



PROGRAMMING CLUB

INTRO TO GIT



Presented by Veer and Sujal

What is Git!?

Have you ever named your assignments like
final.pdf, final_final.pdf,
onelasttime_final_final.pdf

And of course ended up wasting storage and
your precious time

What is Git!?

This is where git comes in play

Definition – Git is a version control system

What is version control!?

What is Version Control!?

Version control is a system that keeps track of changes in your project files, like a smart history log. It remembers who changed what, when they changed it, and lets you go back to any previous version if needed.

What if!?

What if the same assignment was actually a group project and you need to share files among your mates and track changes made by everyone at the same time

What if!?

What if the same assignment was actually a group project and you need to share files among your mates and track changes made by everyone at the same time

GitHub



GitHub

That's where GitHub comes in. GitHub is like the cloud and collaboration layer built on top of Git. It lets us put our repositories online, so we can share, collaborate, review each other's work, and even automate workflows

Git and GitHub

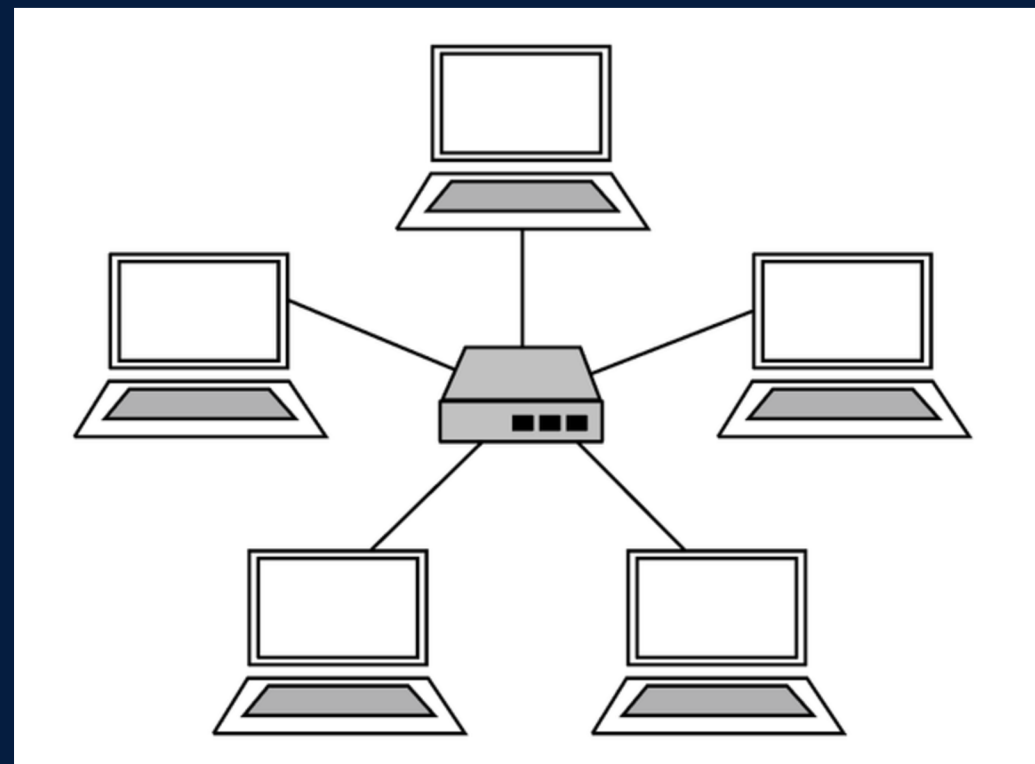
Imagine Git is your phone's Gallery app, which has all photos you clicked on your phone. Where you can add filters, crop and edit images

Now GitHub is basically your Google Photos, where you can upload, make changes, comment(resolving issues), share(forking a repo) and much more for the same photo, even downloading and editing them.

