



Before we start...



Git Installed



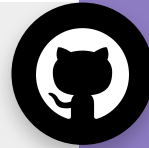
GitHub account



Your favorite editor



Introduction to Git and GitHub





So, What is Git?

A Distributed **Version Control System**.

Any project which uses Git will have a **.git** folder which stores all the history of the project.



History:

Know exactly which files changed, who made those changes, and when those changes occurred.



Backup:

Ability to have different versions of the code in different places.



Collaboration:

Collaborate easily with other people on the same project by uploading and receiving changes



What is GitHub?

GitHub is a website that allows us to use **git** and create repositories **online**. It can also store all your projects online for free.

Let's git
started!



An overview of the
GitHub interface



Let git know who you are



```
$ git config --global user.name "your_username"  
$ git config --global user.email "hello@mail.com"
```

GitHub uses the email address set in your local Git configuration to **associate commits pushed from the command line with your GitHub account.**



My First Repo!


What is a repository?

A repository is a container that houses your project and its' history.

In simpler words, it is a space used to store the files as well as all the changes associated with them.



Initializing a new repository



```
# creating a new folder for our project
$ mkdir MyProject
# changing directory to our project folder
$ cd MyProject
# initializing the current folder as a repository
$ git init
```

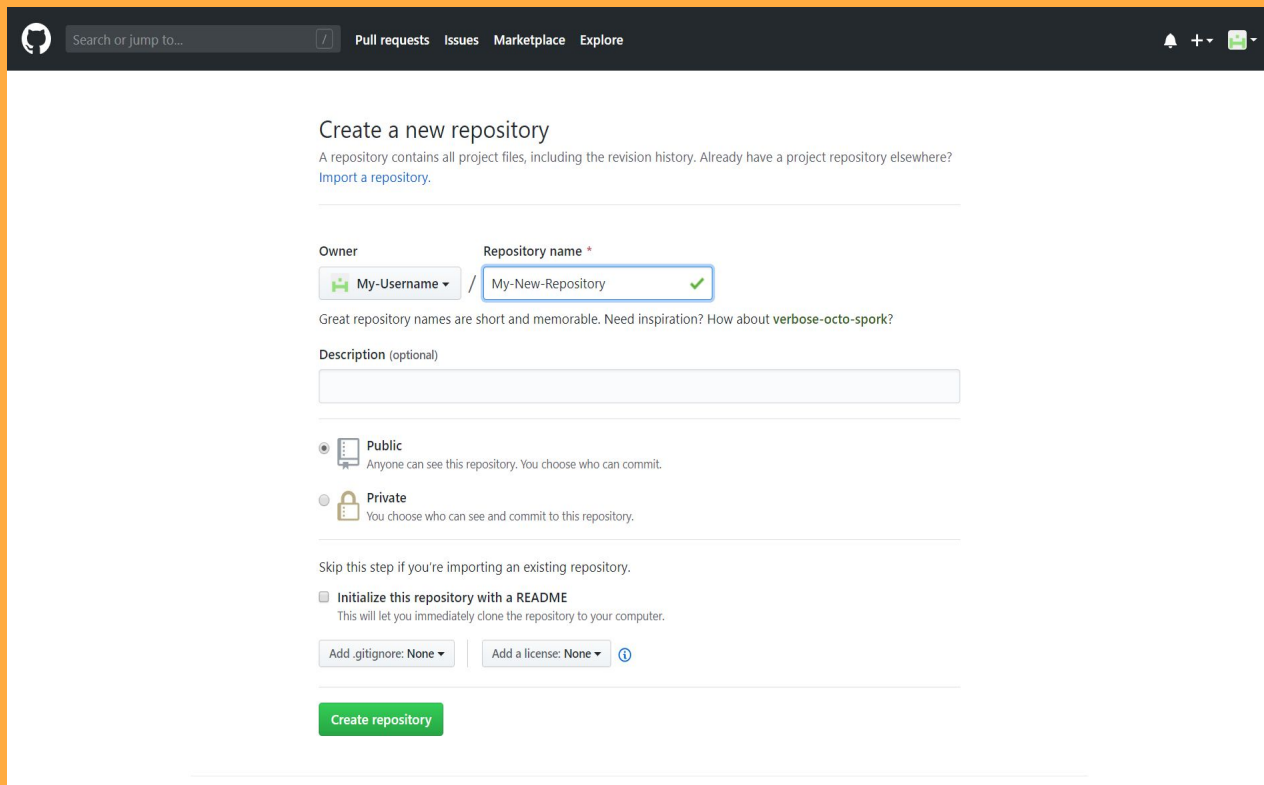
```
Initialized empty Git repository in /home/user/MyProject/.git/
```

