

LAB # 1

Introduction to GitHub

OBJECTIVE

Introduction to GitHub by creating Git repository and cloning its configuration to eclipse project.

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.

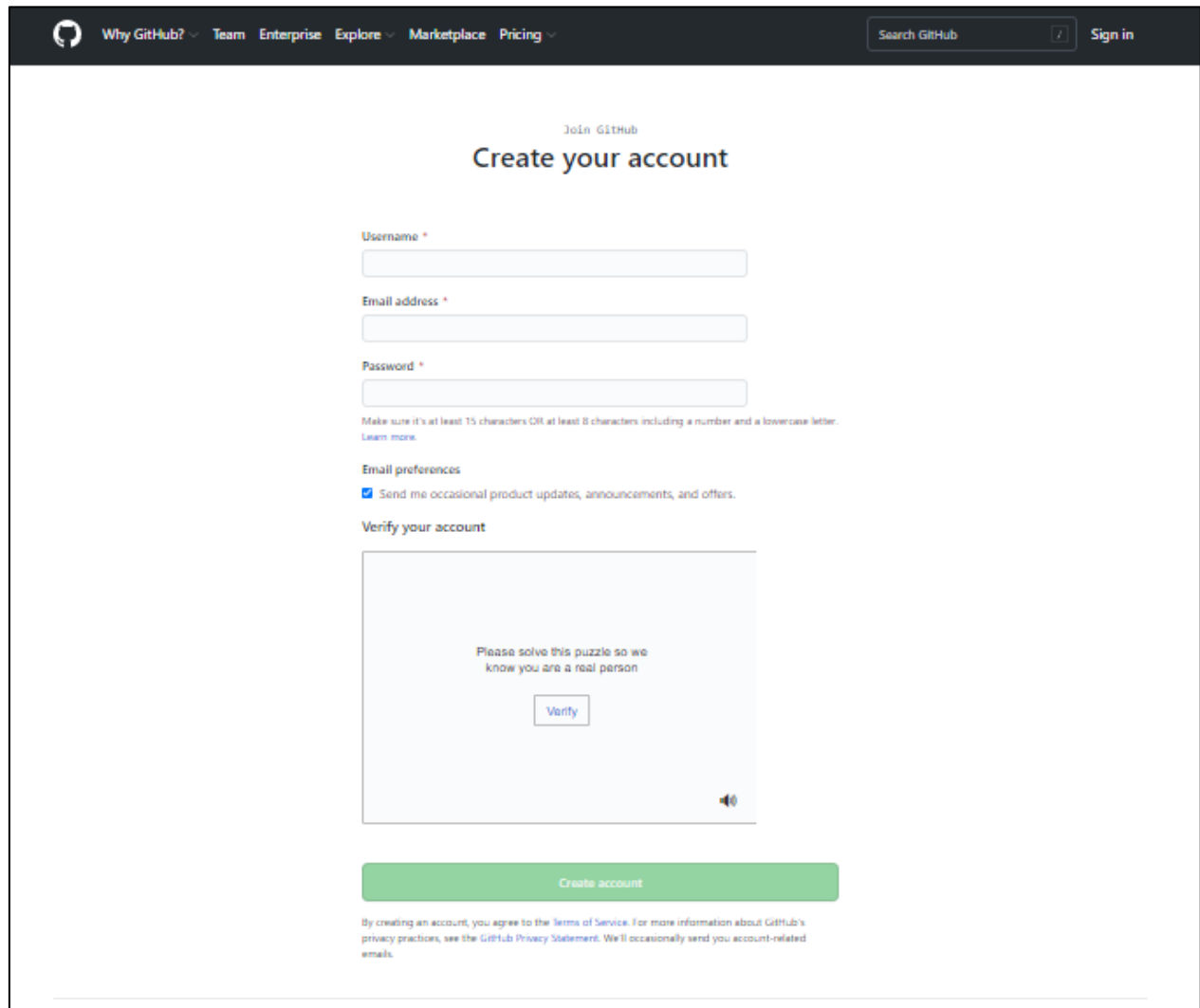
Merge: after a pull request is approved, the commit will be pulled in (or merged) from one branch to another and then, deployed on the live site

