GitHub Tutorial For Beginners.

Prepared By: Bipin Gupta

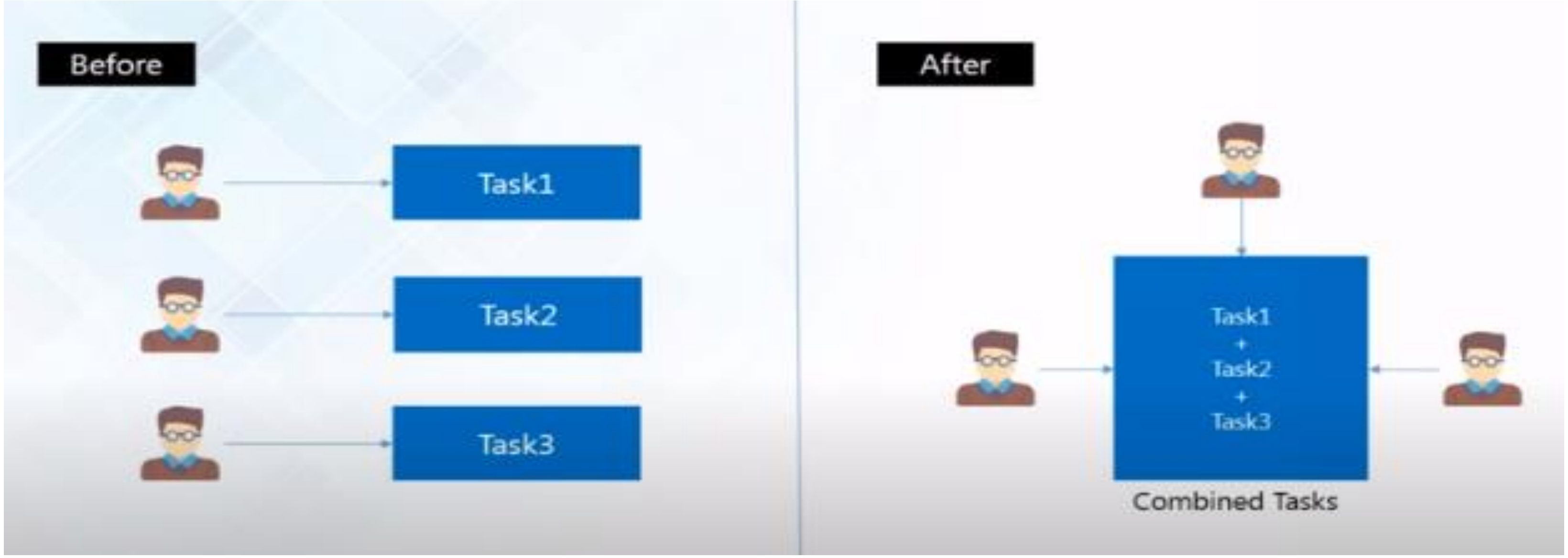
Profile: Quality Analyst

Table Of Contents

1. What Is Version Control System?
2. What is GitHub?
3. What is a Repository in GitHub?
4. How to create Repository in GitHub?
5. What is branching in GitHub?
6. How to create a branch in GitHub?
7. How to use Commit Command in GitHub?
8. How to use Pull Command in GitHub?
9. How to use Merge Command in GitHub?
10. How to Clone and Fork GitHub Repository?

❑ What Is Version Control System?

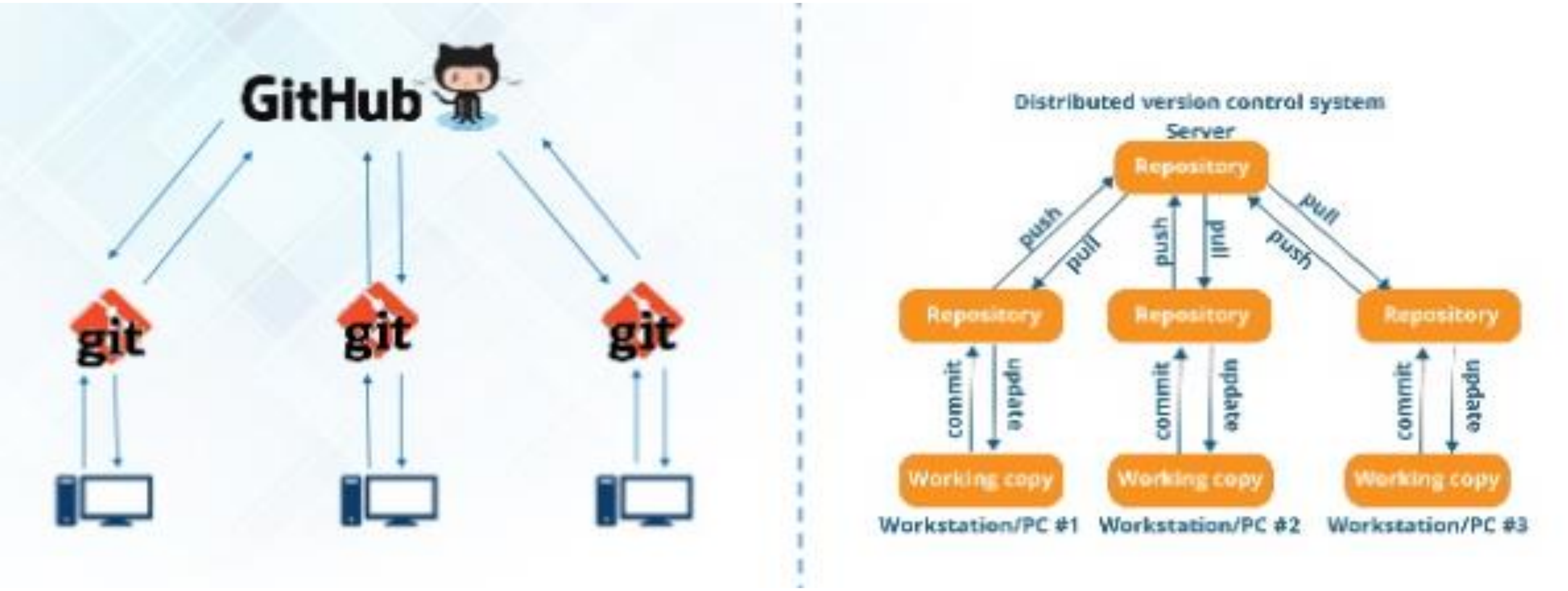
- Version control system is the management of changes to documents, computer programs, Websites, and other information's.
- These changes are usually termed as “Versions”.
- If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members which is the main advantage of version control system.



❑ What is GitHub?

