git

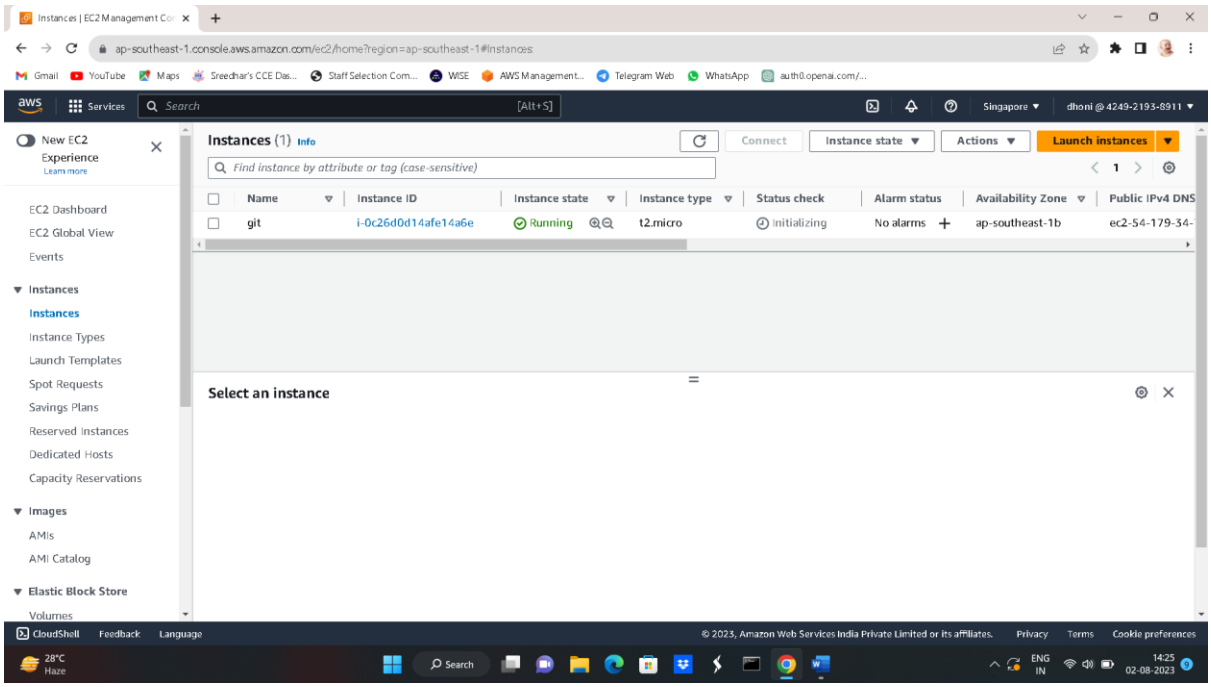
T.AJAY

6301925313

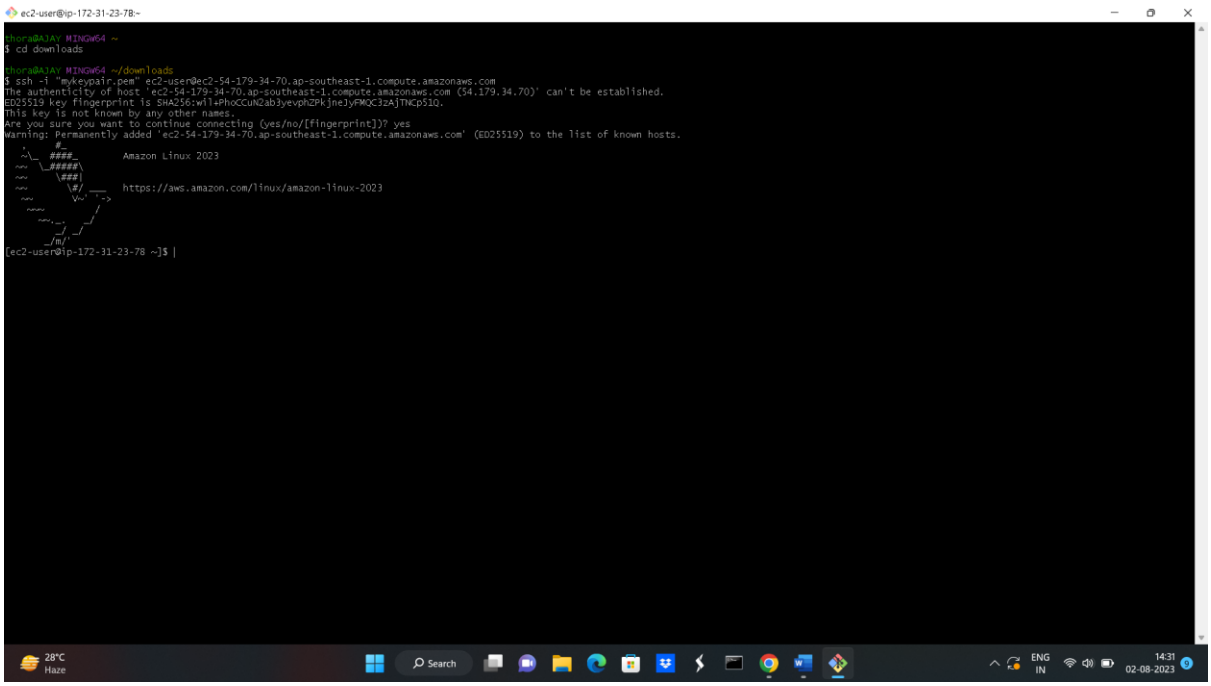
LAB-1

CREATING EC2 INSTANCE

- Login to the aws console.
- Created a server with amazon linux .



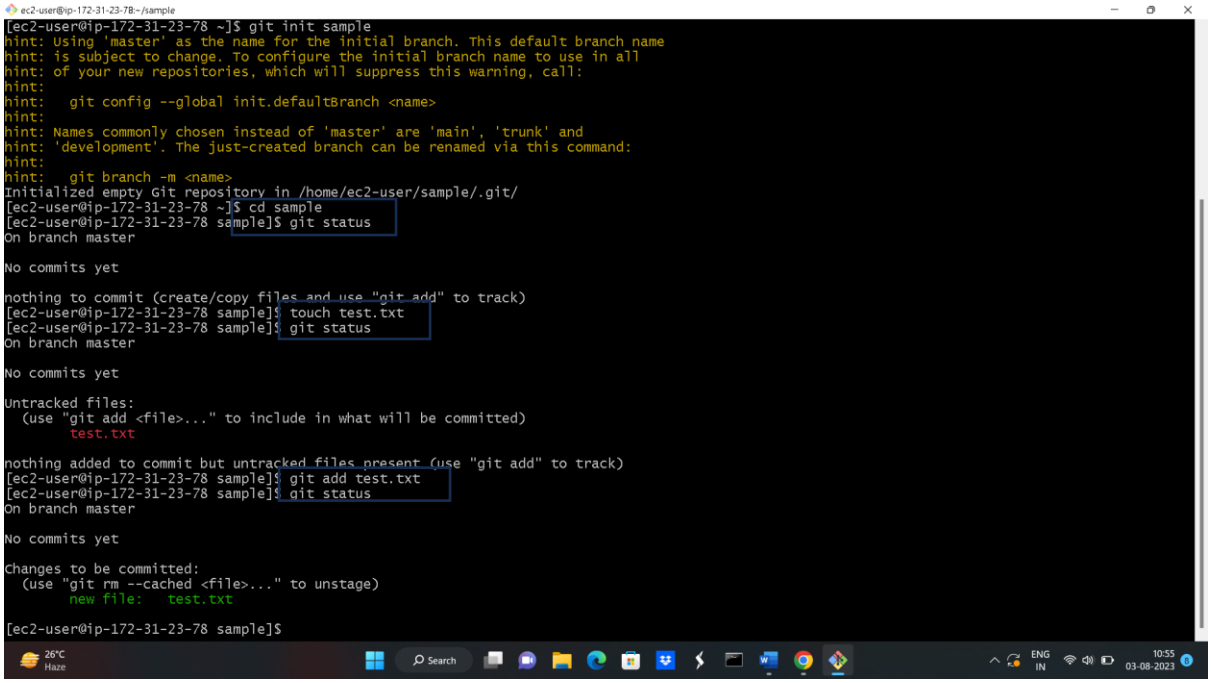
- Connecting the server using git bash.