Creating a new Repository

Go to

github.com/new

Choose a name for
your new
repository and hit
Create repository



The screenshot shows the GitHub 'Create a new repository' page. At the top, there's a dark navigation bar with the GitHub logo, a search bar, and links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The main heading is 'Create a new repository', followed by a subtext: 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'

The form fields include:

- Owner:** A dropdown menu showing 'My-Username'.
- Repository name:** A text input field containing 'My-New-Repository' with a green checkmark icon to its right.
- Description (optional):** A large text area for entering a description.
- Visibility:** Two radio button options: 'Public' (selected) and 'Private'. The 'Public' option is accompanied by the text 'Anyone can see this repository. You choose who can commit.' The 'Private' option is accompanied by 'You choose who can see and commit to this repository.'
- Initialization:** A checkbox labeled 'Initialize this repository with a README' is checked. Below it, a note says 'This will let you immediately clone the repository to your computer.'
- Additional Options:** Two dropdown menus: 'Add .gitignore: None' and 'Add a license: None', with an information icon to the right.

At the bottom of the form is a green button labeled 'Create repository'.

Quick setup — if you've done this kind of thing before

or

HTTPS

SSH

`https://github.com/Gunnika/abc.git`



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# abc" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/Gunnika/abc.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/Gunnika/abc.git
git push -u origin master
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.


Import code



ProTip! Use the URL for this page when adding GitHub as a remote.

Remote

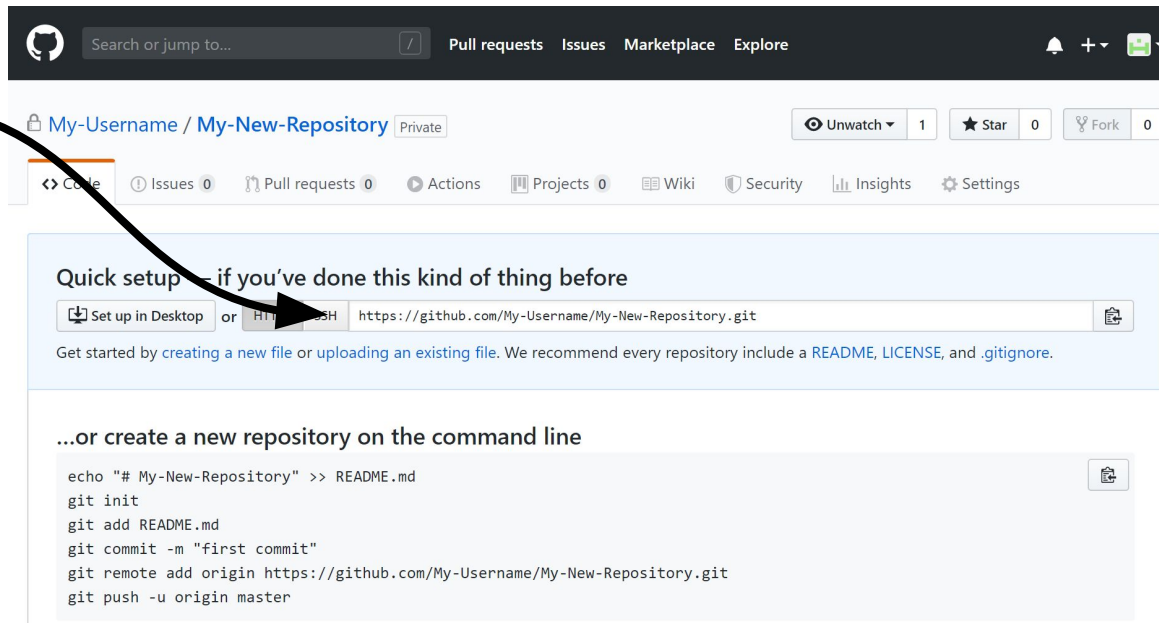
Remotes are the repositories that are available online. They are all referred to by a name so that we don't have to type the whole URL every time.



```
# To set a new remote
$ git remote add <remote-name> <repository-url>
# To view all the remotes
$ git remote -v
```

Pushing changes to your new repository

Copy this URL
and set it as one
of your remote.



