**LAB # 13**

**Introduction to GitHub**

**OBJECTIVE**

Introduction to GitHub by creating Git repository and cloning its configuration to eclipse project. Also Performing commit, push, and pull operations from eclipse repository and analyzing version changes on Git repository.

**You will learn about:**

• Basics of how GitHub works.

GitHub allows multiple developers to work on a single project at the same time, reduces the risk of duplicative or conflicting work, and can help decrease production time. With GitHub, developers can build code, track changes, and innovate solutions to problems that might arise during the site development process simultaneously. Non-developers can also use it to create, edit, and update website content.

Some of the common terms teams will need to understand when using GitHub are:

**Repository**: Every project on GitHub is called repository. It is a folder for your project that contains all files and their revision histories.

**Branch**: a workspace in which you can make changes that won’t affect the live site.

**Commit Changes**: a saved record of a change made to a file within the repository.

**Pull Request (PR)**: the way to ask for changes made to a branch to be merged into another branch that also allows for multiple users to see, discuss and review work being done.

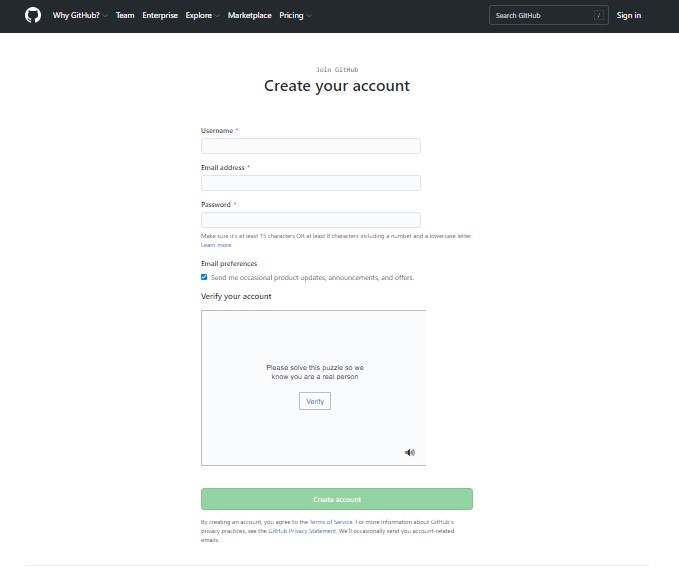
**Merge**: after a pull request is approved, the commit will be pulled in (or merged) from one branch to another and then, deployed on the live site

**Issues**: how work is tracked when using git. Issues allow users to report new tasks and content fixes, as well as allows users to track progress on a project board from beginning to end of a specific project.

**STEPS OF CREATING GIT REPOSITORY:**

**Step # 01**: Create a GitHub account by clicking on link and sign up.

[**https://github.com/**](https://github.com/)



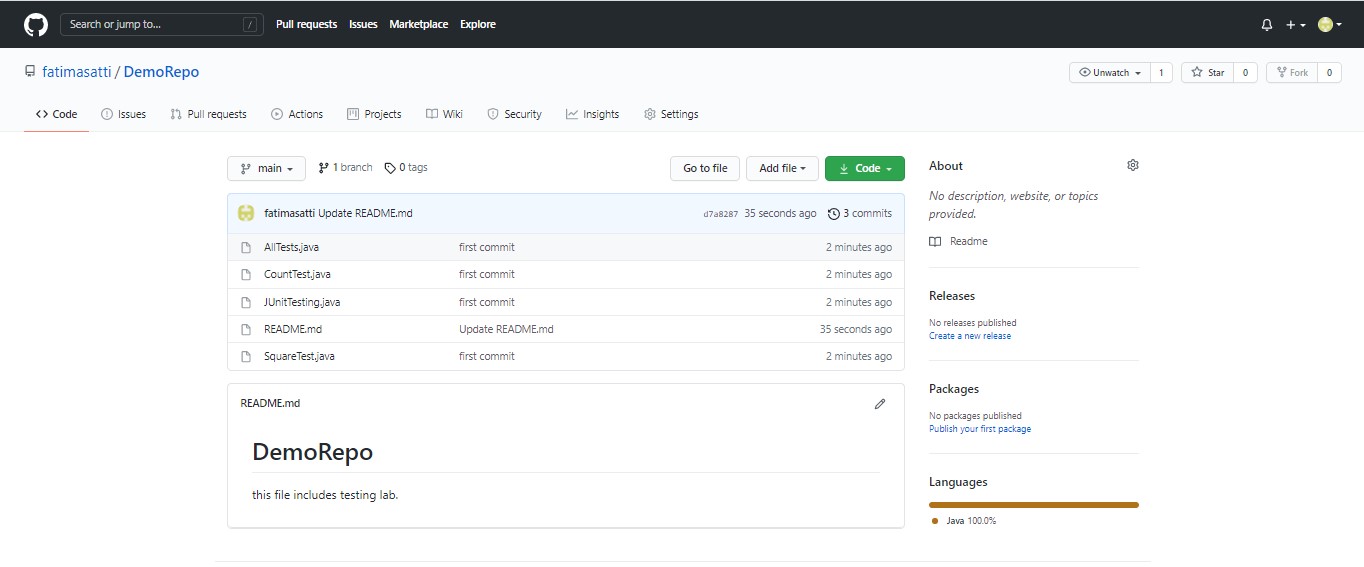
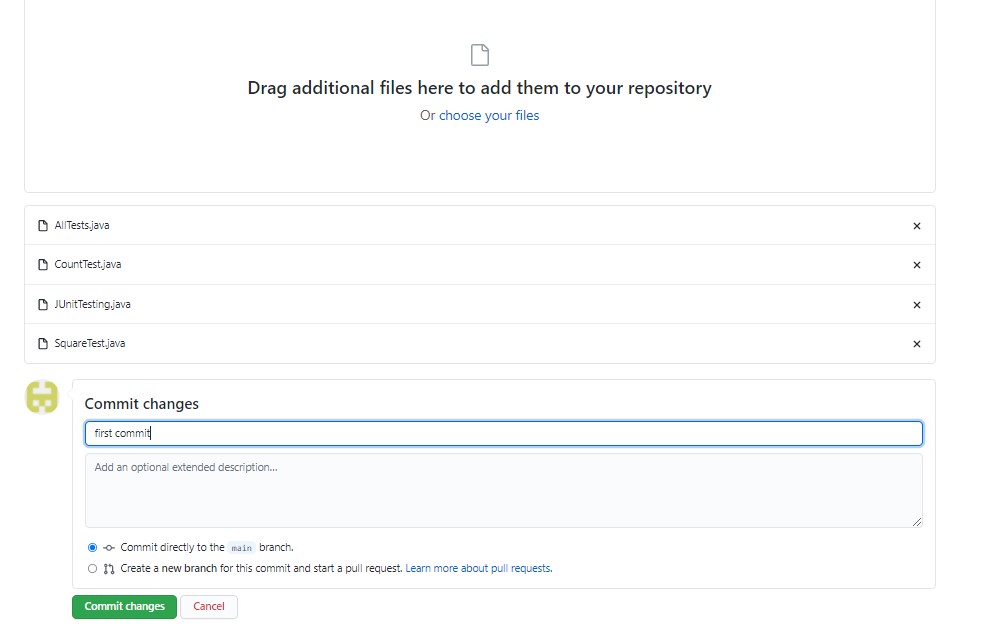
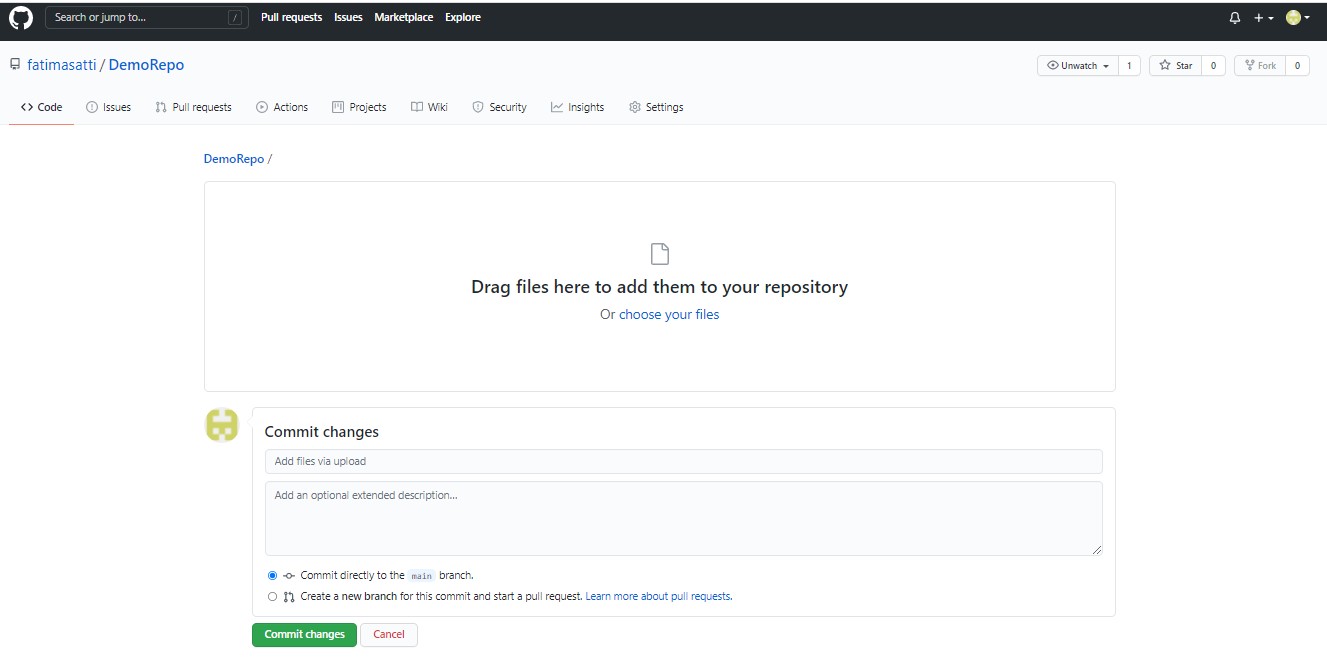
**Step # 2: Create a repository by clicking (+) sign or by clicking the link below:**