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/>

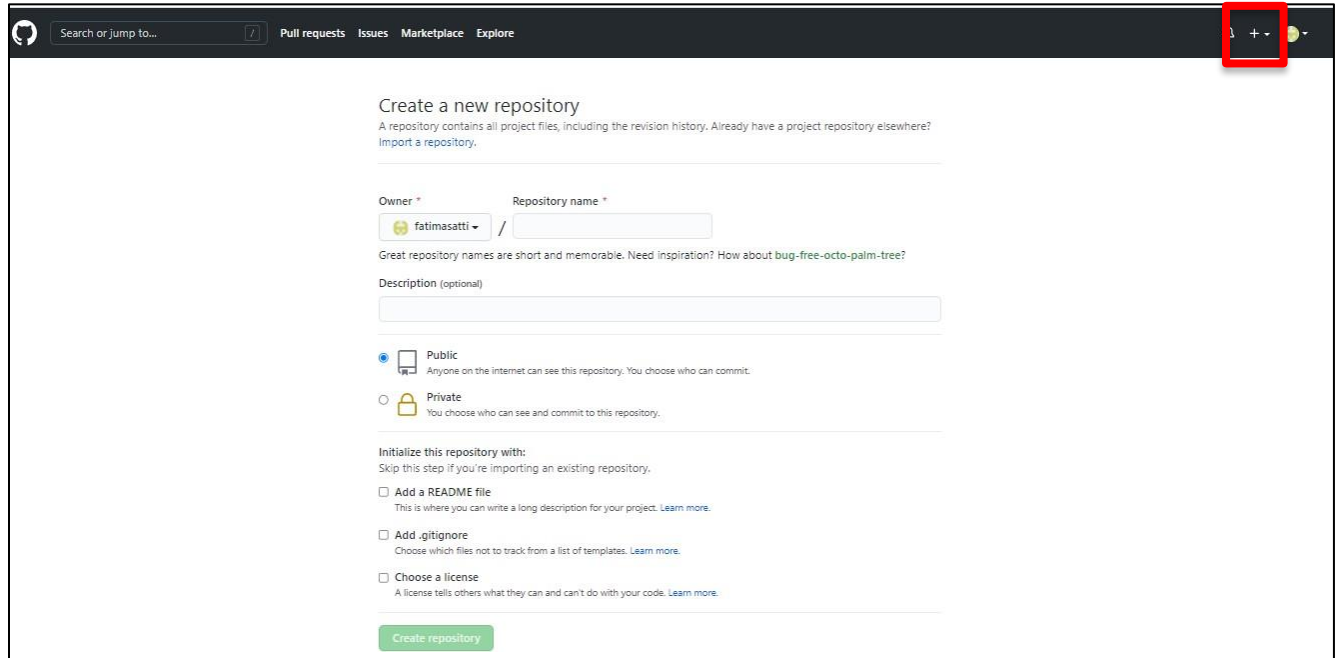


The screenshot shows the GitHub 'Create your account' page. At the top, there is a navigation bar with links: 'Why GitHub?', 'Team', 'Enterprise', 'Explore', 'Marketplace', and 'Pricing'. A search bar and a 'Sign in' link are also present. The main heading is 'Join GitHub' followed by 'Create your account'. The form includes three input fields: 'Username *', 'Email address *', and 'Password *'. Below the password field, there is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)'. There is a section for 'Email preferences' with a checked checkbox 'Send me occasional product updates, announcements, and offers.' Below this is a 'Verify your account' section with a large box containing the text 'Please solve this puzzle so we know you are a real person' and a 'Verify' button. At the bottom of the form is a green 'Create account' button. Below the button, there is a disclaimer: 'By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.'

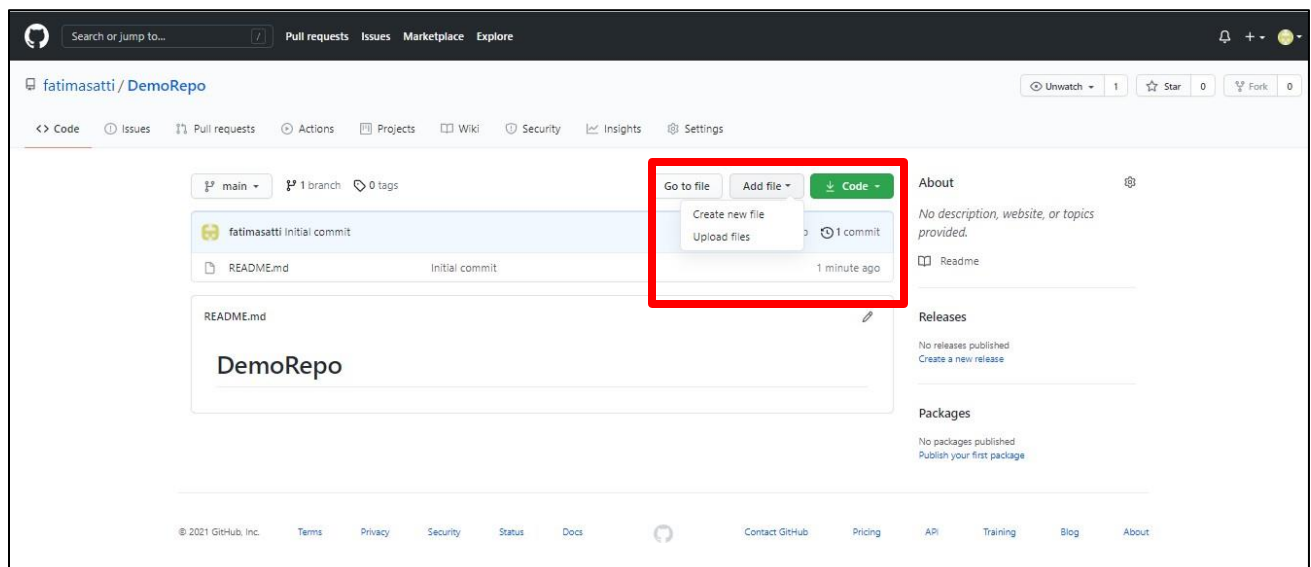
Step # 2: Create a repository by clicking (+) sign or by clicking the link below:

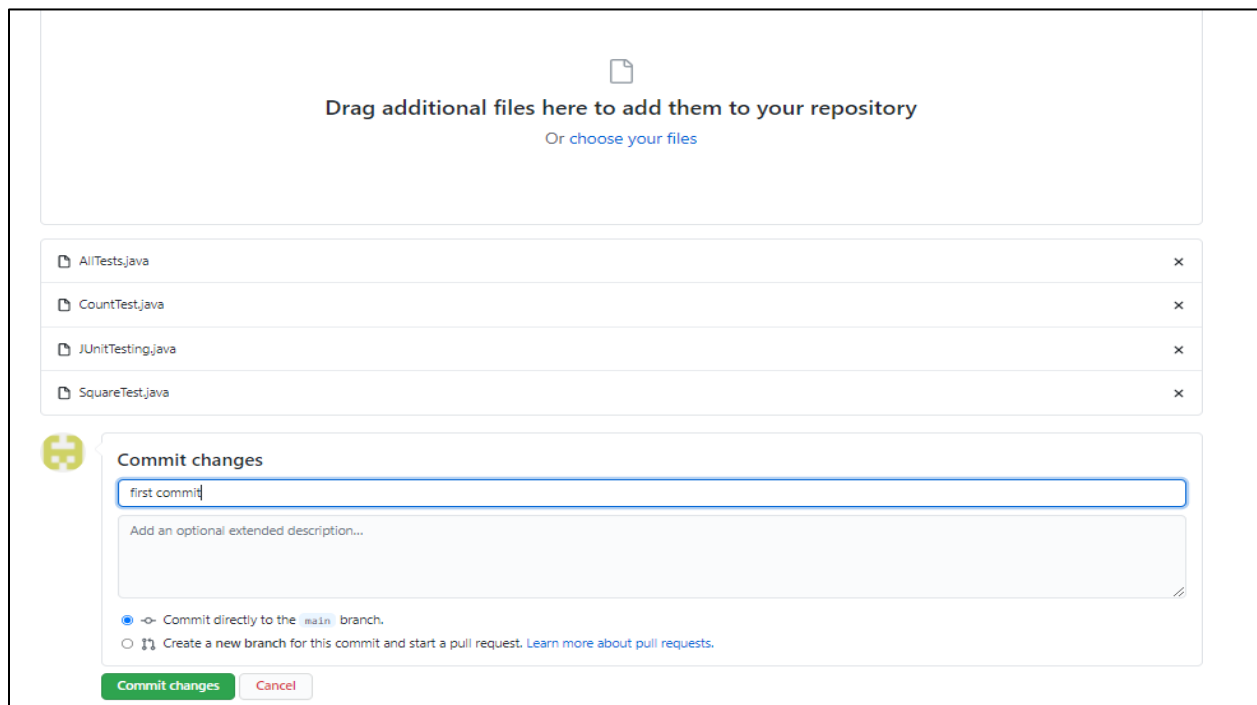
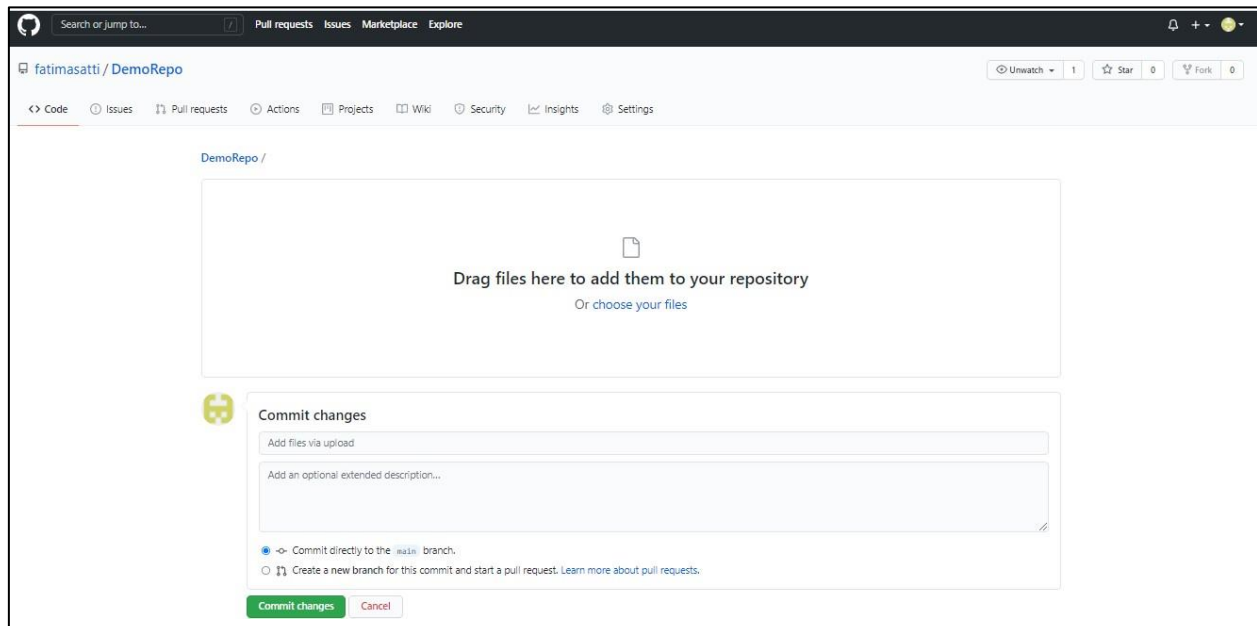
<https://github.com/new>

