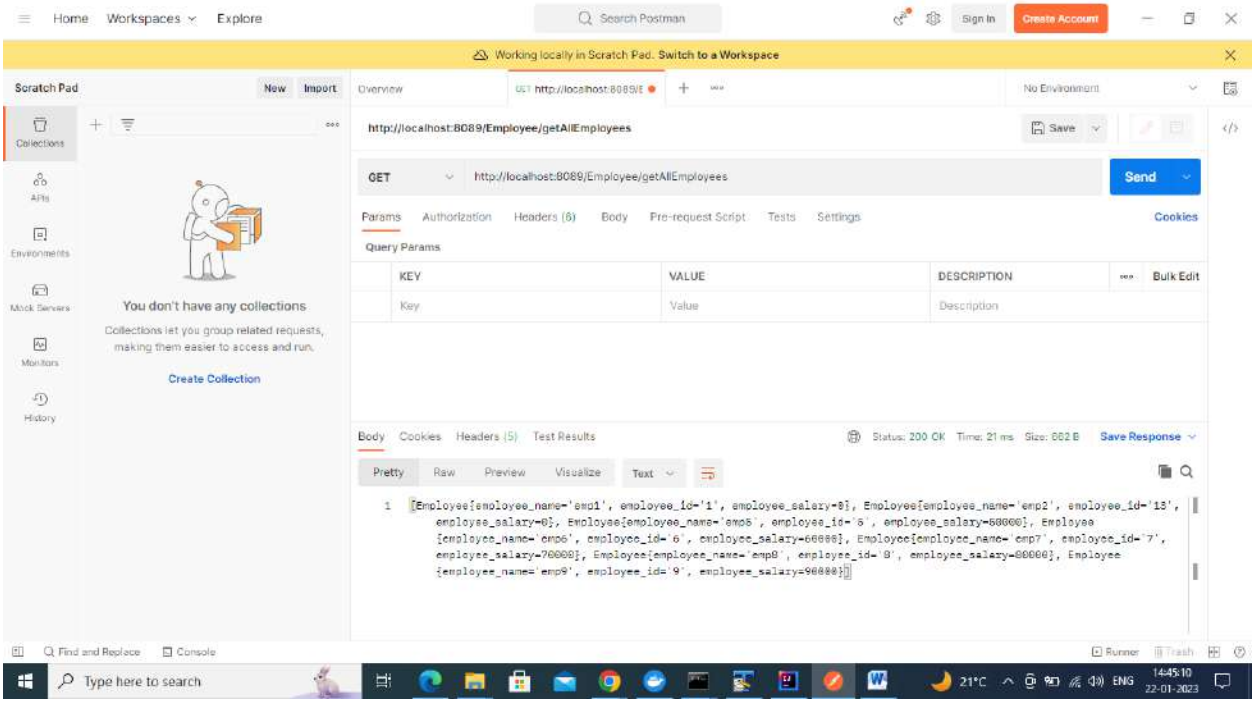


Task 2: Verify the following Urls on Postman

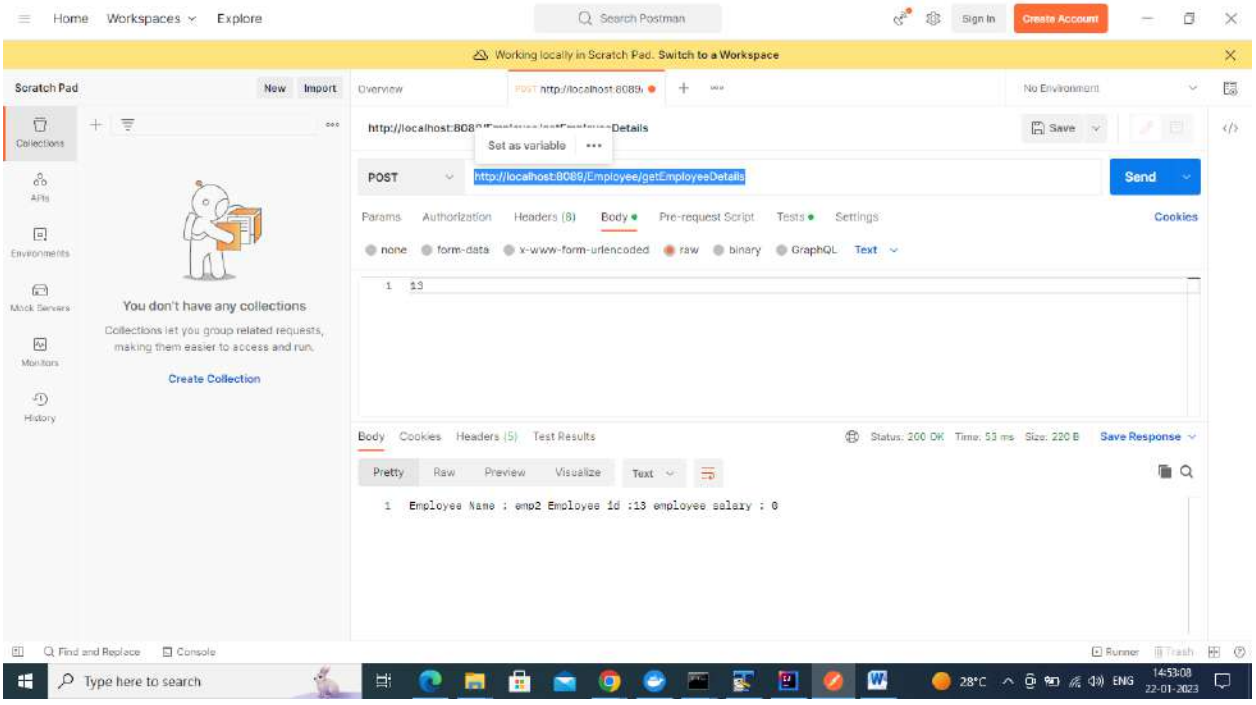
http://localhost:8089/Employee/welcome



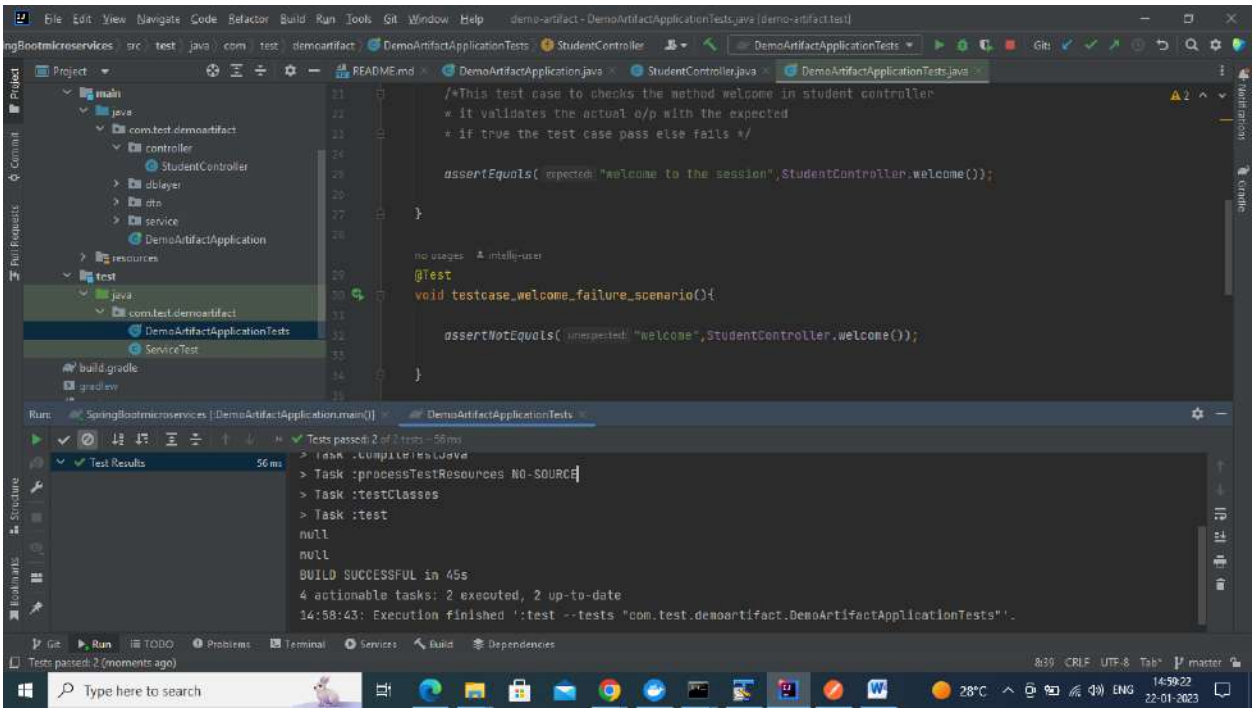
http://localhost:8089/Employee/getAllEmployees



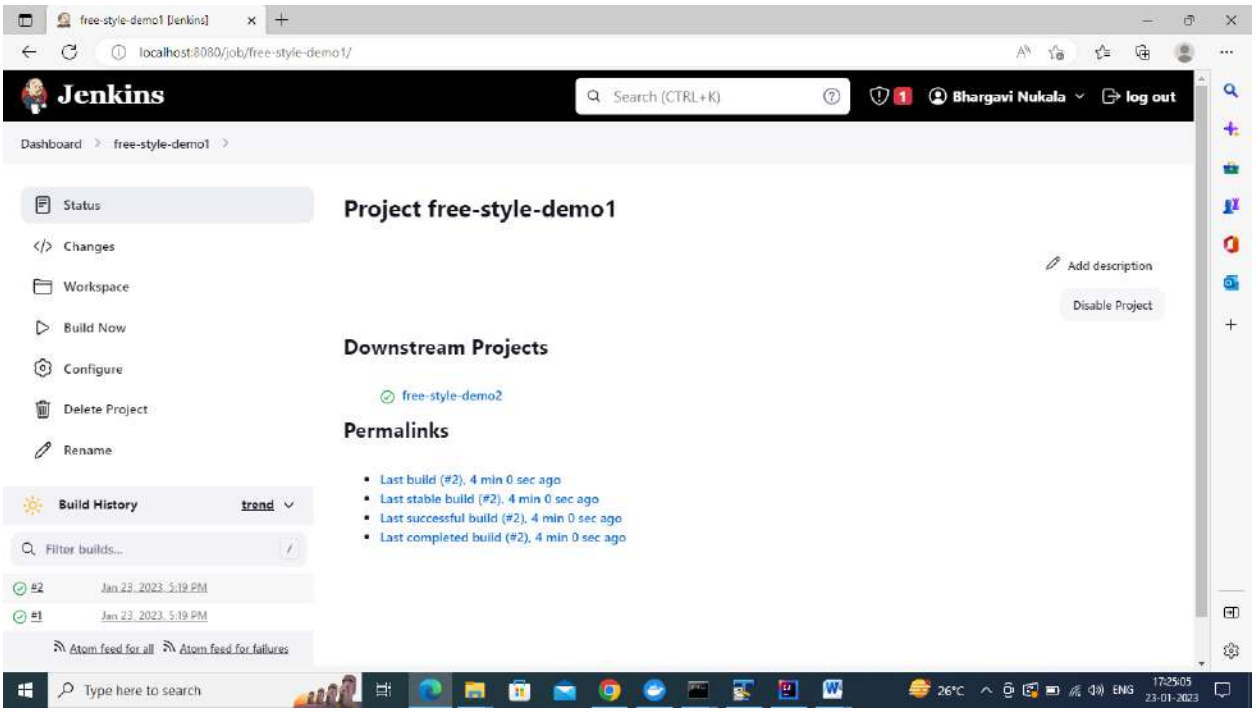
http://localhost:8089/Employee/getEmployeeDetails -- POST method , passed empid=13 in body