[**https://github.com/new**](https://github.com/new)

* Enter your Repository name.
* You can also add description (optional).
* Choose whether you want to make your repository Public or Private.
* Add a README file (optional).
* Click on Create repository.

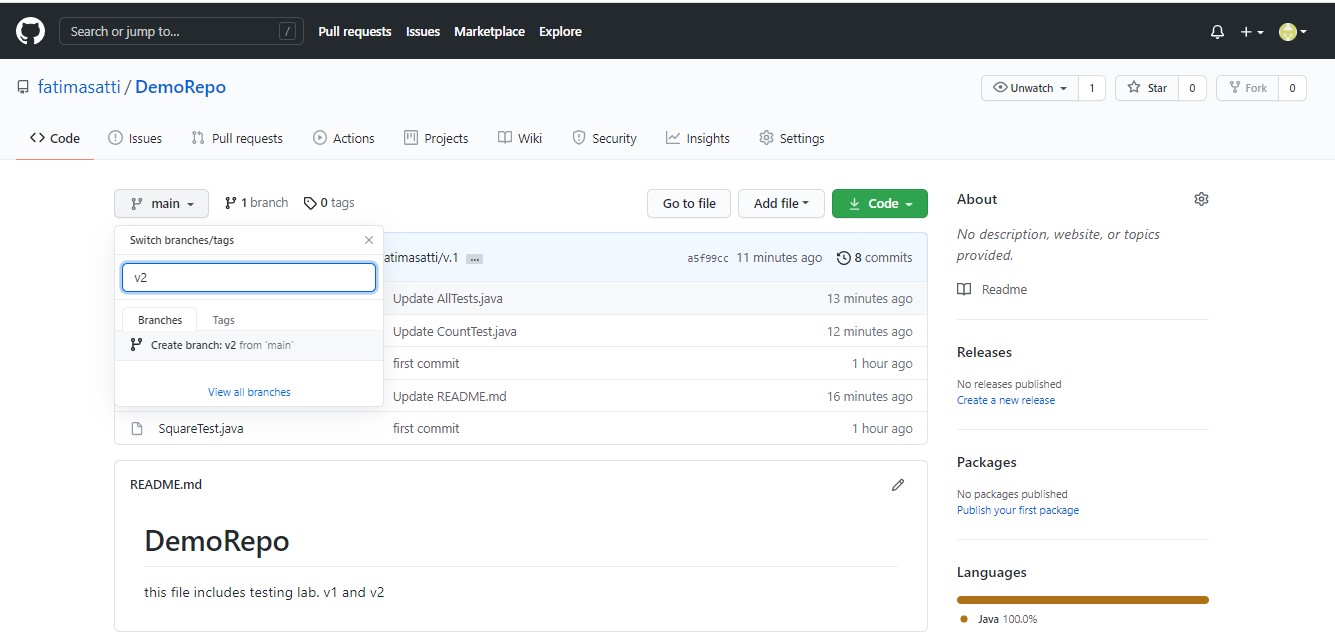


**Step # 3: Upload files from any eclipse workplace folder and commit changes.**



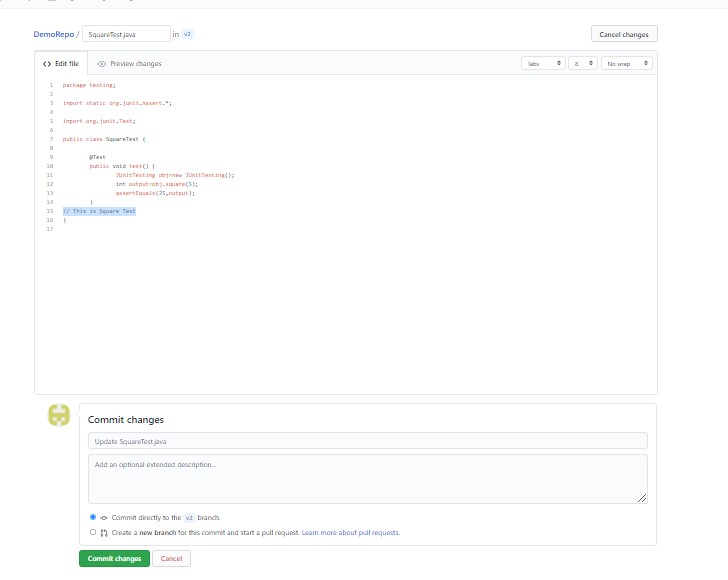
You can now see that all the files are uploaded in your current repository.