Two step process

You have a cloud (GitHub) and git on your local computer. You make visible changes on the cloud as two steps

- **Update the version in the local computer**
- **Ask Github to make these changes which is currently stores in your laptop**

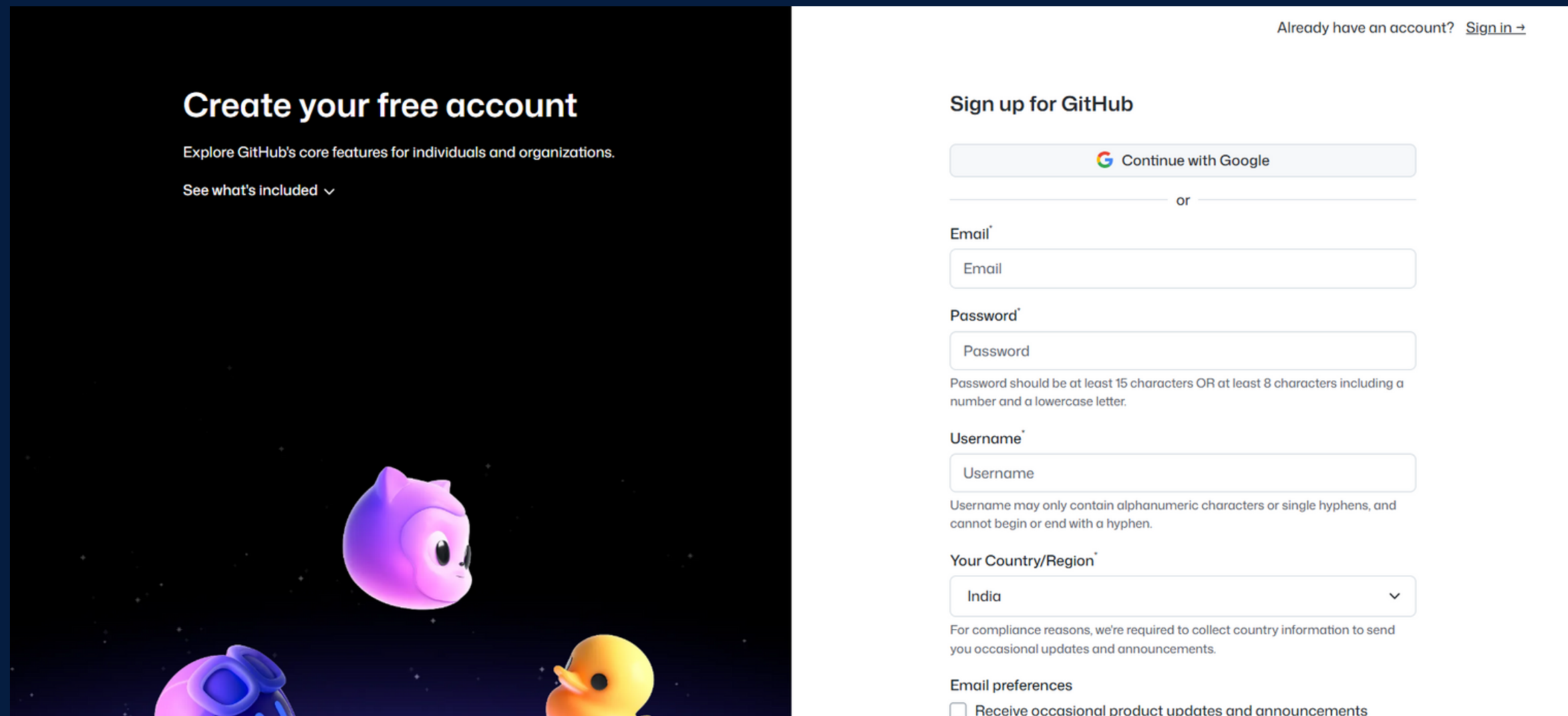


What is a repo

**A GitHub repository (repo) is a centralised storage location which has bunch of folders and files
Managed by git version control system, it serves as the source of collaboration and development**

OK, Cool Stuff

Please set up your GitHub accounts. Its a pretty simple and fast process and please choose a username that you would be ok with for your future purposes.



The image shows a side-by-side comparison of the GitHub account creation process. The left panel, titled 'Create your free account', has a dark background with GitHub's mascot characters (Octocat, Blob, and Duck) at the bottom. It includes a link to 'See what's included'. The right panel, titled 'Sign up for GitHub', is a white form with the following fields: 'Email', 'Password' (with a note: 'Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.'), 'Username' (with a note: 'Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.'), and 'Your Country/Region' (a dropdown menu currently showing 'India'). There is also an 'Email preferences' section with a checkbox for 'Receive occasional product updates and announcements'. A 'Sign in' link is visible at the top right of the right panel.

Already have an account? [Sign in →](#)

Create your free account

Explore GitHub's core features for individuals and organizations.