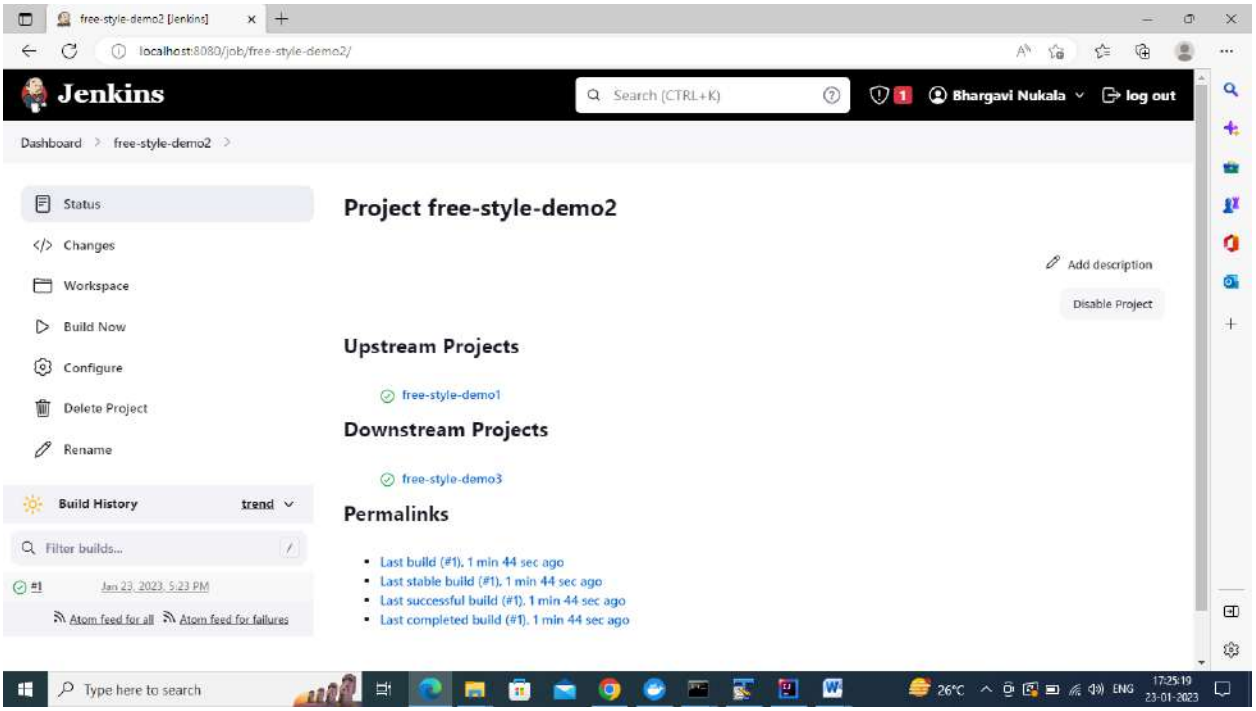


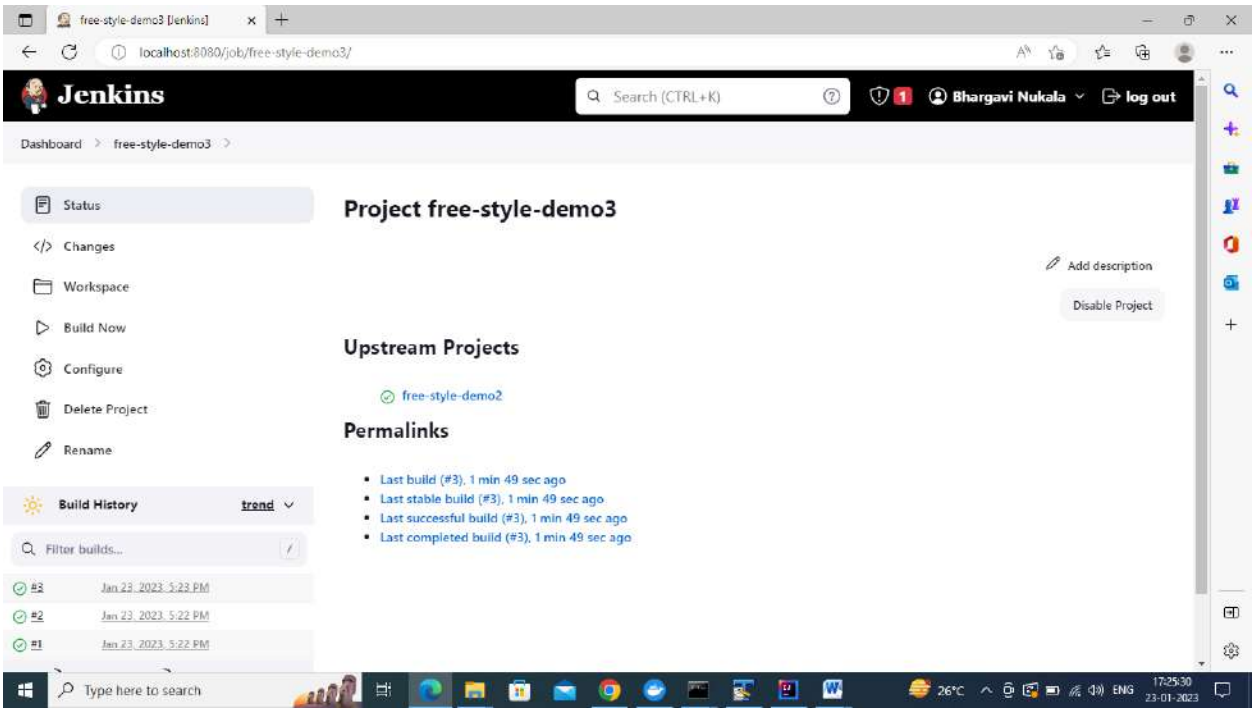
Task 3: Run any test-cases



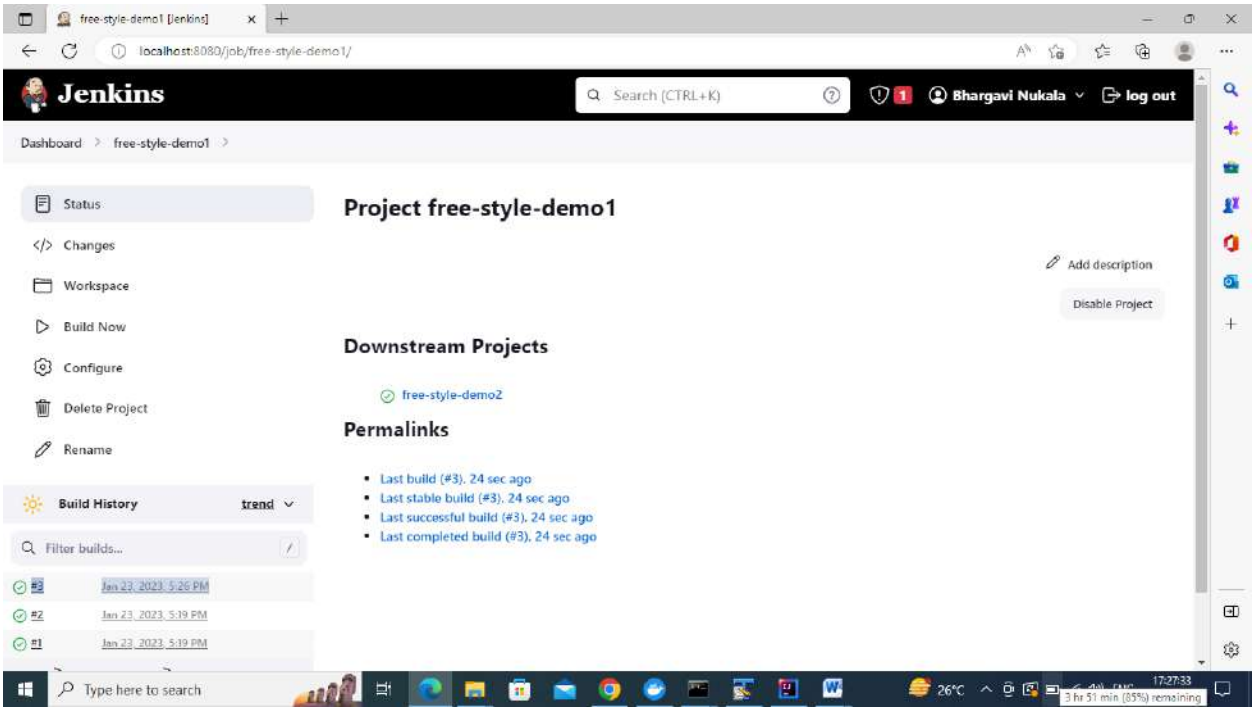
Jenkins project:
Task 1:
In jenkins create 3 free style projects,
free-style-demo1
free-style-demo2
free-style-demo3

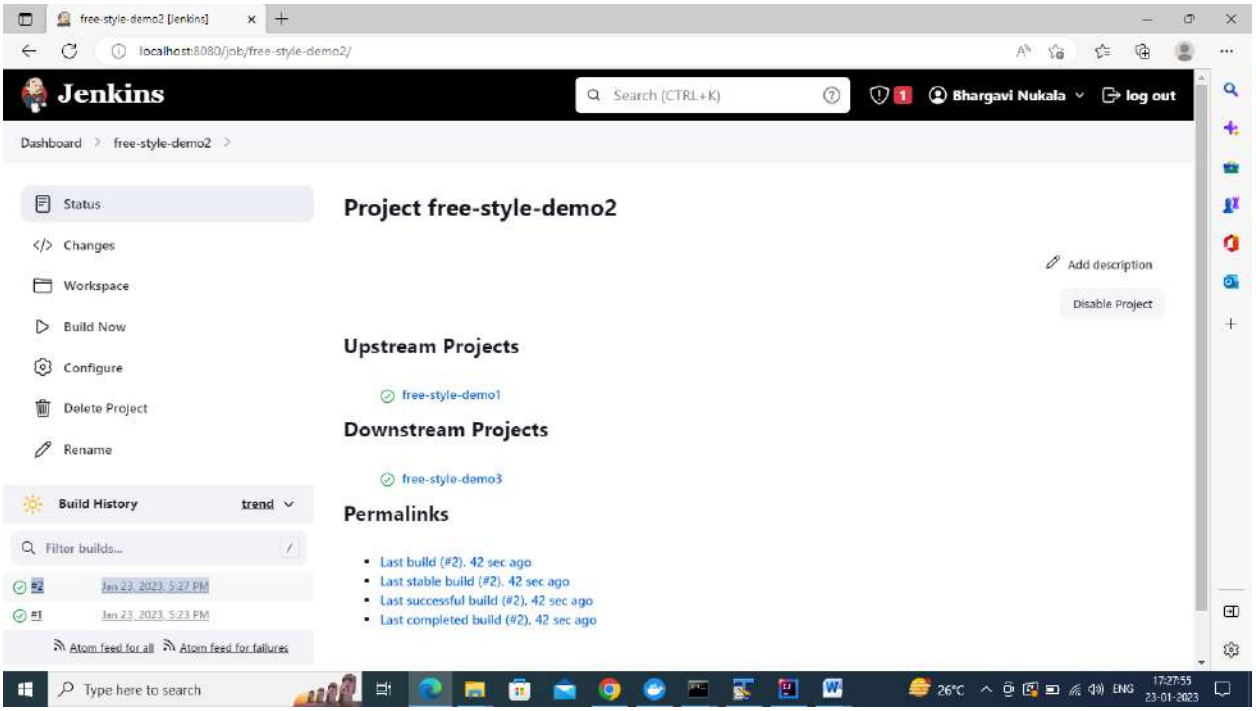


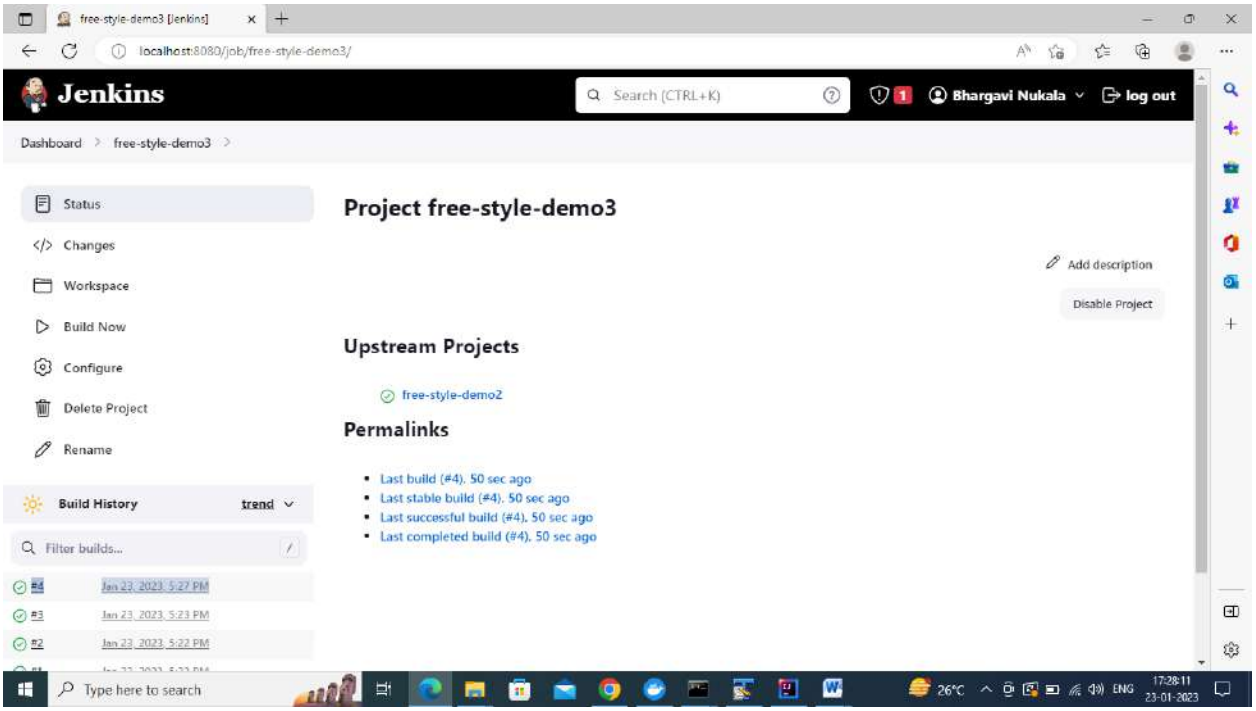

free-style-demo2 [Jenkins]
localhost:8080/job/free-style-demo2/
Jenkins
Search (CTRL+K)
Bhargavi Nukala
log out
Dashboard
free-style-demo2
Status
Changes
Workspace
Build Now
Configure
Delete Project
Rename
Build History
trend
Filter builds...
#1
Jan 23, 2023, 5:23 PM
Atom feed for all
Atom feed for failures
Project free-style-demo2
Add description
Disable Project
Upstream Projects
free-style-demo1
Downstream Projects
free-style-demo3
Permalinks
Last build (#1), 1 min 44 sec ago
Last stable build (#1), 1 min 44 sec ago
Last successful build (#1), 1 min 44 sec ago
Last completed build (#1), 1 min 44 sec ago
Type here to search
26°C
ENG
17:25:19
23-01-2023