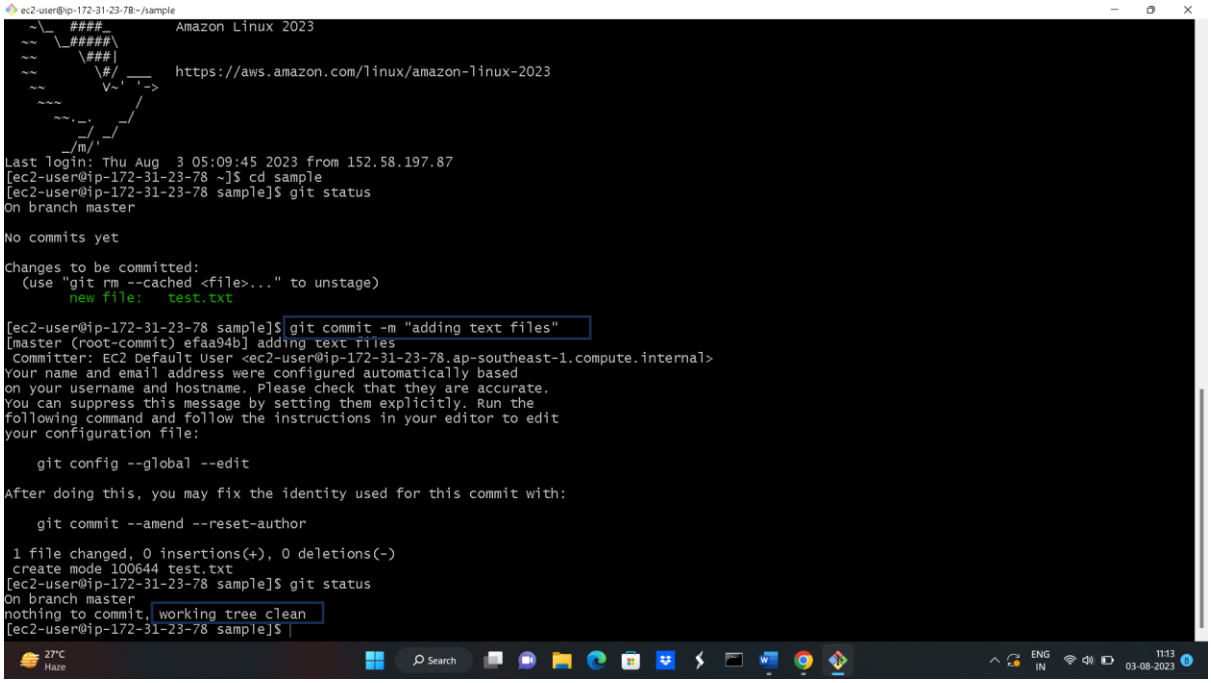
LAB-2

CREATE REPO IN LOCAL MACHINE

- Here iam going to create one folder(directory) and named it as sample and initializing this folder by using git init sample command.
- In the next step go inside the folder and check the status by git status command.
- Now iam going to create one test file using touch command and run git status command once again it shows one untracked file that means file is not tracked by git.
- In order to track your git file by using you need to add this file into staging area by using the command git add filename. now your file is tracked by git.



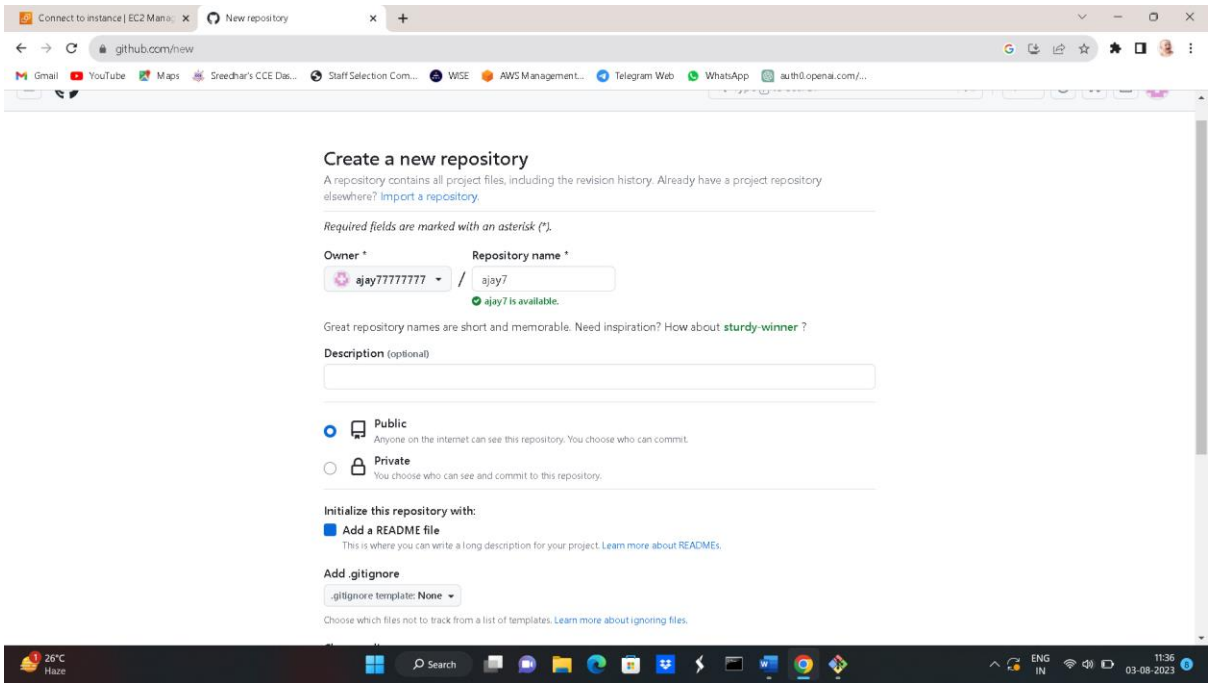
- Now I am committing my changes from staging area to local repository by using command git commit -m “message”.
- Run git status once again it will show that working tree is clean



