

WHAT IS GIT??

- Git is the version control system.
- The version control system is used to track the changes that are occurring in the code.
- When we are supposed to create a small-scale or a large-scale project, the version control system tracks the entire project.

FEATURES OF USING GIT:

- **Popular**
- **Fast and scalable**
- **Free and the Open Source**

NOTE: GIT IS USED IN COLLABORATION ON LARGER BASIS.

WHAT IS GITHUB??

- GitHub is the website where the developers manage their code using Git. Generally, the code of the projects is stored in the folder which is also termed as the repository in the language of Git.
- README is a special type of file wherein the details of the project we are making are included. Markdown(.md) is the extension used for specifying the project details.

- **Initial commit i.e.** commitment(change/updation) done in an initial phase.
- In Git, adding of the changes takes place but in Github, Commitment (Final execution) to the added change will take place.
- The basics of HTML can be used **to commit the changes.**
- **Some Basic Commands of Git:**

I) ls: Provides the list of folders, files, and recent directories.

II) clear: Clears all the working directories.

III) pwd: Moving to the current directory.

- **Git is configured at 2 levels: Global(For all the existing repositories)and local level(For the specified repository).**

METHODS TO CONFIGURE(In which account are we going to make changes) THE GIT:

- **git config - -global user.name (Name of the username)**
- **git config - -global user.email (Email Address)**
- **git config - -list**

The Tilde Symbol (~) represents the root directory(folder). GitHub contains the credential helper(A storage where credentials are stored). Credentials are the basic details of a user who has created an account on GitHub.

CLONE AND STATUS

- **Clone** - Cloning the repository on a local machine.

For Eg: A place named remote(Repositories of GitHub are stored) and another place named local(Laptop/Personal Computer). So, Keyword Clone is used to copy the repositories of the GitHub on the laptop/PC.

Syntax: **git clone** "Project ka link"

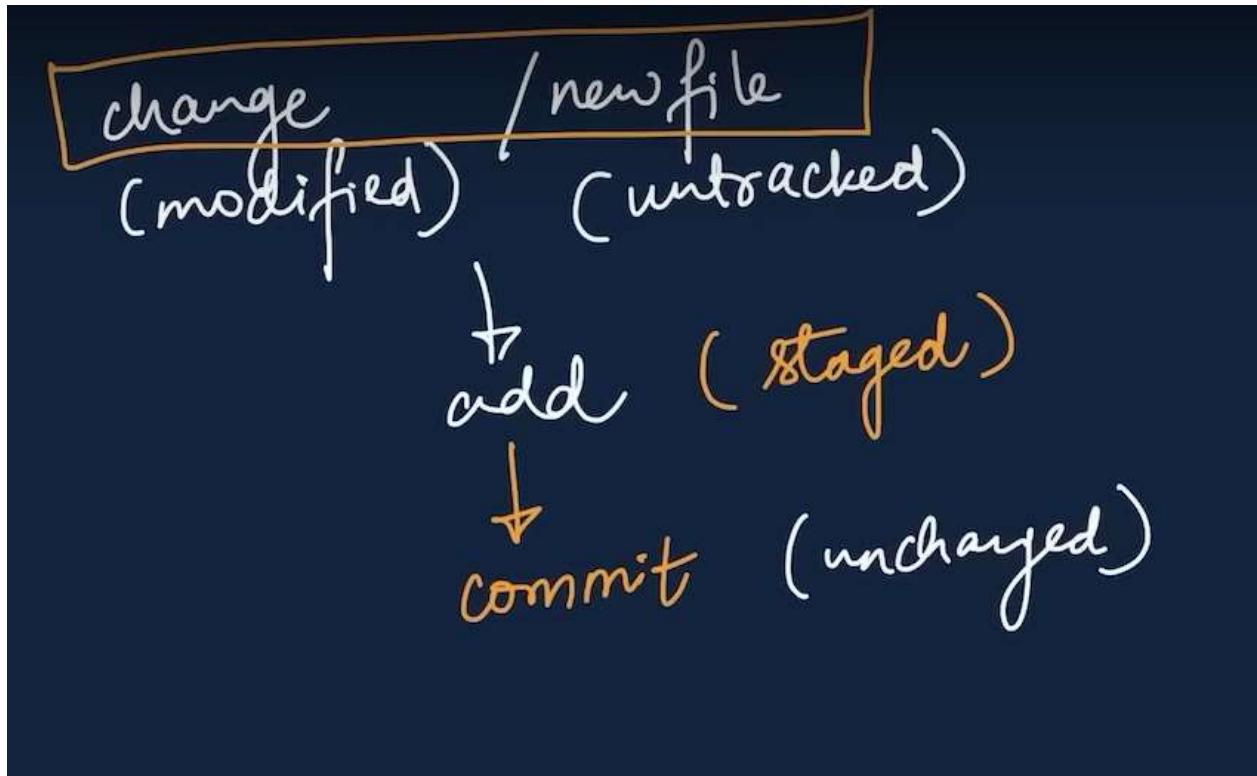
- **cd** is the **change in the directory**. **ls** is the command to list **the existing files and the directories**. **ls -a** is the command to list the hidden and existing files and the directories(In Mac).
- **Status**: Displays the status of the code present in the repositories.