- Git is an open-source version control system that was started by Linus Torvalds—the same person who created Linux. Git is similar to other version control systems like Subversion, CVS, and Mercurial.
- Version control systems keep these revisions straight, storing the modifications in a central repository. This allows developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute.
- GitHub is a Git repository hosting service, but it adds many of its own features. While Git is a command line tool, GitHub provides a Web-based graphical interface. It also provides access control and several collaboration features, such as a wikis and basic task management tools for every project.
- The flagship functionality of GitHub is “forking” – copying a repository from one user’s account to another. This enables you to take a project that you don’t have write access to and modify it under your own account. If you make changes you’d like to share, you can send a notification called a “pull request” to the original owner. That user can then, with a click of a button, merge the changes found in your repo with the original repo.
- These three features – fork, pull request and merge – are what make GitHub so powerful.

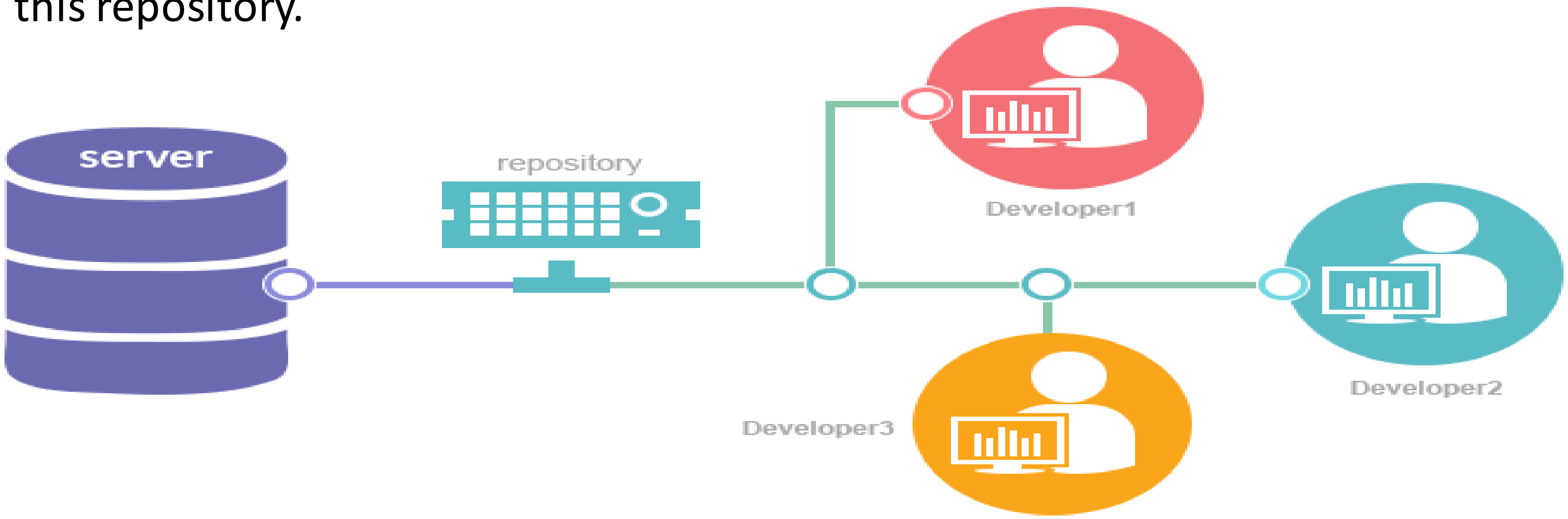
- If you look at the image below, GitHub is a central repository and Git is a tool which allows you to create a local repository. Now people usually get confused between git and GitHub but its actually very different. Git is a version control tool that will allow you to perform all kinds of operations to fetch data from the central server or push data to it whereas GitHub is a core hosting platform for version control collaboration. GitHub is a company that allows you to host a central repository in a remote server.



❑ What is a Repository in GitHub?

- Repository is a directory or storage space where your projects can live. Sometimes GitHub users shorten this to “repo.” It can be local to a folder on your computer, or it can be a storage space on GitHub or another online host. You can keep code files, text files, image files, you name it, inside a repository.
- Repository are of two types:

1. Central Repository: It is typically located on Remote Server. It consists of “.git” repository folder. It is used for the development teams to share and exchange data.
2. Local Repository: It is typically located on local machine. It stays as a “.git” folder inside your project’s root. Only the Owner/ Admin of the machine can work with this repository.