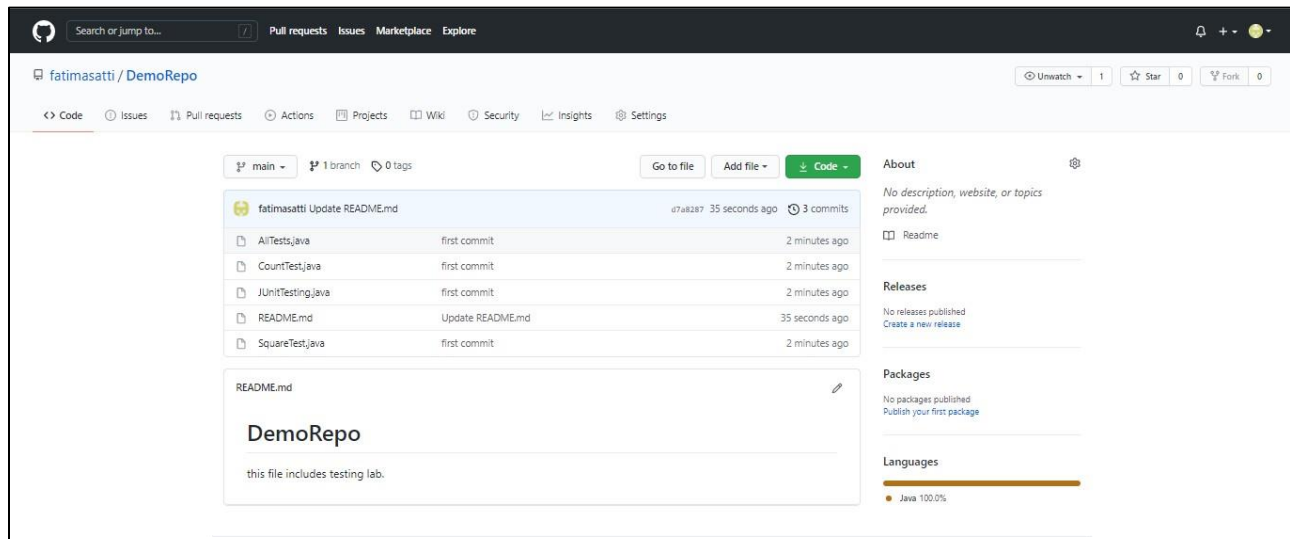
- Enter your Repository name.
- You can also add description (optional).
- Choose whether you want to make your repository Public or Private.
- Add a README file (optional).
- Click on Create repository.