[See what's included](#) ▾

Sign up for GitHub

[Continue with Google](#)

or

Email*

Password*

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username*

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

Your Country/Region*

India ▾

For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Email preferences

☐ Receive occasional product updates and announcements

Git-Bash

**Git, Please Download Git-Bash, follow
instructions for your respective OS.**

Setting up

```
git config --global user.name "Your Name"
```

```
git config --global user.email "Your email"
```

Setting up

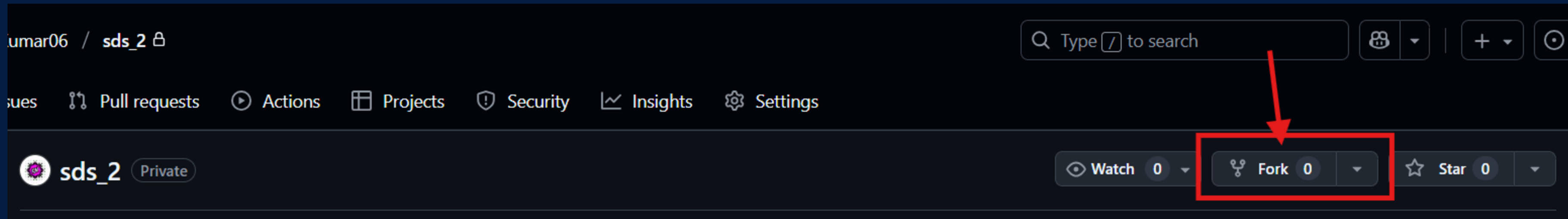
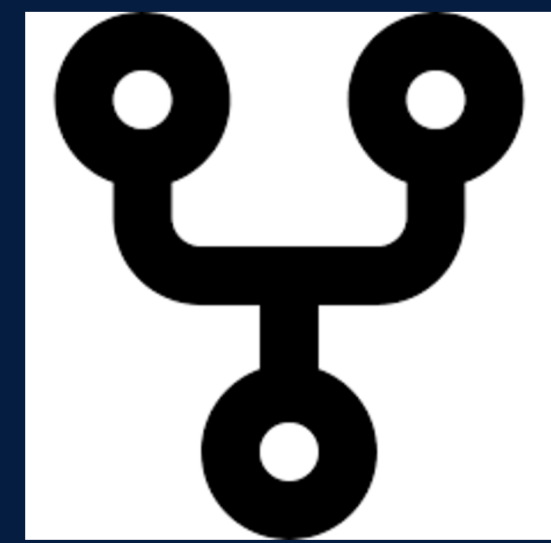
For Windows - It will ask you to set up things during your first push.