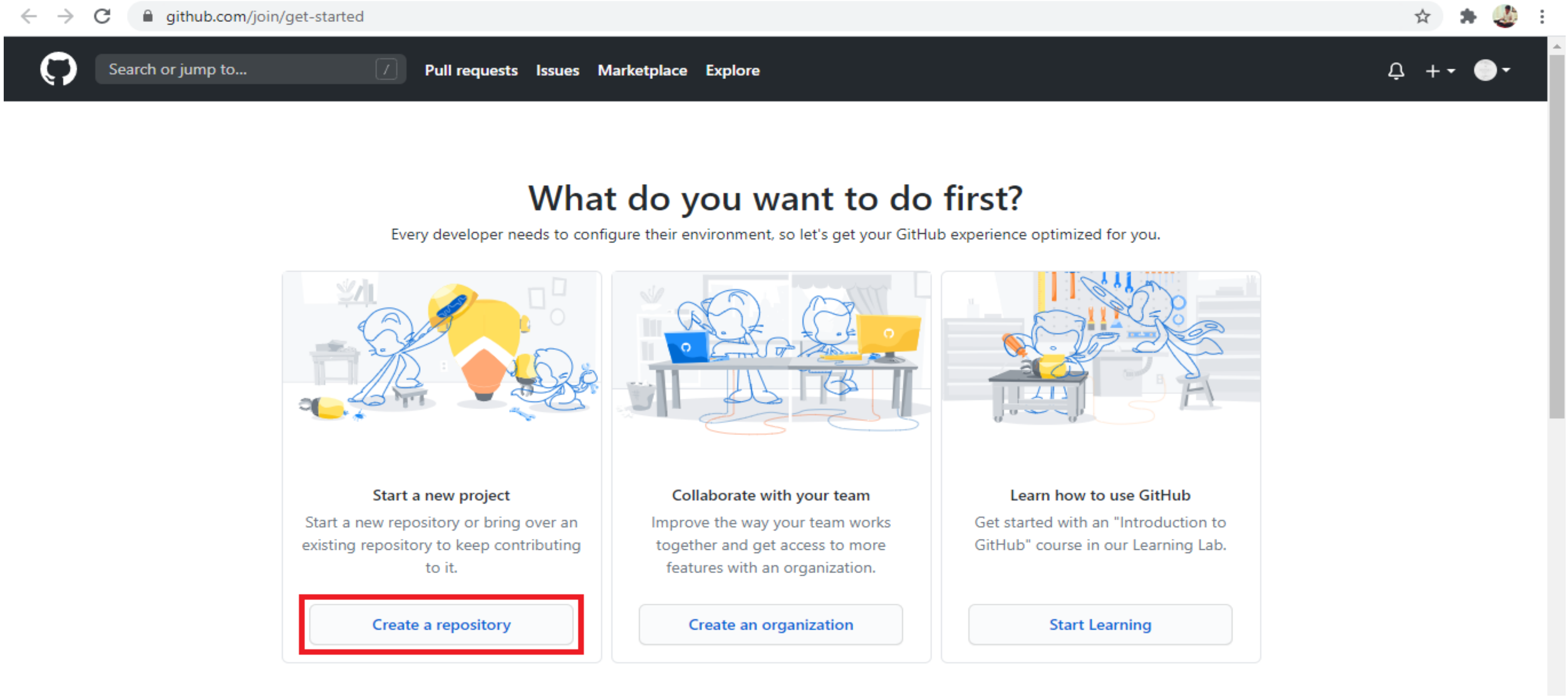


❑ How to create Repository in GitHub?

- You need a GitHub repository when you have done some changes which are ready to be uploaded. The GitHub repository will acts as your remote repository.
- Follow these simple steps to create a GitHub repository:

1. Go to the link: https://github.com/ .
2. Fill the sign up form and click on "Sign up for Github".
3. Click on "Create a repository" button in "Start a new project" tile.

- Refer to the below screenshot to get a better understanding.



github.com/join/get-started

What do you want to do first?

Every developer needs to configure their environment, so let's get your GitHub experience optimized for you.

Start a new project

Start a new repository or bring over an existing repository to keep contributing to it.

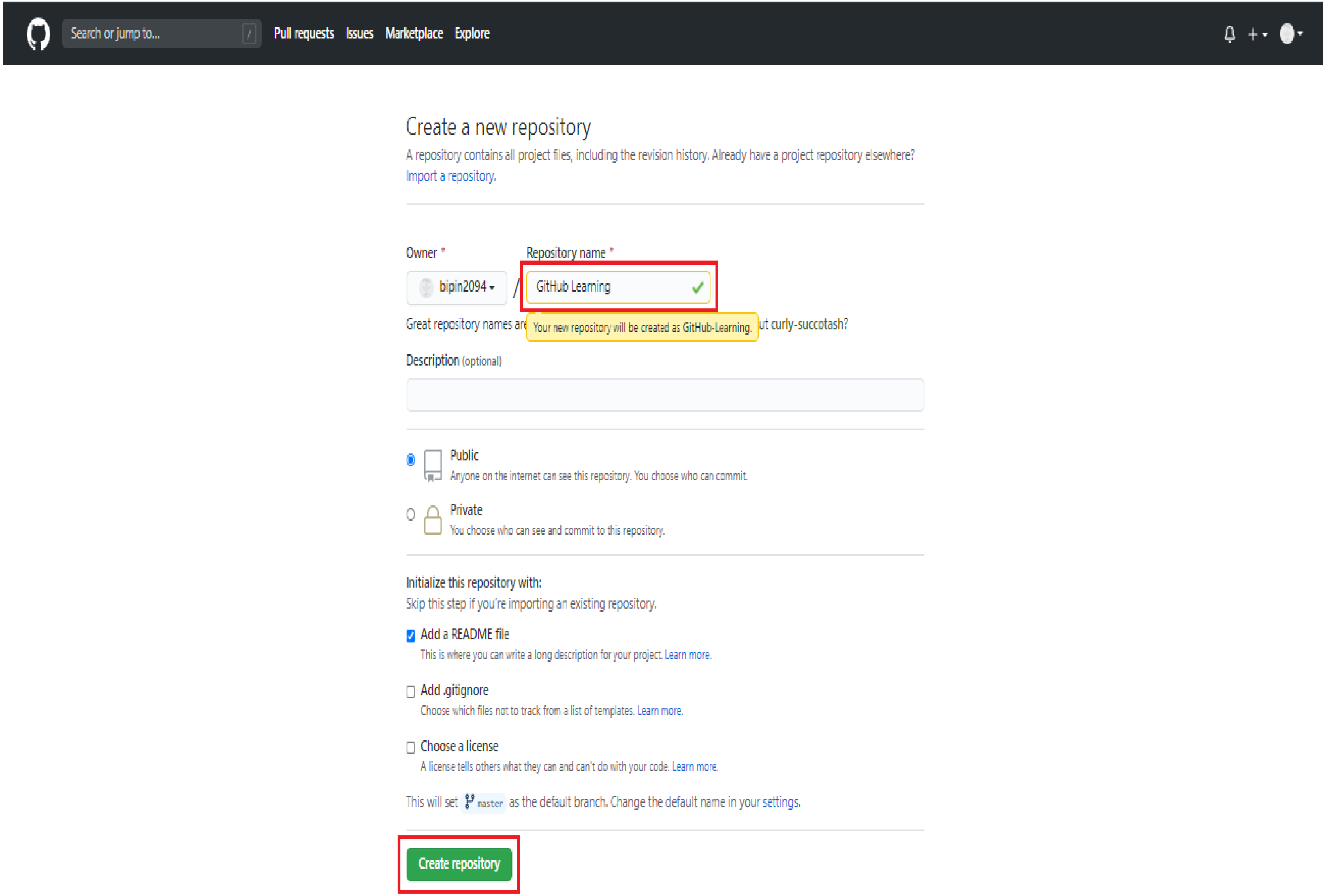
Collaborate with your team

Improve the way your team works together and get access to more features with an organization.

Learn how to use GitHub

Get started with an "Introduction to GitHub" course in our Learning Lab.

4. Enter any repository name and click on “Create Repository”. You can also give a description to your repository (optional).



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Owner * bipin2094 / Repository name * GitHub Learning ✓

Great repository names ar[e] Your new repository will be created as GitHub-Learning. [abo]ut curly-succotash?

Description (optional)

● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☑ Add a README file
This is where you can write a long description for your project. Learn more.