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

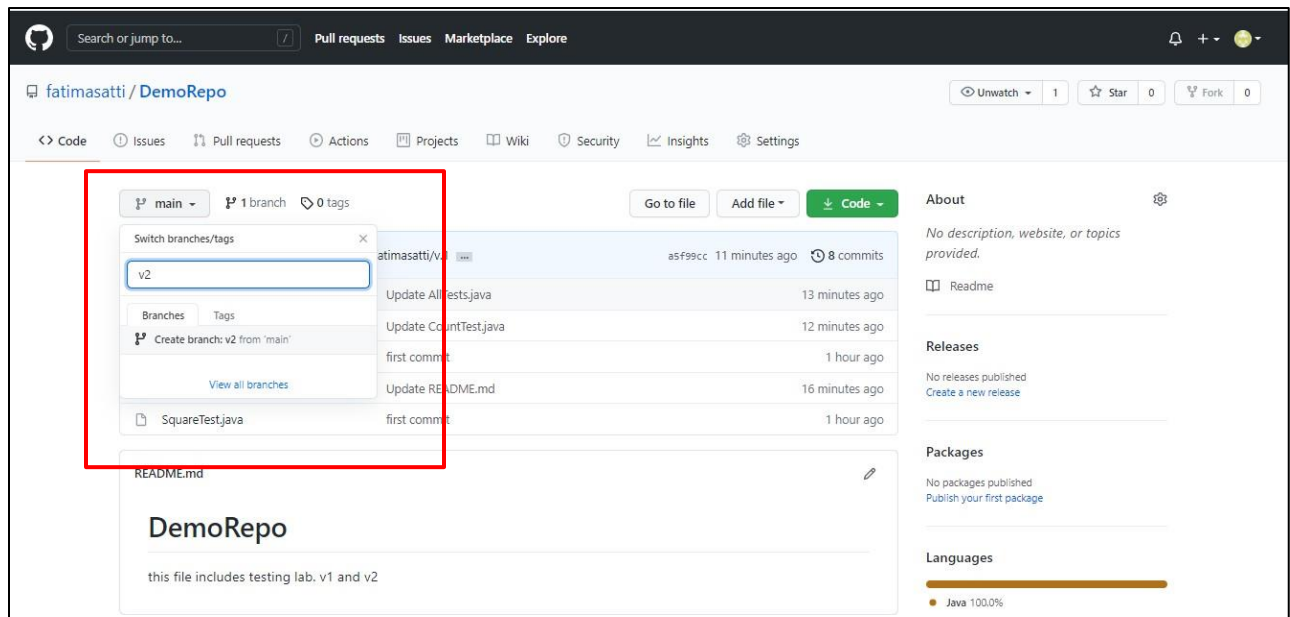






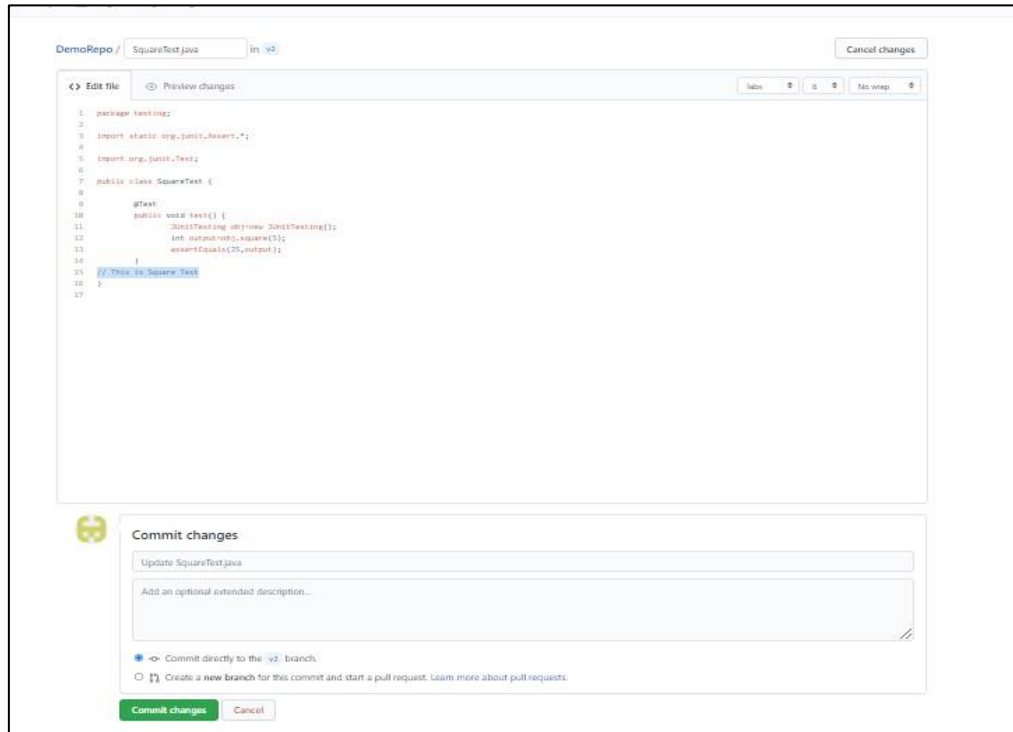
You can now see that all the files are uploaded in your current repository.