For Linux (Ubuntu/Debian) - `sudo apt install gh`

For Mac - `brew install gh`

`gh auth login`

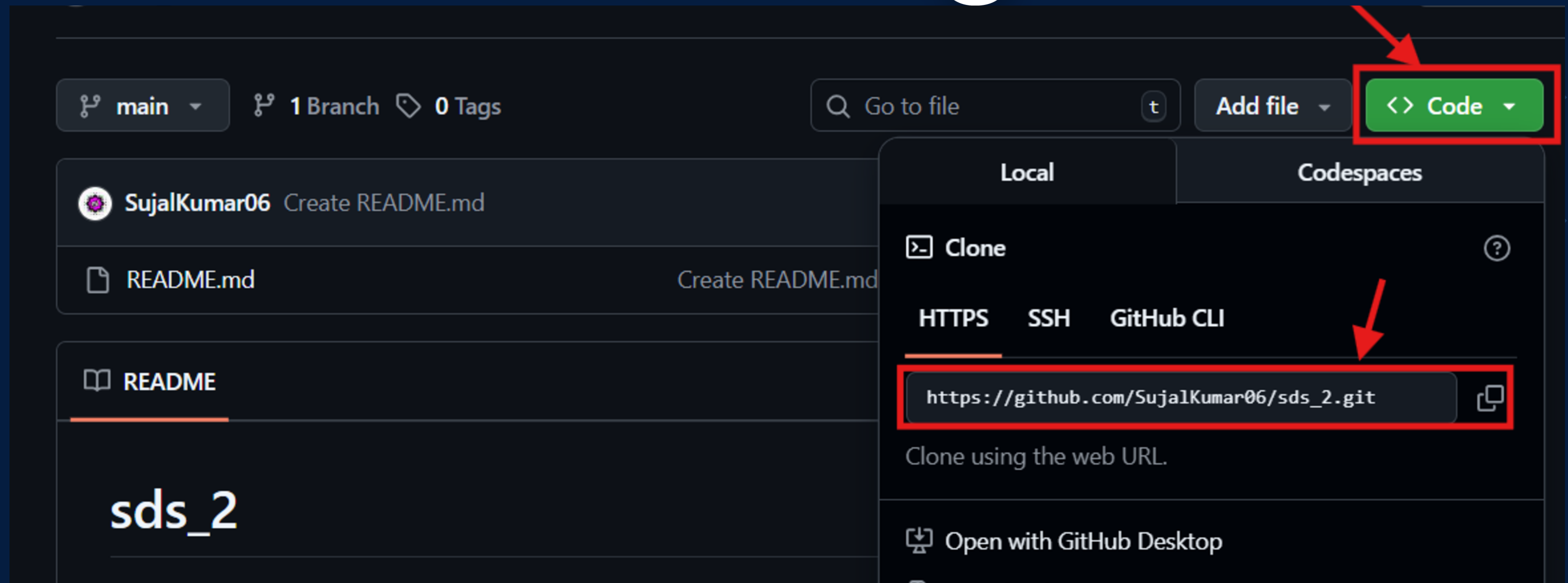
Forking



A fork is your personal copy of someone else's repo.

This copy allows you to freely make changes and experiment without affecting the original repo.