The screenshot shows the GitHub interface for a repository named 'My-New-Repository'. The 'Quick setup' section is highlighted, showing the 'SSH' option selected. The URL `https://github.com/My-Username/My-New-Repository.git` is displayed in a text box with a copy icon. Below this, there are instructions to get started by creating a new file or uploading an existing file. Further down, the section '...or create a new repository on the command line' provides a series of terminal commands to initialize a repository, add a README file, commit, and push to the remote.

```
echo "# My-New-Repository" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/My-Username/My-New-Repository.git
git push -u origin master
```

Commits

Checkpoints/Snapshot of the state of your repository (project) at a particular time.



Creating your first commit



```
# shows the state of the working directory and the staging area.  
$ git status  
# Add the files to staging area  
$ git add fruits.txt  
# Commit the changes into the repository  
$ git commit -m "Add fruits.txt"
```

Downloading/Updating repos



```
# To download a remote repository
```

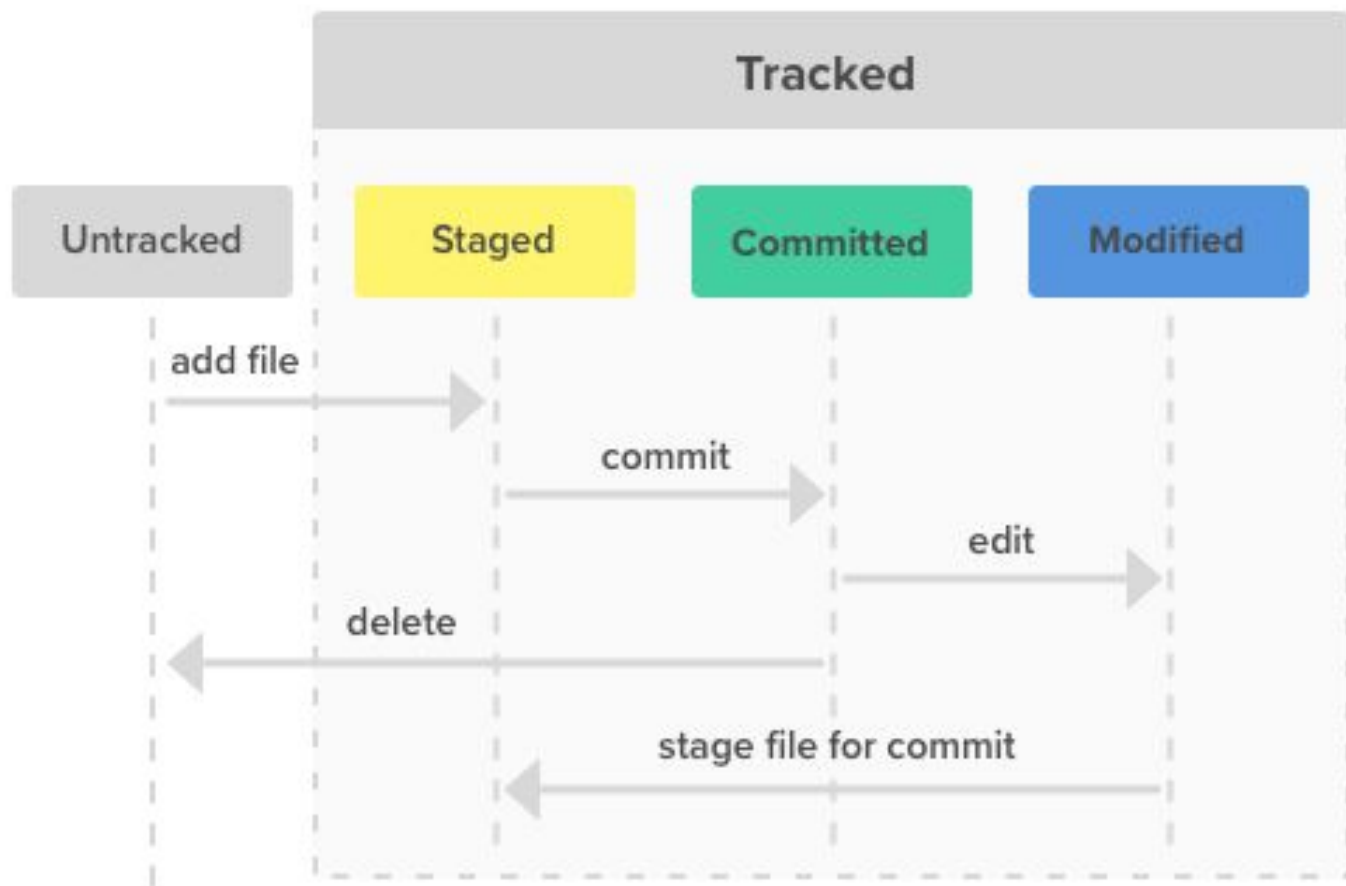
```
$ git clone <repository url>
```

```
# To grab changes from a remote repository and add to yours
```

```
$ git pull
```

```
# To submit your changes to remote repository
```