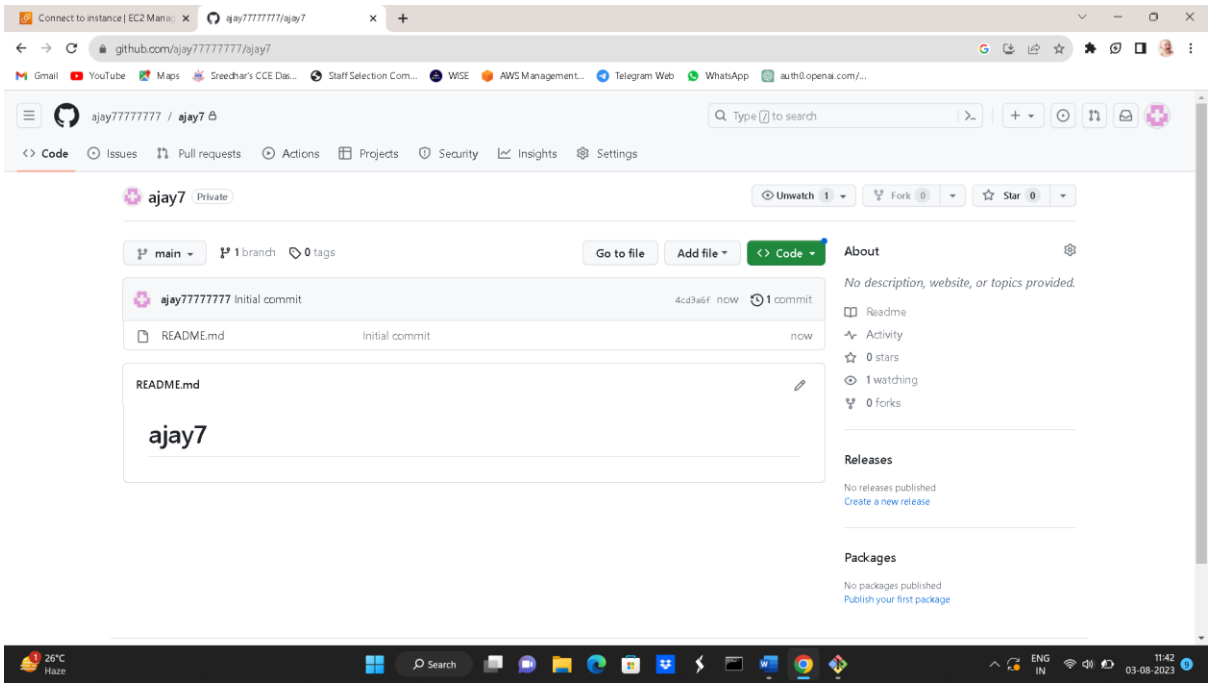
LAB-3

CREATING REPO IN REMOTE LOCATION -GITHUB

- In remote location github iam creating one new repository by providing repository name and make bucket private and initializing the repository by adding README.md file and click on create repo.



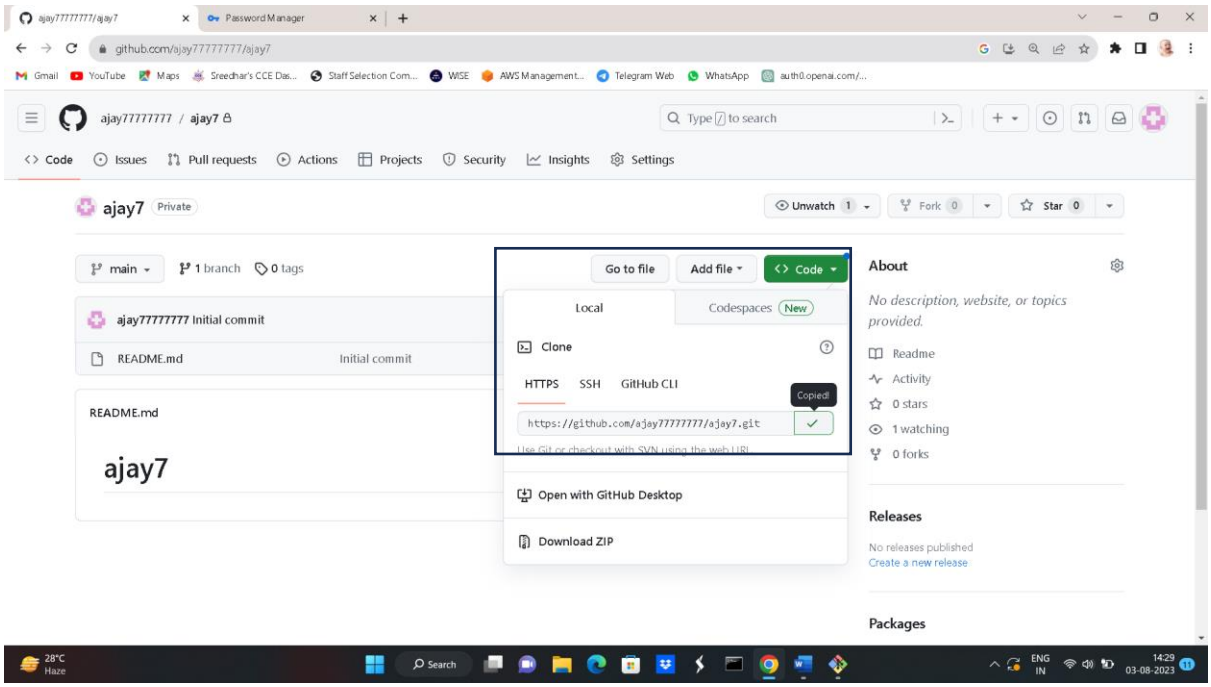
- My repository created successfully.