Cloning



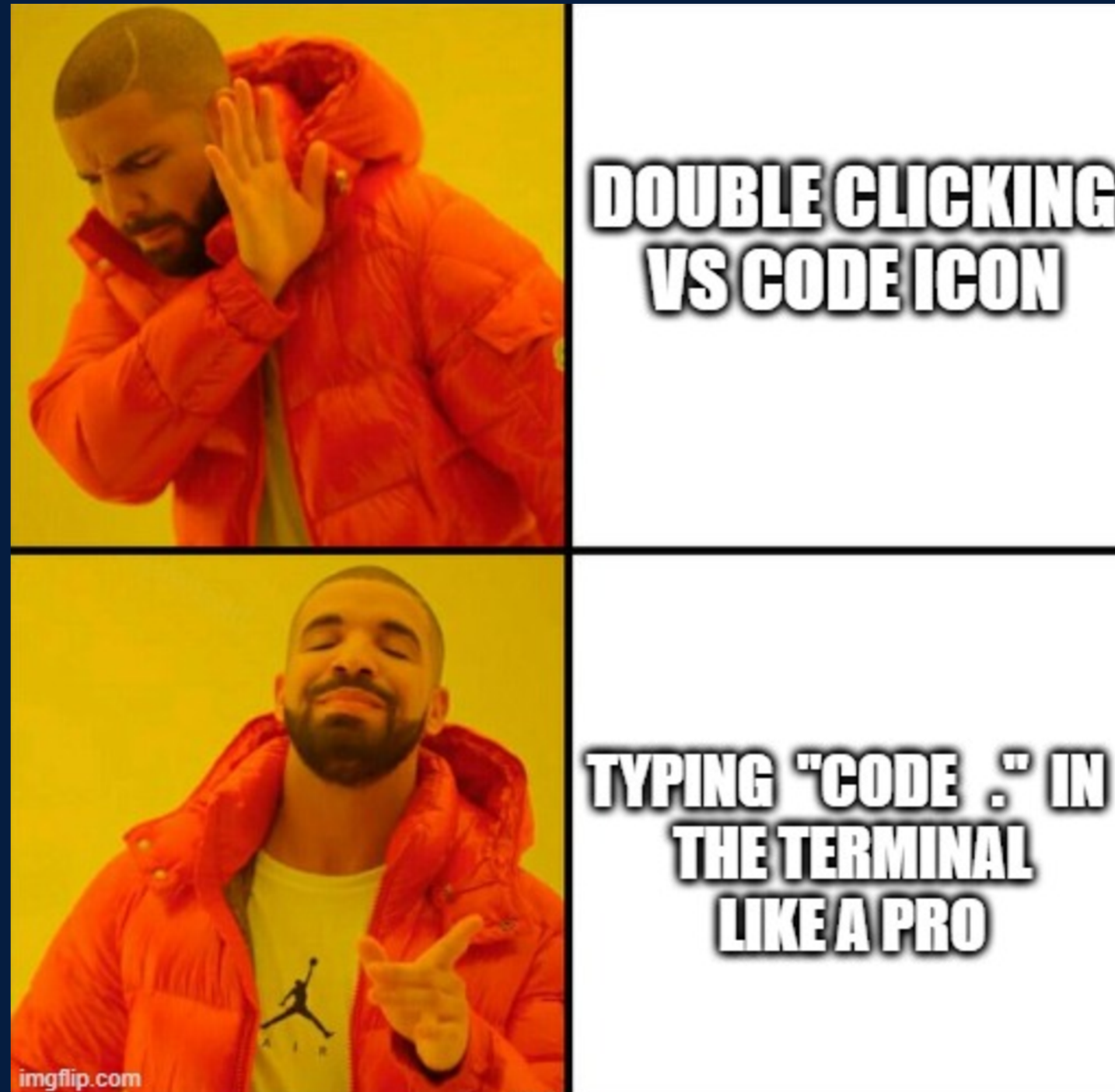
git clone <link>

Making a local copy of your repo.

Opening your local repo

`cd sds_2` → opens the directory in your terminal

`code .` → Opens the current directory in VS Code



Fun Time

Go Ahead and write a Joke/~~Pickup Line~~/Guess
where Yoogi lives/Compliment about Veer
in the designated position in README.md

How to save these changes

git diff → shows the changes that you have done

git add <filename> → Tells git which files or changes you want to keep track of.

git commit -m "commit mssg" → Records the changes that were stages and creates a new version in version history.

Note: git add . → This will stage all the files in the current directory

Now you want everyone to see the changes

git push → pushes all your commits to upstream

Now check your forked GitHub Repo, you would be
able to see the

What If!?

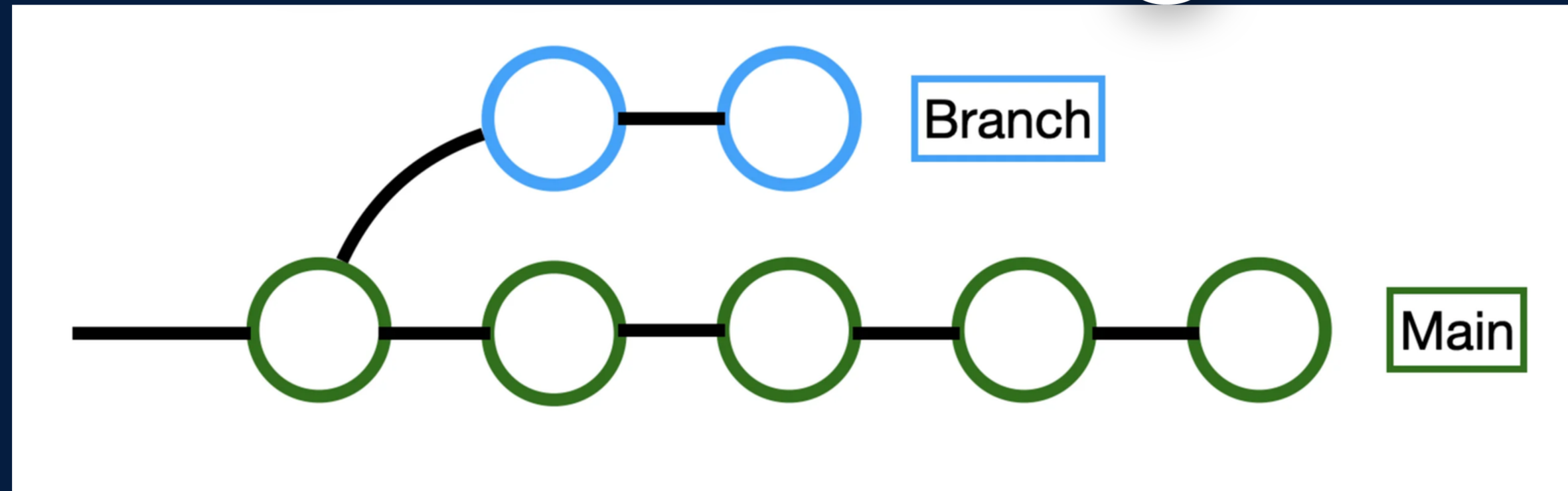
What if you have an assignment and you are doubtful about a question, and you don't want to mess with the assignment file just for that question!?