**Step # 4: Create a new branch and make changes in any file. Perform Commit, Pull requests and merge.**



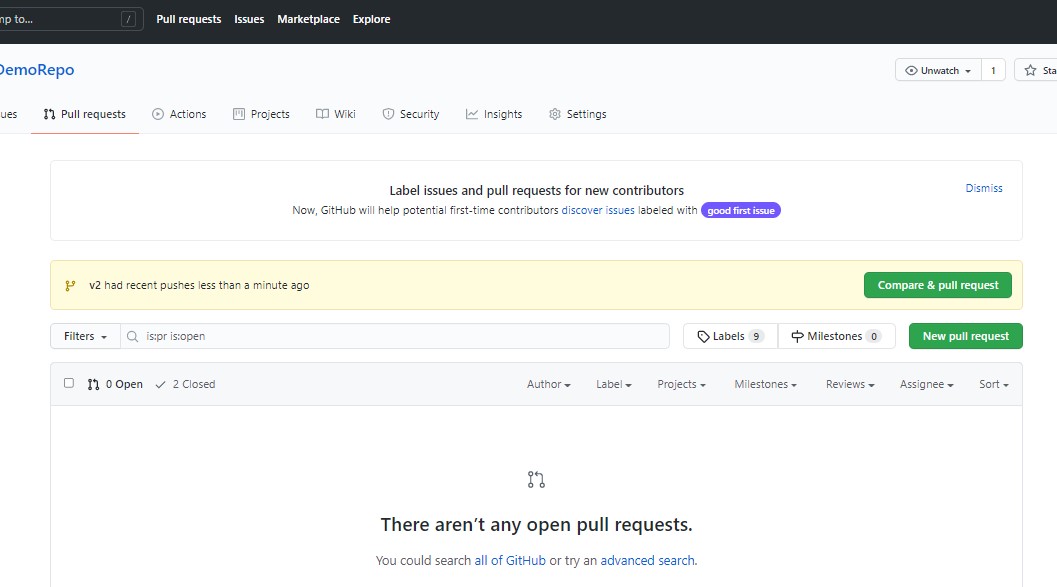
(In above screeshot, you can see a branch named **V2** is created).

Update any file in V2, then select Commit changes.



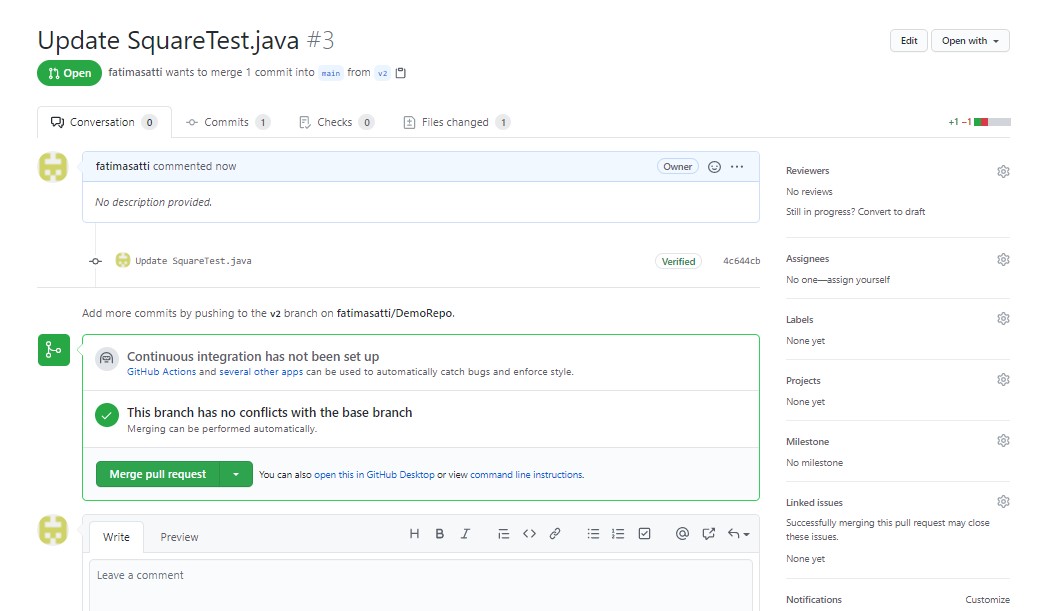
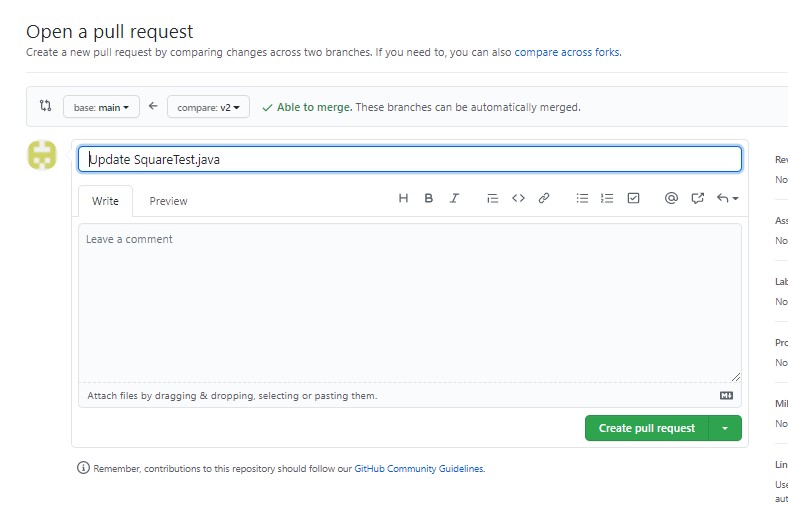
**Go to Pull Requests tab, select Compare and Pull request.**

// This will compare your V2 edited file with main branch and then will commit the changes.