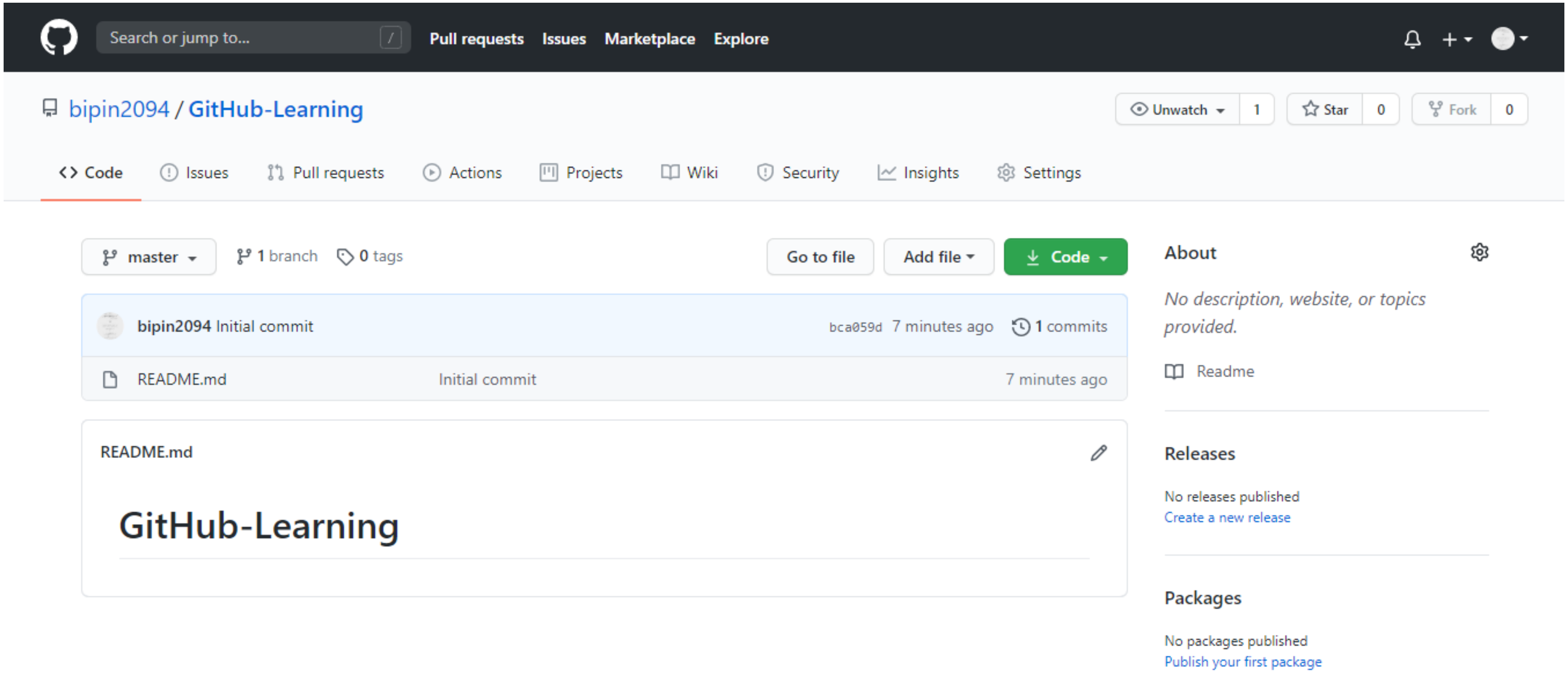
□ Add .gitignore
Choose which files not to track from a list of templates. Learn more.

□ Choose a license
A license tells others what they can and can't do with your code. Learn more.

This will set master as the default branch. Change the default name in your settings.

5. Initialize the repository with a README file. This file contains the description of the file and once you check this box, this will be the first file inside your repository.

6. After successful creation of repository it will look like the below screenshot:



bipin2094 / GitHub-Learning

bipin2094 Initial commit bca059d 7 minutes ago 1 commits

README.md Initial commit 7 minutes ago

README.md

GitHub-Learning

No description, website, or topics provided.

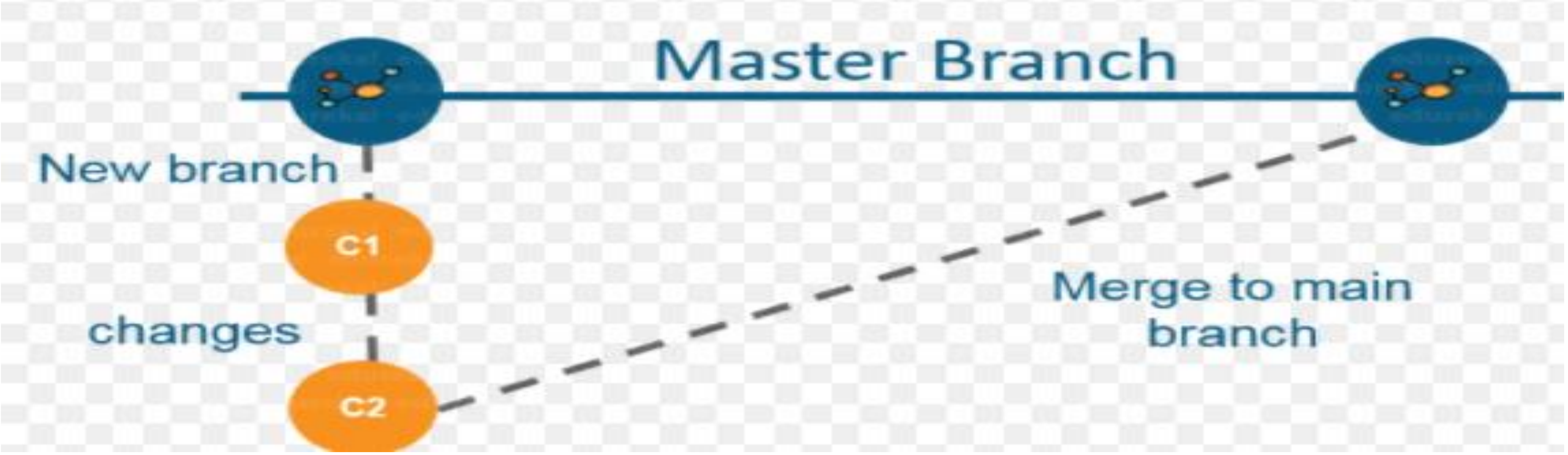
No releases published
Create a new release

No packages published
Publish your first package

7. After successful creation of central repository you are ready to commit, pull, push and perform all the other operations.

❑ What is branching in GitHub ?