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



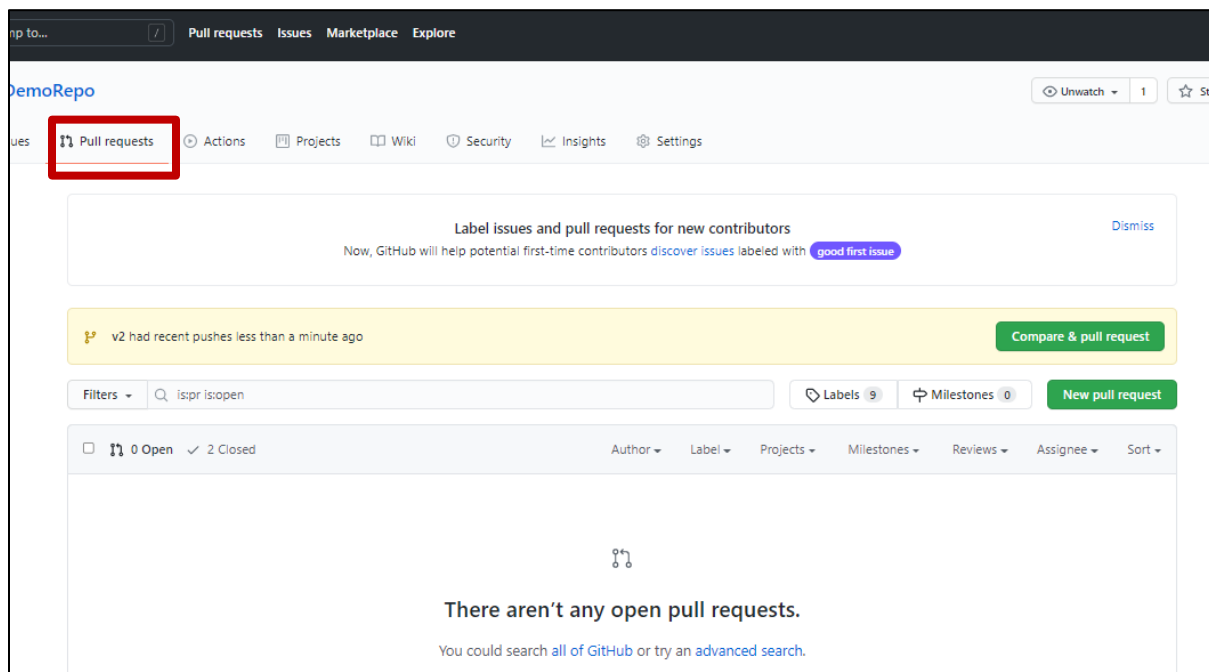
(In above screenshot, you can see a branch named **V2** is created).

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

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base: main ← compare: v2 ✓ Able to merge. These branches can be automatically merged.

Update SquareTest.java

Write Preview

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Select Merge Pull request

Update SquareTest.java #3

fatimasatti wants to merge 1 commit into main from v2

Conversation 0 Commits 1 Checks 0 Files changed 1

fatimasatti commented now

No description provided.

Update: SquareTest.java Verified 4c644cb

Add more commits by pushing to the v2 branch on fatimasatti/DemoRepo.

Continuous integration has not been set up
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request You can also open this in GitHub Desktop or view command line instructions.

Write Preview

Leave a comment

Reviewers: No reviews. Still in progress? Convert to draft.

Assignees: No one—assign yourself.

Labels: None yet.