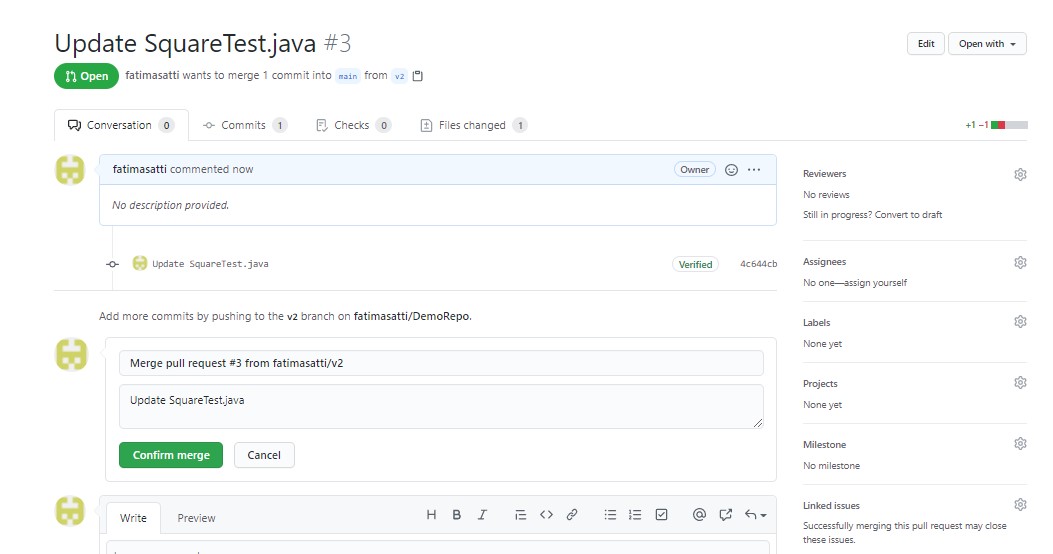


**Enter create pull request**

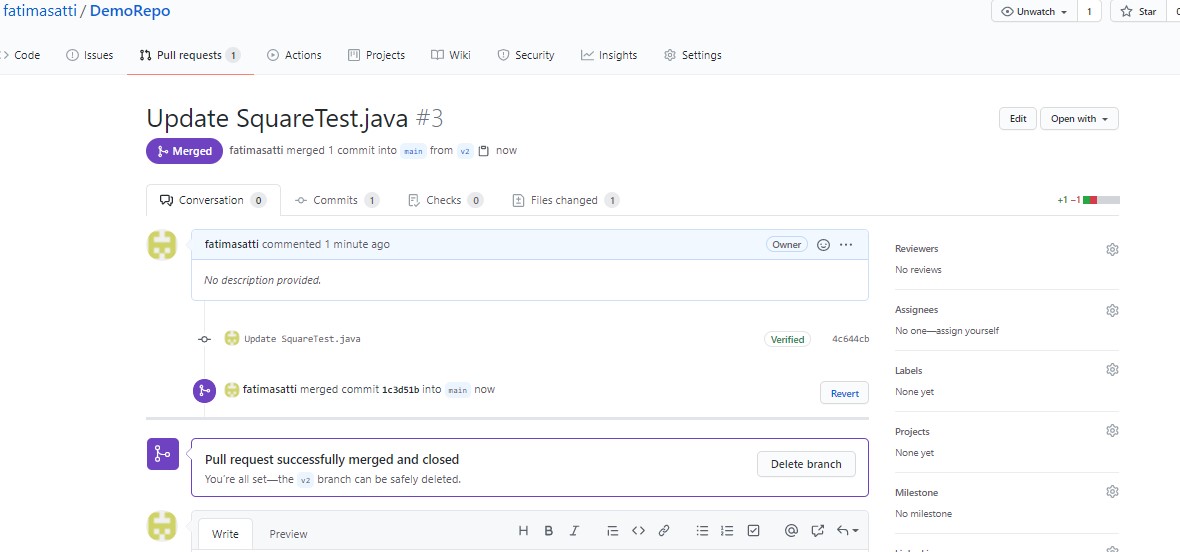
**Select Merge Pull request**



**Confirm Merge**

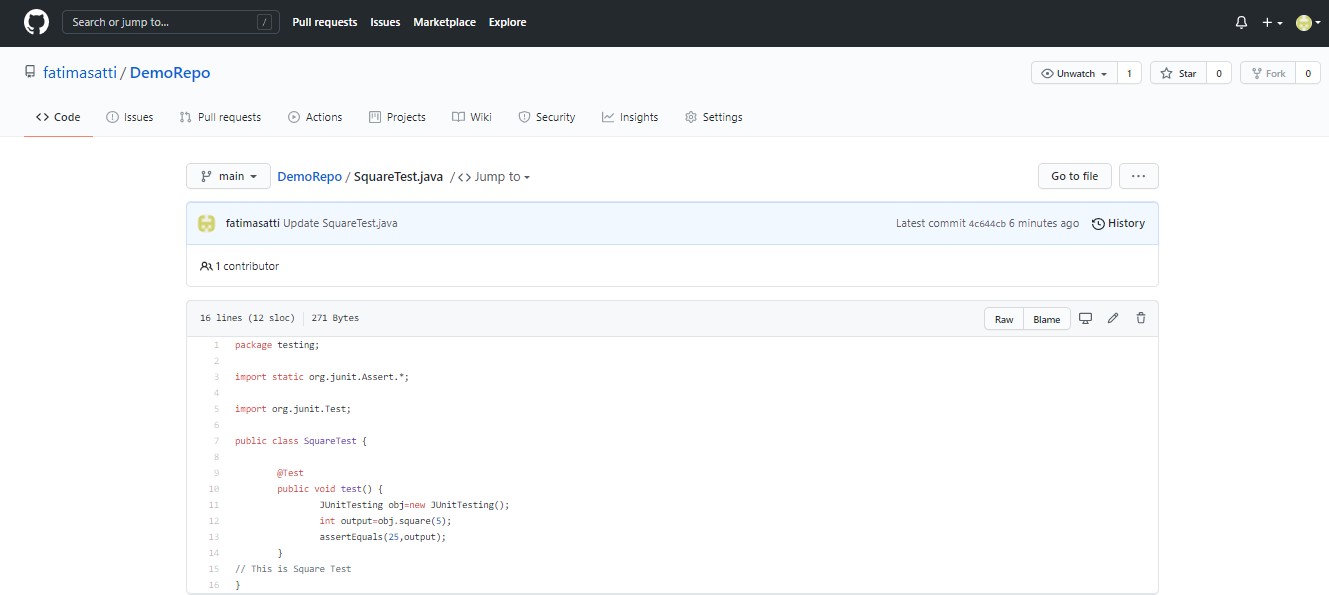
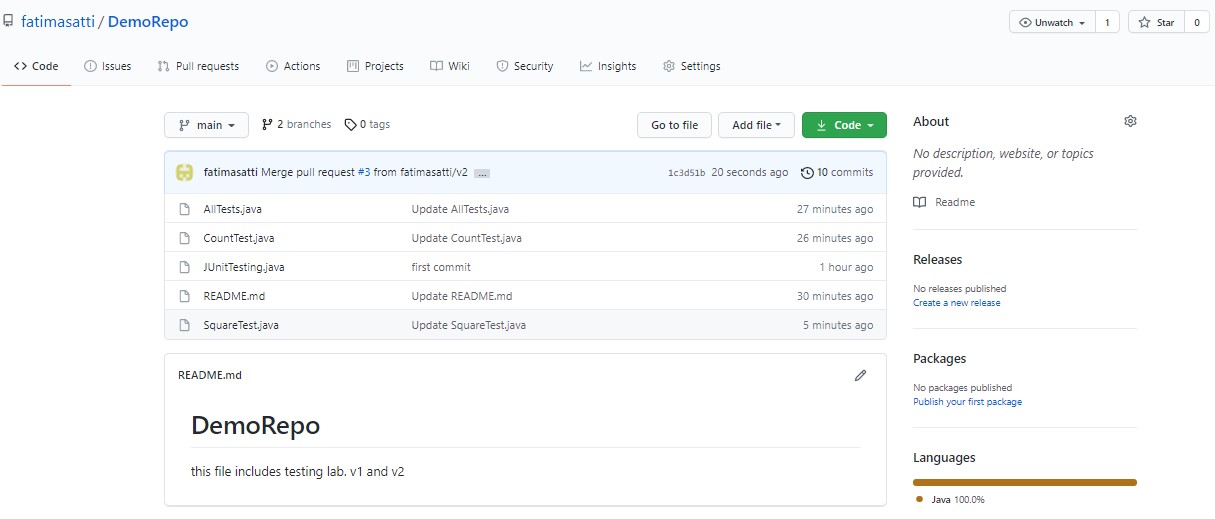


**You can see that Pull request is successfully merged and closed.**



You can now delete V2 branch as it is successfully merged into main branch.

SquareTest file is now updated.



**You must know:**

* How to create GitHub repository

**What you’ll learn:**

* How to clone repository in eclipse.
* How to add eclipse project to GitHub repository.
* How to commit, push and pull the changes.

