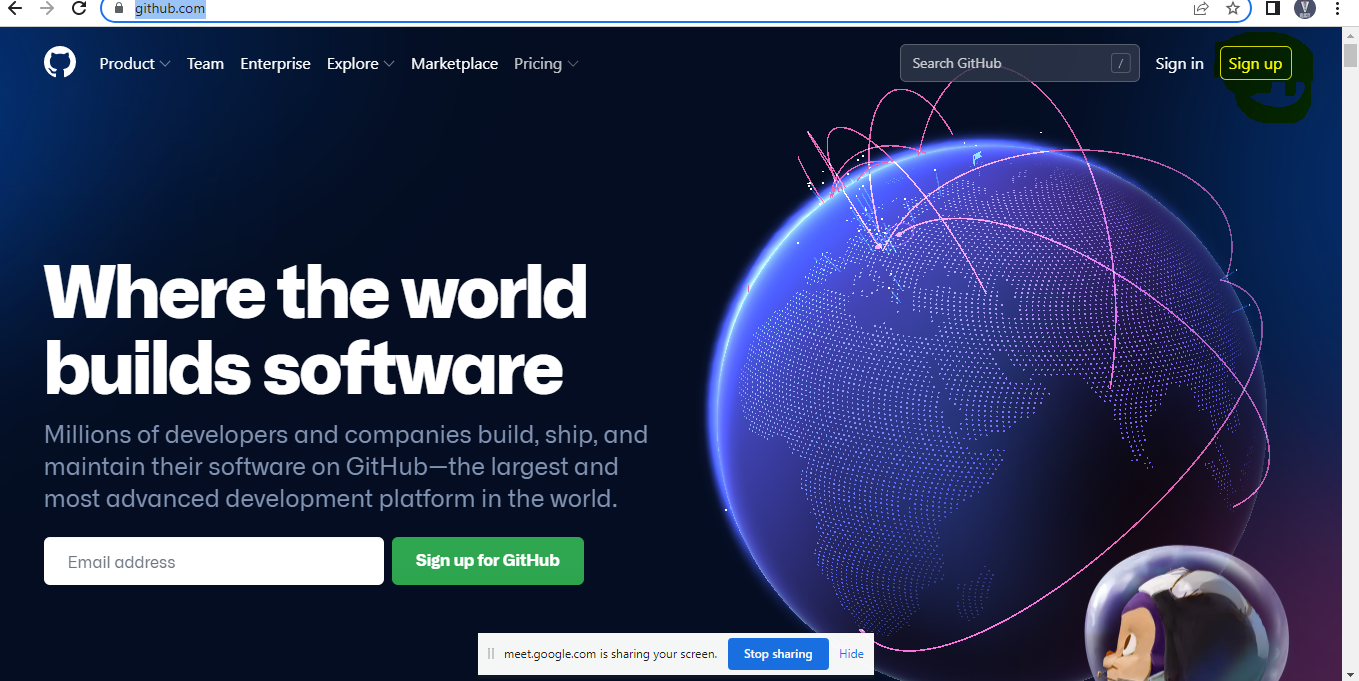
Git Account creation:

1. Go to Github.com

<https://github.com/>

2. Click on Sign up:

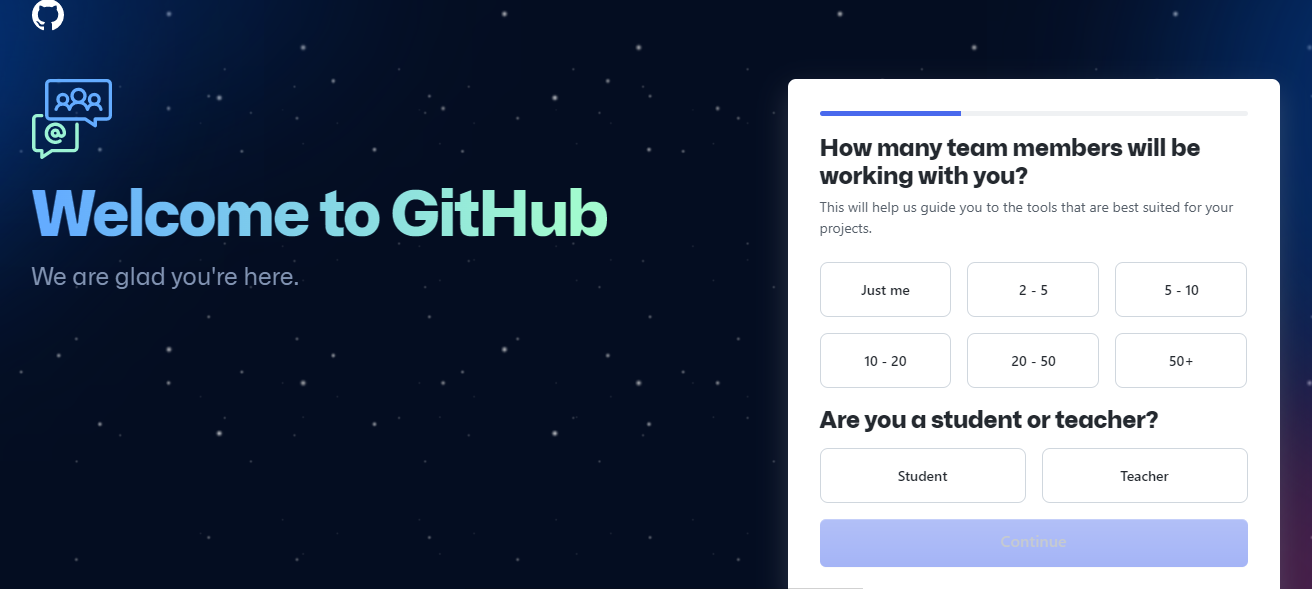


3. Fill the information



4. Enter the verification code

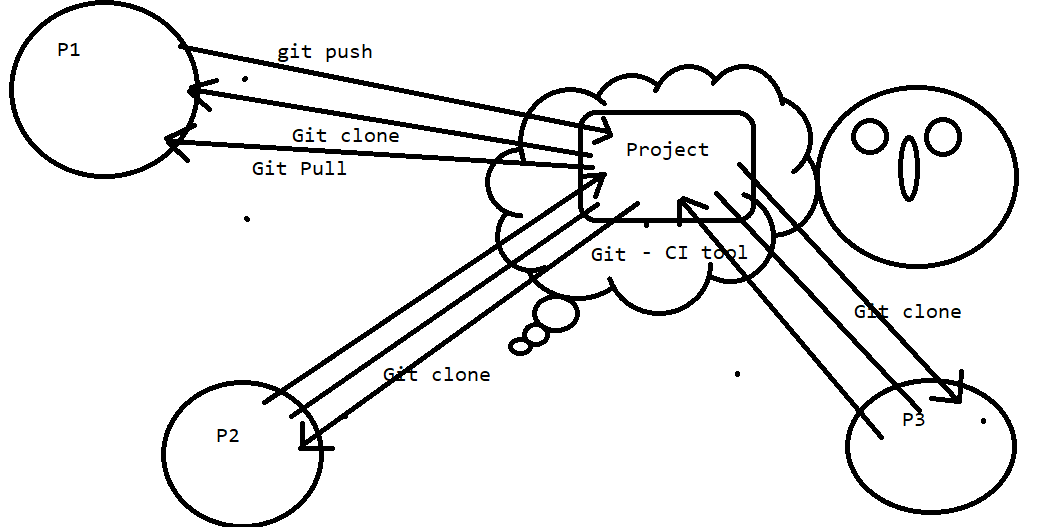
5. Choose any option from the below image:



5. Choose Git for Automation CI / CD

6. Choose continue for free and account will get created.

Local repository: A repository(Project) which is available at local system and it is ready to go to the git server.



Commands in git:

1. Add: to add the files which wants to put into the git server.

2. commit: Through this command we are moving our code to the staging area.

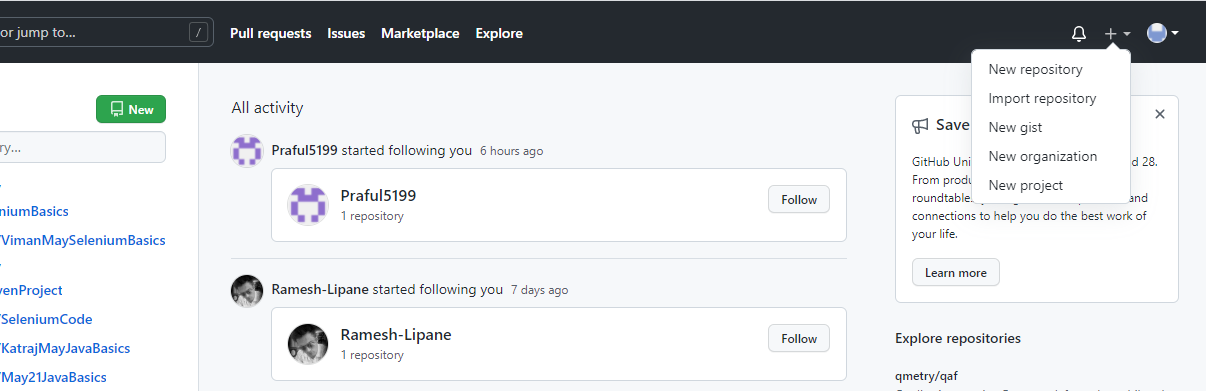
3. push: to push the code to the git.

4. pull: To take the latest updated code into the system we have to perform pull command.