free-style-demo3 [Jenkins]
localhost:8080/job/free-style-demo3/
Jenkins
Search (CTRL+K)
Bhargavi Nukala
log out
Dashboard
free-style-demo3
Status
Changes
Workspace
Build Now
Configure
Delete Project
Rename
Build History
trend
Filter builds...
#3
Jan 23, 2023, 5:23 PM
#2
Jan 23, 2023, 5:22 PM
#1
Jan 23, 2023, 5:22 PM
Project free-style-demo3
Add description
Disable Project
Upstream Projects
free-style-demo2
Permalinks
Last build (#3), 1 min 49 sec ago
Last stable build (#3), 1 min 49 sec ago
Last successful build (#3), 1 min 49 sec ago
Last completed build (#3), 1 min 49 sec ago
Type here to search
26°C
ENG
17:25:30
23-01-2023

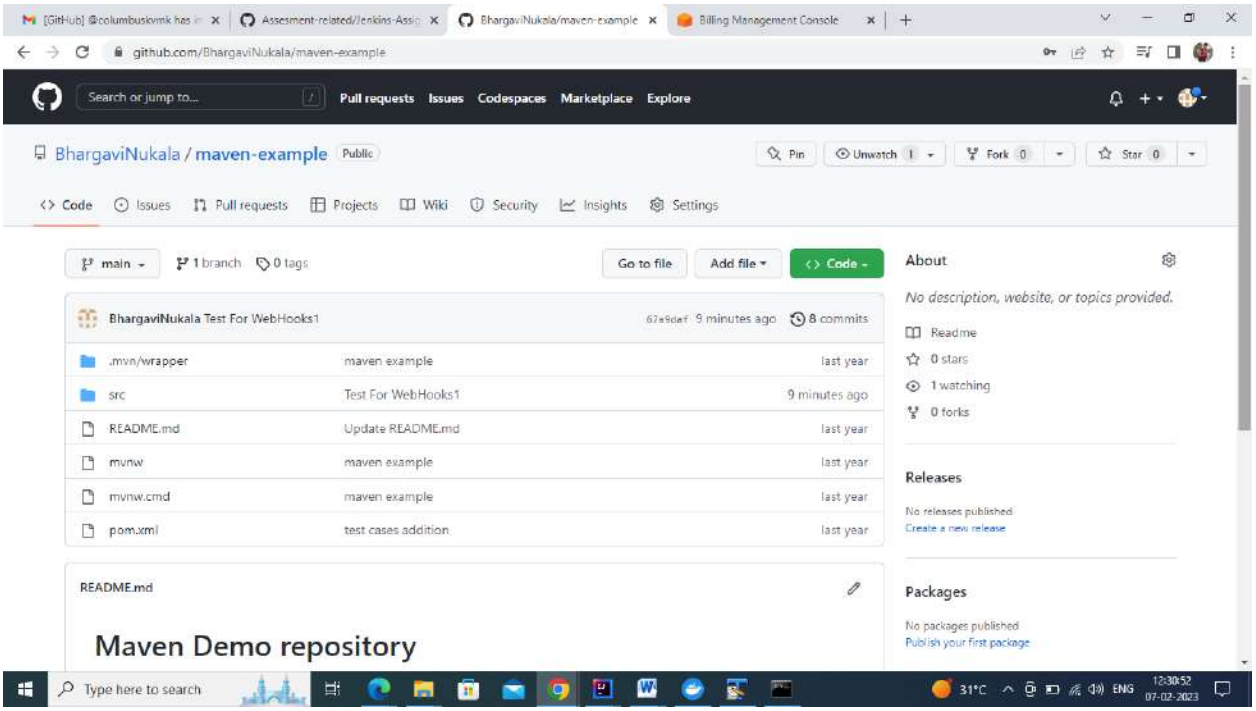
Once free-style-1 is built then free-style-demo2 and later free-style-demo3 should build.



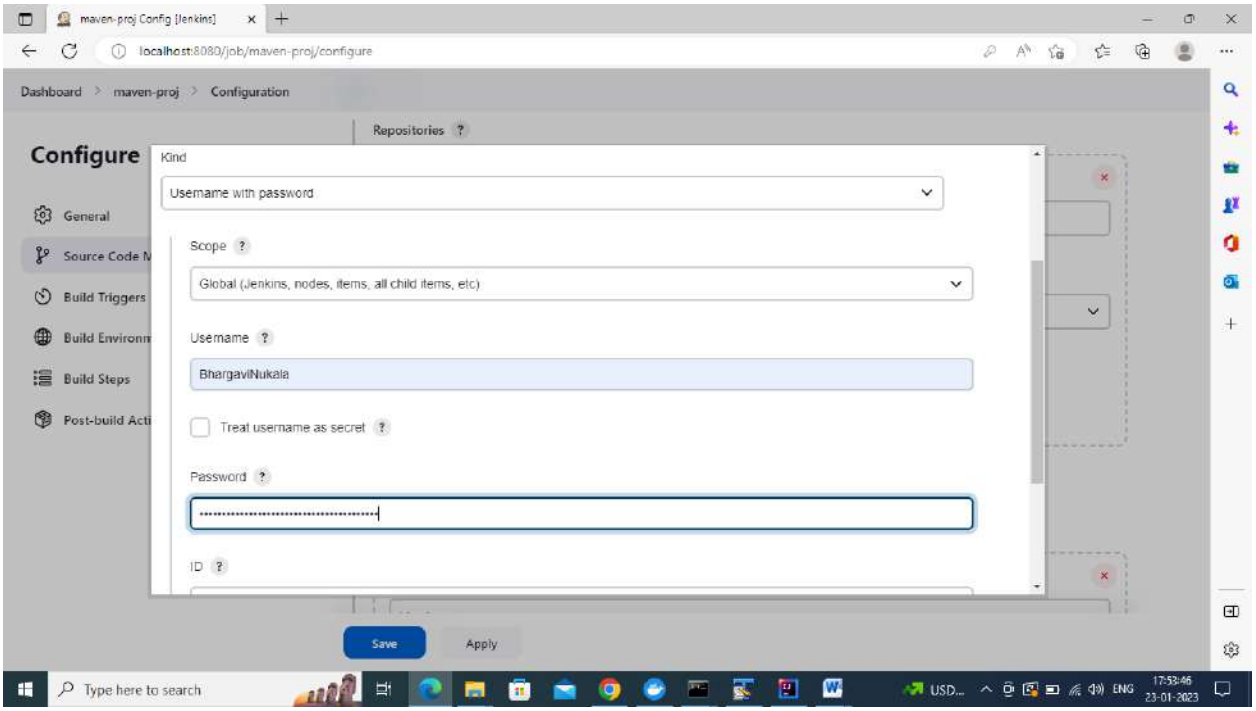


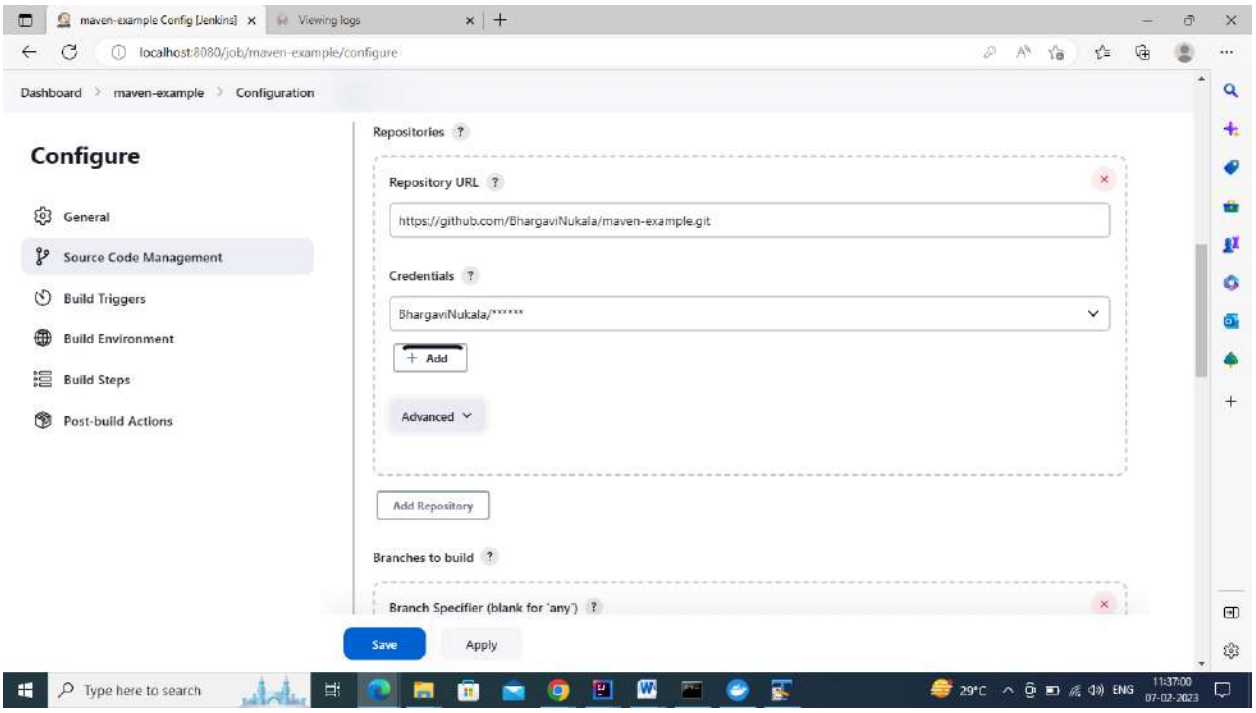


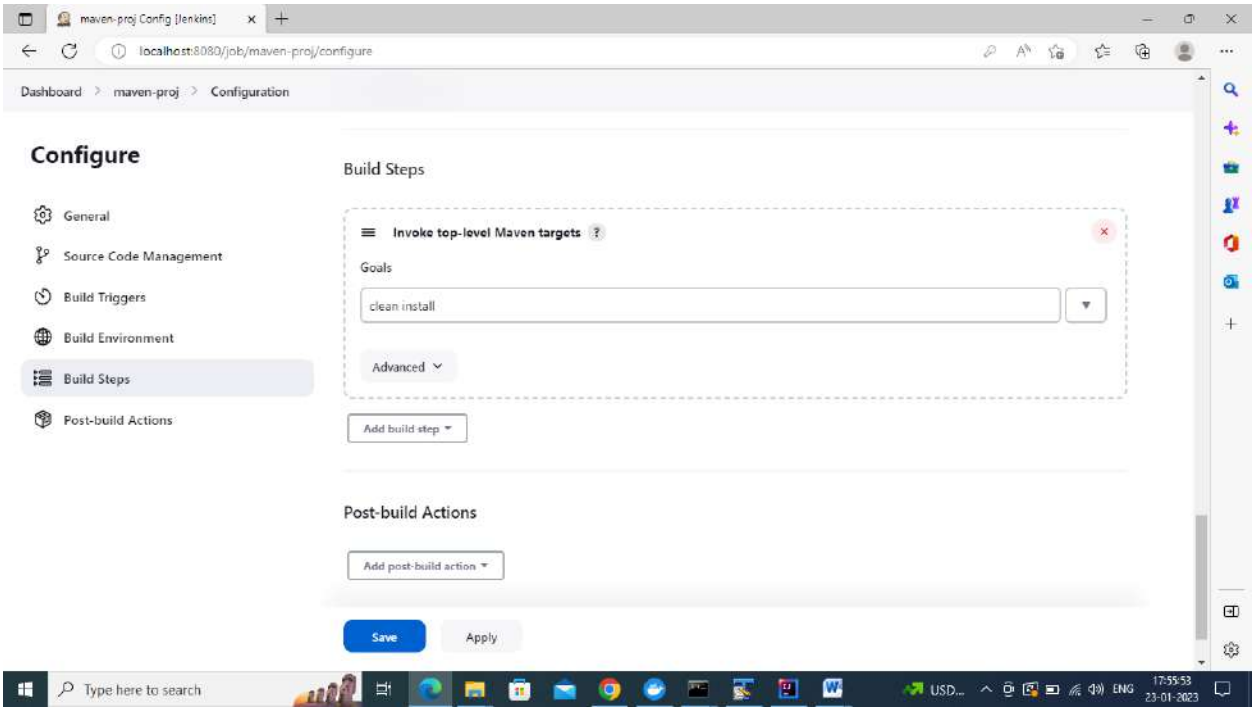
Task 2:
Create a new repo in your login and checkin the maven project.