What If!?

What if you have an assignment and you are doubtful about a question, and you don't want to mess with the assignment file just for that question!?

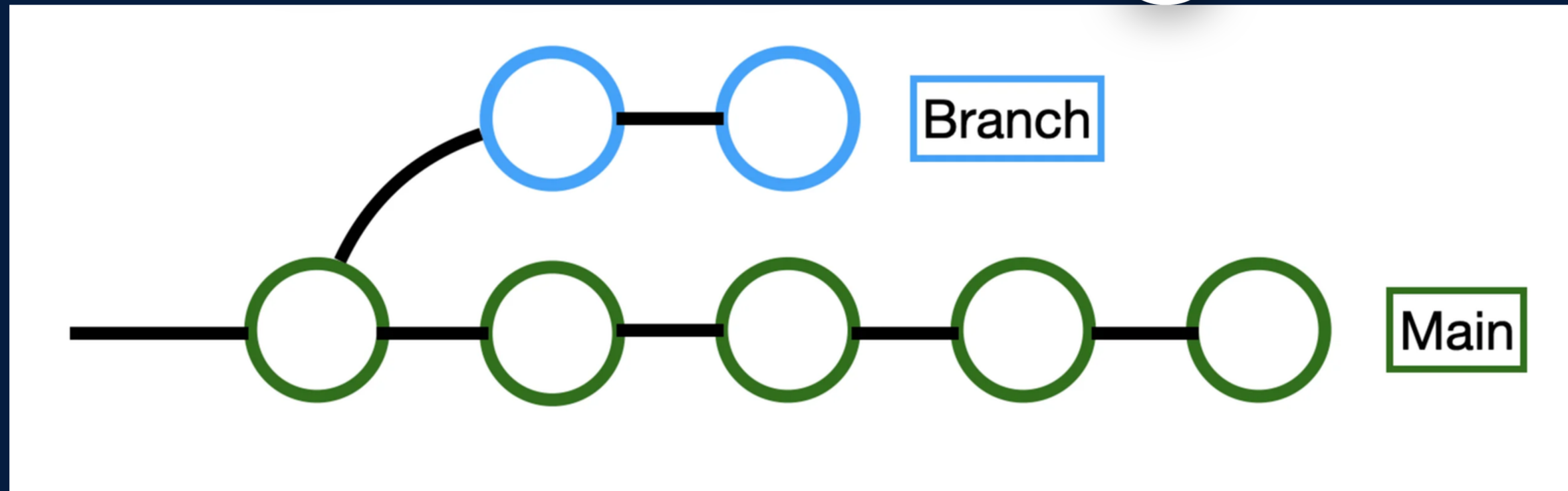
Here is where **branching** comes into play

Branching



Branching is the creation of a separate line of development that allows changes to be made without affecting the main codebase.

Branching



git branch -c "new_branch" → creates a new branch named "new_branch"

git branch → shows the branches and which branch you are on

git switch new_branch → switches to the new_branch

Shoutout time!?

So everyone has written something, now
you want to tell us what you have written.

How do we do that!?

Shoutout time!?

So everyone has written something, now you want to tell us what you have written.

How do we do that!?

Here is where **pull request** comes into play

Pull Request

A pull request is like asking to add your changes into the main repo. Other people review it, give feedback and approve before the code is merged

Pull Request

The screenshot shows a GitHub repository named 'sds_2' which is public and forked from 'SujalKumar06/sds_2'. The repository has 1 branch (main) and 0 tags. A red arrow points to the 'Contribute' button, which is highlighted with a red box. Below this, a modal box displays the message 'This branch is 1 commit ahead of SujalKumar06/sds_2:main' and 'Open a pull request to contribute your changes upstream.' The 'Open pull request' button in this modal is also highlighted with a red box and pointed to by another red arrow. The repository's README file is visible, titled 'sds_2', with the content 'Hey guys. You guys are the coolest ones.'