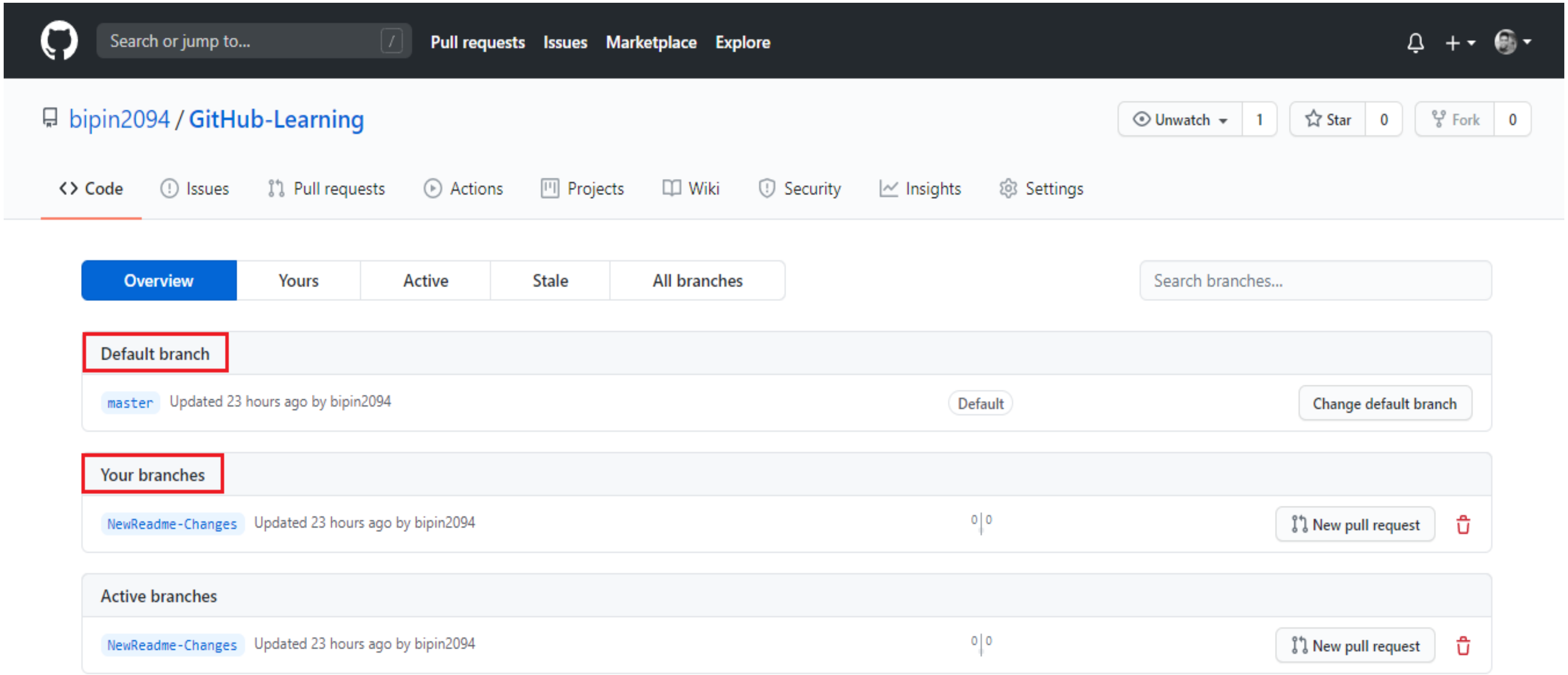
- Branches allow you to move back and forth between the different states/versions of a project. In the above scenario, you can create a new branch and test the new feature without affecting the main branch. Once you are done with it, you can merge the changes from new branch to the main branch. Here the main branch is the master branch, which is there in your repository by default. Refer to the below image for better understanding:
- As depicted in the above image, there is a master/ production branch which has a new branch for testing. Under this branch, two set of changes are done and once it completed, it is merged back to the master branch. So this is how branching works!



❑ How to create a branch in GitHub ?

- Follow the below mentioned steps to create a branch in GitHub:

1. Click on the dropdown “Branch: master”.
2. After clicking on the branch, you can find an existing branch or you can create a new one. In my case, I have created a new branch with a name “NewReadme-Changes”. Please refer the below screenshot for the same.



master Updated 23 hours ago by bipin2094

NewReadme-Changes Updated 23 hours ago by bipin2094 0|0

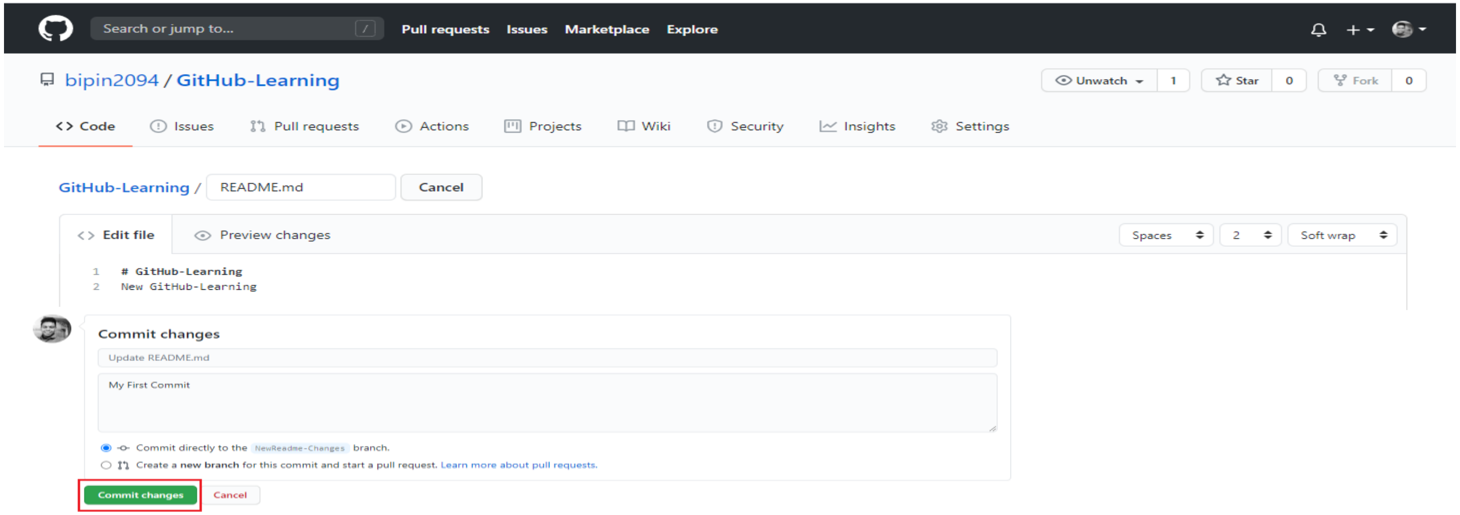
NewReadme-Changes Updated 23 hours ago by bipin2094 0|0

❑ How to use Commit Command in GitHub ?

- This operation helps you to save the changes in your file. When you commit a file, you should always provide the message, just to keep in the mind the changes done by you also follow the below steps our first commit

1. Click on “readme- changes” file which we have just created.
2. Click on the “edit” or a pencil icon in the right most corner of the file.
3. Once you click on that, an editor will open where you can type in the changes.
4. Write a commit message which identifies your changes.
5. Click commit changes in the end.