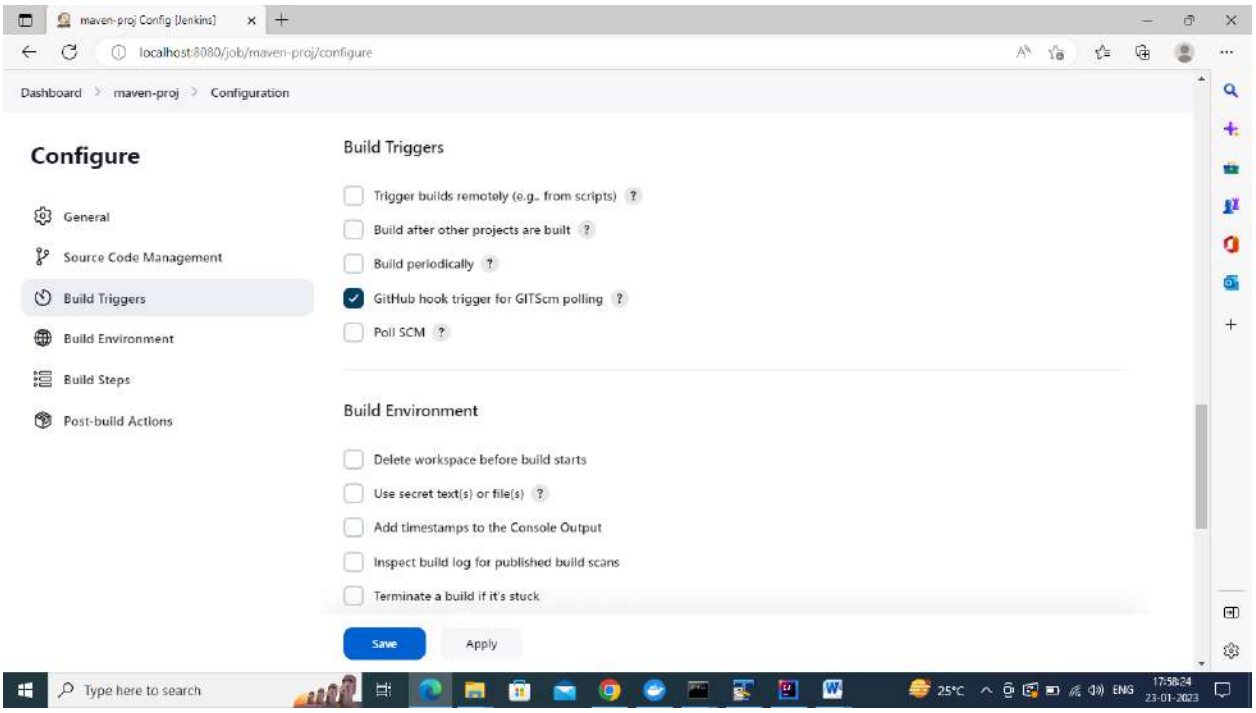
Configure a Maven project from GIT in Jenkins. Configure credentials of Git as username=GIT username and password=GIT token, ID is some random name.


maven-proj Config [Jenkins]
localhost:8080/job/maven-proj/configure
Dashboard > maven-proj > Configuration
Configure
General
Source Code M
Build Triggers
Build Environm
Build Steps
Post-build Acti
Repositories
Kind
Username with password
Scope
Global (Jenkins, nodes, items, all child items, etc)
Username
BhargaviNukala
Treat username as secret
Password
ID
Save
Apply
Type here to search


maven-example Config [Jenkins]
Viewing logs
localhost:8080/job/maven-example/configure
Dashboard > maven-example > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Repositories
Repository URL
https://github.com/BhargaviNukala/maven-example.git
Credentials
BhargaviNukala/******
Add
Advanced
Add Repository
Branches to build
Branch Specifier (blank for 'any')
Save
Apply
Type here to search

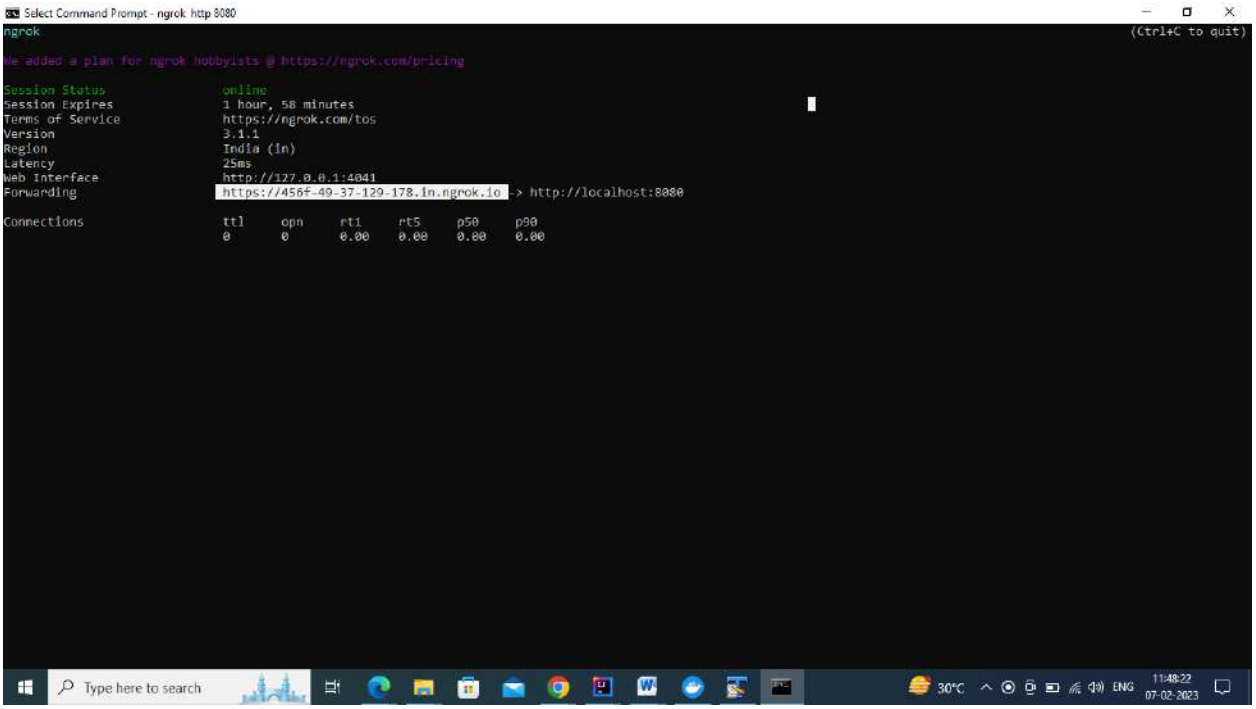


Select Github hook trigger option under Build Triggers

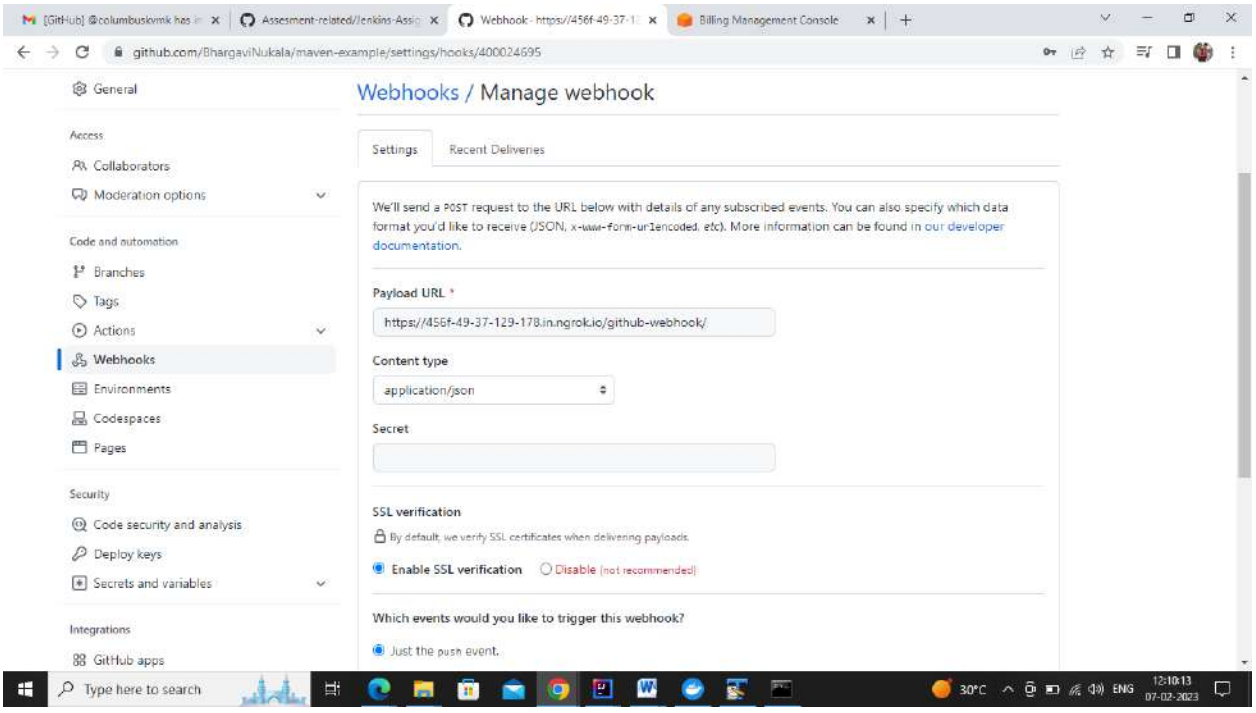


Download ngrok from https://ngrok.com/download
Extract to a folder.
From the cmd, go to the extracted folder and run the command
ngrok http 8080

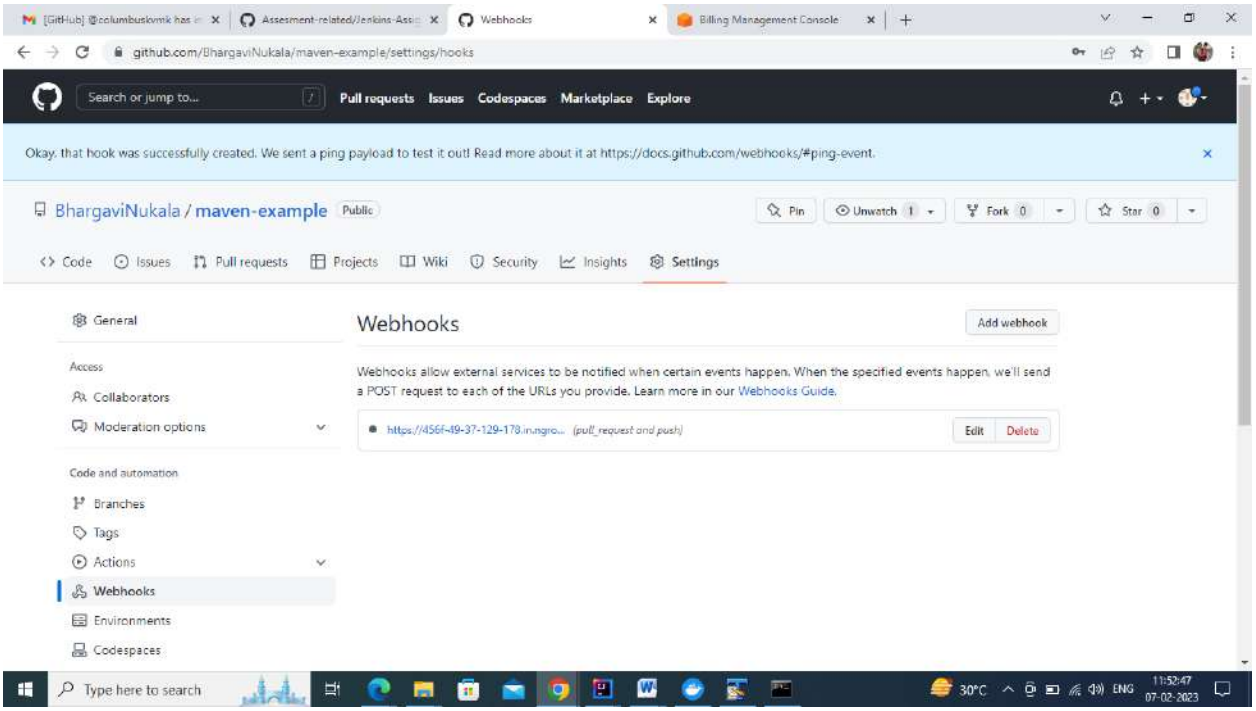
Copy the Forwarding entry in Command Prompt