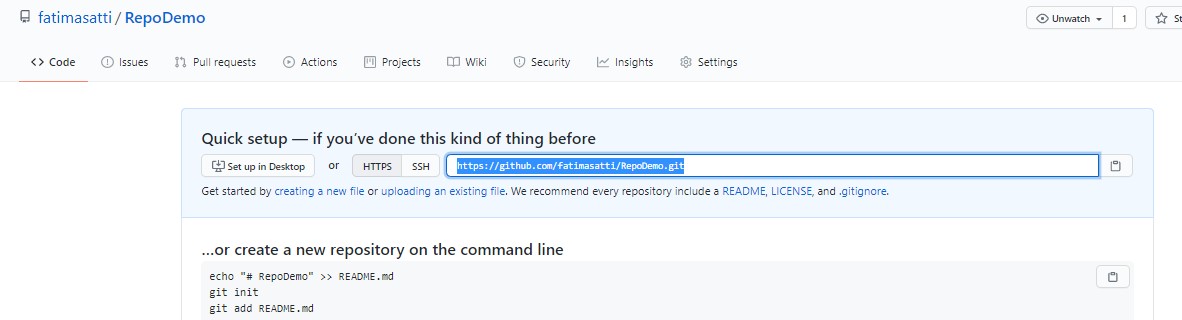
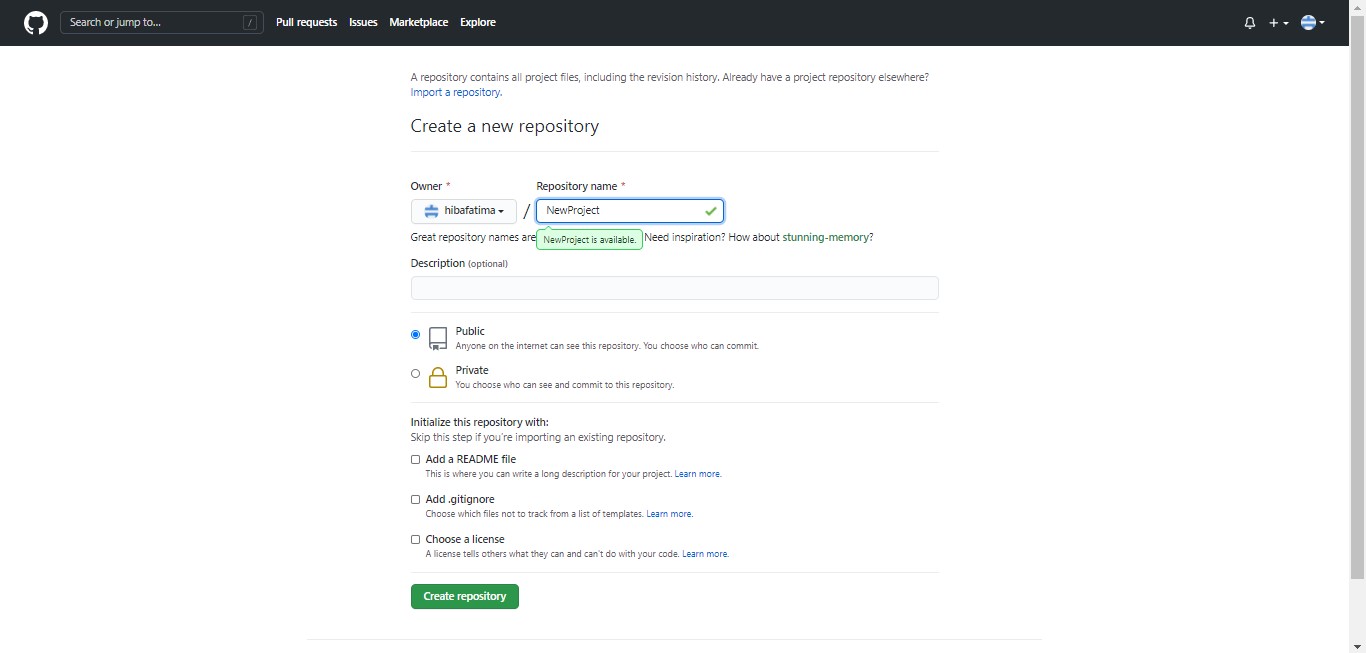
**Step 01: Create GitHub Account and SignIn**

**Step 02: Create a Repository.**

After

creating repository, copy the URL

.



**Step 03: Start Eclipse.**

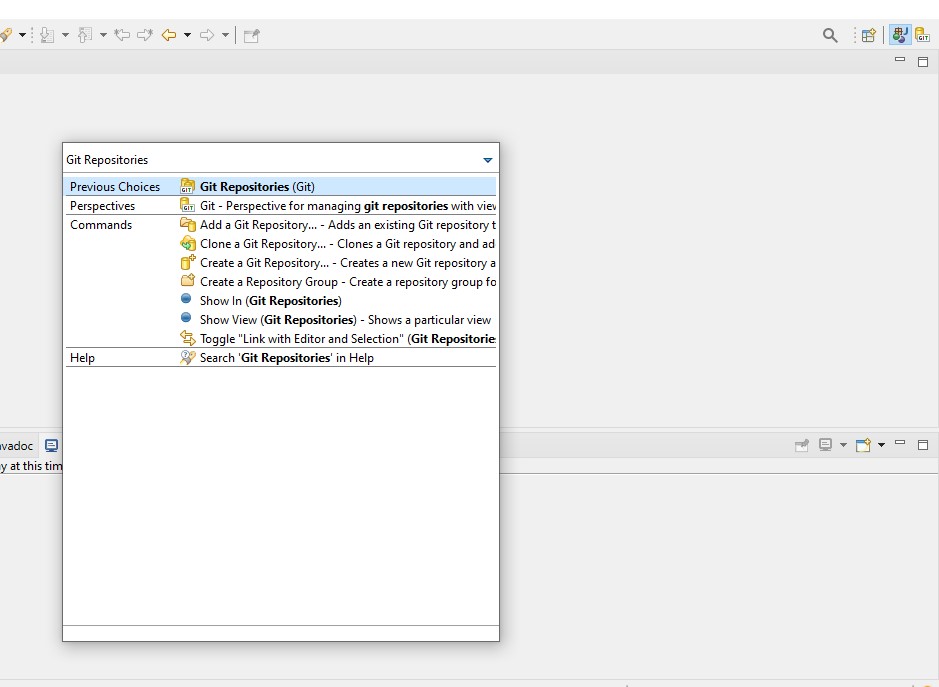
**Step 04: Go to Perspective** → **Git Repositories** → **Add Git Repositories**

**Or**

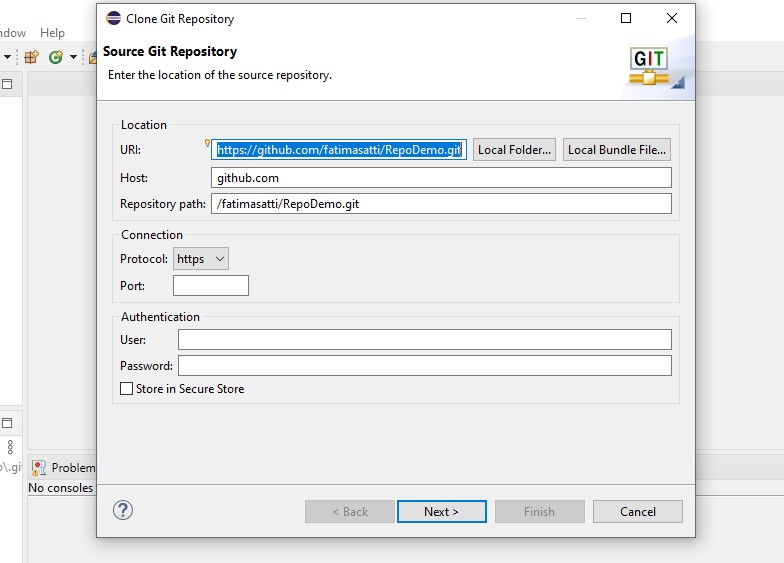
**Window** → **Perspective** → **Open Perspective** → **Other** → **Add Git**