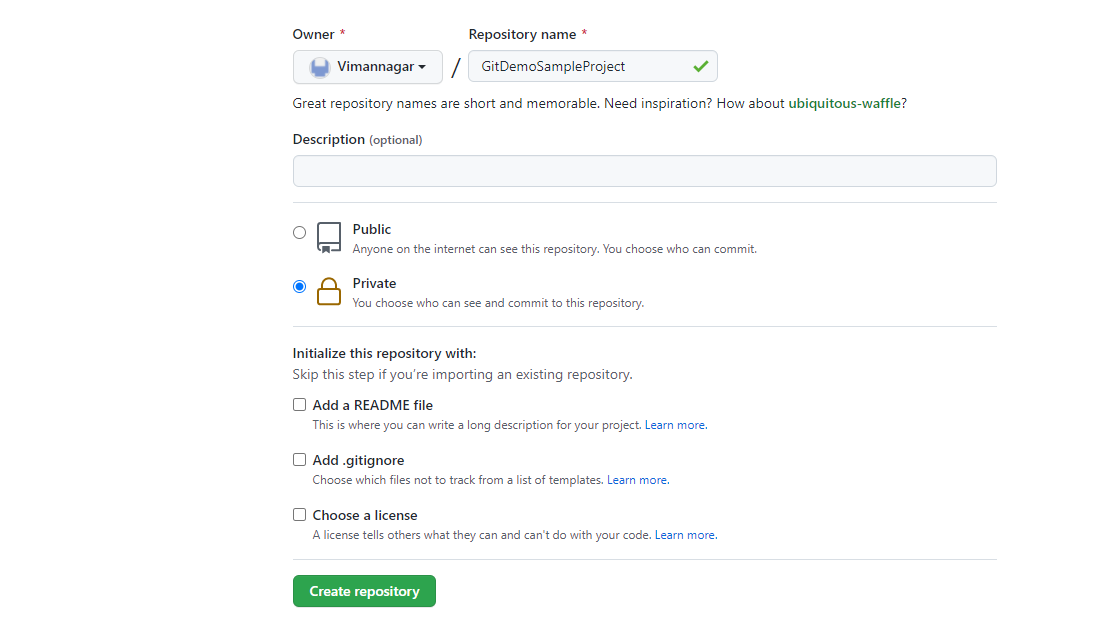
5. clone: This command is used to take the code from git repository to local repository for the first time.

Git configuration:

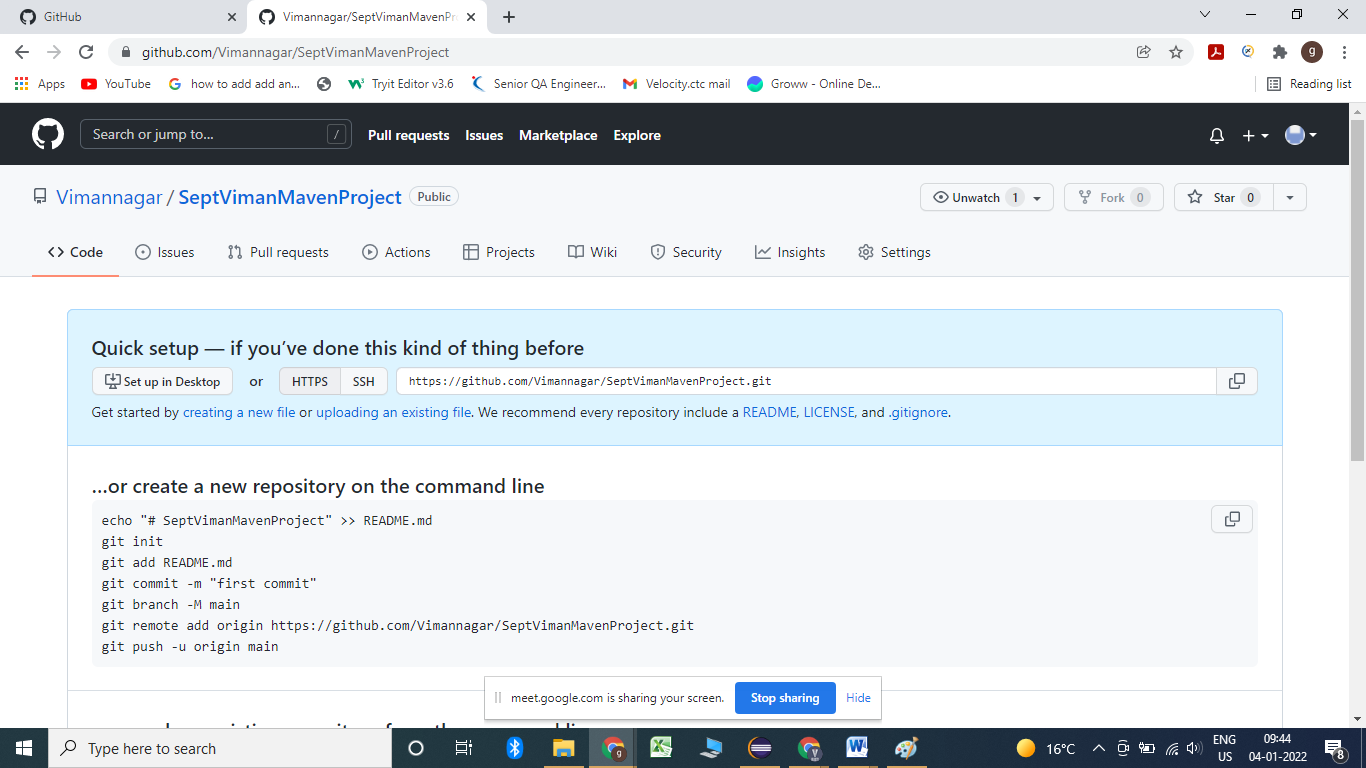
Step 1: Create a new repository by clicking on ‘+’ or click on green button.



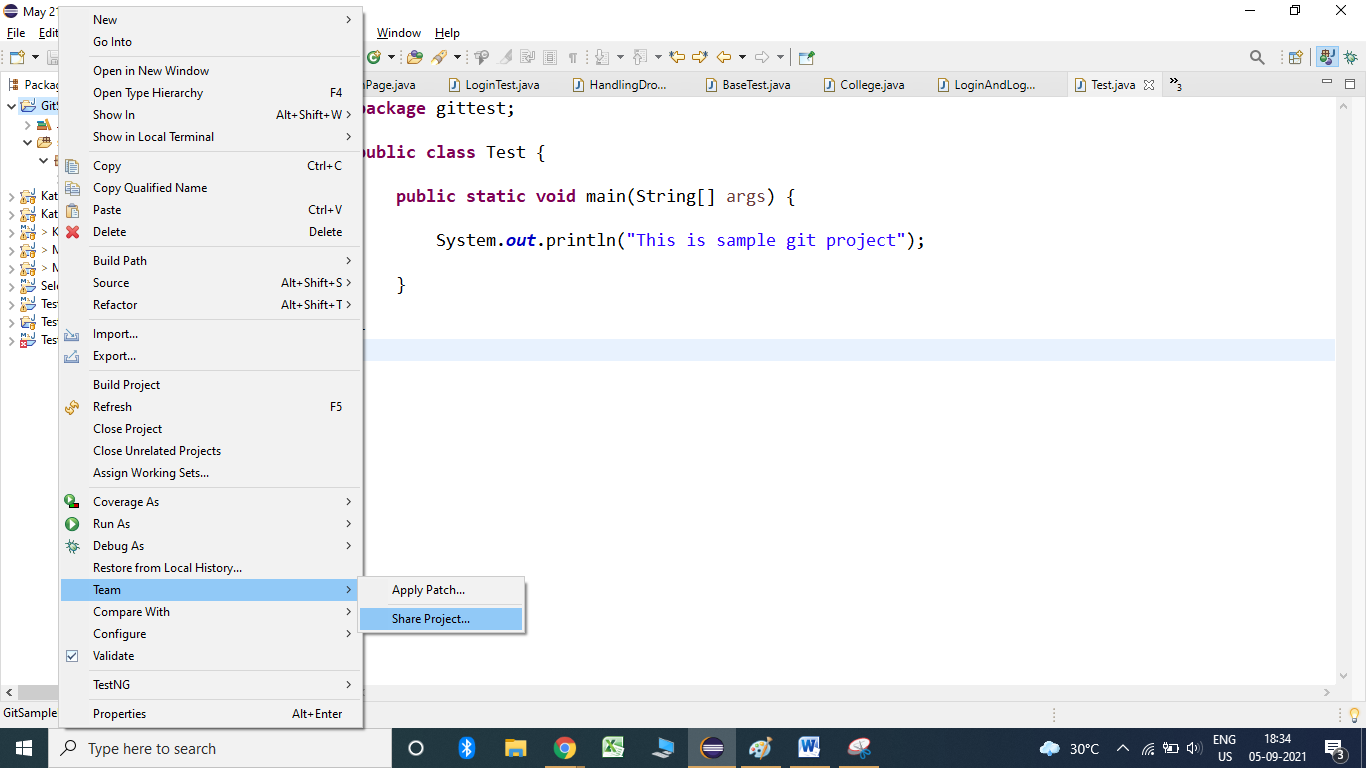
Step 2: Fill the details and click on create repository:



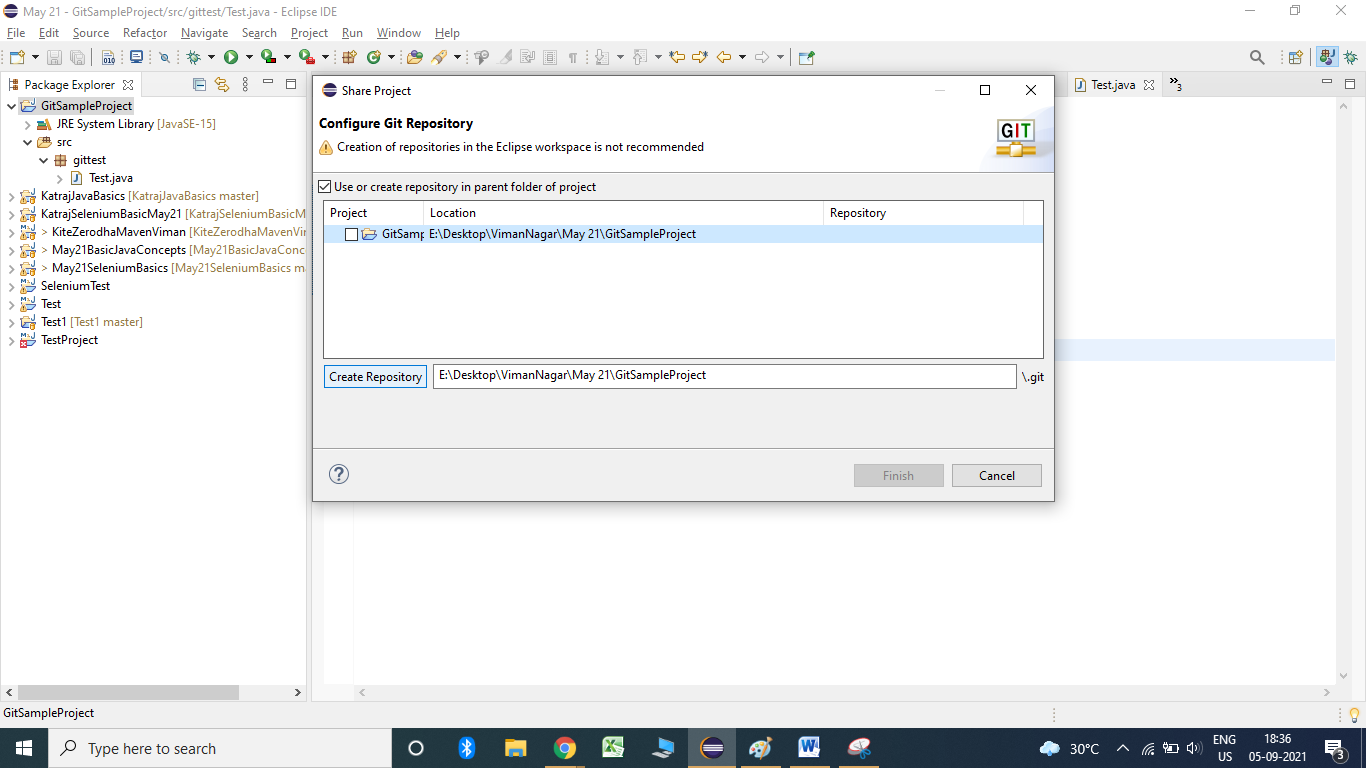
After the creation of git repository we need to copy the url



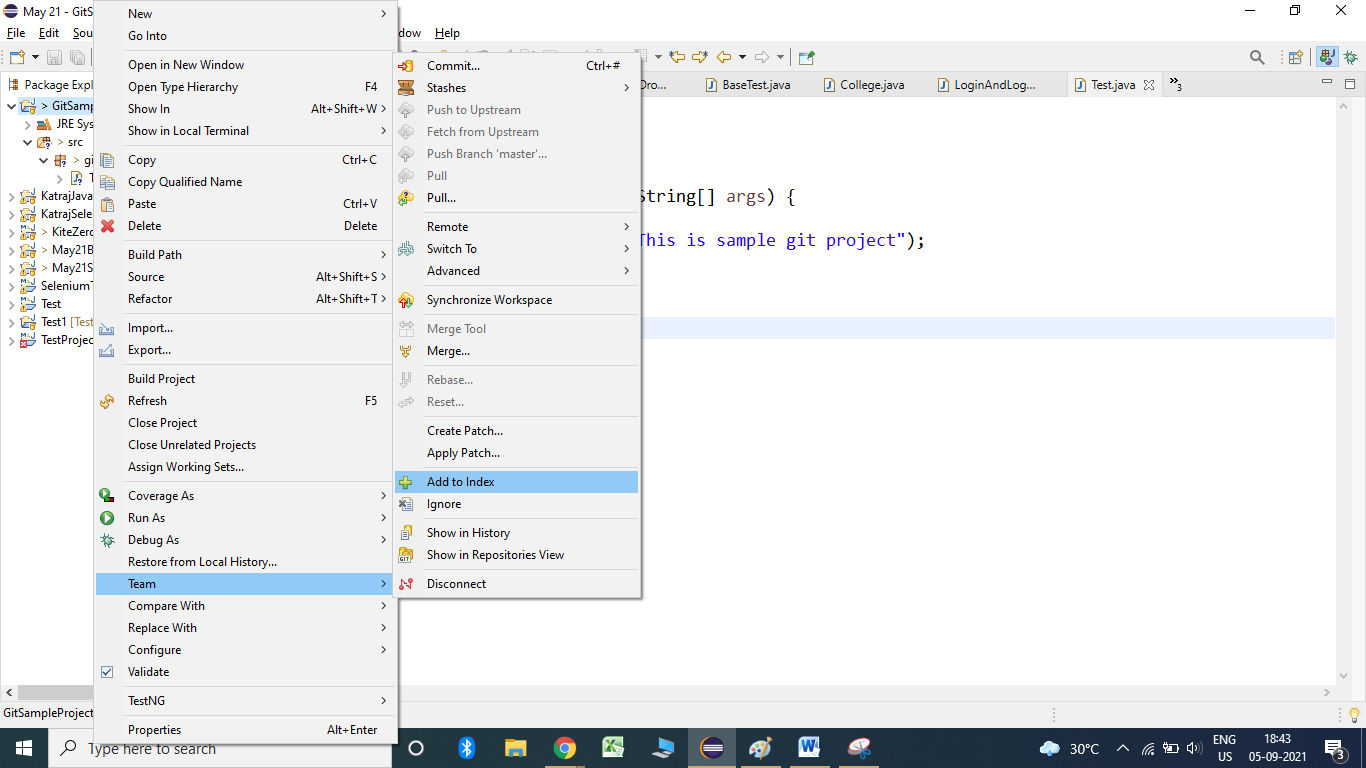
Step 3: Go to eclipse and right click on the project which we wants to push over the git server and select team --- share project.



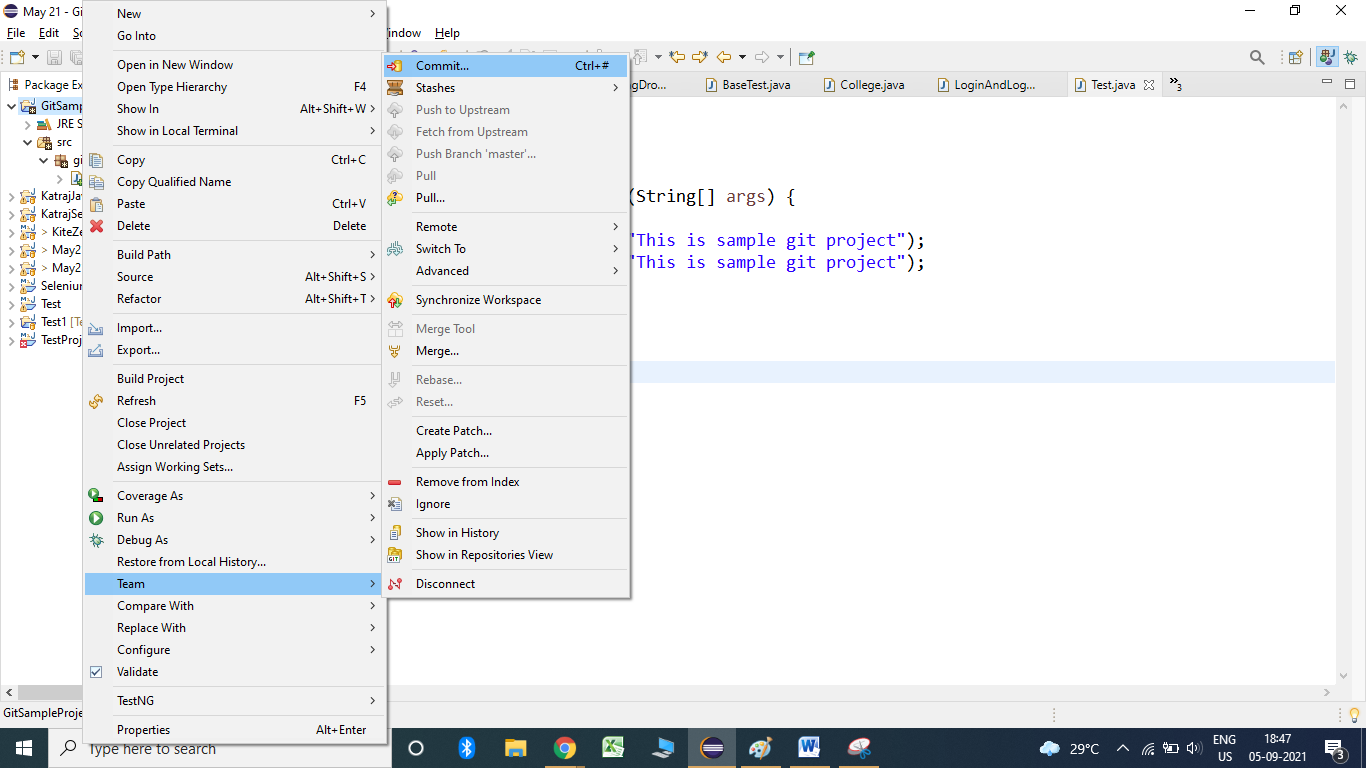
Step 4: Click on the share project click on the check box of below dialogue box and click on the project check box as well and click on create repository and click on finish.