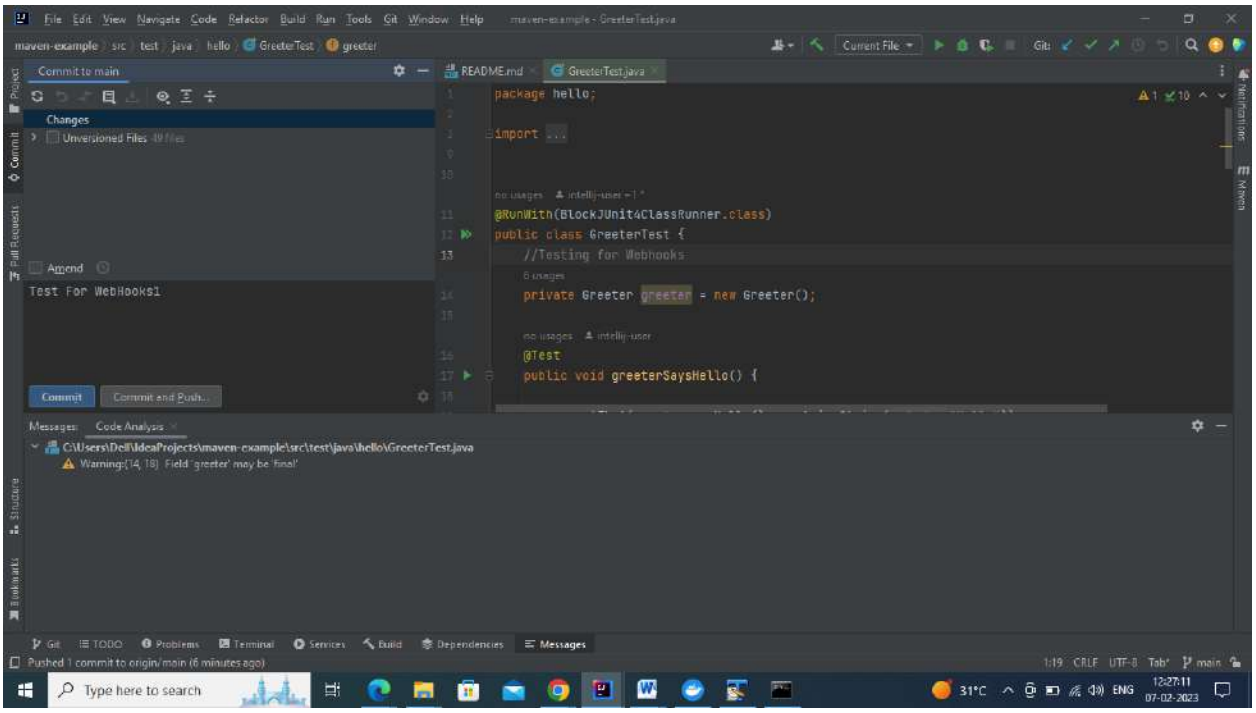
In Git Hub Maven project, under Settings tab ->Click on WebHook and add a new WebHook.
Payload URL => URL copied from ngrok/github-webhook



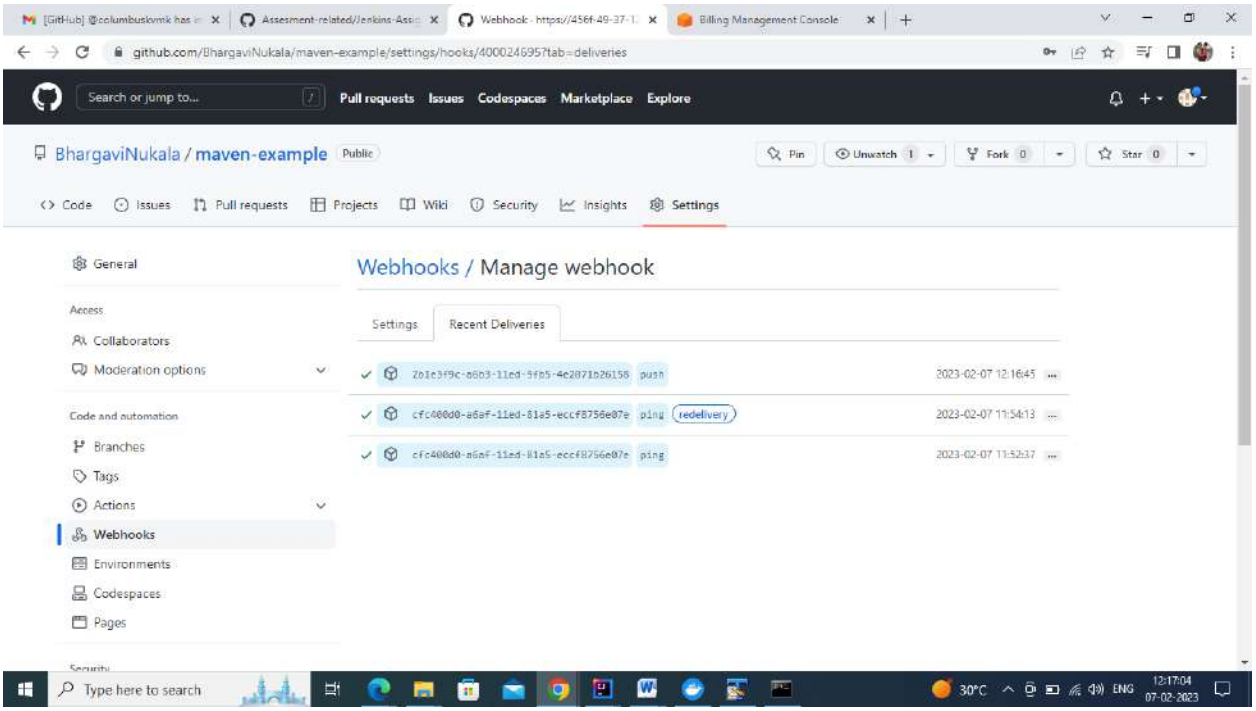
A new WebHook is added



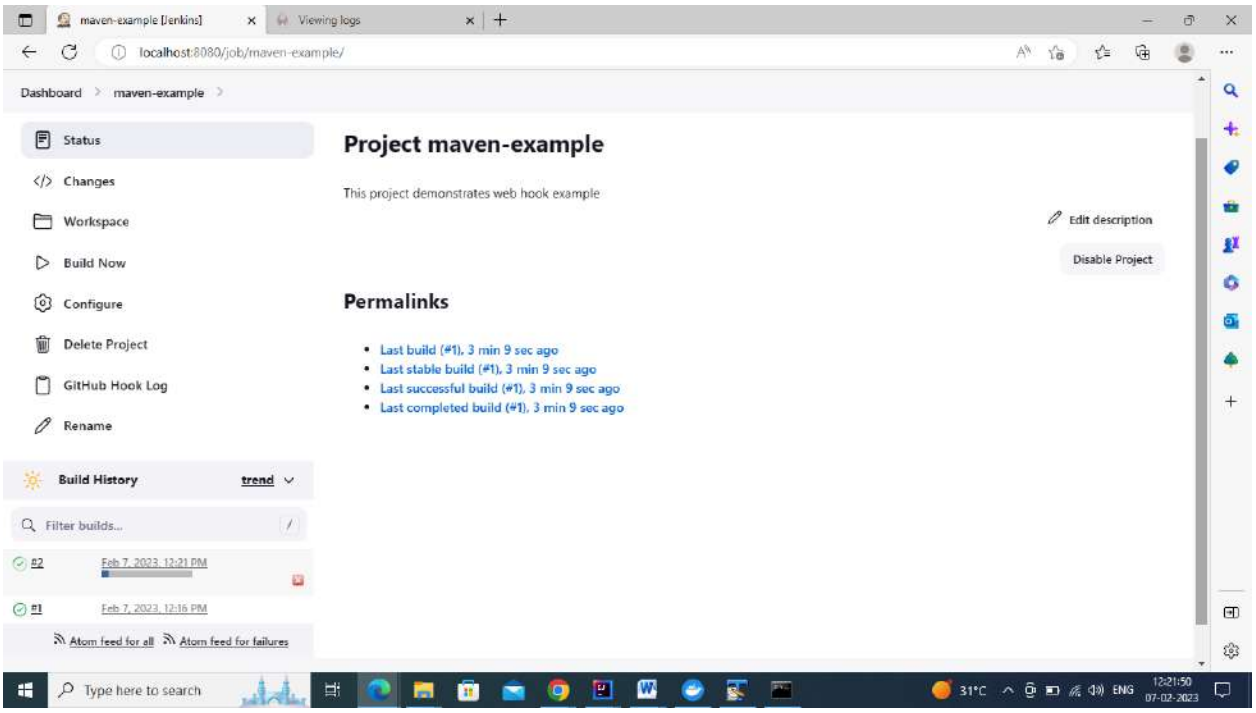
Make a change in the project and commit and push the changes to the repo

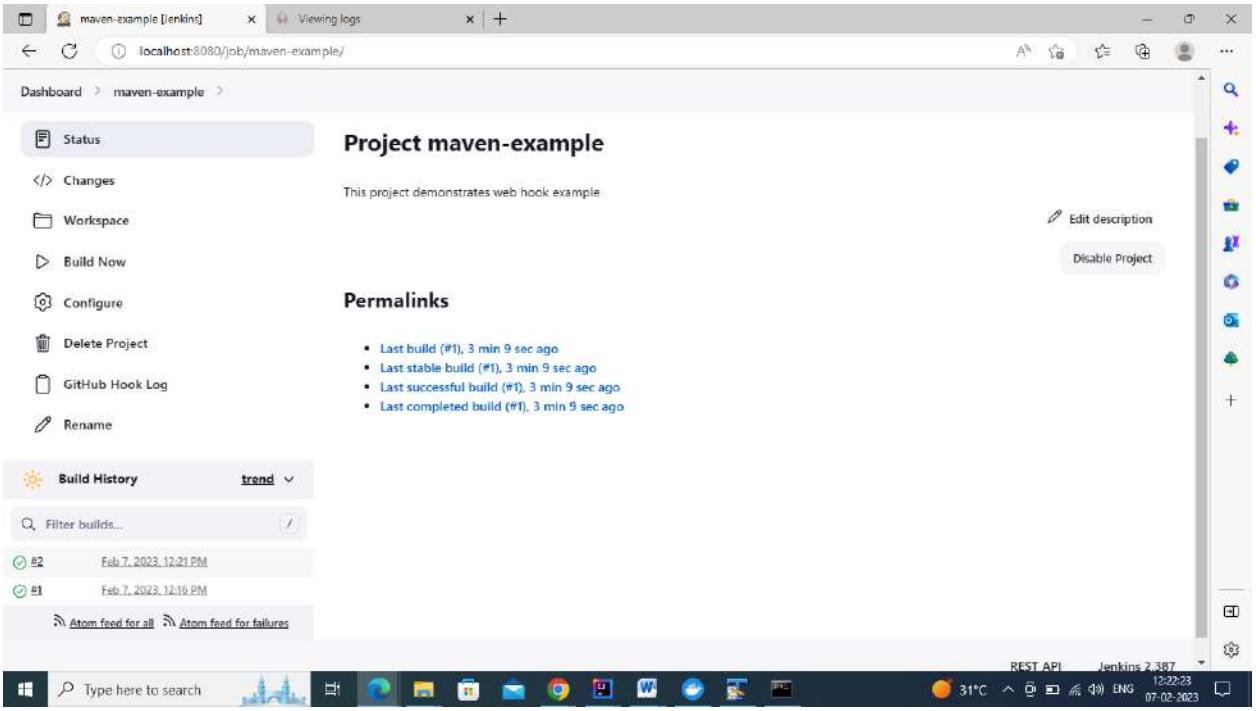


If we check in the GitHub WebHooks, a new entry for the push can be seen.