**Or**

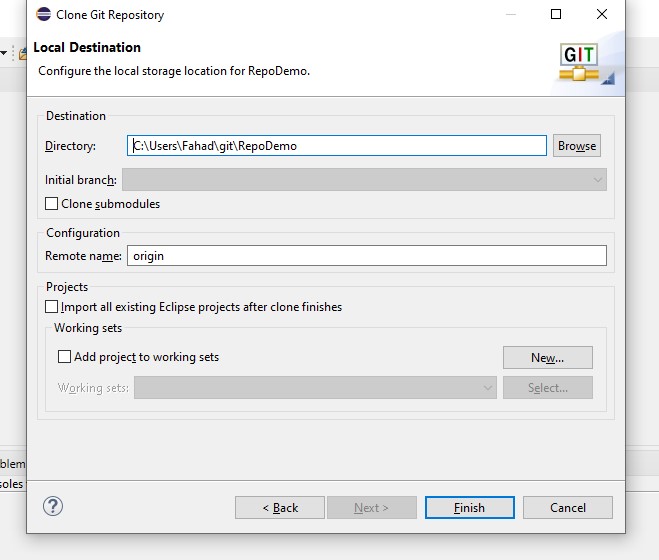
**Search for Git Repositories**



As we have already created repository in GitHub, we will now clone that repository in Eclipse.

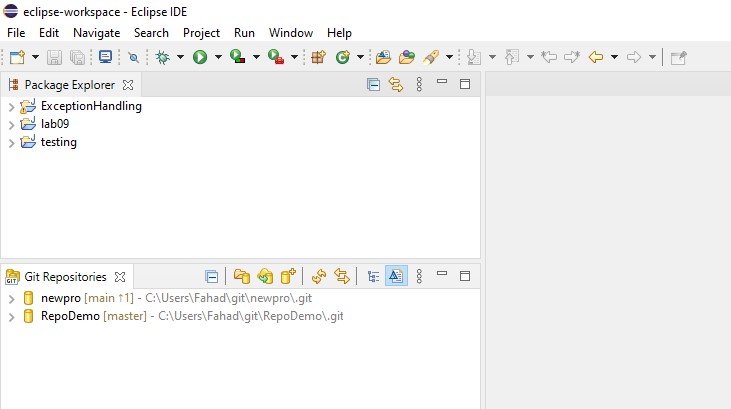


Enter the URL copied from GitHub and then Provide login credentials and enter next.



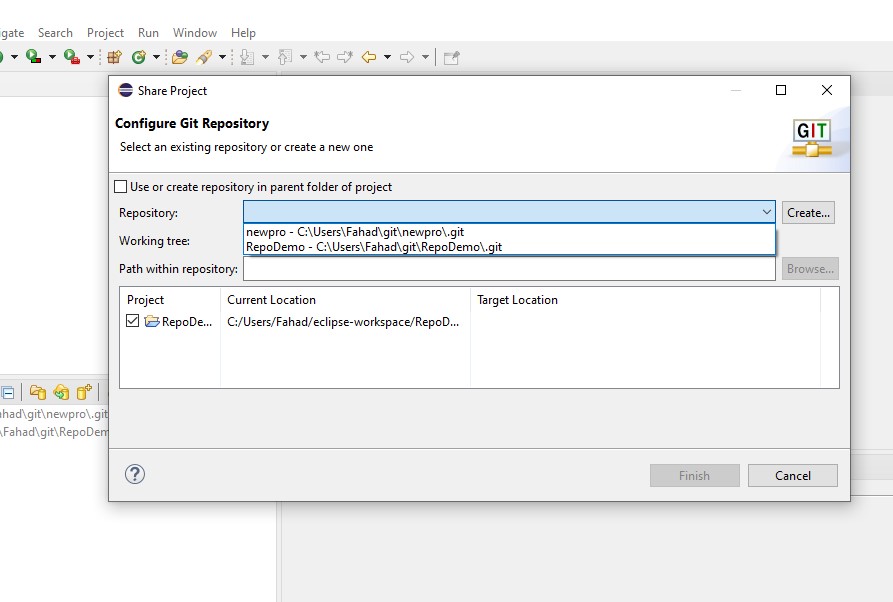
Enter Finish

We got our repository from eclipse.



**Step 05: Create a Project in Eclipse.**

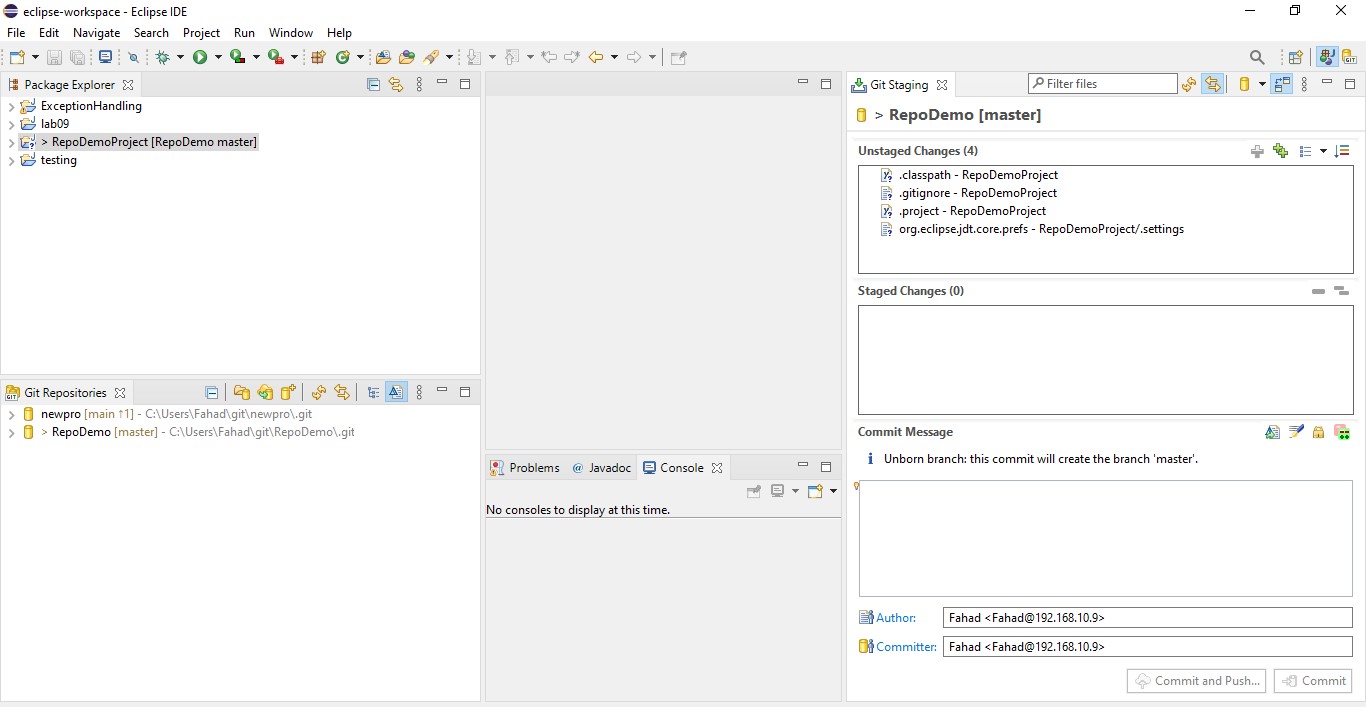
**Step 06: Right click on Project** → **Team** → **Share** → **Add Git Repository (the one you created)**



Select your current repository and enter Finish. The project will be now entered in your repository.