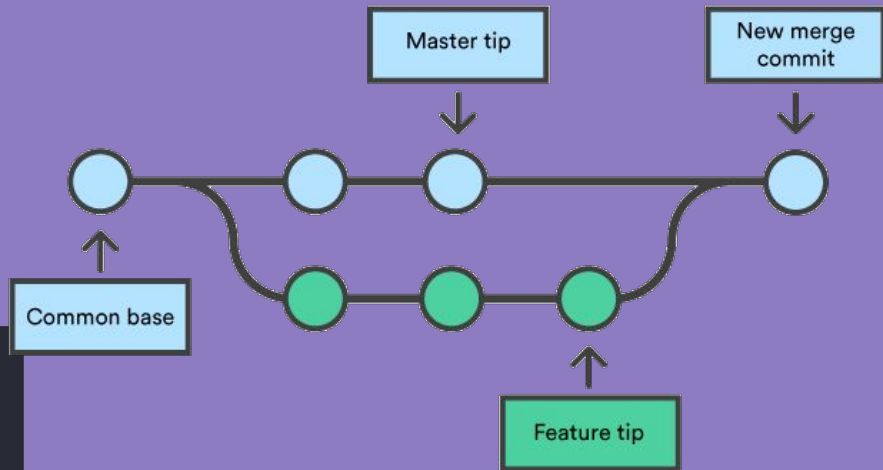
```
$ git push
```



Git Branches



```
# To list all branches
$ git branch
# To create a new branch
$ git branch MyBranch
# To change the control to new branch
$ git checkout MyBranch
# To merge two branches together
(master)$ git merge MyBranch
```



Fork and Clone

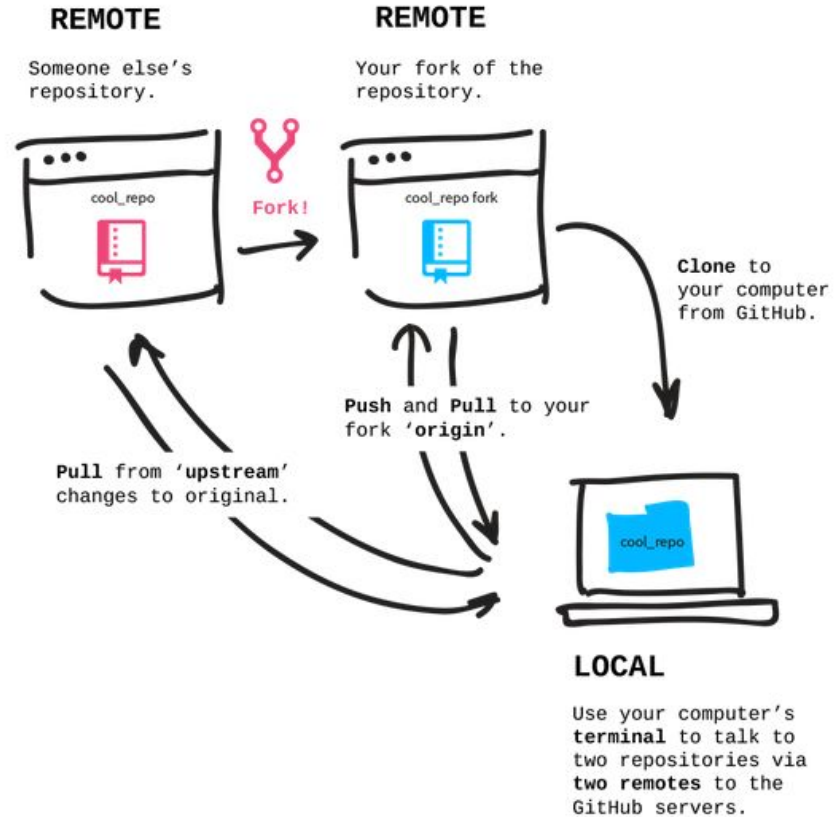
Both mean making copies, so how are they different?