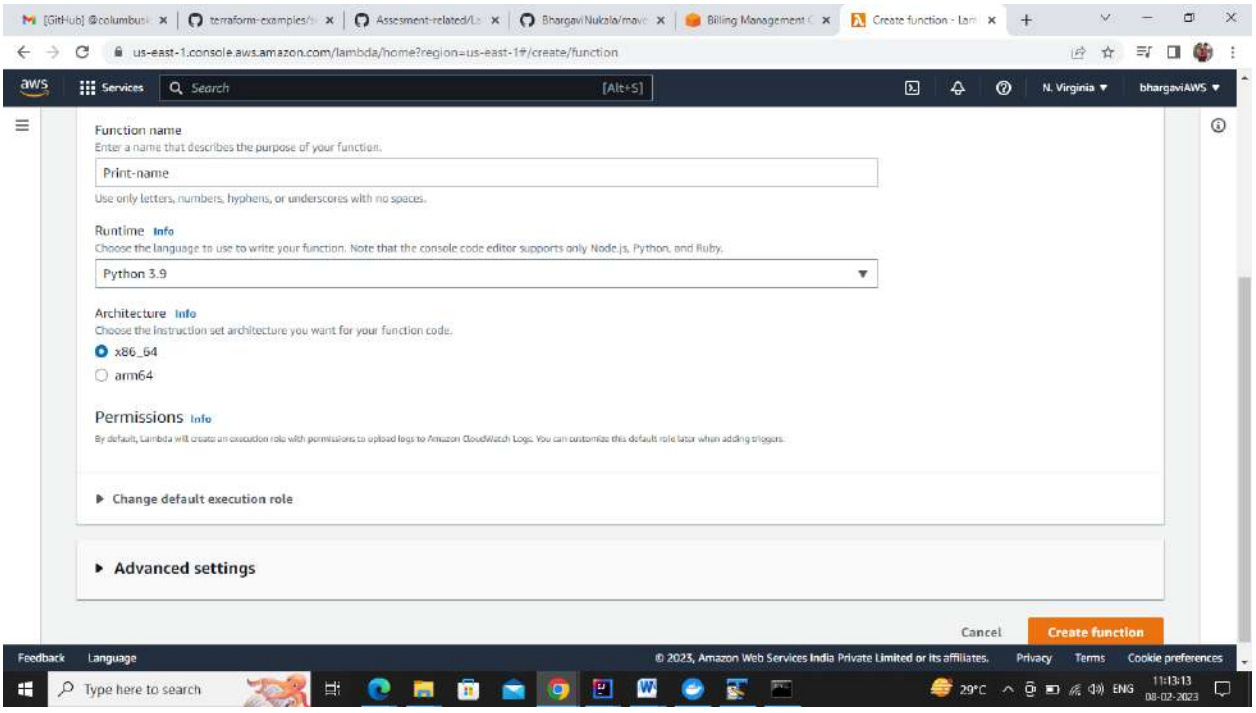
And at the same time, we can see that a new build is triggered in Jenkins.



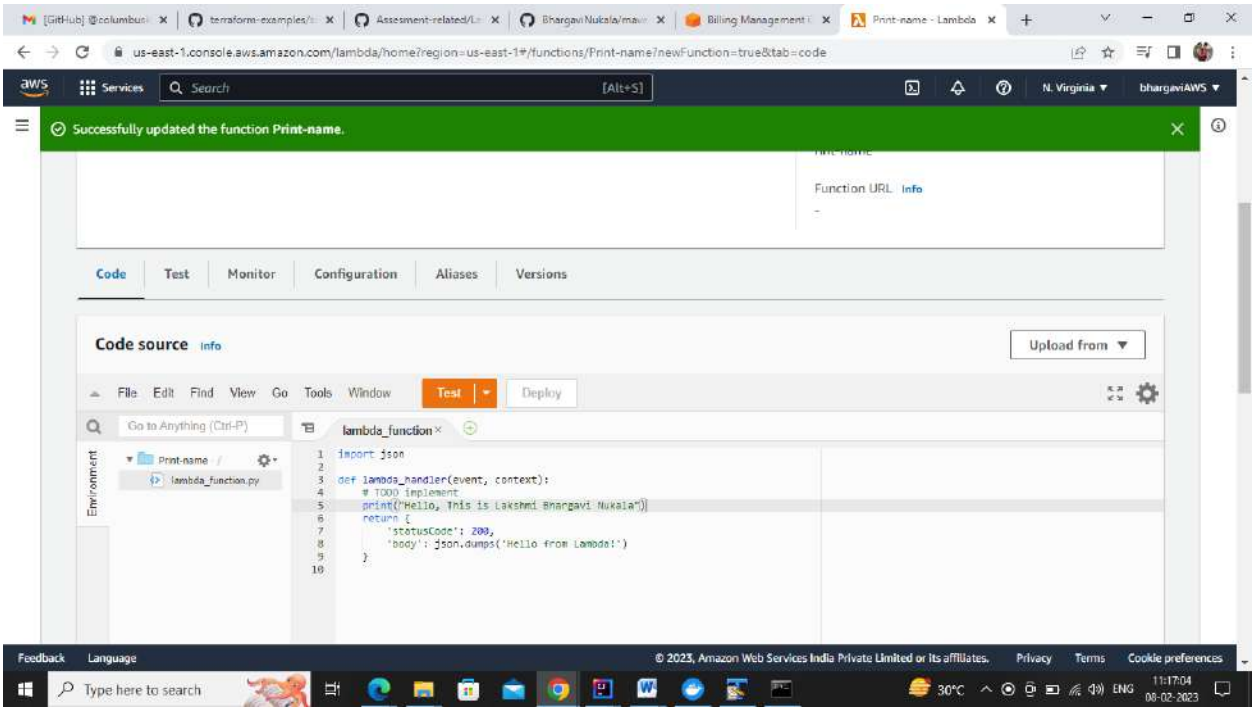


Lambda:

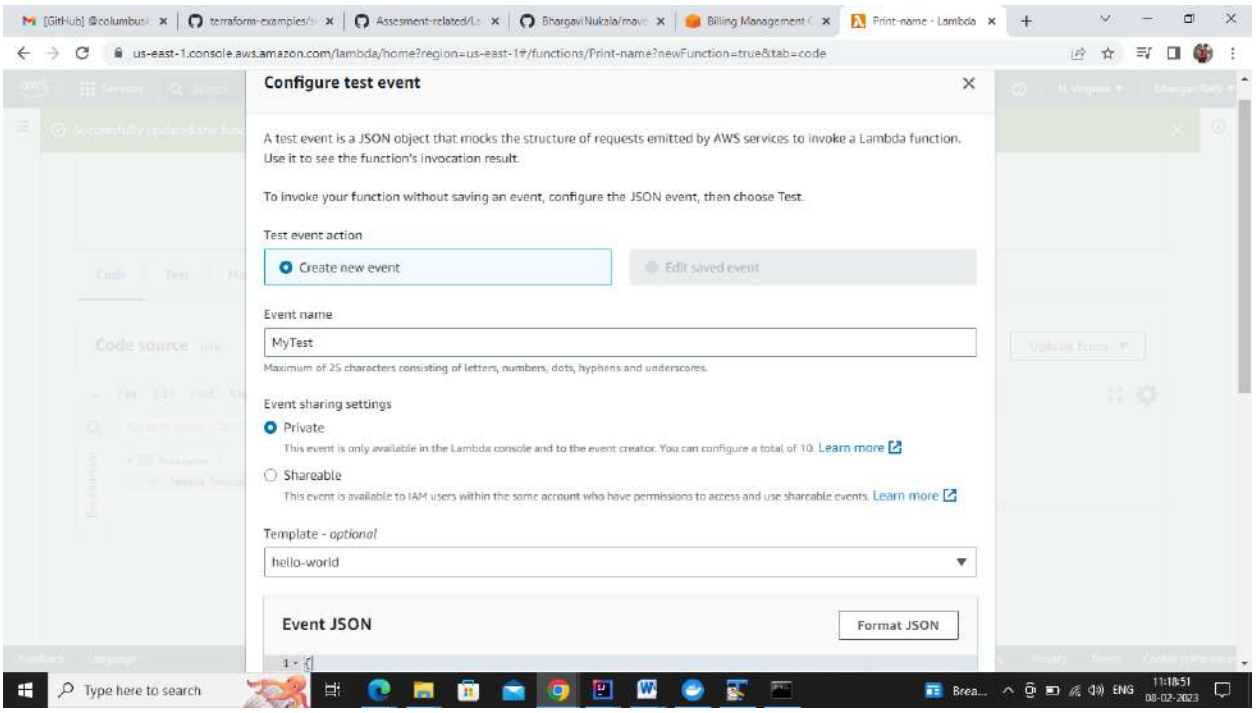
Task -1: Create a Lambda function that prints your name.



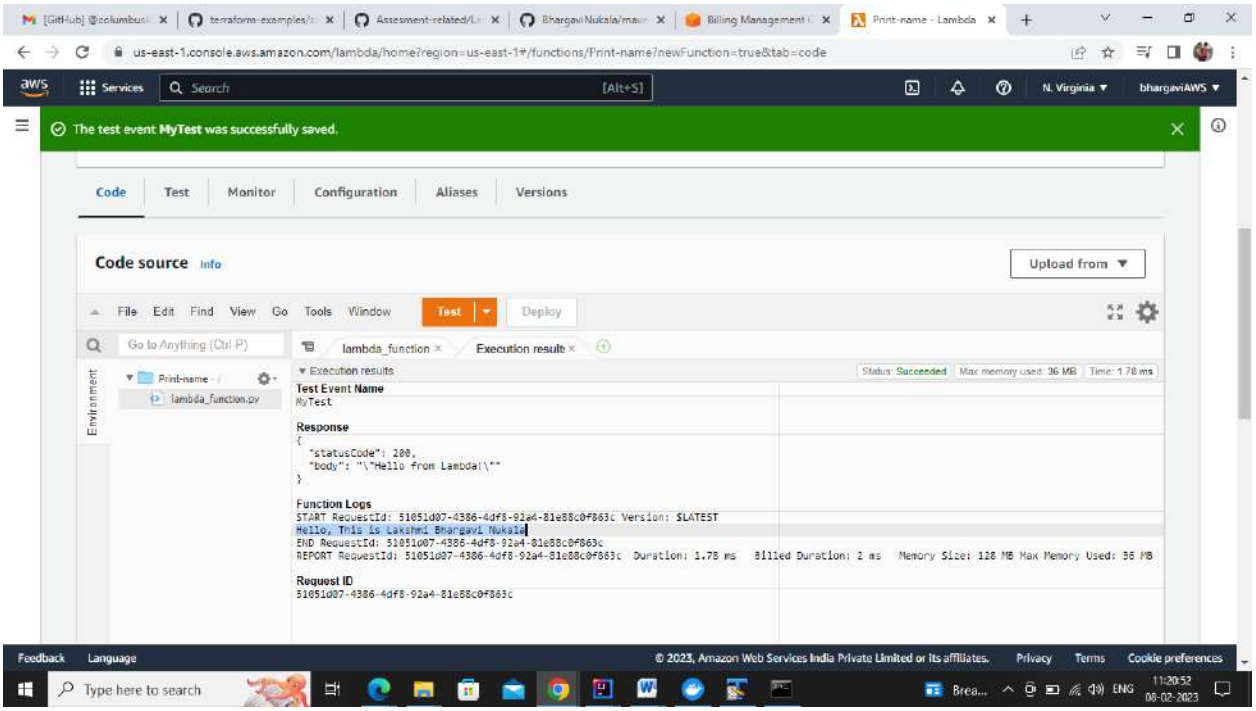
Add a line printing your name in the code. Then Deploy the code.



Now click on Test to configure a test event.