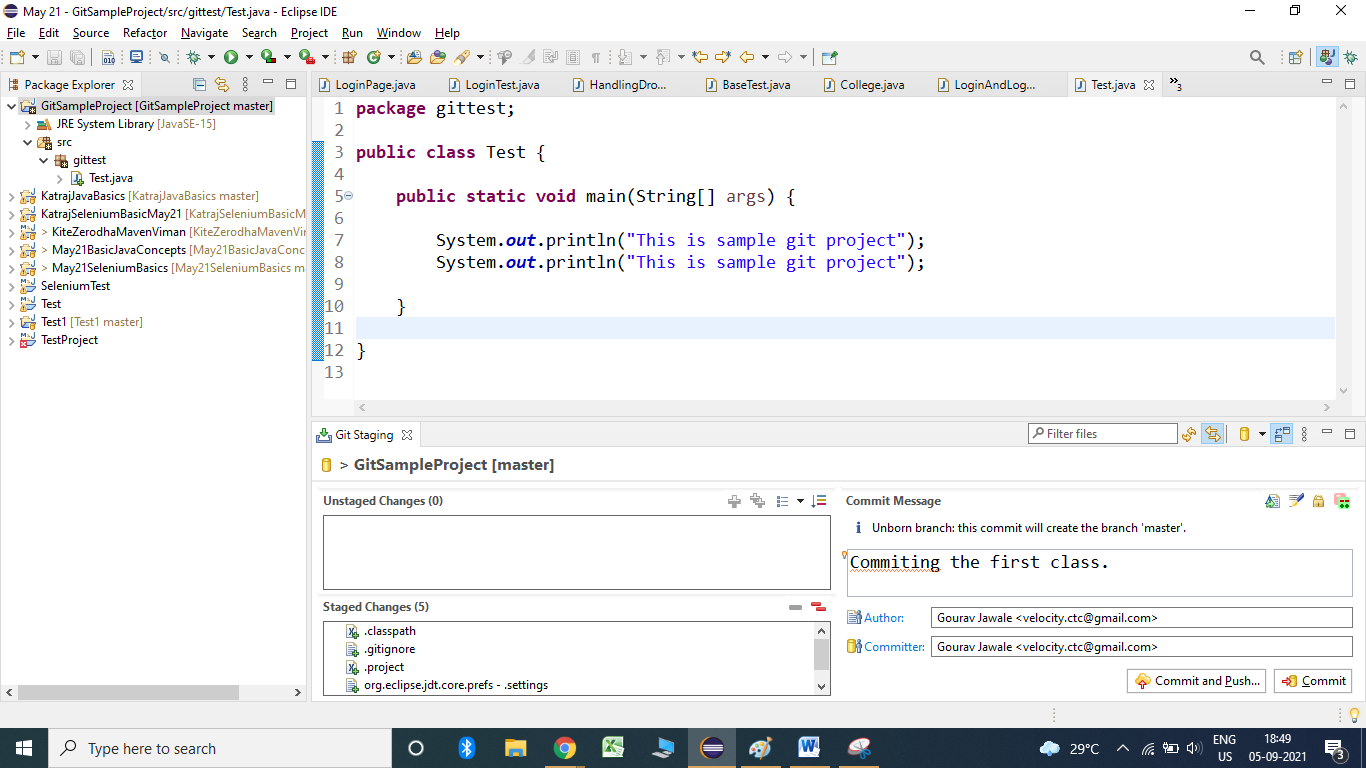
Step 5: To add the newly created files/updated files we have to perform add operation.



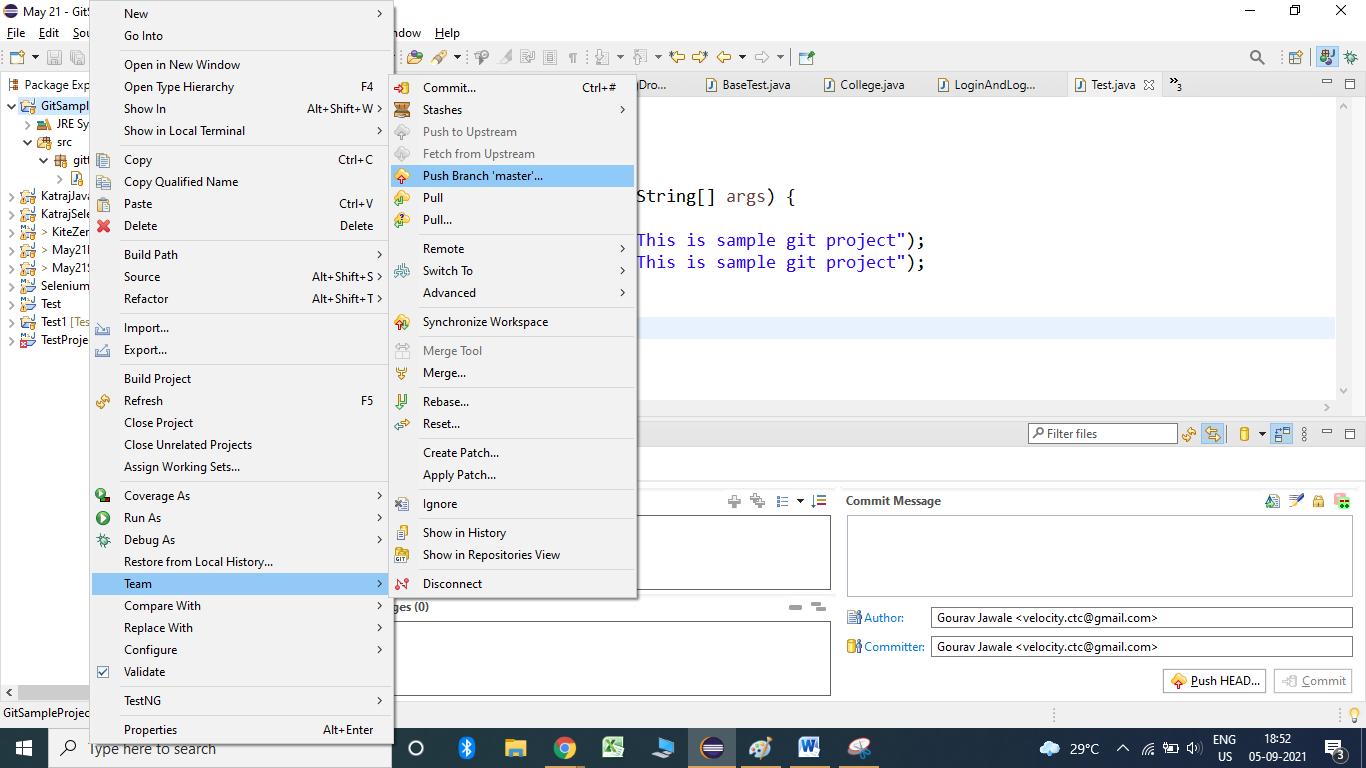
Step 6: to commit the project right click on project --- Team--- commit

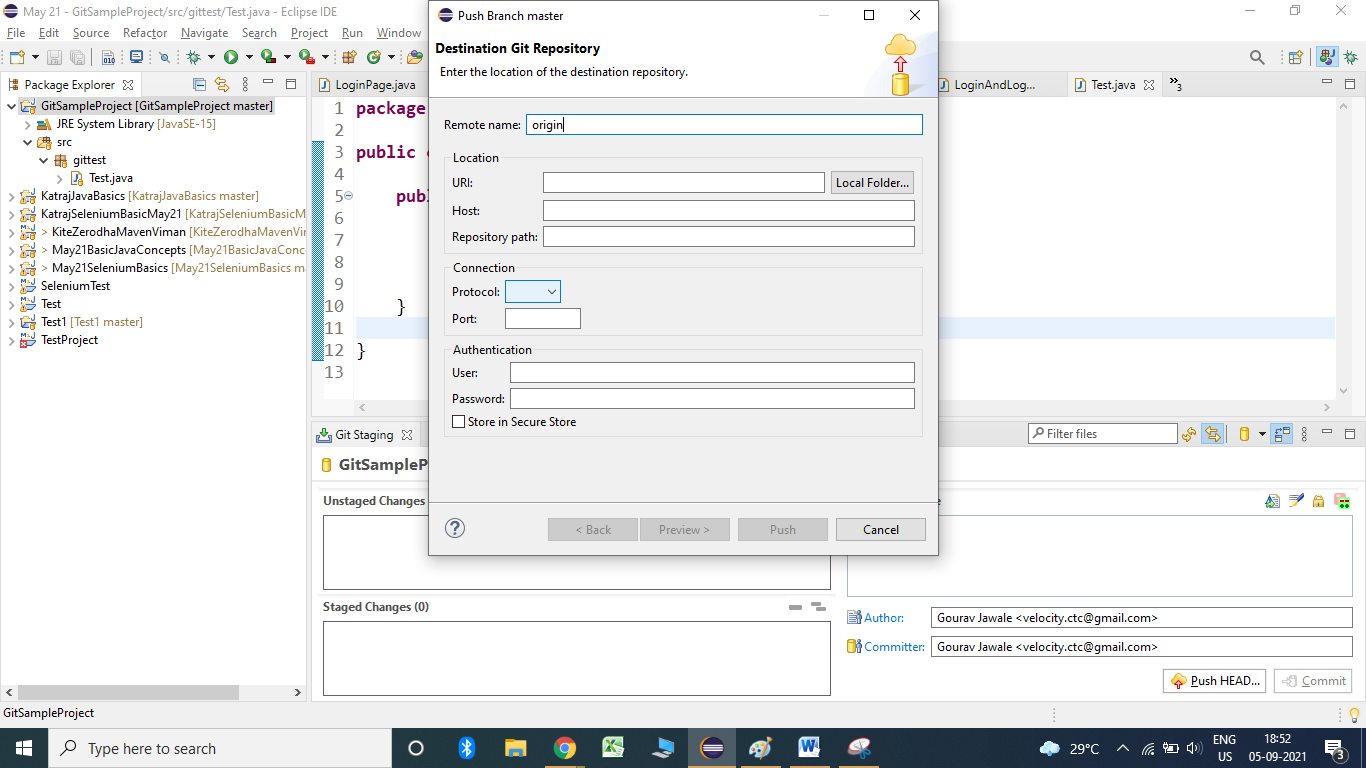


Step 7: Write down the message for the commit process and click on commit.



Step 8: To push code Right click on project --- Team--- push.