Syntax: **git status**

Primarily, there are 4 types of files in Git:

- **Untracked**: New files in git that don't yet track
- **Modified**: Changed
- **Unmodified**: Unchanged
- **Staged**: The file is ready to be committed.

Firstly, files that are under the category of Untracked/Changed(Changes done in the code present in the repo) need to be staged(File is ready to commit) and finally i.e. once the process of commitment is done, is shifted to Unmodified(Unchanged).



Add and Commit:

Add & Commit

add - adds new or changed files in your working directory to the Git staging area.

```
git add <- file name ->
```

commit - it is the record of change

```
git commit -m "some message"
```

```
graph LR; A[ ] --> B[add (engage)]; B --> C[commit (wudu)];
```

NOTE: In the terminal of VS Code, if the color for any of the file names becomes green, then that file is staged. If we need to add all the files present in the git repository, then Syntax is: “git add .”

This means that Your branch is ahead of 'origin/main' by 2 commits You need to modify the 2 updates on the git repo/ Ahead of the 2 commits on VS code as compared to the main repo.

PUSH COMMAND: Used to upload the local content i.e. content present on the VS code to the remote repository(main repository).

To come out of the current directory, Syntax is: **cd - -**

To make the new directory, the Syntax is **mkdir “Name of the new directory”**.

If the directory is created with the help of the mkdir command, and if it is not the git directory, **then the syntax is git init**. The meaning of init is: Create a new git repository.

Before using the syntax: git push origin main, Following steps to be followed :

- 1) git remote add origin(Link of the new remote repository),
- 2) git remote -v (Verify the new remote)
- 3) git branch (to check branch)
- 4) git branch -M main(to rename branch)
 - Where main is the new name of the branch.
- 5) Then use this syntax to push the local repo to the remote repo,
Syntax is: git push origin main.

WORKFLOW:

I) Creating the GitHub Repo

II) Clone to the Local Repo

III) Add the changes

IV) Commit the changes

V) Push again to the GitHub Repo.

BRANCH COMMANDS:

`git branch` (to check branch)

`git branch -M main` (to rename the branch)

`git checkout <Name of the second branch>` (to navigate {to move from one branch to another branch} the branch)

`git checkout -b <Name of the branch>` (to create the new branch)

`git branch -d <Name of the branch which is to be deleted>`

NOTE: We cannot delete the branch on which we are currently present.

PULL REQUEST

METHOD - 1:

1) `git diff <Name of the branch with whom to be compared>` (Used to compare commits, branches, and files).

2) `git merge <Name of the branch to be merged>`

METHOD - 2:

- 1) Create a Pull Request using GitHub (It tells the others that the changes have been pushed to a branch in the repository of GitHub.)

PULL COMMAND

It is used to fetch and download the content from the remote repository and immediately update the local repository.

RESOLVING MERGE CONFLICTS:

An event occurs when git is unable to automatically resolve the differences in the codes between the commits.

UNDOING CHANGES:

CASE - I: Staged changes(Files are added but not committed)

SYNTAX: **git reset / git reset <!Name of the file>**

CASE - 2: Changes are committed(For one commit)

SYNTAX: **git reset HEAD~1**

CASE - 3: Changes are committed(For more than one commit)

SYNTAX: **git reset <!commit hash> / git reset - -hard
<!commit-hash>**

git log: Used to check the updations done in the files.

FORK: Fork is the rough copy of the repository that is used to share the code and the visibility settings of the original upstream copy.