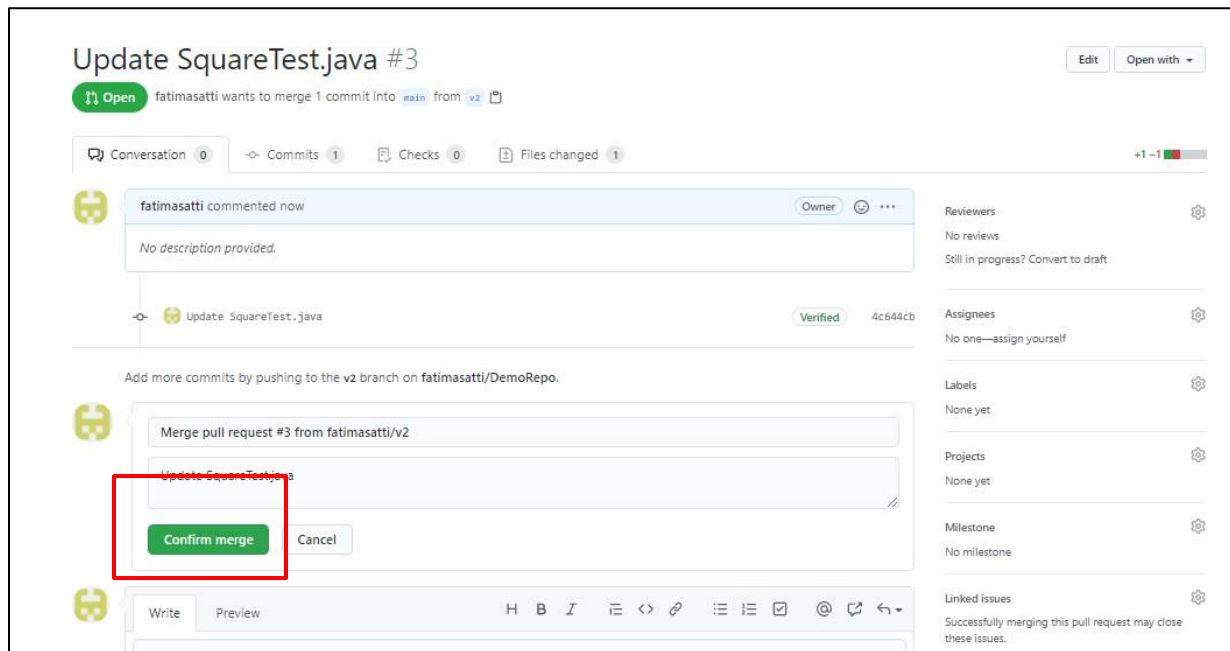
Projects: None yet.

Milestone: No milestone.

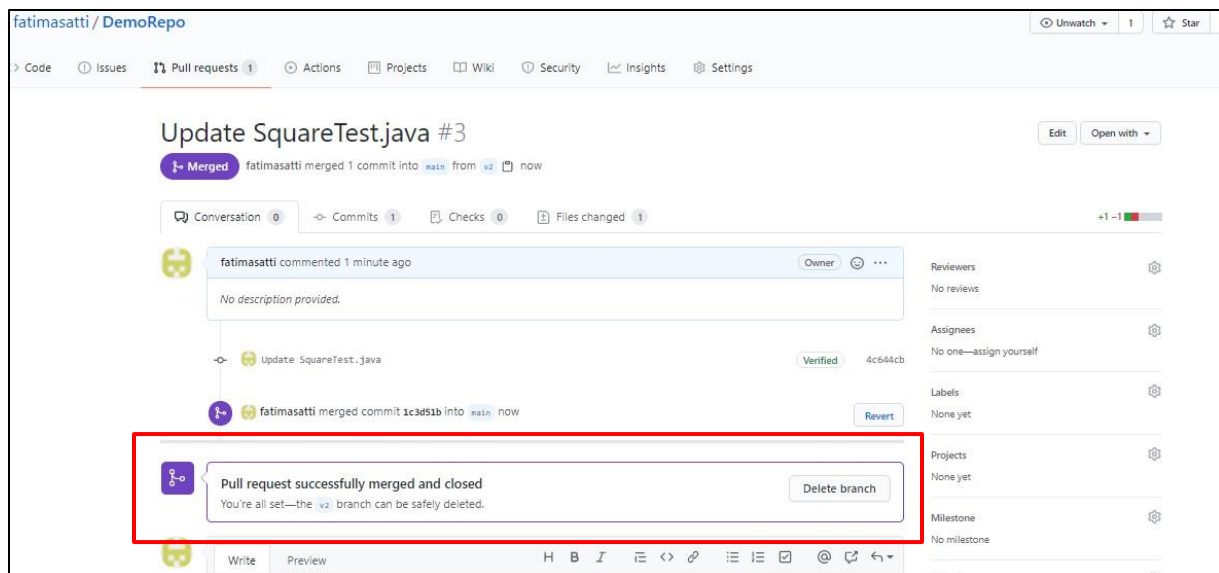
Linked issues: Successfully merging this pull request may close these issues. None yet.

Notifications: Customize.