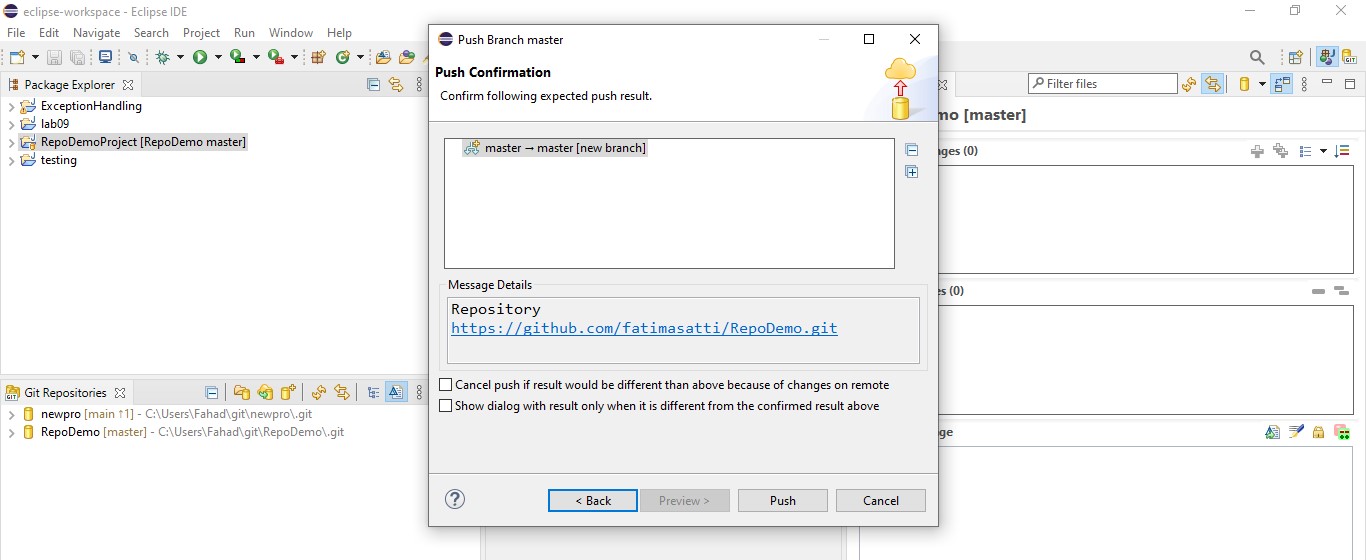
**Step 07: Commit and Push the project to the repository.**

**Right Click** → **Team** → **commit**



**Drag and drop files from Unstaged changes to staged changes, write some commit message and enter Commit and push.**

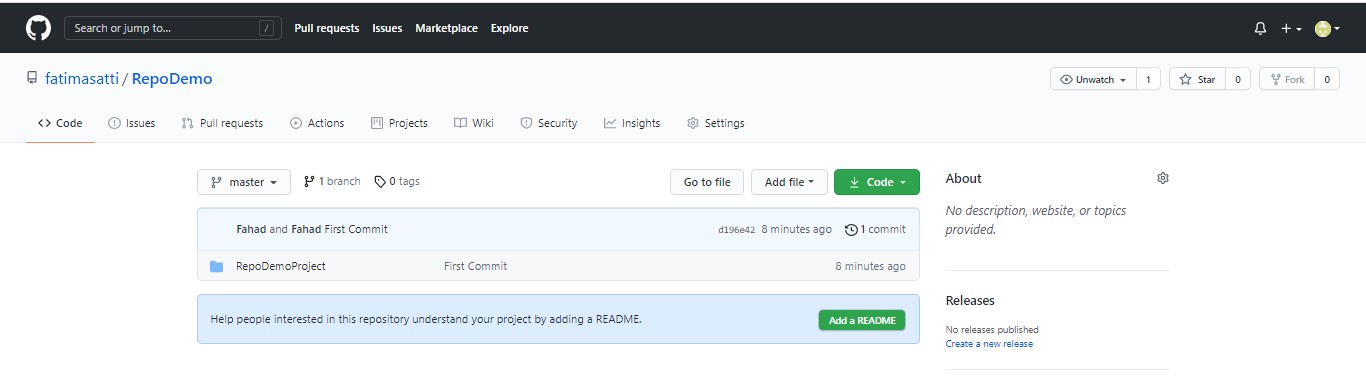
**It will ask for username and password then enter finish.**



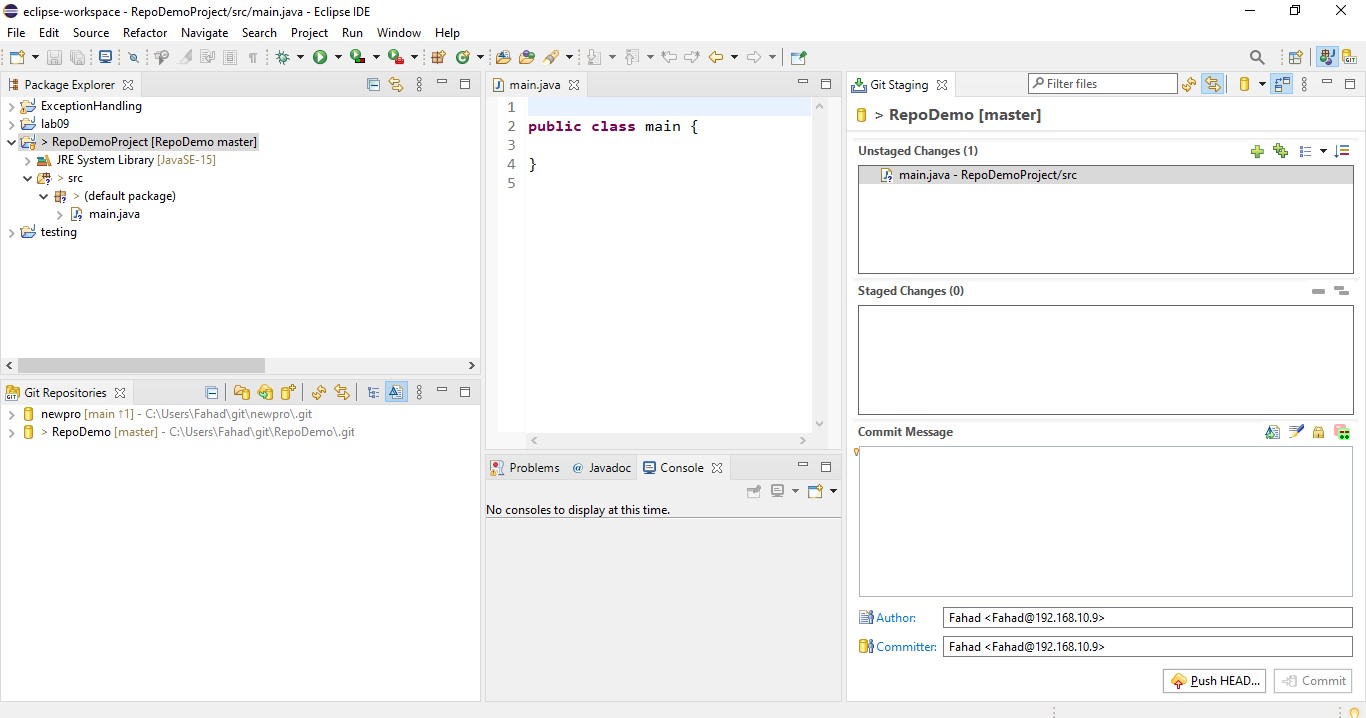
Enter Push.