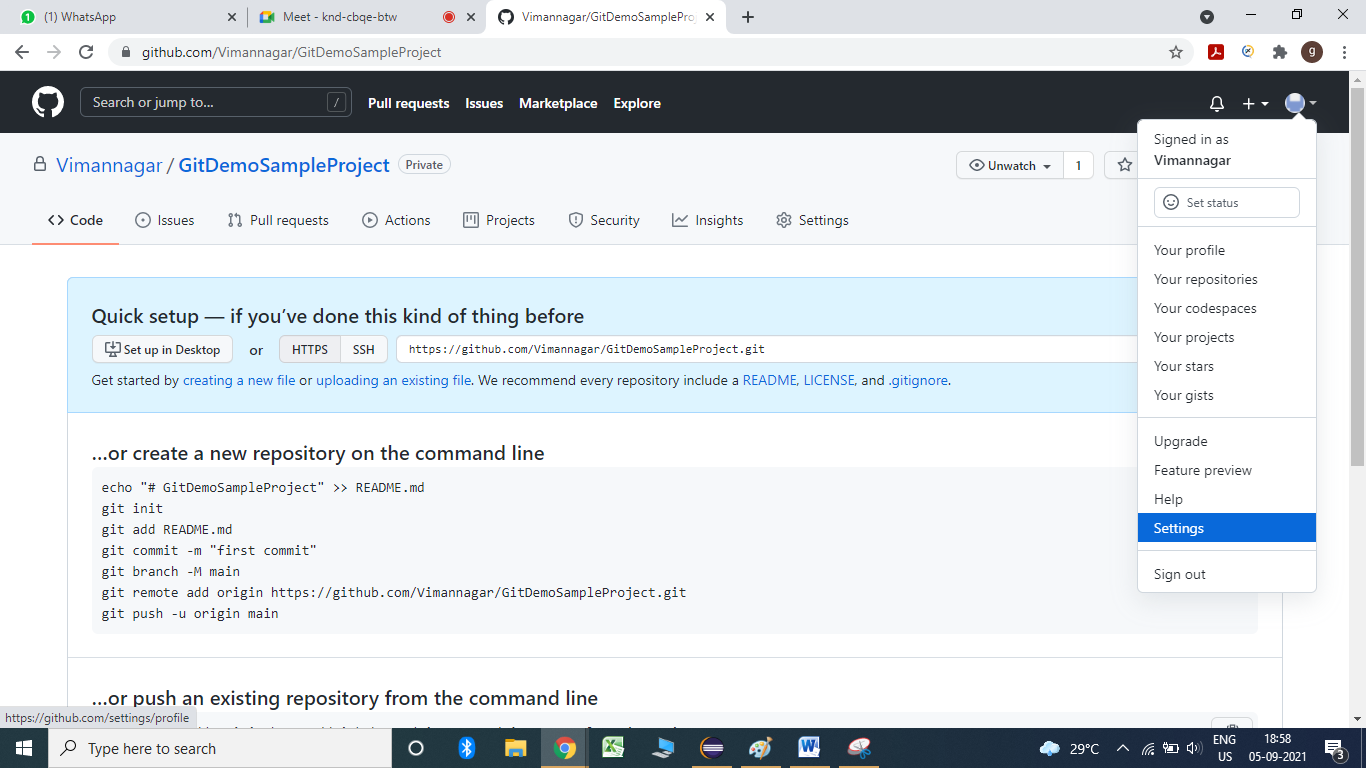


Step 9: Paste the url from the git into the URI path and enter the username into the field.

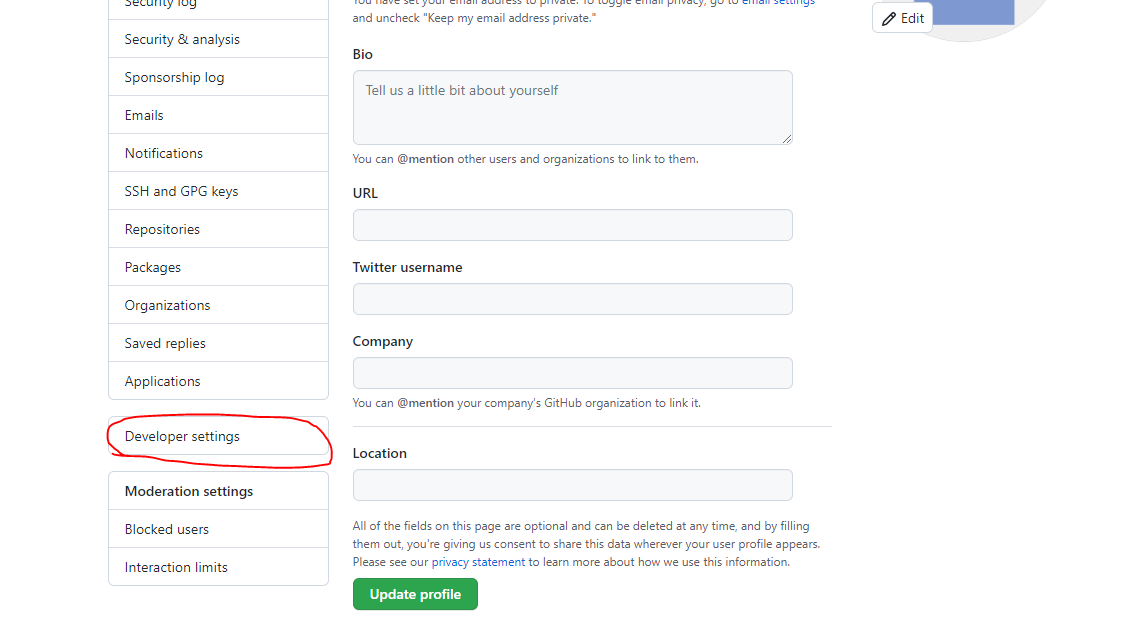
Step 10: Generate the personal access token from git account that has to be used for password.

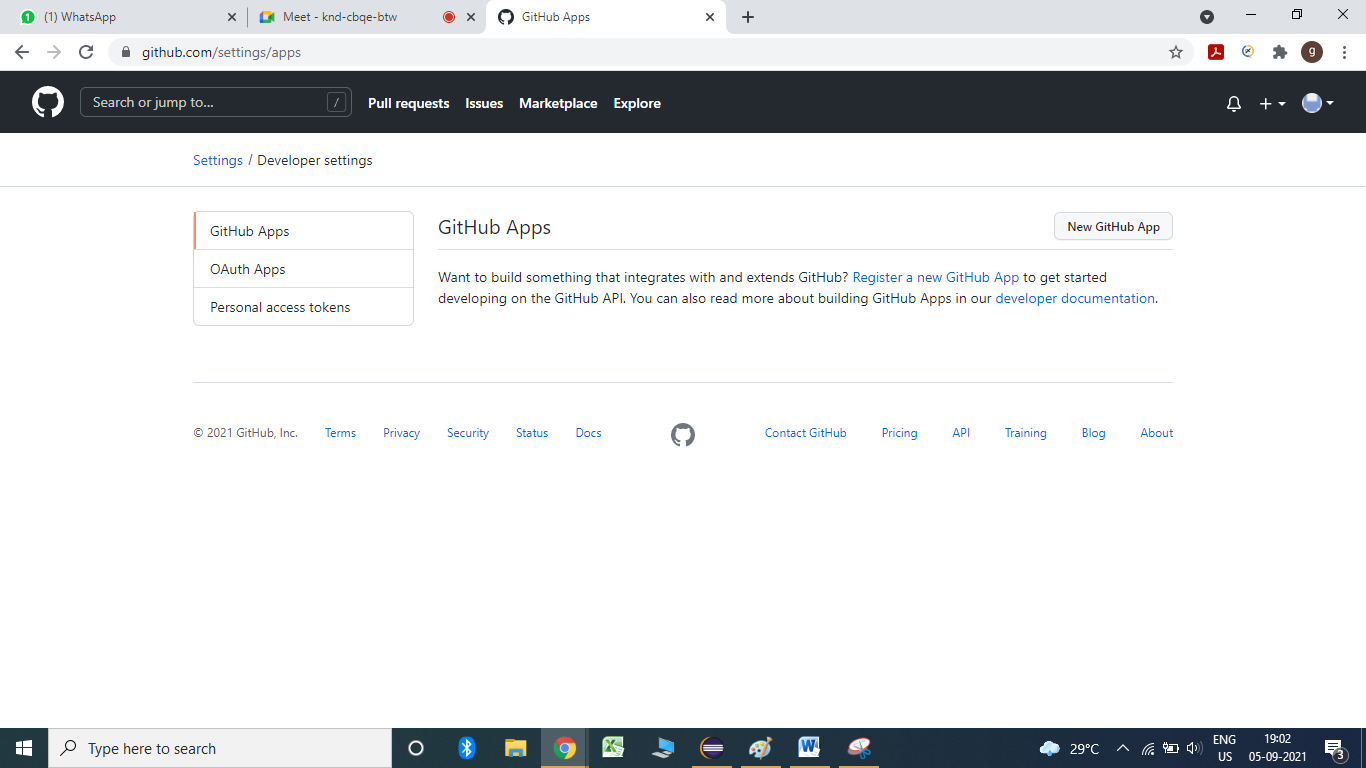
To generate the token:

1. click on the profile avatar and select settings:



2. Click on developer setting:



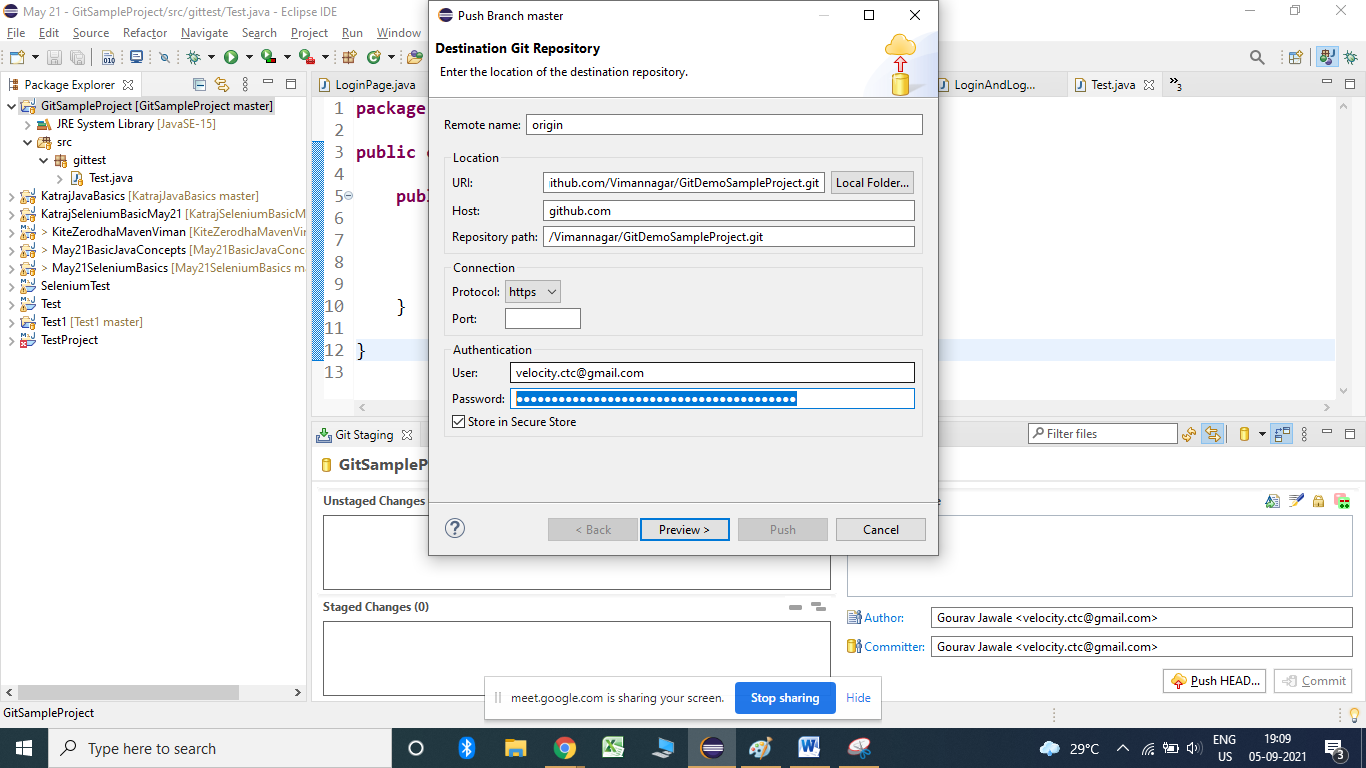


3. Click on personal access token:

4. Click on generate new token.

5. After the generation of new token right click on the project—Team—Push to branch master:

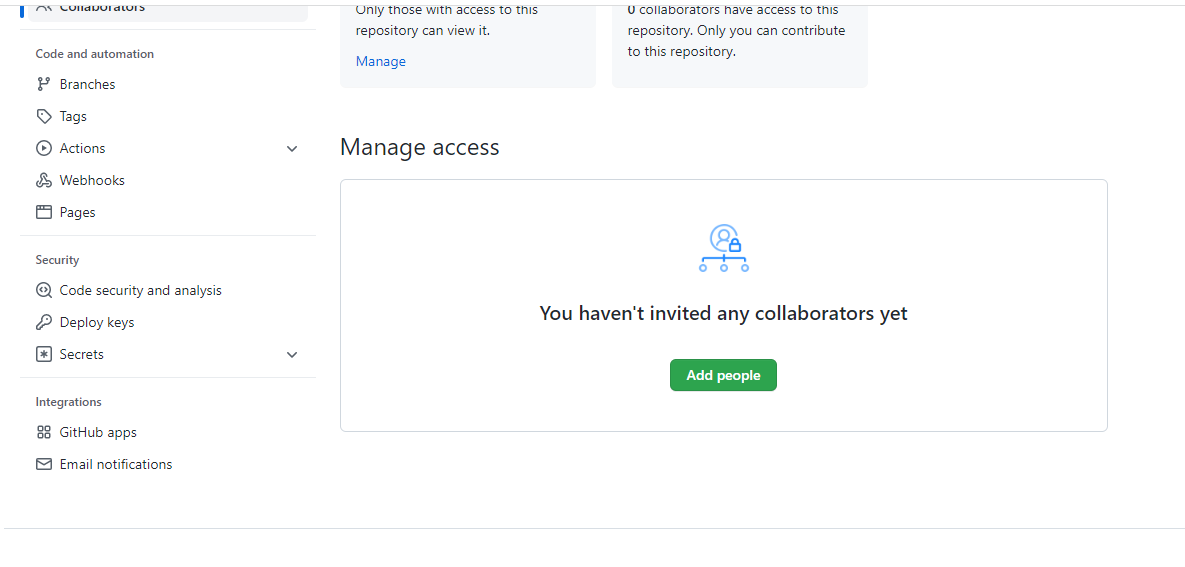
Paste the url and enter the username of git and personal access token as password followed by click on secure storage check box.



Click on the preview button and click on Push.

To provide acess to the user:

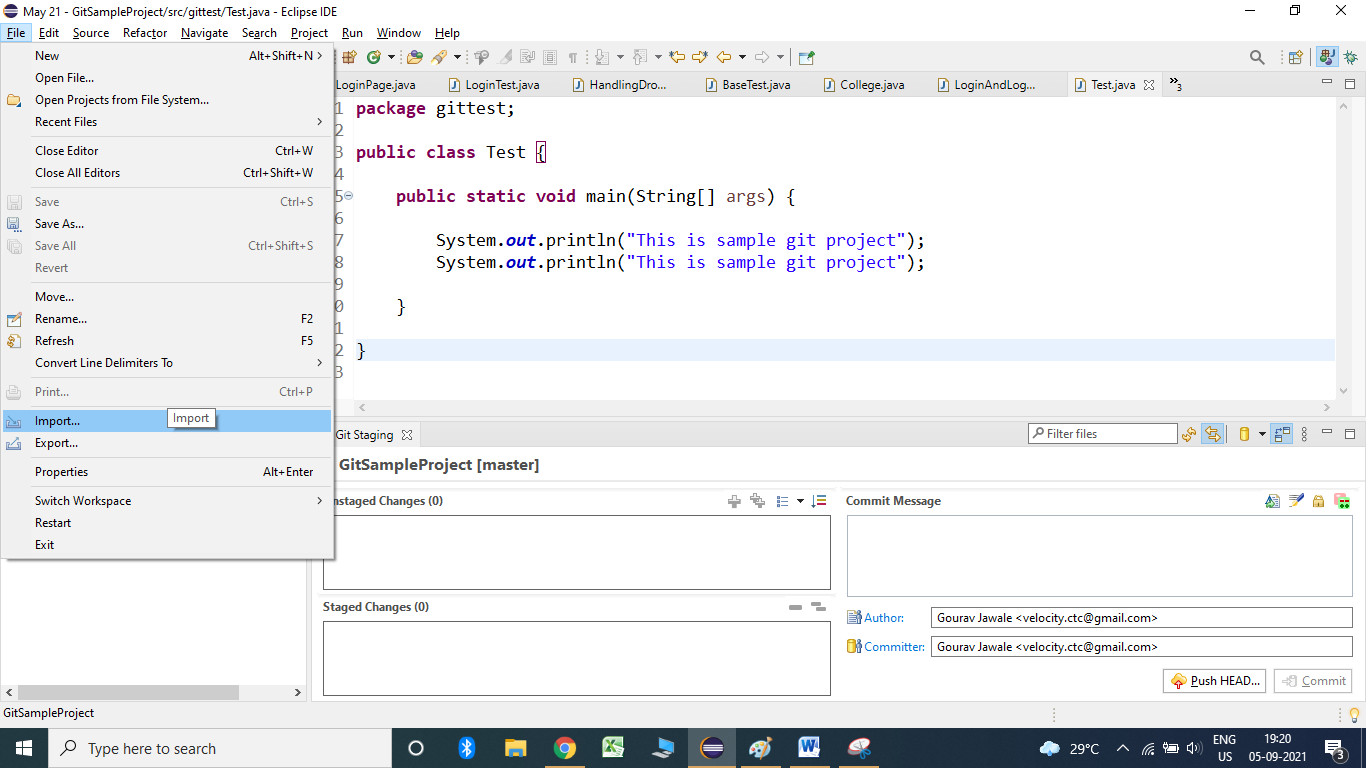
Click on settings in the git hub and click on Collaborators:

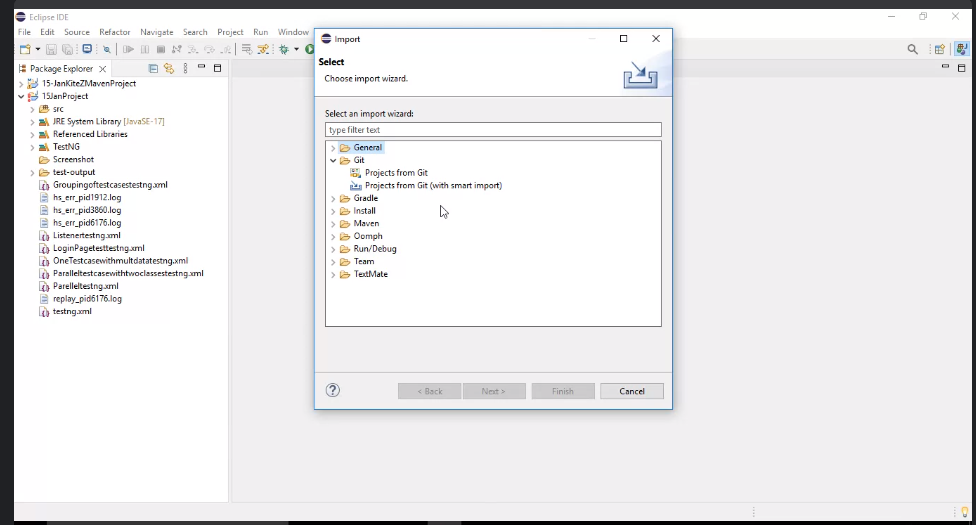


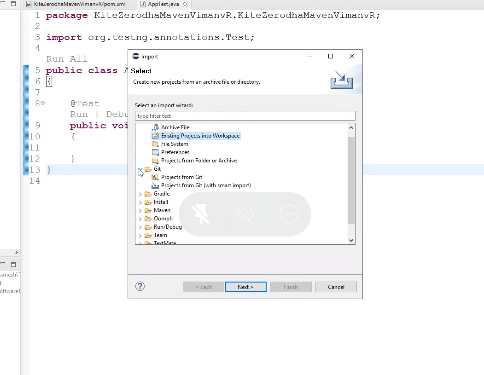
Click on Add people:



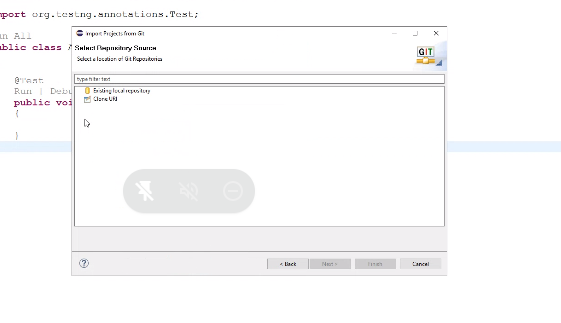
Clone a git project:



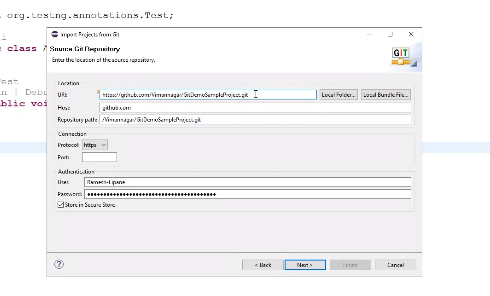




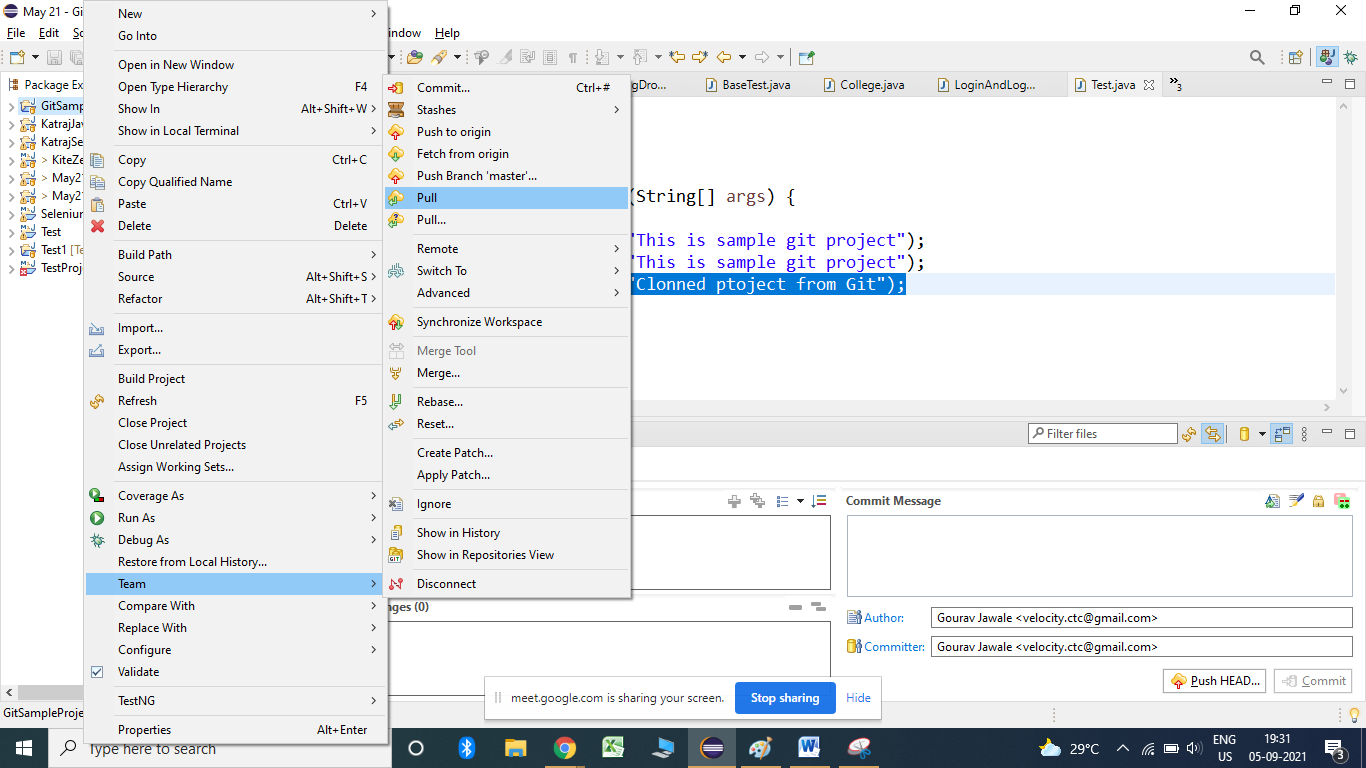
Click on Project from Git wiith smart import



Click on clone URI and click on Next

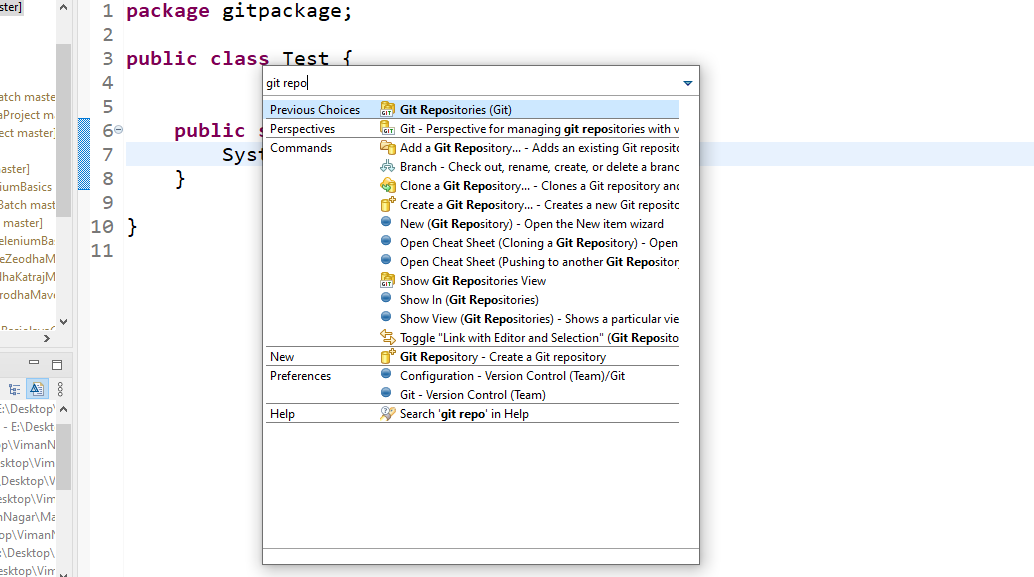


Pull process:



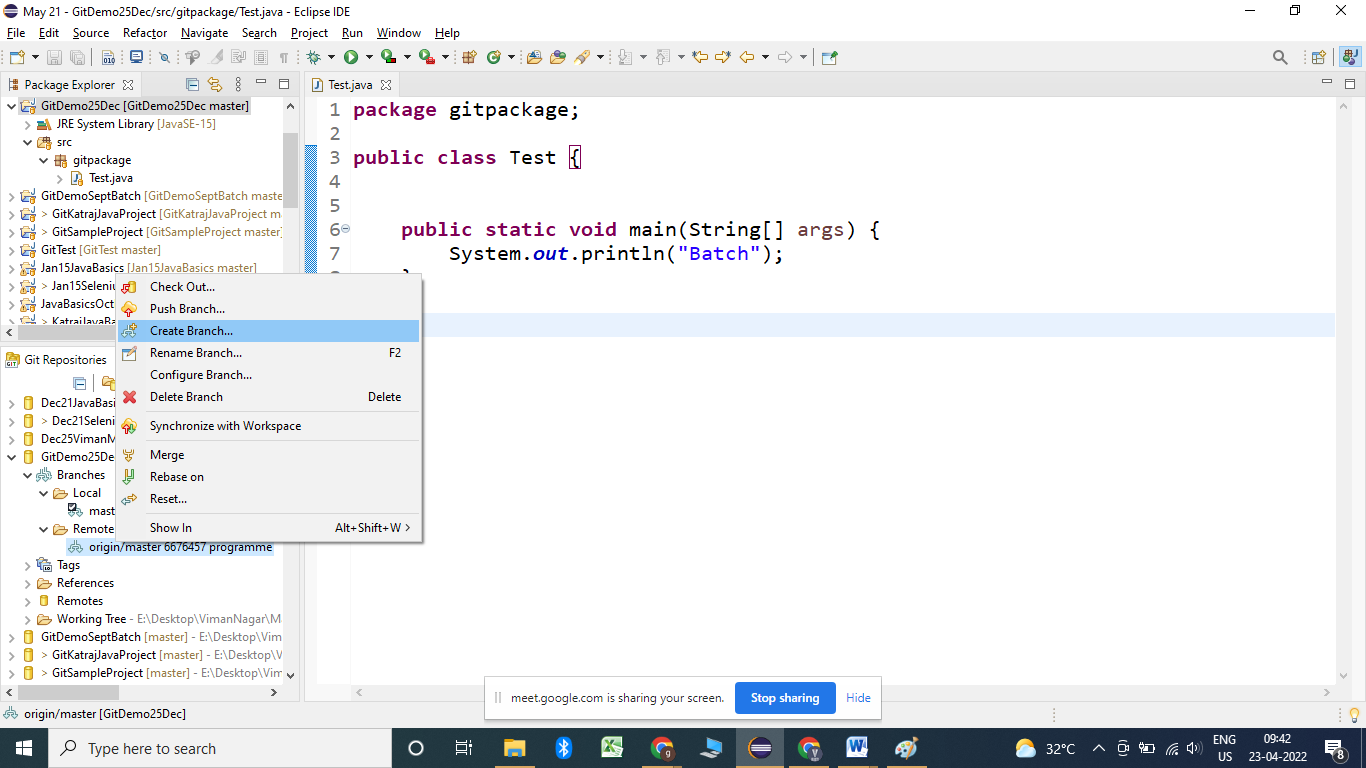
Conflicts and merging of branch in git:

Branching in Git: To create a mirror image of master branch we create a branch in the git.