- Refer to the below screenshot for better understanding:



GitHub-Learning / README.md

```
# GitHub-Learning
New GitHub-Learning
```

Commit changes

Update README.md

My First Commit

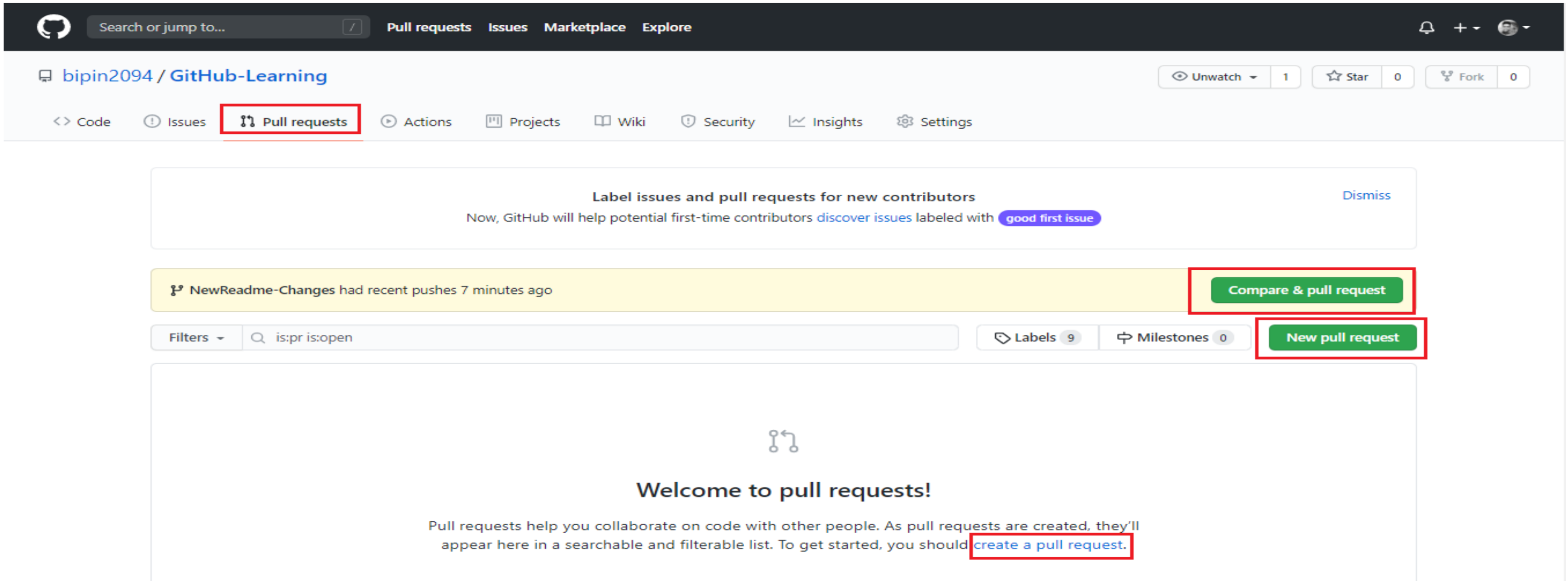
● Commit directly to the NewReadme-Changes branch.

○ Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

❑ How to use Pull Command in GitHub ?

- Pull command is the most important command in GitHub. It tell the changes done in the file and request other contributors to view it as well as merge it with the master branch. Once the commit is done, anyone can pull the file and can start a discussion over it. Follow the below steps involved to pull request in GitHub.

1. Click the ‘Pull requests’ tab.
2. Click ‘New pull request’.
3. Once you click on pull request, select the branch and click ‘readme- changes’ file to view changes between the two files present in our repository.
4. Click “Create pull request”.
5. Enter any title, description to your changes and click on “Create pull request”.
6. Refer to the screenshots in next slide for better understanding.



Label issues and pull requests for new contributors

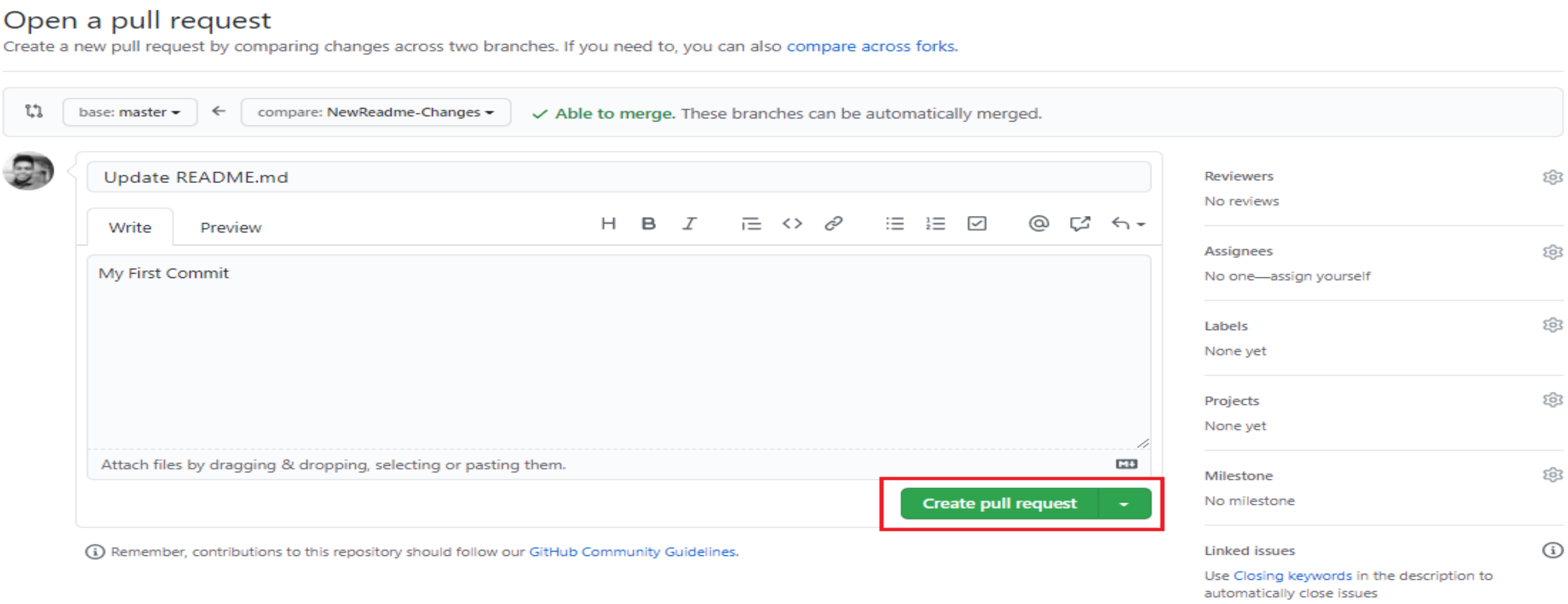
Now, GitHub will help potential first-time contributors discover issues labeled with good first issue

NewReadme-Changes had recent pushes 7 minutes ago

is:pr is:open

Welcome to pull requests!