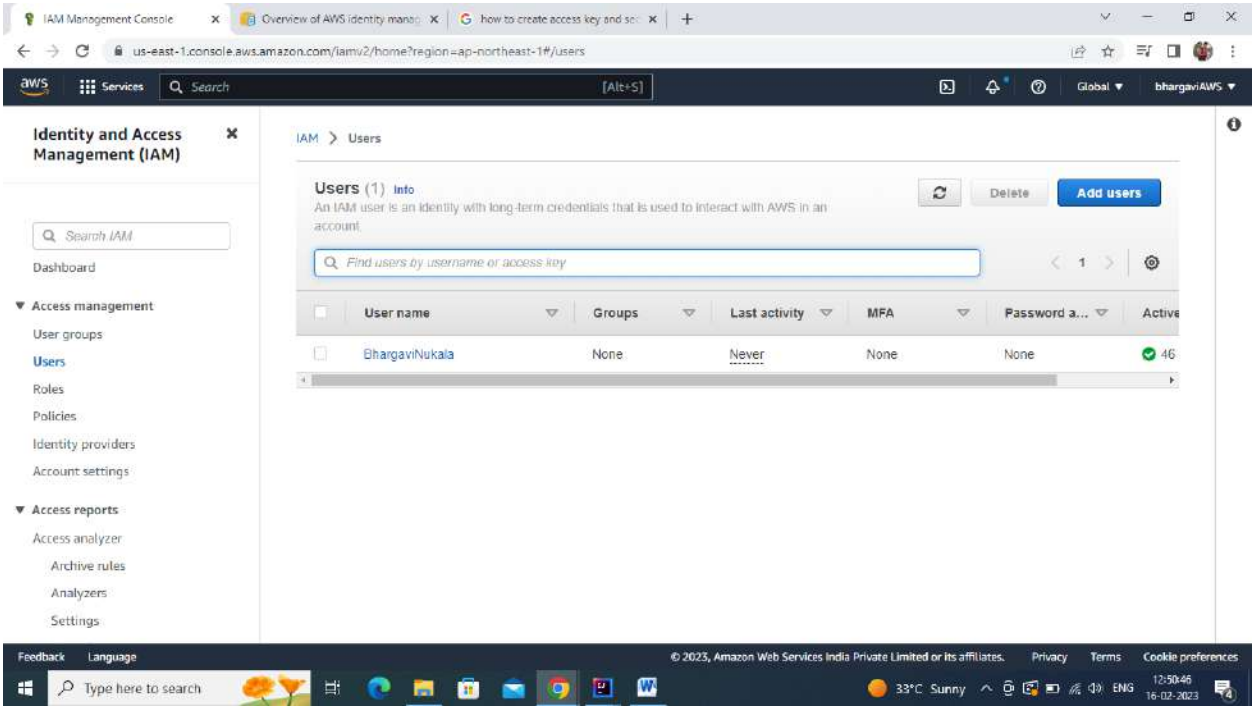


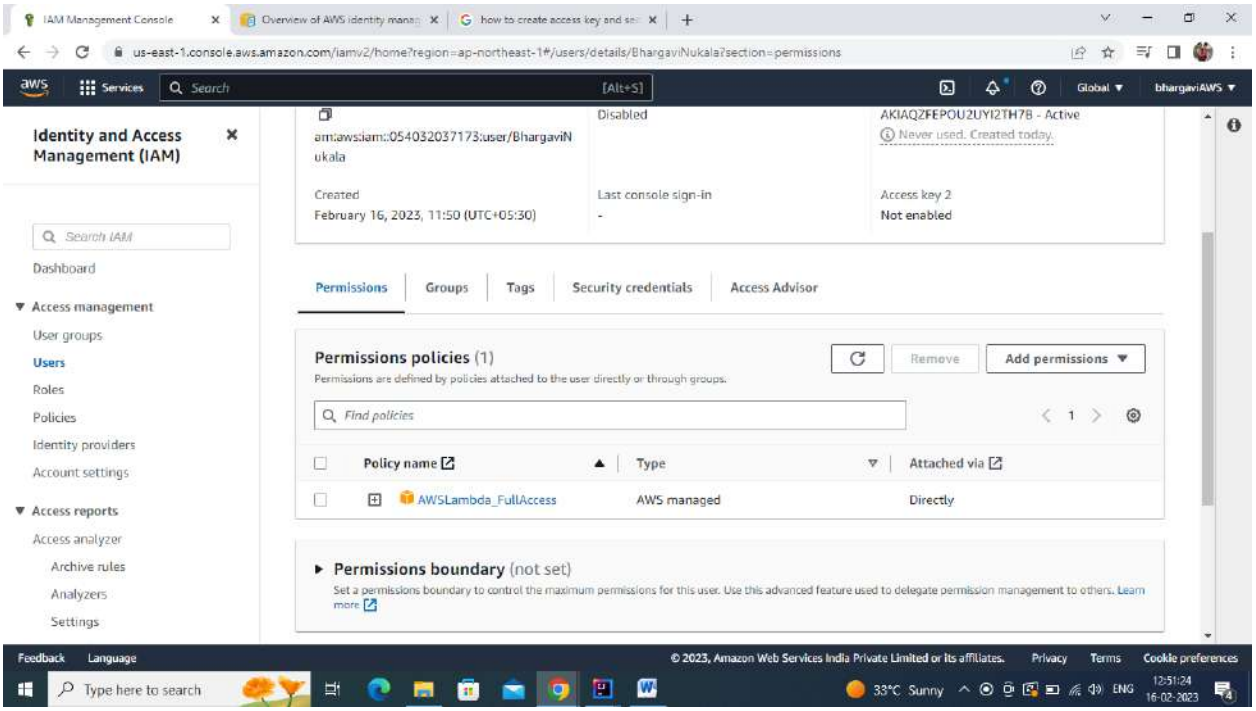
Then Click on Test to run the event and print your name.



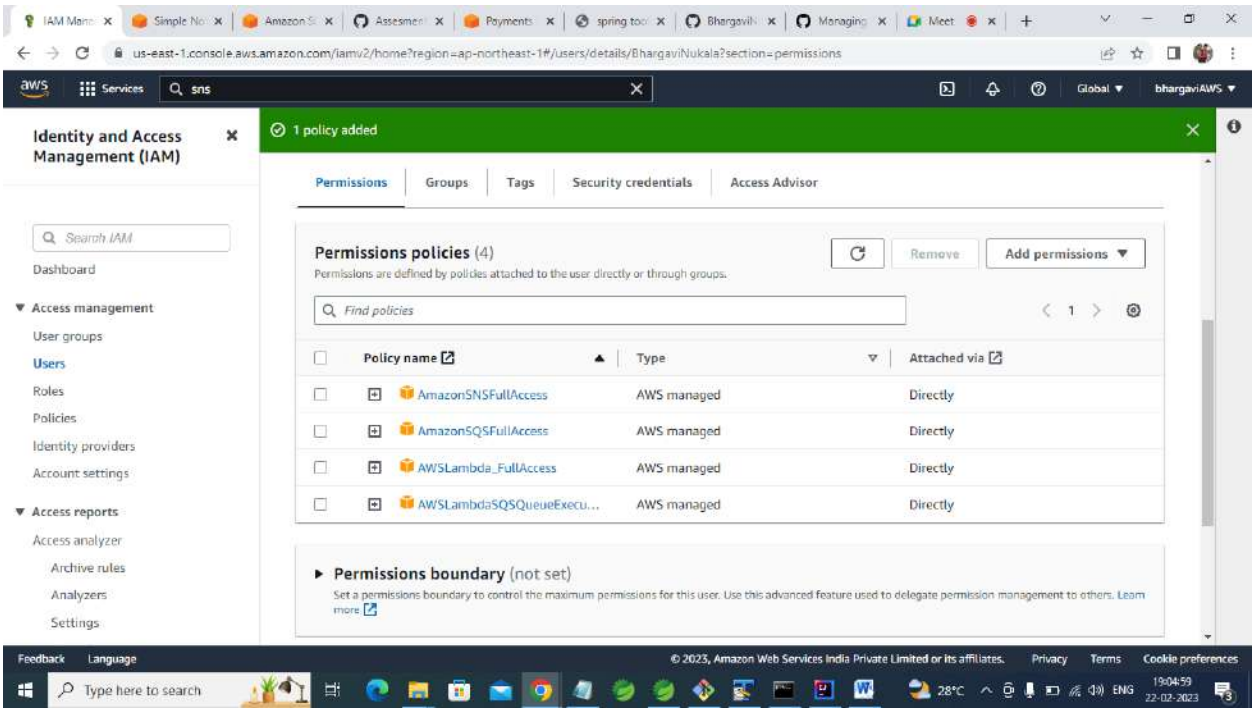
Task-2:

Create an IAM user BhargaviNukala in AWS and give full access to Lambda permission to this user.