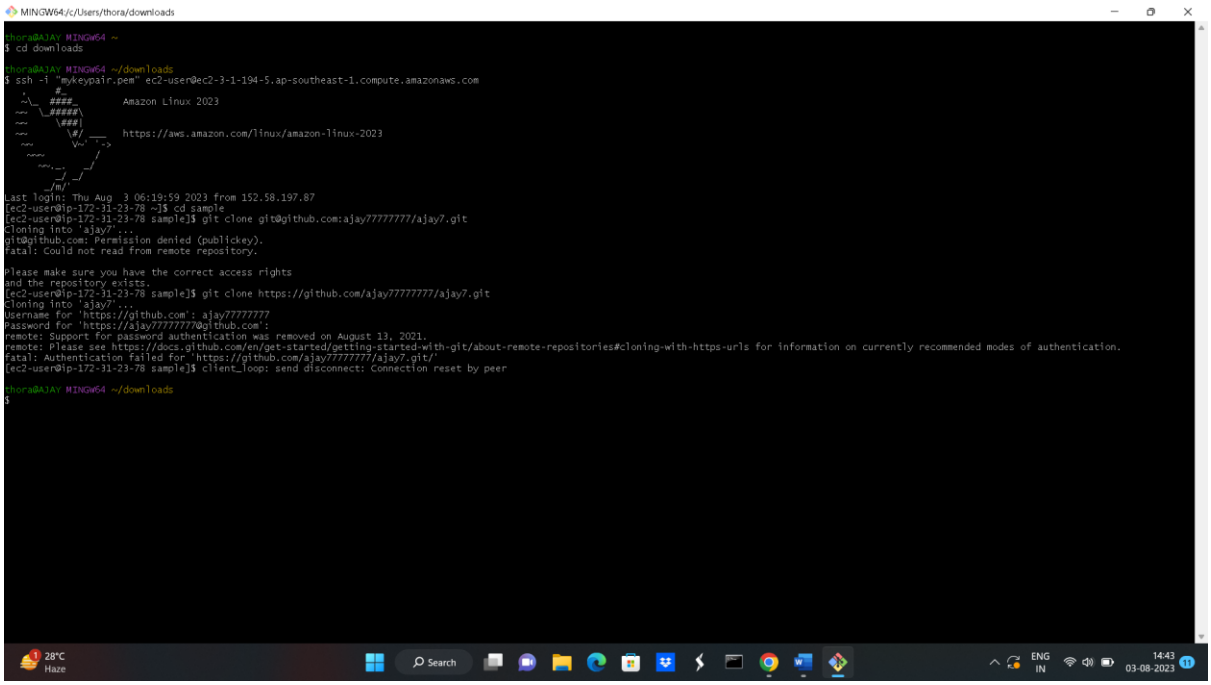
LAB-4

WORKING WITH REMOTE REPO

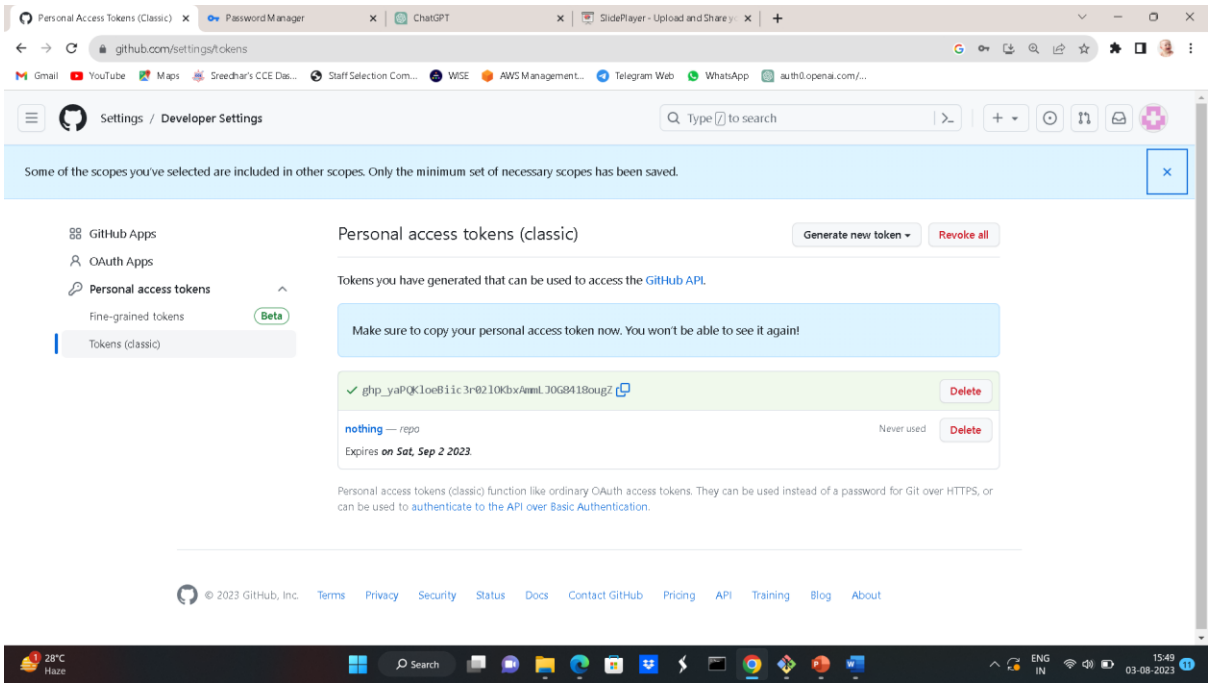
- Copying the URL of the repository from the GitHub repo.



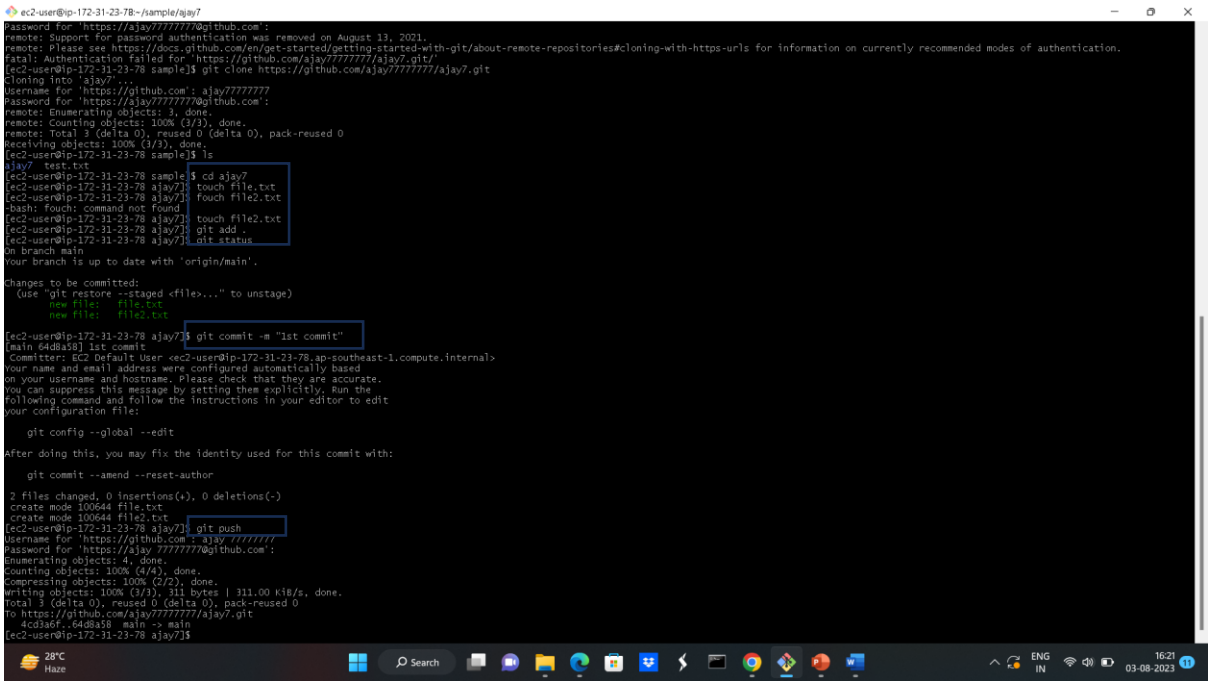
- Clone the repository by using repository git clone URL.



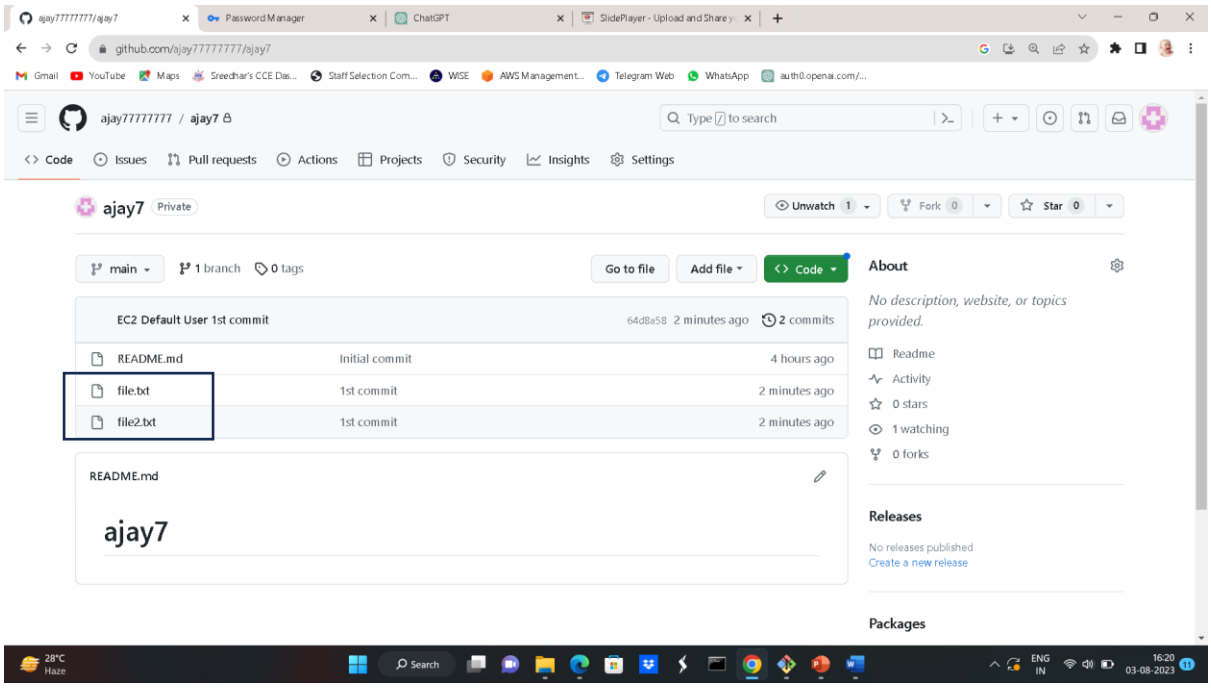
- Now we need to create the personal access token to work with repo
- Click on user icon → goto settings → developer settings → personal token → click on generate token → provide a note → set expiry date and select the scope repo→ generate token → copy the token



- Clone the repository and again give username and in the password you have to give the token.
- After cloning into the repository go inside the repository and created two files by using touch command
- Stage this changes by git add command and commiting this files into local repo by using git commit command.
- Push this changes to remote repo by using gitpush command.