Pull requests help you collaborate on code with other people. As pull requests are created, they'll appear here in a searchable and filterable list. To get started, you should create a 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.

✓ Able to merge. These branches can be automatically merged.

Update README.md

My First Commit

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

Remember, contributions to this repository should follow our GitHub Community Guidelines.

No reviews

No one—assign yourself

None yet

None yet

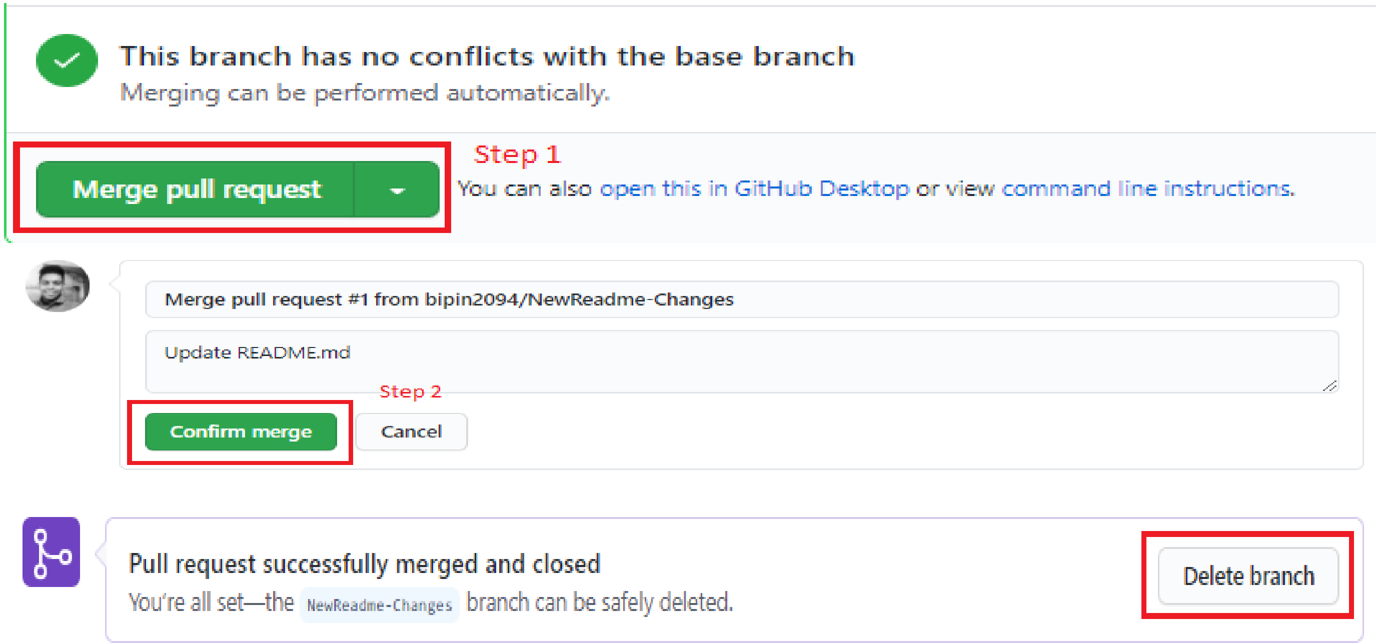
No milestone

Use Closing keywords in the description to automatically close issues

❑ How to use Merge Command in GitHub ?

- Merge command is used to merge the changes into the main master branch.
- Follow the below steps to merge pull request:

1. Click on “Merge pull request” to merge the changes into master branch.
2. Click “Confirm merge”.
3. You can delete the branch once all the changes have been incorporated and if there are no conflicts. Refer to the below screenshots.



This branch has no conflicts with the base branch

Merging can be performed automatically.