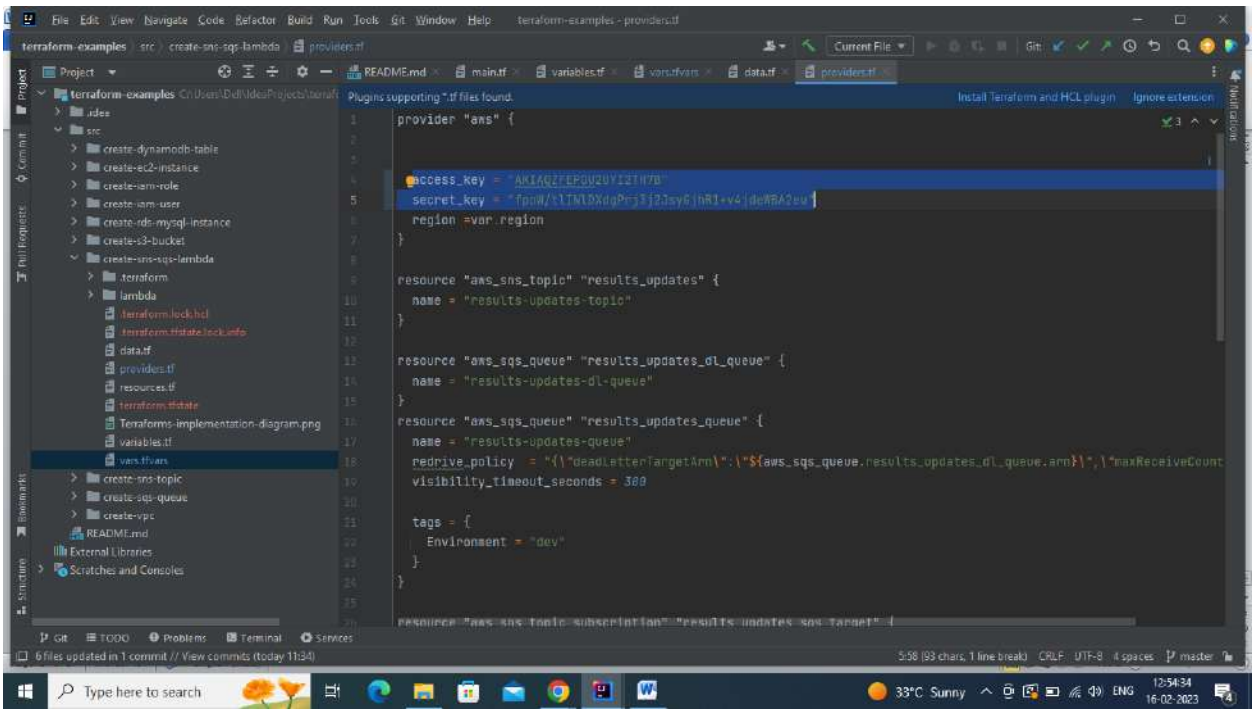


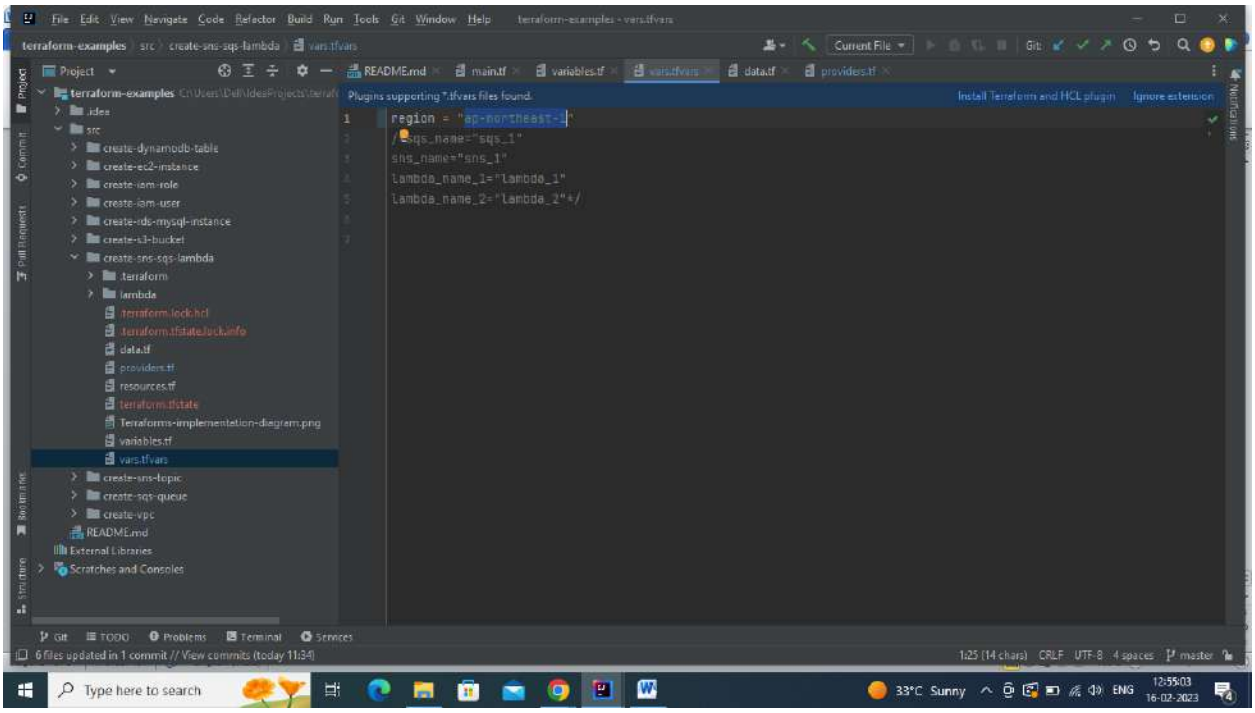


Also give additional permissions for SQS and SNS in Permissions tab

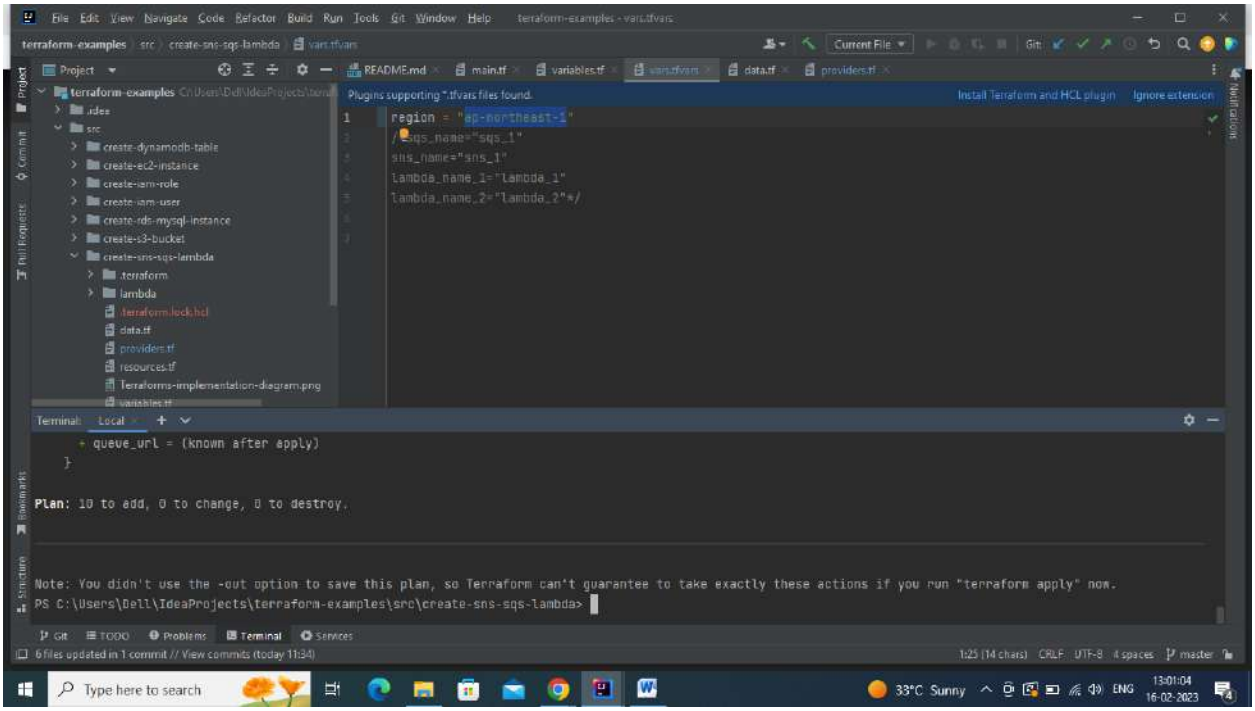


Check out the project terraform-examples into IngtelliJ and set accesskey and secret access code of your AWS IAM user in providers.tf and region in vars.tf

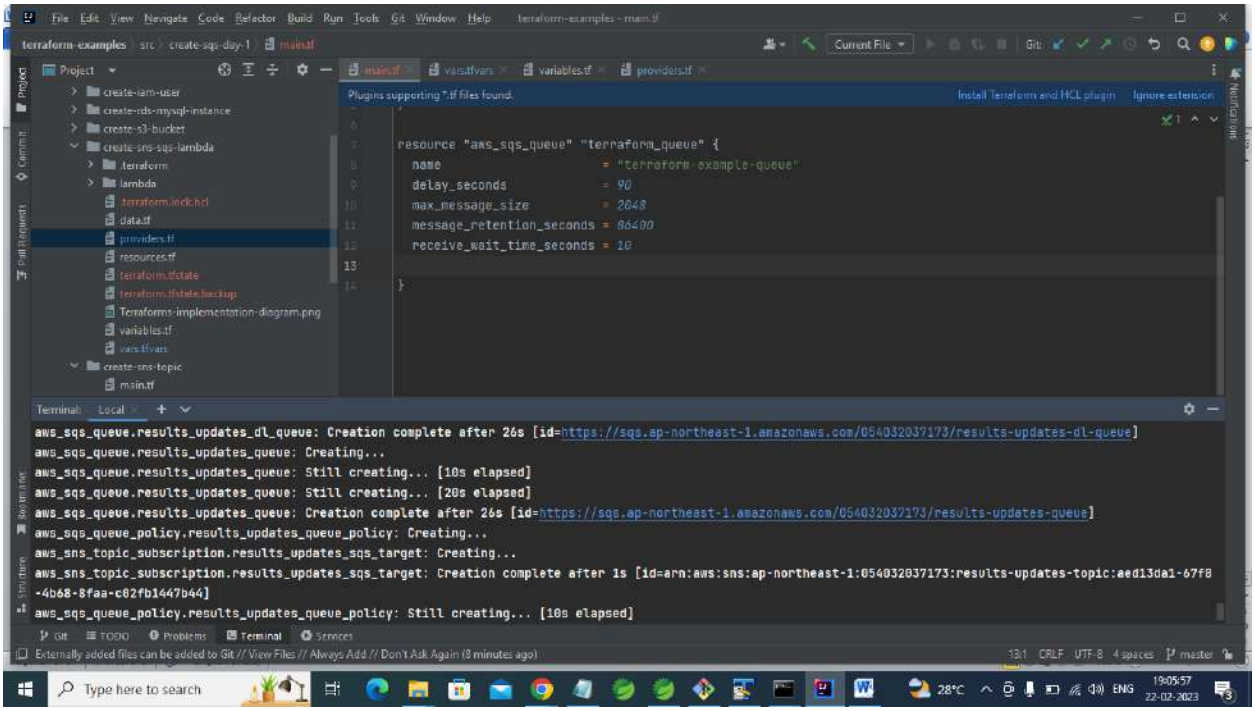




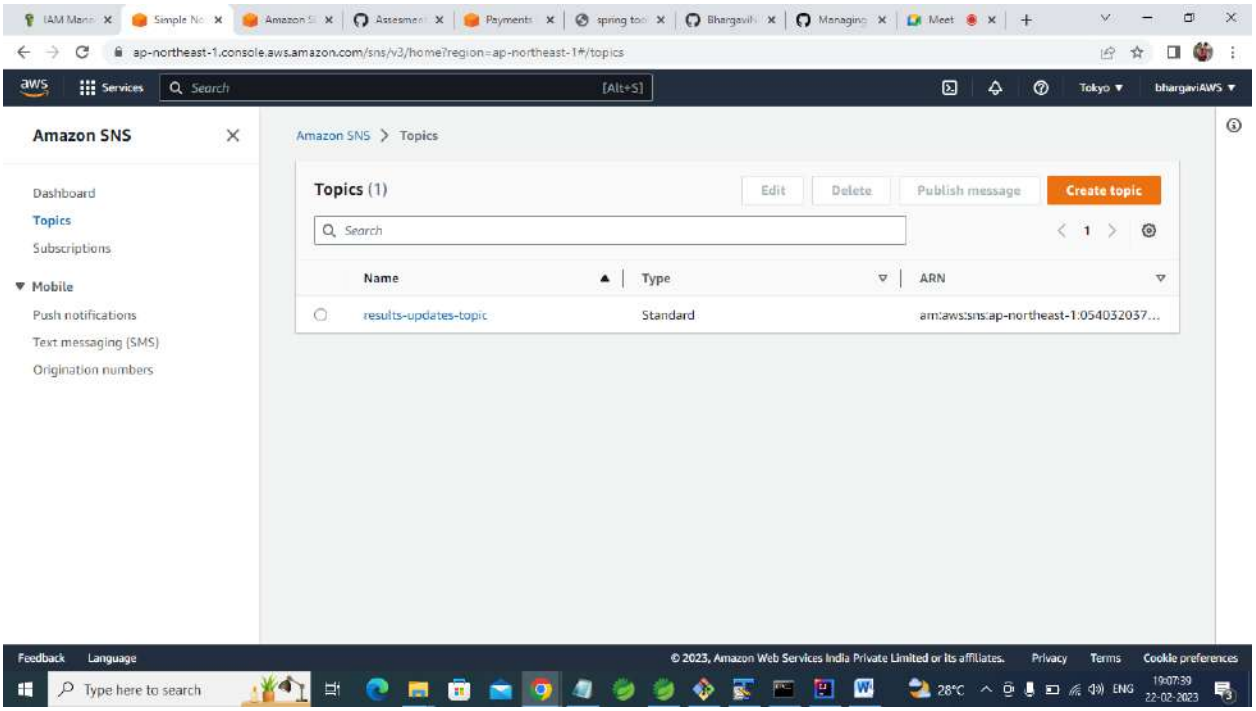
Run “terraform init” and “terraform plan” in the folder

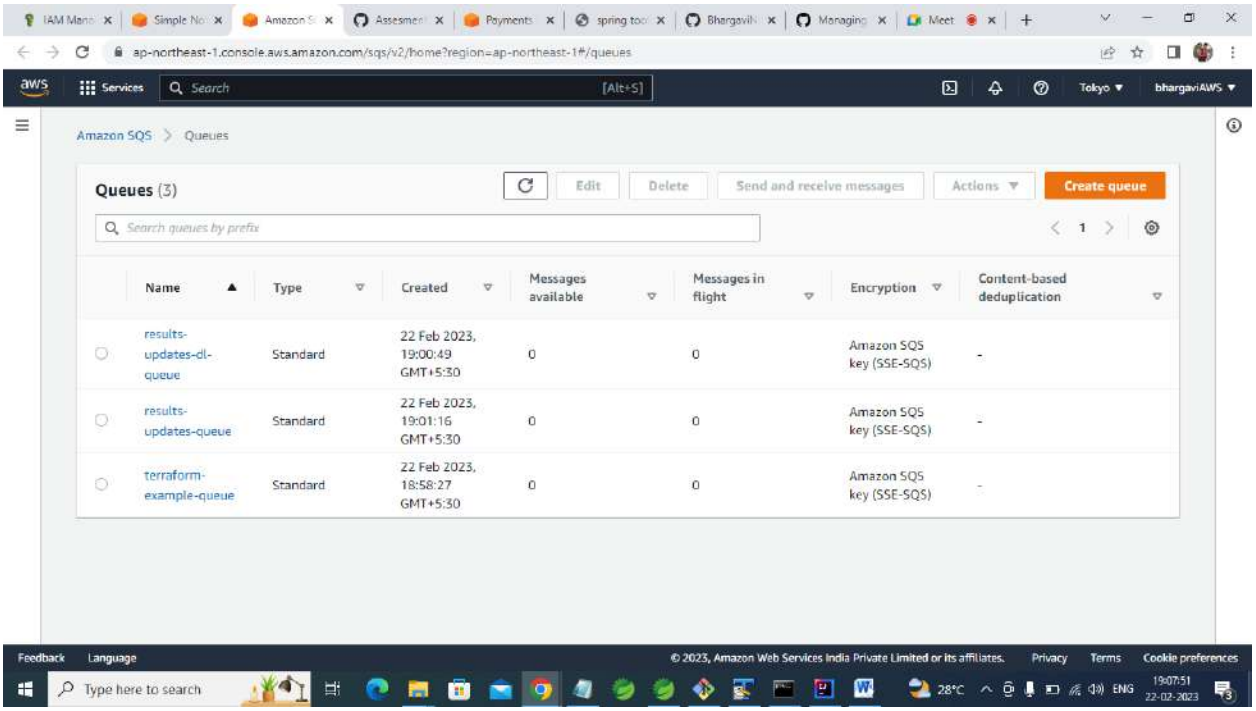


Now run “terraform apply” command. It will ask to enter region and approve by entering yes. Then it will start creating Queue and Topic.



We can see that a Topic and Queue has bveen created in AWS


ap-northeast-1.console.aws.amazon.com/sns/v3/home?region=ap-northeast-1#/topics
Services
Search
[Alt+S]
Tokyo
bhargaviAWS
Amazon SNS
Dashboard
Topics
Subscriptions
Mobile
Push notifications
Text messaging (SMS)
Origination numbers
Amazon SNS > Topics
Topics (1)
Edit
Delete
Publish message
Create topic
Search
Name
Type
ARN
results-updates-topic
Standard
arn:aws:sns:ap-northeast-1:054032037...
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
28°C
ENG
19:07:39
22-02-2023


ap-northeast-1.console.aws.amazon.com/sqs/v2/home?region=ap-northeast-1#/queues
Services
Search
[Alt+S]
Tokyo
bhargaviAWS
Amazon SQS > Queues
Queues (3)
Edit
Delete
Send and receive messages
Actions
Create queue
Search queues by prefix
Name
Type
Created
Messages available
Messages in flight
Encryption
Content-based deduplication
results-updates-dl-queue
Standard
22 Feb 2023, 19:00:49 GMT+5:30
0
0
Amazon SQS key (SSE-SQS)
-
results-updates-queue
Standard
22 Feb 2023, 19:01:16 GMT+5:30
0
0
Amazon SQS key (SSE-SQS)
-
terraform-example-queue
Standard
22 Feb 2023, 18:58:27 GMT+5:30
0
0
Amazon SQS key (SSE-SQS)
-
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
28°C
ENG
19:07:51
22-02-2023