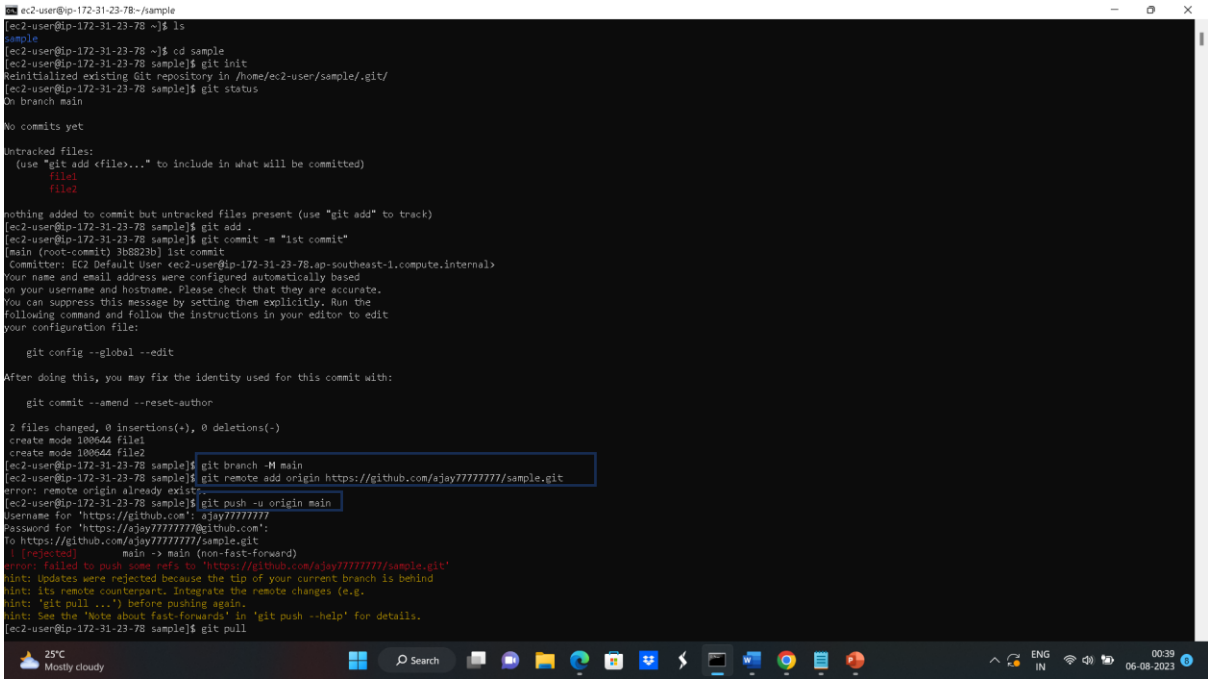
- Goto remote repo and see we will be able to see the changes.

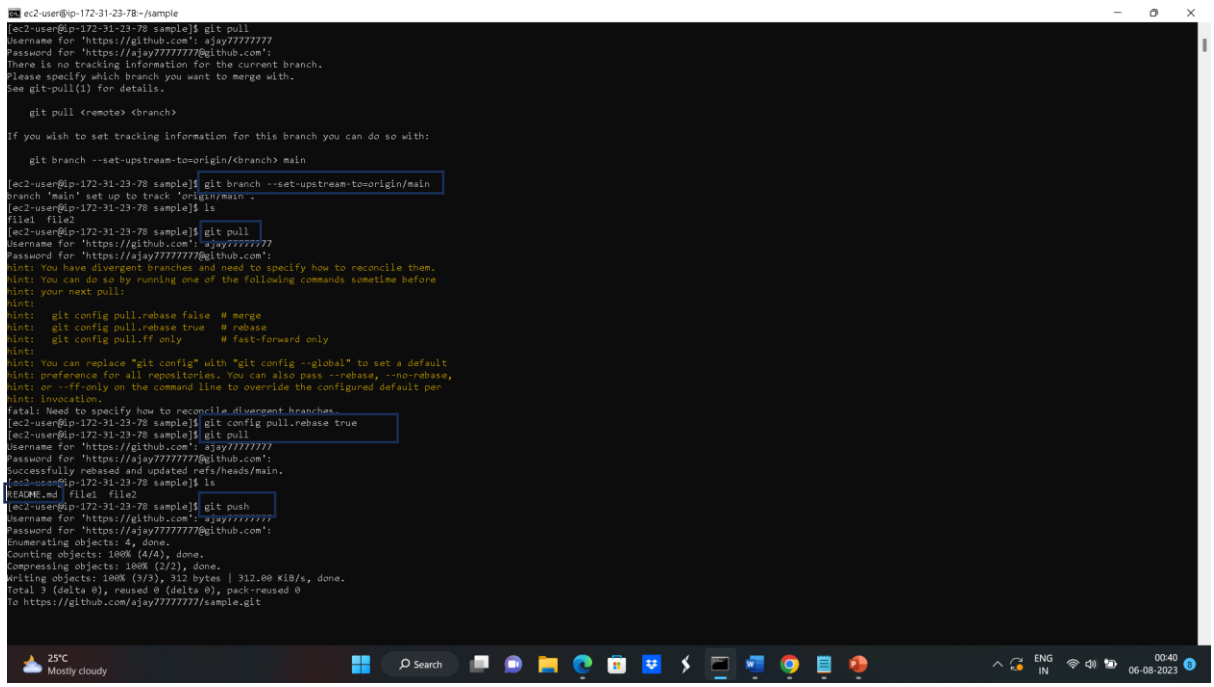


LAB-5

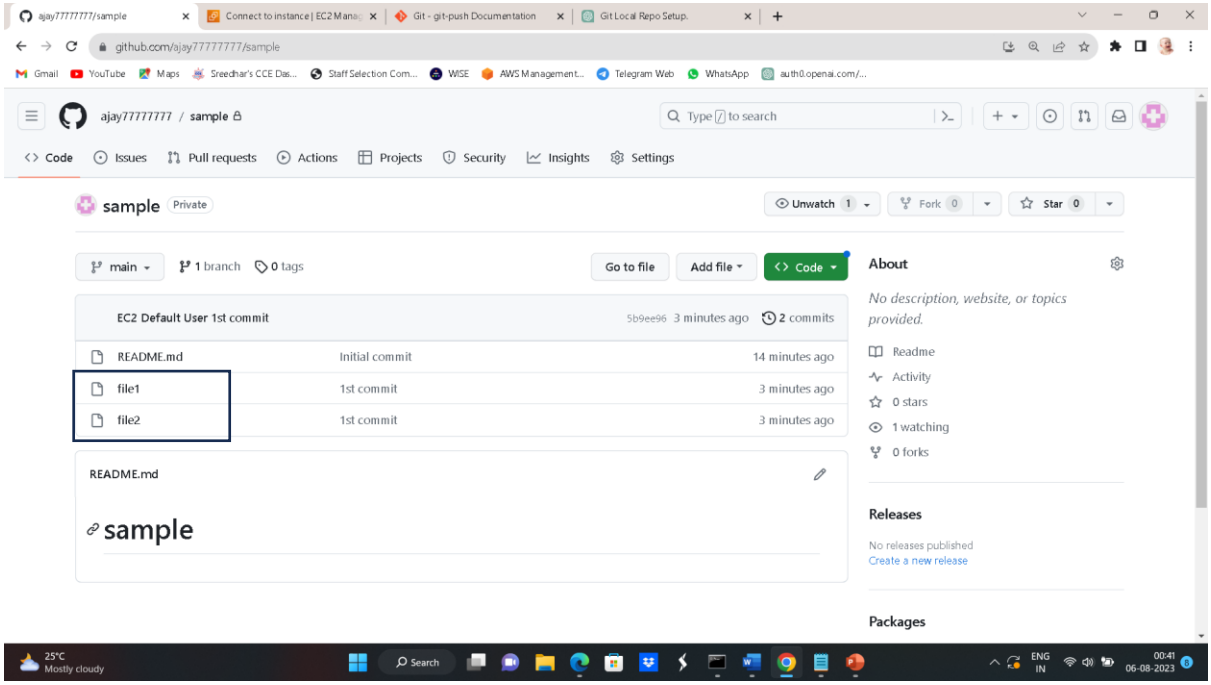
PUSHING A LOCALLY CREATED REPO INTO GITHUB:

- Created one repo in our local machine and initialize it locally.
- Create one remote repo with the same name as local repo in github and do not initialize it.
- Run the below commands.
  - git branch -M main (to change the name of branch as master branchis known as master branch)
  - git remote add origin <URL>
  - git push -u origin main (to push your local branch to remote repo)





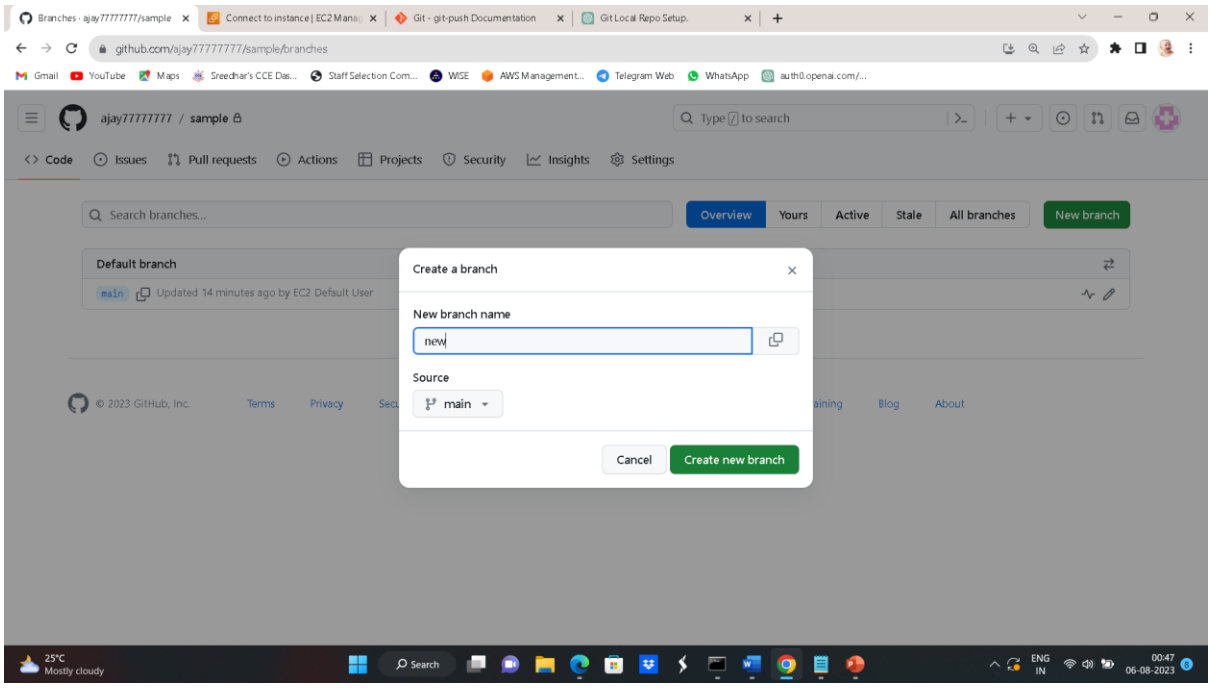
- Here we can see the changes in local repository pushes added in remote repository.



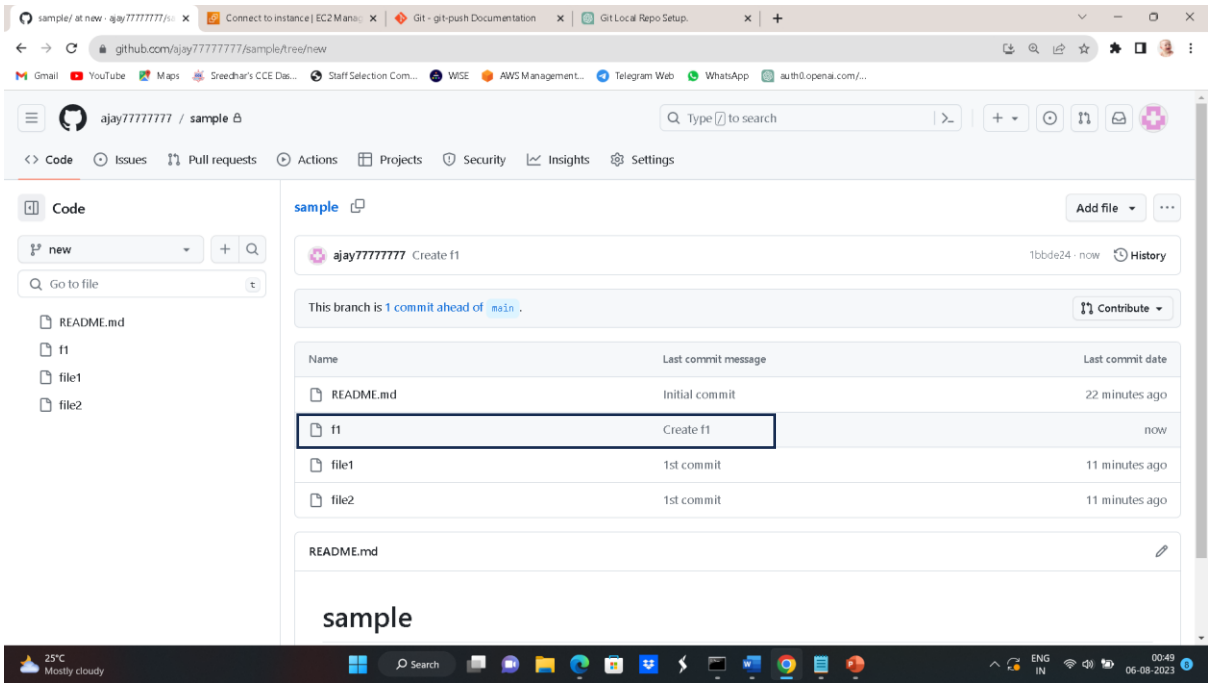
LAB-6

CREATING A NEW BRANCH FROM YOUR MAIN BRANCH

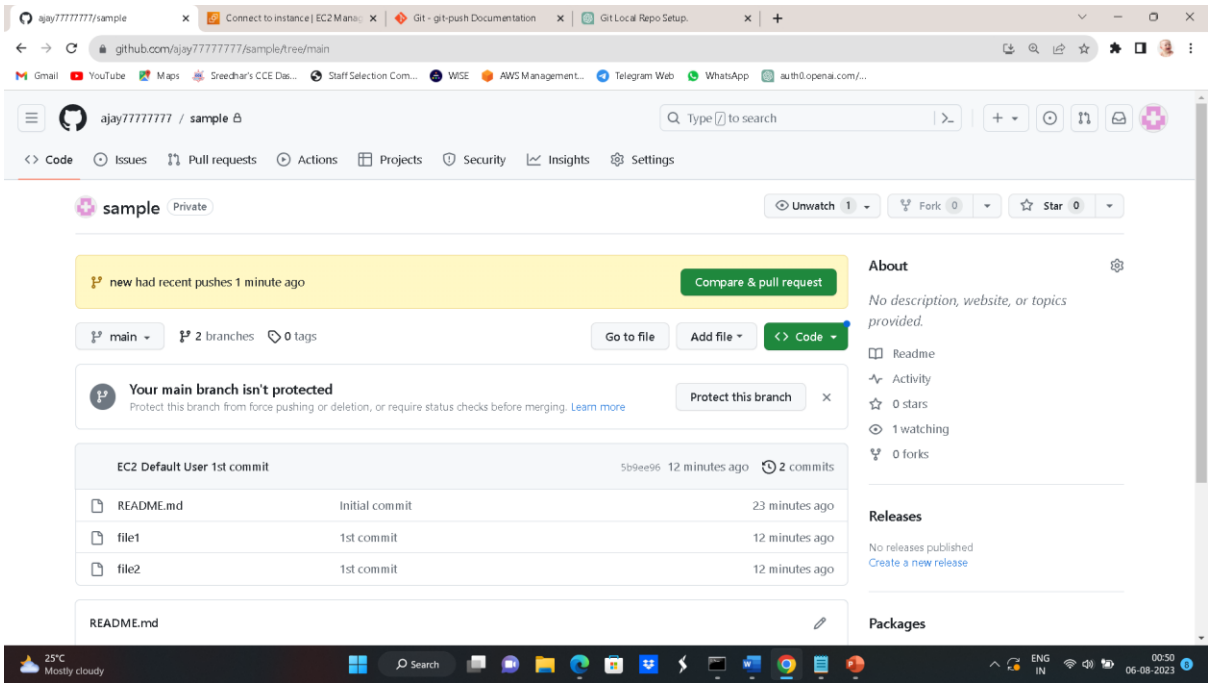
- Creating a new branch



- In the created branch added one file.