Refresh GitHub, now you can see your project their in the repository you created.

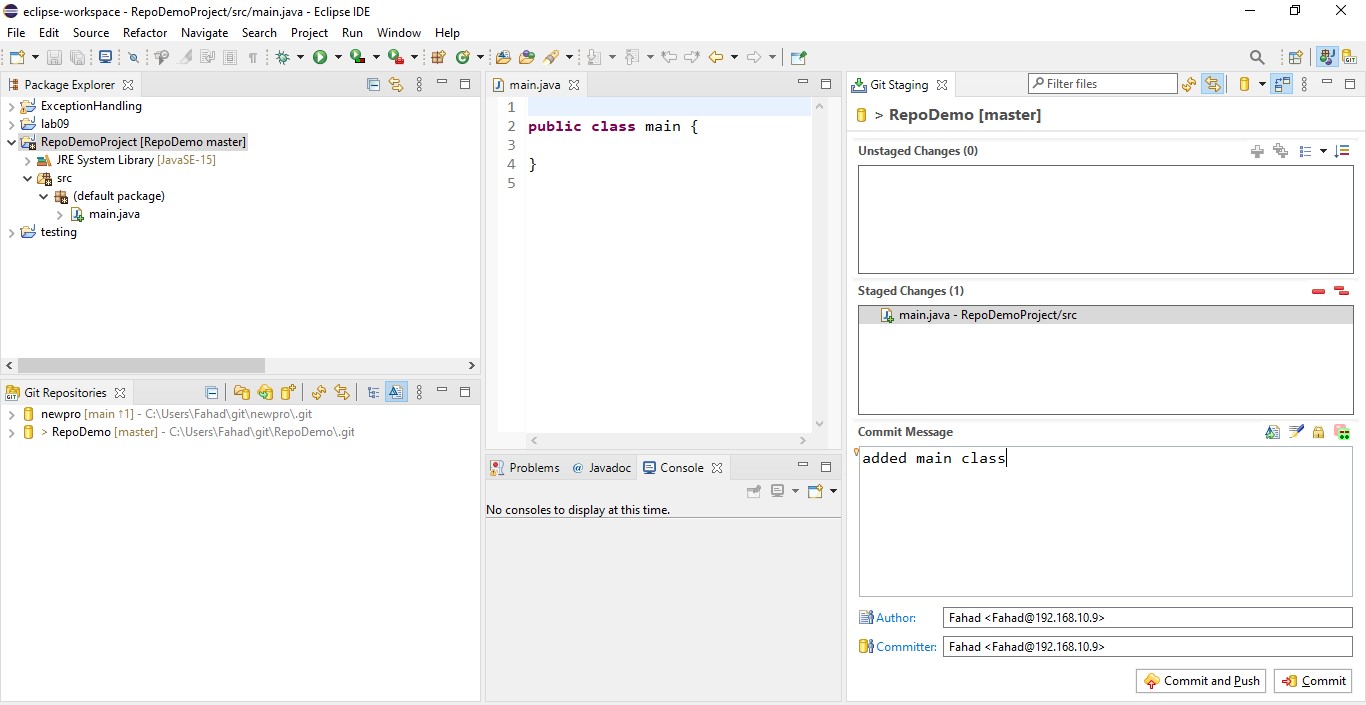
**Step 08: Adding classes.**

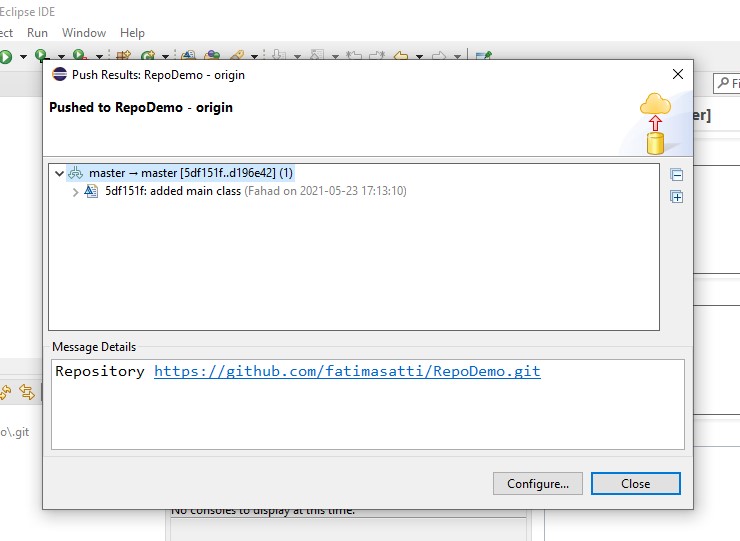


Create a new class in eclipse. Go to team, perform commit.



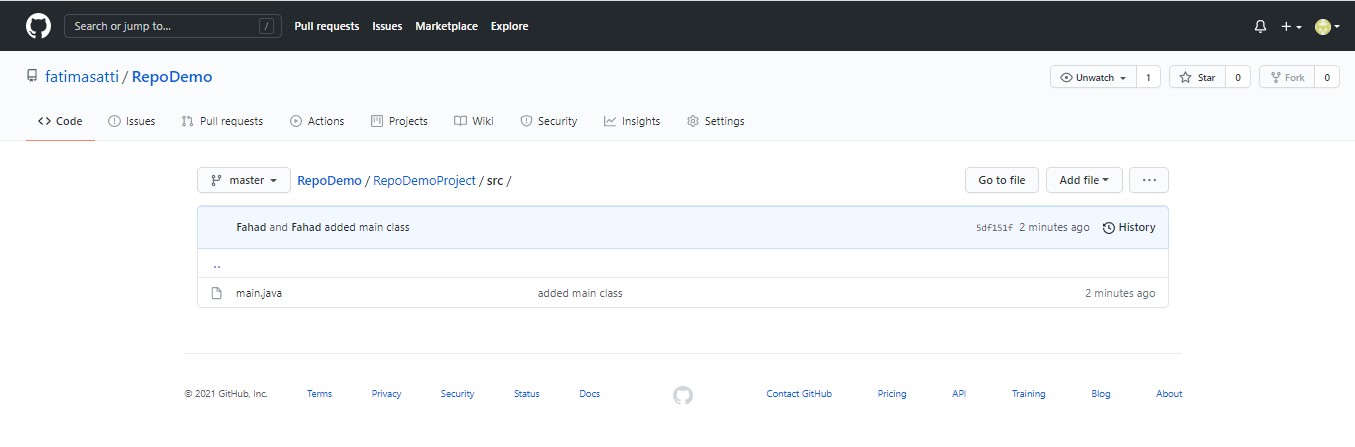
**Drag and drop files from Unstaged changes to staged changes, write some commit message and enter commit and push.**





Go to GitHub, refresh your page.

You will see that your class has been added and saved in your repository. (Here src folder is added which contains our java class)



**Lab Task:**

1.Create account on GitHub.

2.Upload any of your Java Project files and perform commit, push, pull and merge operations.

3.Create account on GitHub, add a clone git repository in eclipse and push the java project on GitHub.