Fork = A copied repository

A bridge between the original repository and your personal copy where you can contribute back to the original project using Pull Requests.

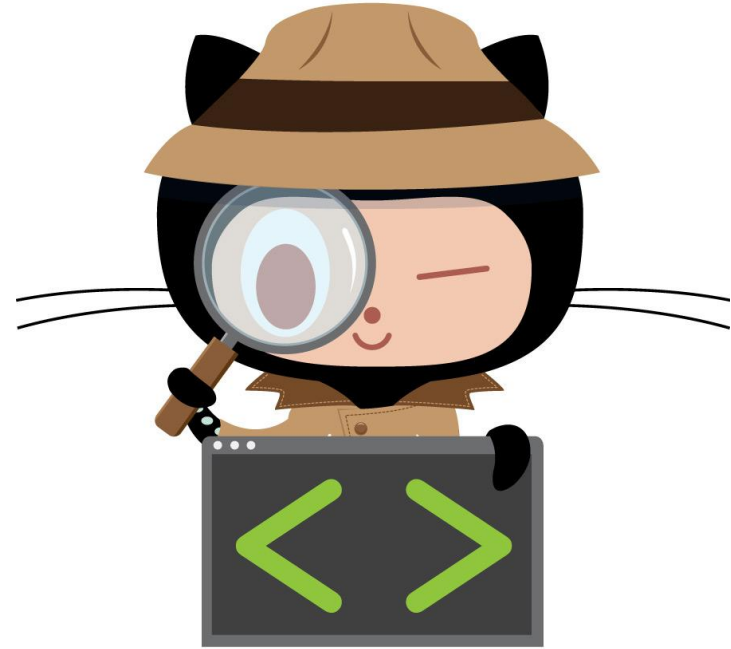
Clone = A local copy on YOUR machine

If the project is owned by someone else you won't be able to contribute back to it unless you are specifically invited as a collaborator.



Pull Request

Pull requests are proposed changes to a repository submitted by a user and accepted or rejected by a repository's collaborators.



But why not 'push' request?

When you file a pull request, all you're doing is requesting that the project maintainer **pulls** a branch from **your repository** into **their repository**.



What can we do with
GitHub?
Well, what not?

GitHub Education

Programs and **technology leadership** at your school



GitHub
Student Developer Pack



GitHub Classroom



GitHub
Campus Experts



GitHub Campus Advisors



GitHub

Student Developer Pack

Gives students free access to the best developer tools in one place so they can 'learn by doing'.

This year, it has been **doubled** in size with **21 new partners**—and offering **nearly \$45,000 in savings available to students for free.**

- Unique links
- Coupon codes
- GitHub login or direct links to grant access

<https://education.github.com/students>



With Canva, anyone can create professional looking graphics and designs. Featuring thousands of templates and an easy to use editor.

Benefit Free 12 month subscription of Canva's Pro tier.

aws  **educate**

Access to the AWS cloud, free training, and collaboration resources

Benefit Student Developer Pack members receive up to \$110 in bonus AWS credits for a total of \$75-\$150.



A hackable text editor for the 21st Century

Benefit Open Source by GitHub, free for everyone.



A powerful domain extension to convey that you belong to the technology industry.

Benefit One .TECH domain free for 1 year.



GitHub
Campus Experts

Learn skills to enrich the technology community at your campus.

- Start your application
- Application approved
- Complete 7 modules
- Create a pull request
- Improve with reviews
- Pull Request merged
- GitHub Campus Expert!

<https://githubcampus.expert/training>



EVENTS BY GitHub



<https://githubuniverse.com>




<https://hacktoberfest.digitalocean.com>




<https://itch.io/jam/game-off-2019>





Lead - Women Techmakers
BVP 

Campus Expert - GitHub 

Data Analytics Intern, Campus
Lead- SHEROES 

 @ShauryaAg

 ShauryaAg

 /in/shauryaag

 @GunnikaBatra

 Gunnika

 Gunnika Batra

 Web Executive - DSC BVP