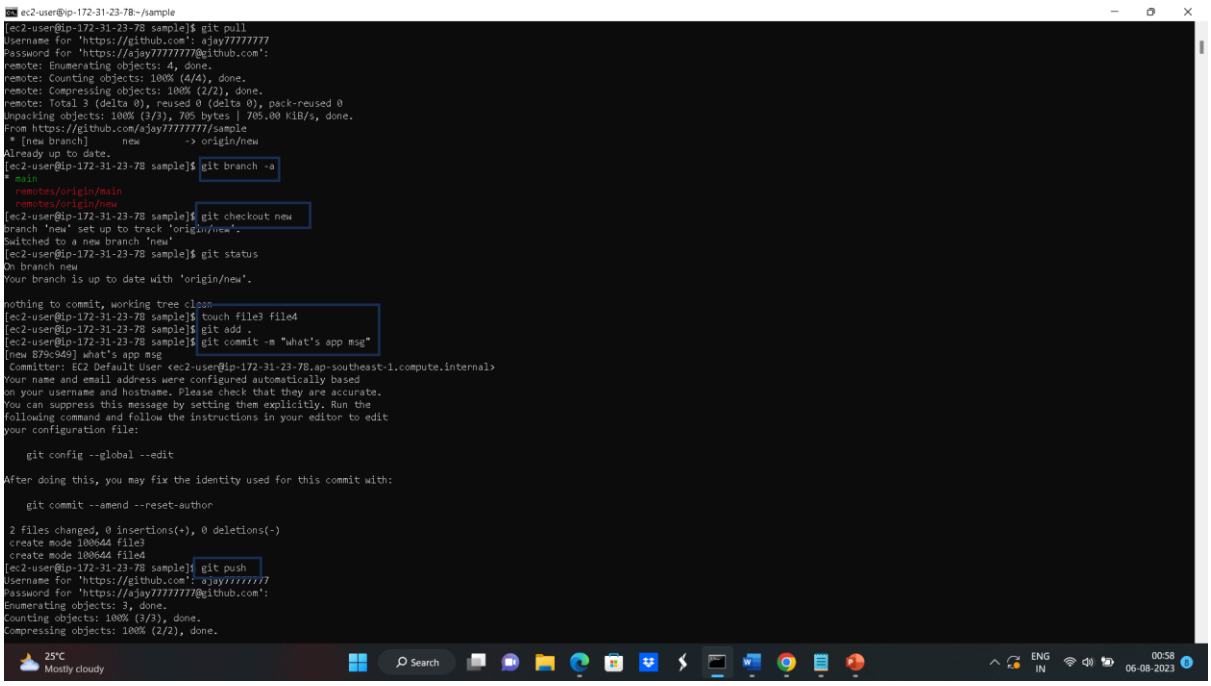
- Here you will see that your changes are only applied to your new branch but not to the main branch.



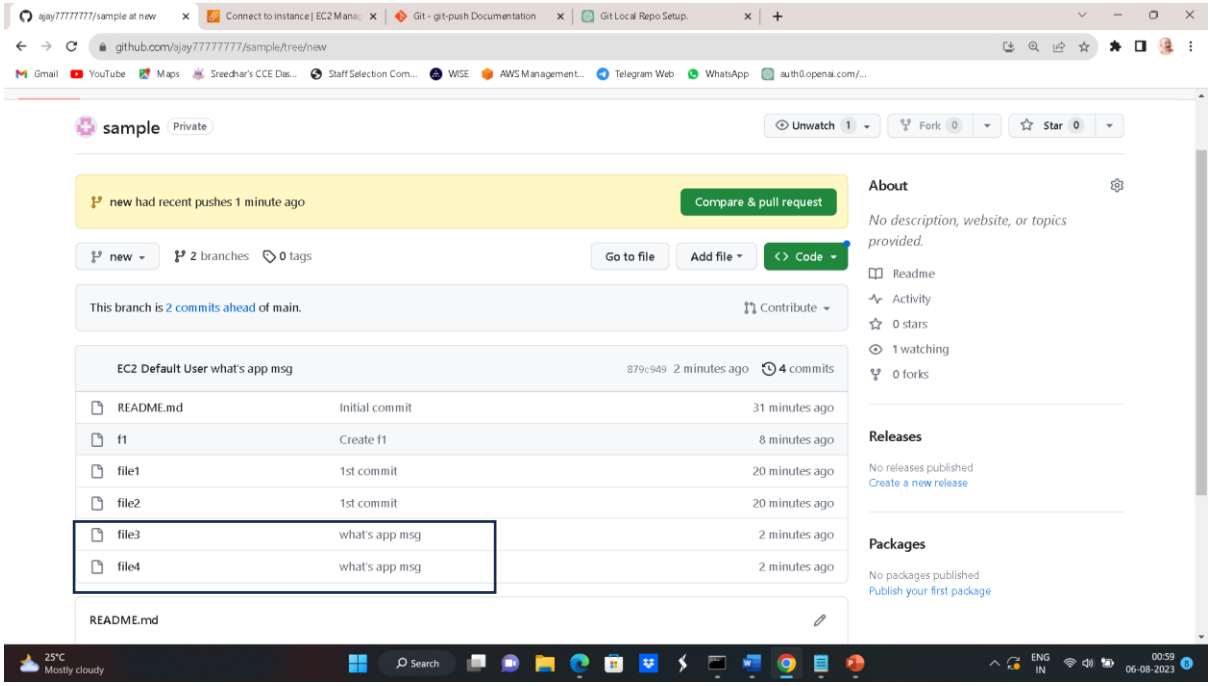
LAB-7

PULL ALL THE BRANCHES IN YOUR LOCAL MACHINE

- In the local machine pull all new changes such as branches from the remote location.and list the branches by using git branch -a command.
- Switch into the created branch by using git checkout <branch name> command. and create some text files and add to staging and commit it.
- Push the changes to remote repository by using git push command.