Confirm Merge

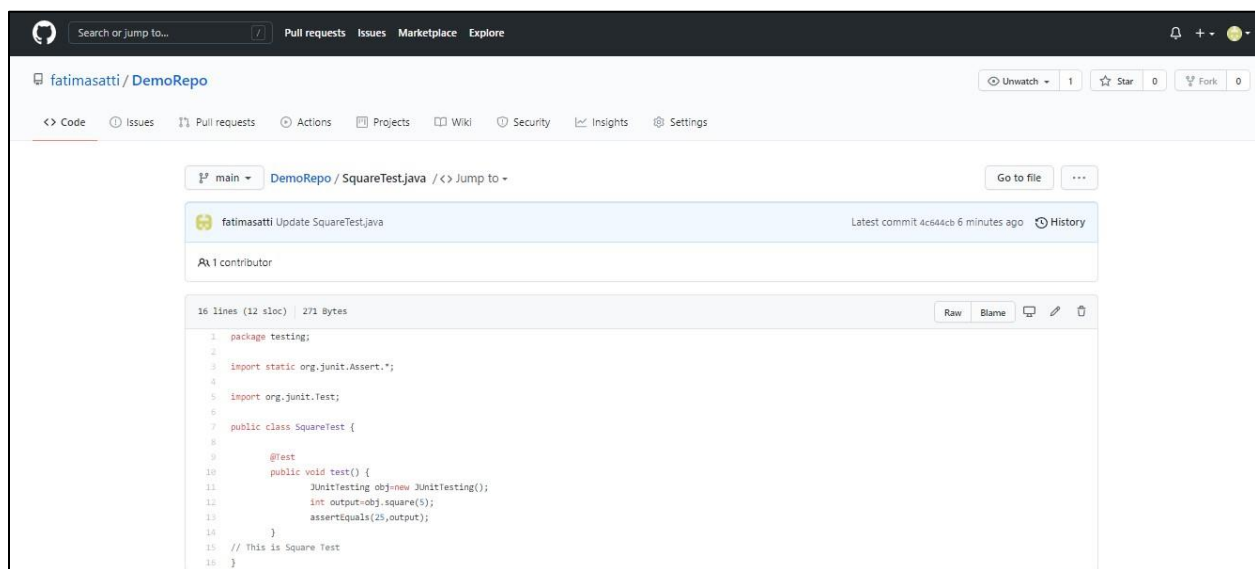
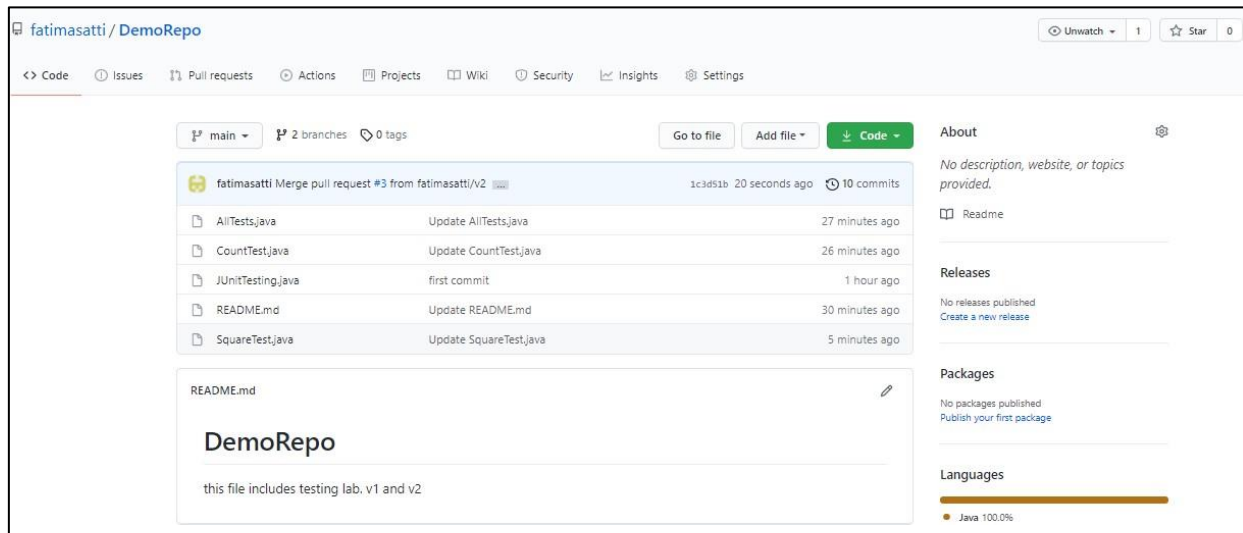


You can see that Pull request is successfully merged and closed.



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

SquareTest file is now updated.



Lab Task:

Create account on GitHub.

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