Step 1

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

Merge pull request #1 from bipin2094/NewReadme-Changes

Update README.md

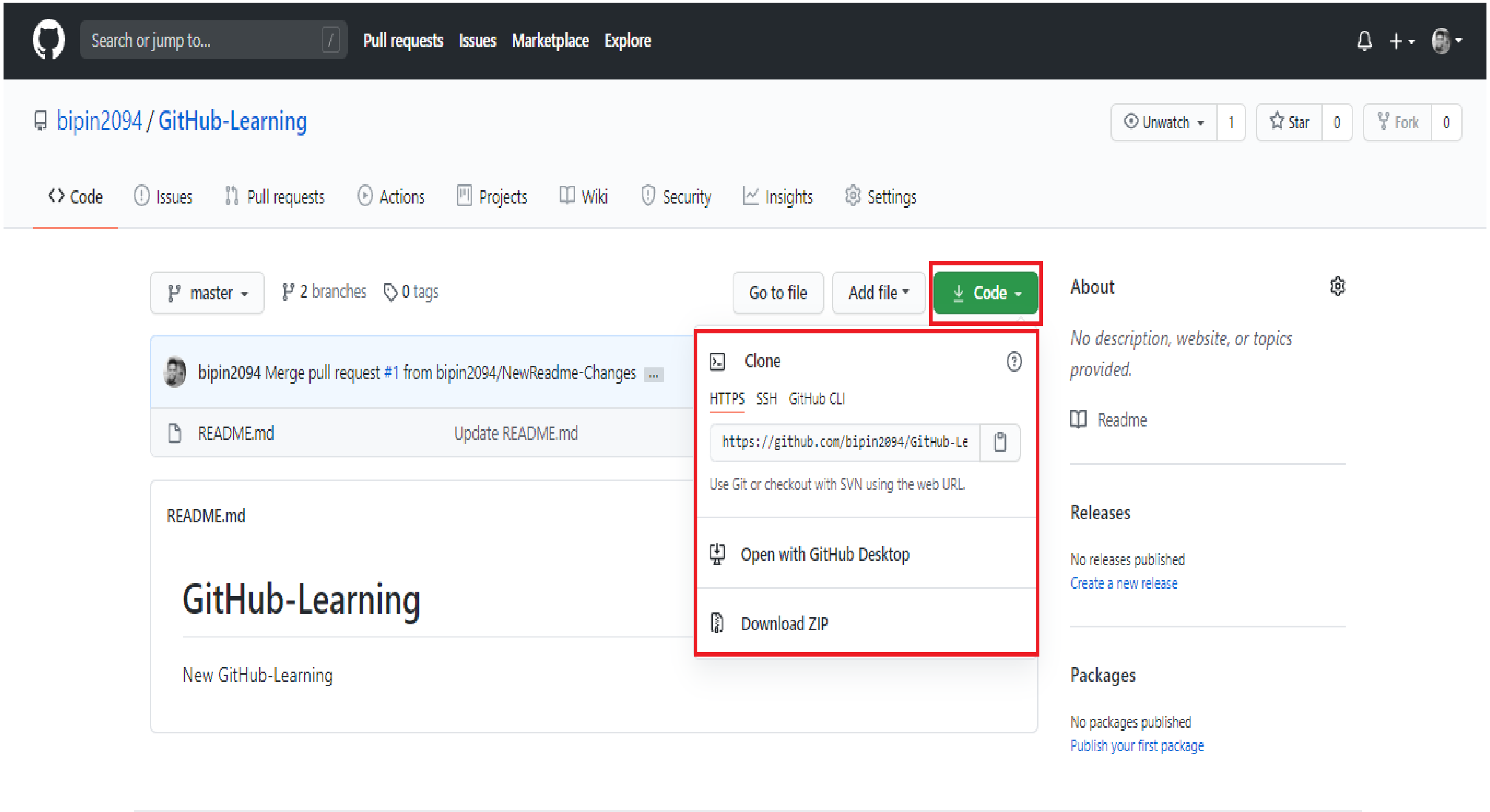
Step 2

Pull request successfully merged and closed

You're all set—the NewReadme-Changes branch can be safely deleted.

❑ How to Clone and Fork GitHub Repository ?

- **Cloning** is done to use some code which is present in a public repository, where you can directly copy the contents by cloning or downloading.
- Refer to the below screenshot for better understanding.



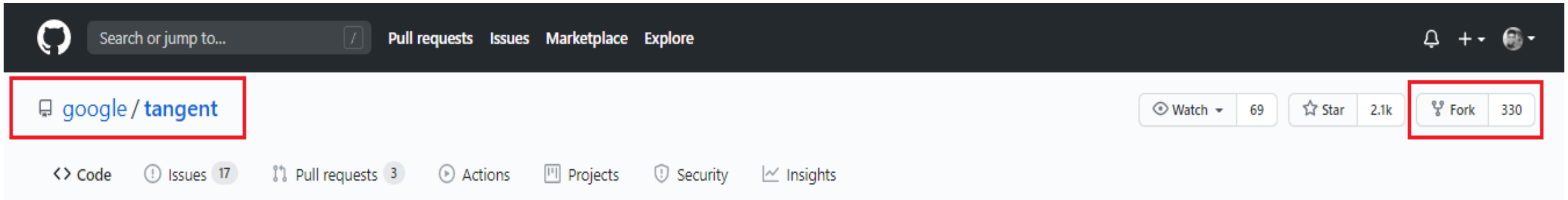
- **Forking:** After using GitHub by yourself for a while, you may find yourself wanting to contribute to someone else's project. Or maybe you'd like to use someone's project as the starting point for your own. This process is known as forking.
- Before we get started with forking, there are some important points which you should always keep in mind.

A. Changes done to the original repository will be reflected back to the forked repository.

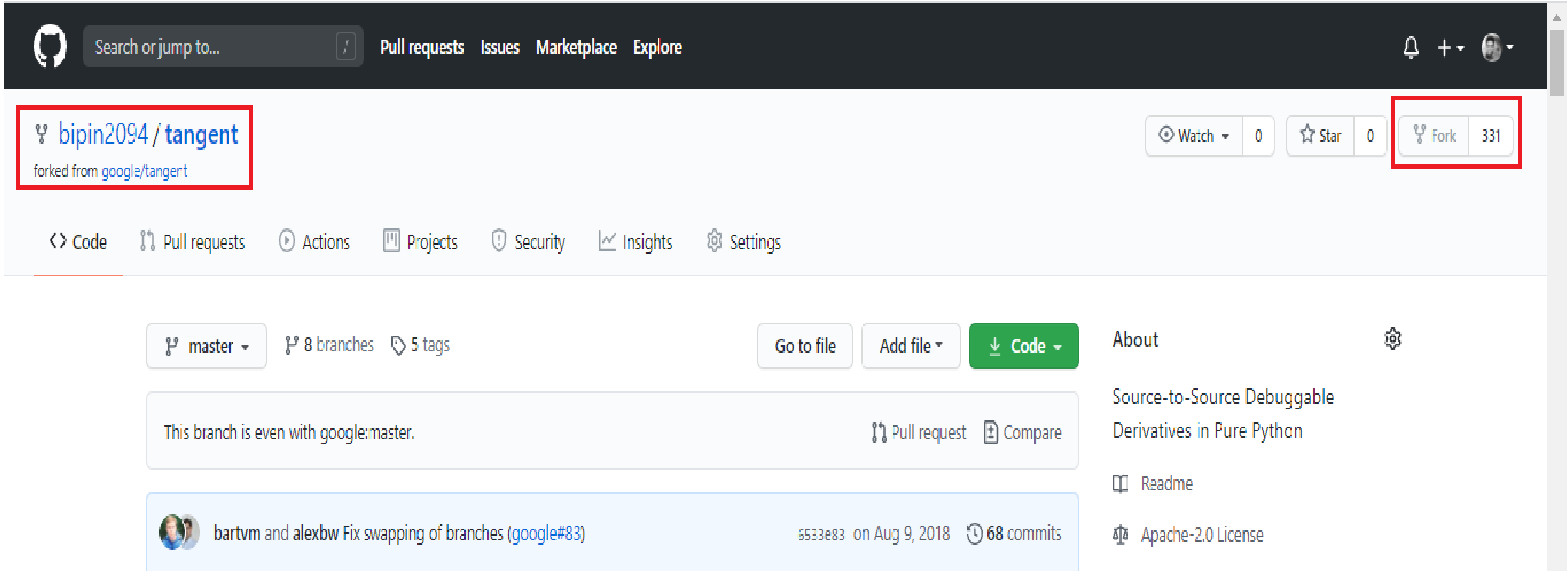
B. If you make a change in forked repository, it will not b reflected to the original repository until and unless you have made a pull request.

- Follow the below steps to fork a repository:

1. Go to Explore and search for public repositories.
2. Click "fork". Note that this "tangent" repository is already forked 330 times and it is under "google" account. Refer the below image for better understanding.



- As soon as you click on “Fork”, it will take some time to fork the repository. Once done you will notice that the repository name is under your account. For reference, you can have a look at the below screenshot.



bipin2094 / tangent

forked from google/tangent

This branch is even with google:master.

bartvm and alexbw Fix swapping of branches (google#83) 6533e83 on Aug 9, 2018 68 commits

About

Source-to-Source Debuggable Derivatives in Pure Python

Thank You!