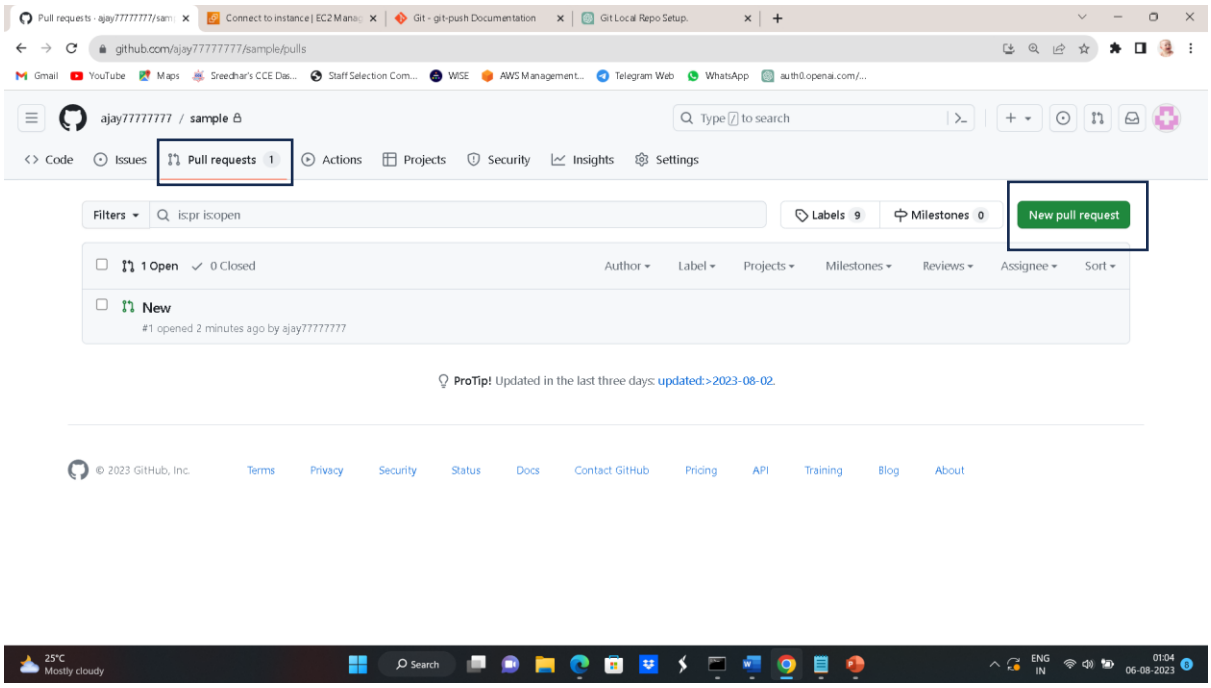
- Here we see that the new changes are only available in your feature branch but not in the main branch.



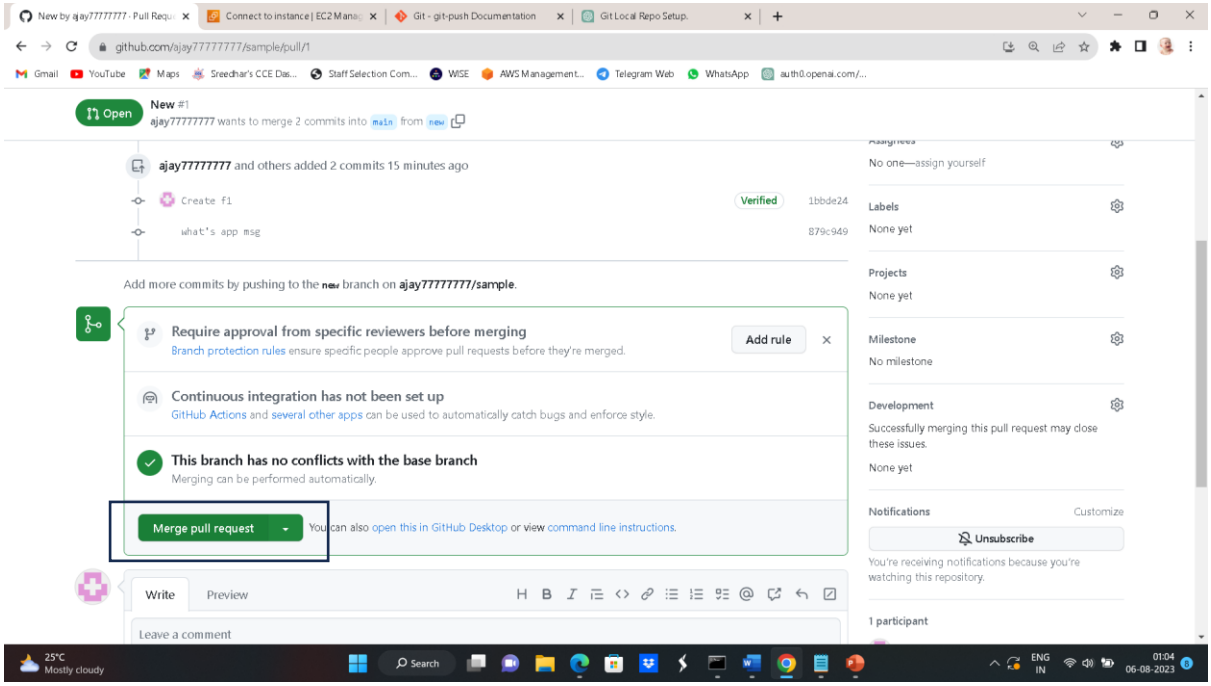
LAB-8

MERGE OUR FEATURE BRANCH WITH MAIN BRANCH

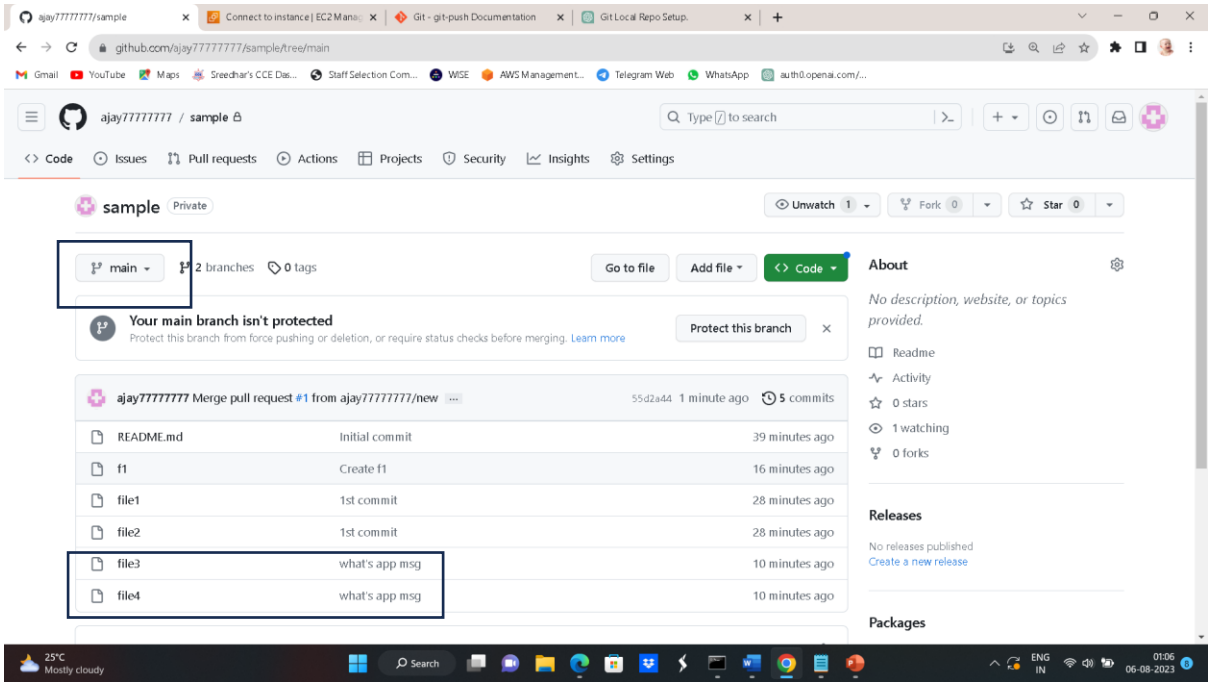
- In the git hub repository go to the pull request and create pull request in that put your main branch and feature branch in the respective blocks and click on create pull request.



- Click on merge pull request.



- Now go to your code in main branch and see changes are now visible here.



LAB-9

GOTO LOCAL MACHINE AND SEE CHANGES AVAILABLE ONLY IN MAIN BRANCH

- Goto your local machine and switch into the master branch and run the git pull command .
- Here we see that the new changes are only available in your main branch.