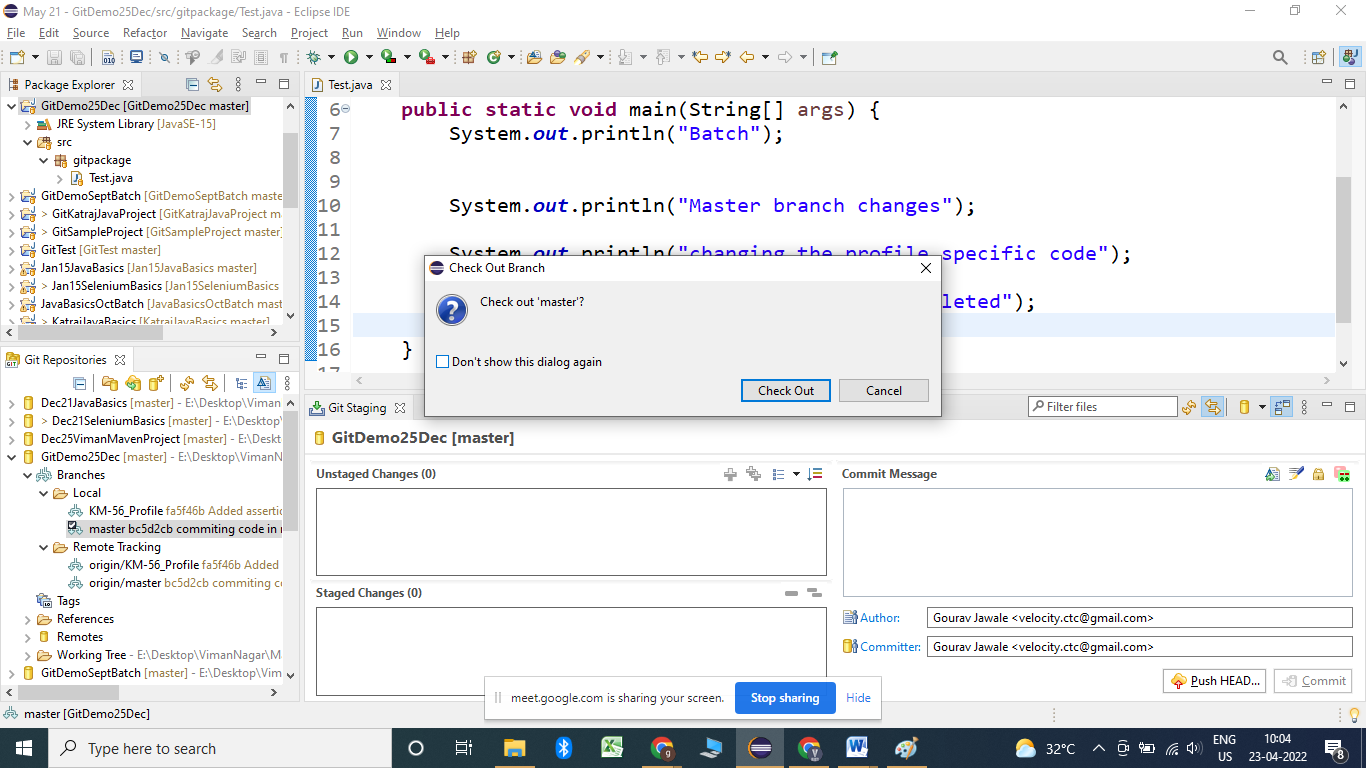
Click on Git repository

Right click on the Remote branch and select create branch option



After the commit and push if we wants to merge it with master

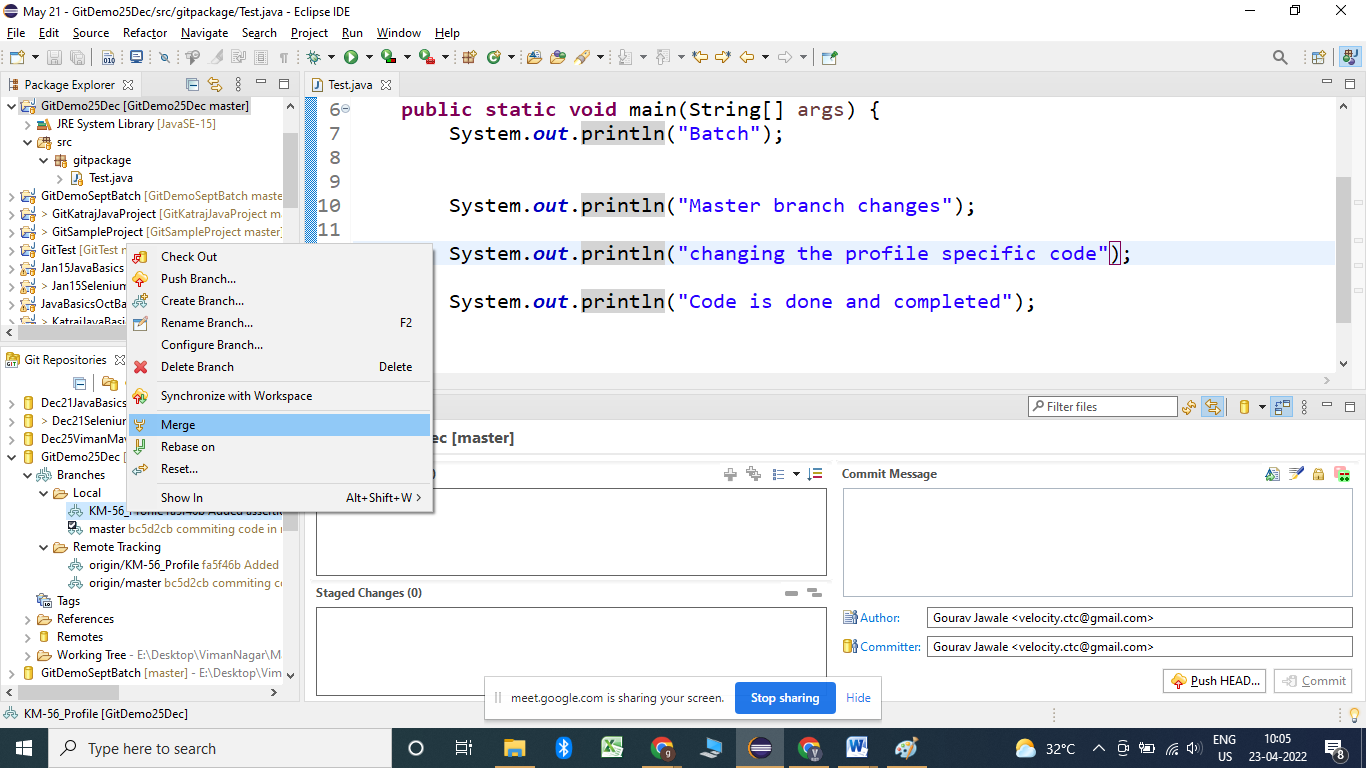
Checkout to master—double click on master branch



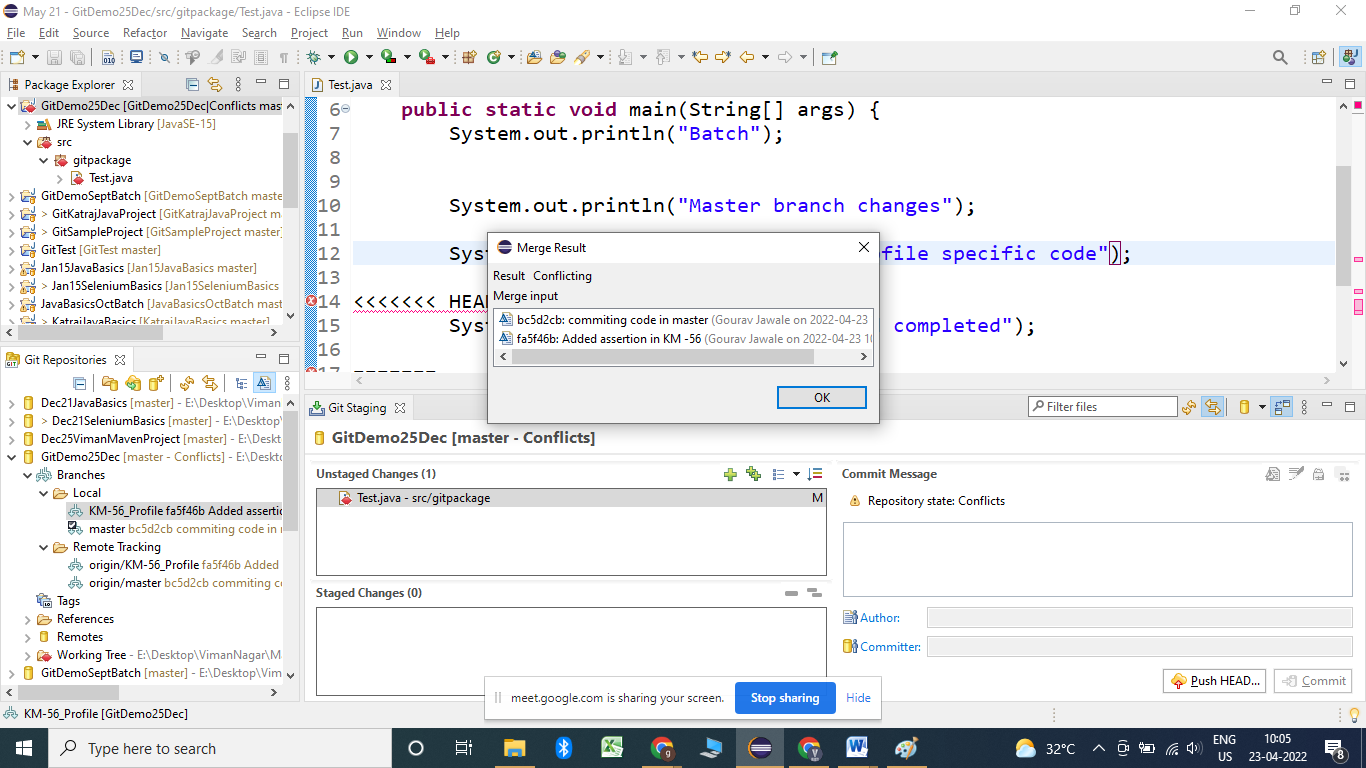
Click on checkout

Merging the code of KM -56 with master :

Right click on KM 56 branch and select merge



Once we click on merge we get conflicts as the content of the Test.java is different for both the branches



Once conflict occurs we have to resolve this issue which is getting display as error in the code