sds_2 Public

forked from SujalKumar06/sds_2

main 1 Branch 0 Tags

Go to file t Add file <> Code

This branch is 1 commit ahead of SujalKumar06/sds_2:main . Contribute Sync fork

Veer-19 Update README.md

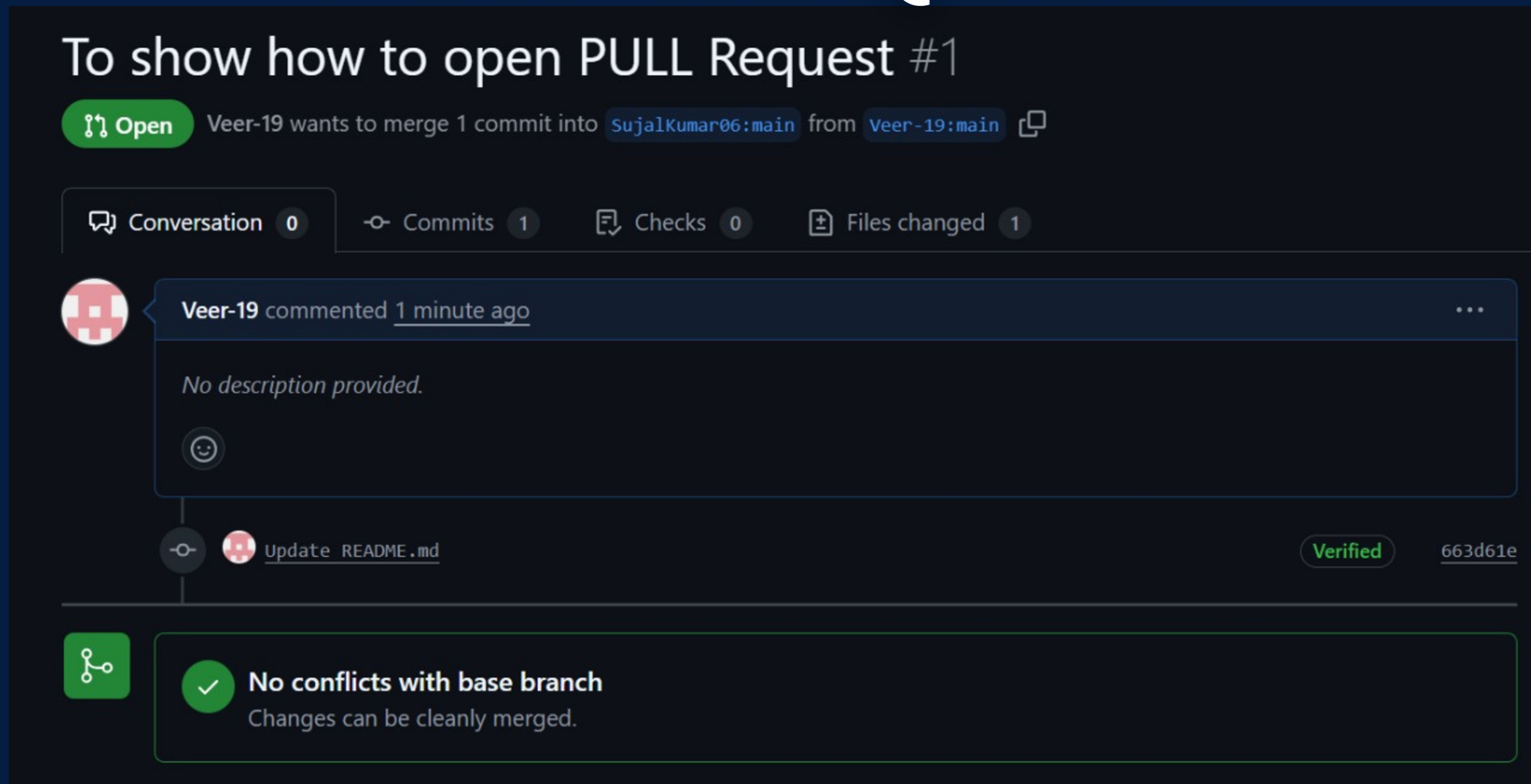
README.md

README

sds_2

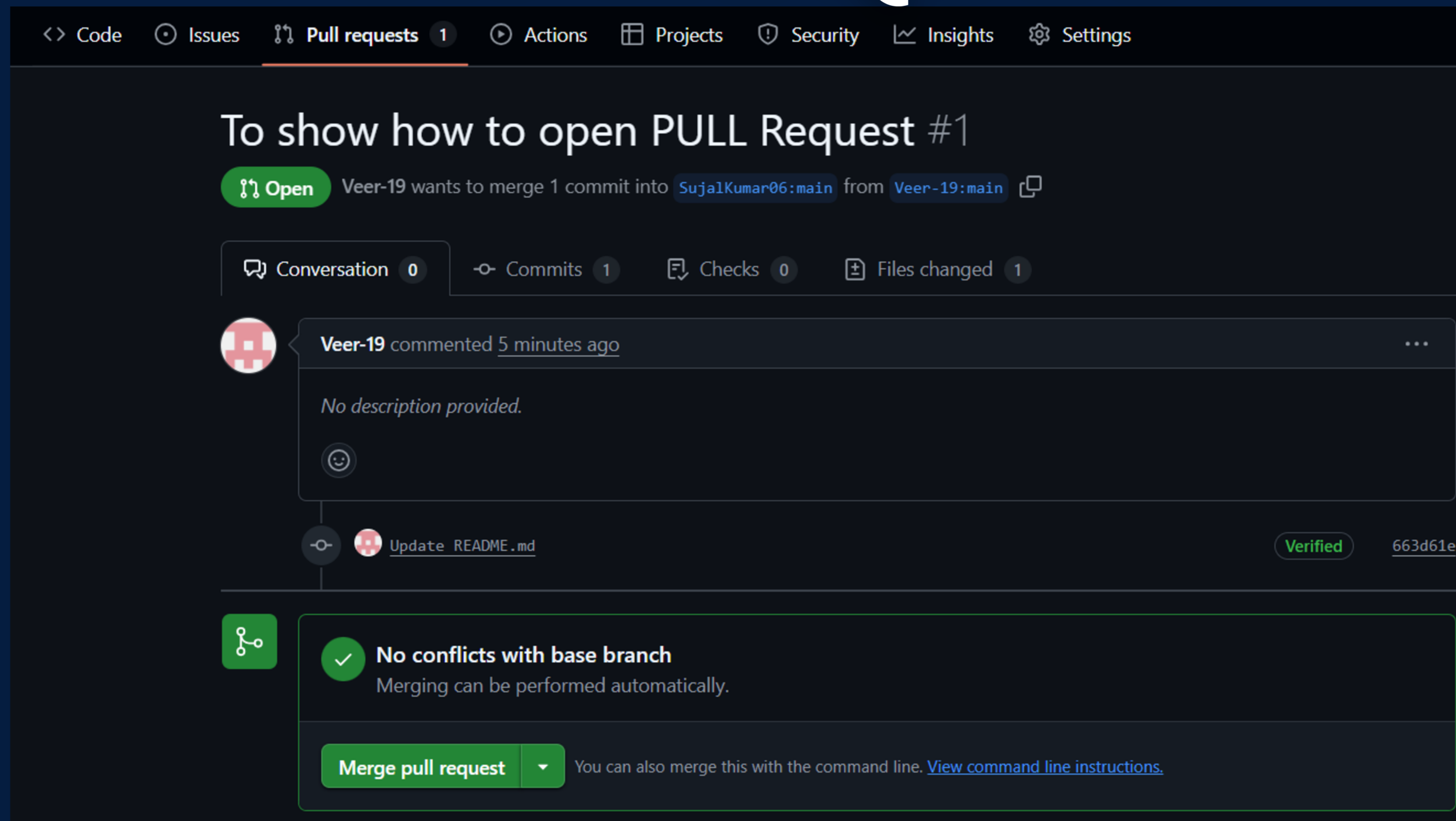
Hey guys. You guys are the coolest ones.

Pull Request



This is how it will look at your screen
(contributor's end)

Pull Request



This is how it will look at our screen
(collaborator's end)

Pull Request

**We have some cool graphs to show you how
these works and looks**

**Open Insights tab in the original repo and
go to Network.**

Pull

Now all of you have opened a pull request to get your joke merged in the repo by the owner but the owner have to choose the best one, so now your local repo has your joke ,not the one which the owner merged in the repo so to keep your local copy in sync with cloud repo(upstream), we use git pull

git pull → Syncs your local repo with your upstream

Are we done!?



No, git has a endless list of functionalities that you can look upon. We just touched upon the most important ones

Though are you excited to see one last thing!?

MERGE CONFLICT!!!!

~~Parijat has given us clear instructions to tell you, if
you ever encounter this, DELETE THE WHOLE REPO!!~~

Jokes Aside.

We will just demonstrate what a merge conflict is.

***Any* Doubts!?**



THANK YOU!!!

