|  |  |
| --- | --- |
| **GITs** | **GITHUB** |
| Git is **software / Tool** | It is a **service**. |
| Linux maintains Git. | Microsoft maintains GitHub. |
| It is a command-line tool. | It is a graphical user interface. |
| You can install it **locally on the system**. | It is hosted on the web. It is exclusively **cloud-based**. |
| It is a **VCS** to manage source code history. | It is a **hosting service** for Git repositories. |
| It is open-source licensed. | It has a free-tier and pay-for-use tier. |

* **SVN** : has a Centralized Model
* **Git :** has a Distributed Model.

1. **Local Repository :** Local copy of the code / laptop/ pc
2. **Remote Repository :** Where the code lives in cloud / remote

* You need authentication to **GitHub / GitLab / Bitbucket** token’s

**Stage of code in local repository**



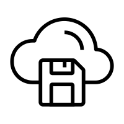
**Working Directory Staging Area Local repository**

$ git add

$ git commit Commit 1

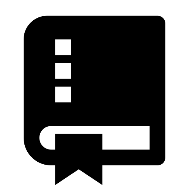
Commit 2

**Stage of code in local repository into remote repository**



GIT GIT GIT GITHUB

**Working Directory Staging Area Local repository remote repository**



$ git init $ git add <files> $ git commit -m $ git push

1. **Working directory:** This is where you make changes to your files. It's the directory on your computer where you edit, add, or delete files according to your project requirements.
2. **Staging Area:** The staging area is an intermediate step where you prepare and organize specific changes before committing them to the repository.
3. **Local repository:** permanently stores the committed changes and maintains a complete history of your project.
4. **remote repository:**  storage your code into cloud

**Windows**: Download Git bash https://git-scm.com/downloads

**Linux:** Download “yum install git -y “ https://git-scm.com/download/linux

**Setup for project:**

$ git init : Create a local repo to initialise the current directory as a repo

**$** git config --global user.name "**Ragesh**" : user name

**$** git config --global user. email "**Ragesh@yourcloud.com**” : user e-mail

**$** git config –list : List of users meta data

$ git remote add <alias> <url>: Add a remote repo

$ git remote : View all remote connections. Add-v flag to view URLS

$ git clone <url> : Download a remote repo

**Working on project**

$ git add : Add a file to staging

$ git add**.** :Stage all the files from the directory

$ git commit – m : Commit all staged file to local repo

$ git commit – m : Add all changes made to tracked files & commit

$ git push : Upload local repo into Remote repo

$ git status : List new or modified files not yet committed, check state of work

$ git log – oneline : List commit history, with oneline

$ git branch <name> : Create a new branch

$ git checkout <branch>: Switch to a branch

$ git merge : used to merge the branches FB to master

$ git rebase : used to integrate changes from one branch onto another

**conflict:**

**Conflict:** a conflict arises when multiple branches or contributors have conflicting changes to the same part of a file, requiring manual resolution to determine which changes should be kept.

**To resolve:**

$ git config --global merge.tool vimdiff

$ git config --global merge.conflictstyle diff3

$ git config --global mergetool.prompt false

$ git mergetool : vi editor to check the code and resolve it

**TASK:**

$ git fetch : updates your local repo with all of the latest changes from a remote

$ git stash : temporarily save in working directory

$ git stash pop : re-apply temporarily save work into staging



**GitHub Repo :** https://github.com/rageshtamizharasu

**Linux file colour in details:**

1. Green : Execution file
2. Blue : Directory
3. Red : zip, tar
4. Black : files

