Git vs Github

Git : command line system to track changes, many users can work on the same file and yet see the changes

Github: acts like a storage place where git repositories ( a location where everything related to a project can be stored . eg: source code files, configuration files, documentation etc)

**# to know git version**

Git – version

**#to set email and name in git**

Git config –global user.name =’anisha’

Git config –global user.email=”nn@gmail.com’

**# to check if the email or name has been set**

Git config –global user.name or user.email

**#make new folder with git in vscode**

Mkdir <folder name>

**#make it a default folder to work on**

Cd <folder name>

**# to make a repository with same folder name**

Git init

**# to give a list of files this repository contains**

ls -a

output:

.git , or any other file extension if exists

**# to know changes made to a git file**

Git status

**# add git file to the repository**

Git add . ---> this command ( adds everything to repository )

Git add <file name with extension > -🡪 this command adds the specific file name

**#STAGING IN GIT**

**#to see how many changes have been done in the file**

Git log

**# to go back to the the level where changes were made**

Git checkout <paste location here>

**# to know the branch of file**

Git branch

**#BRANCHING**

**Master branch is the main branch and from this main branch, you can create any number of branches so that the project feature created in one branch is not affected by other branch.**

**# create a branch in git**

Git branch <branch name > eg: git branch dev

**# to switch to a particular branch**

Git checkout <branch name> eg: git checkout dev -------------🡪 this switches current branch to dev

**# to merge files from another branch to other branch**

**Suppose there is a big team working on different features of a project so we make branches for different features, work in those different branches and the when everything is okay, we merge all to the main or master branch**

**Steps:**

1. Come to the branch where you want files from other branch --🡪 git branch dev
2. Merge the new branch to this --🡪 git merge <new branch name> eg: git merge branch 2

**#to make a gitignore file : these are files which are needed in working of project but we don’t want to share it on github repository publicly.**

touch.gitignore or make it directly as a new file

**#Linking git to github**

**Steps:**

1. Create a new repository in github
2. Public , don’t add gitignore file or readme file
3. Gitignore file and readme file can be created during project
4. Command -> git remote -v

( **git remote -v )-🡪 used to see all remote repositories in github**

1. **Copy the command from git repository created on github and paste it in vscode**

Git remote add origin url

1. Push files to github

Git branch -M master

1. Push the code to github master branch

Git push -u origin master

**#clone a repository from github**

git clone <repository url>

**# copy from someone’s repository and make it your own.**

**Fork the repository of that someone ---🡪 this creates a cloned repository in your account**

* You can make changes changes to this repository from your account

**# to make open source contribution**

Fork the repository-> make changes -> create a pull request so that you can send someone to merge it in their repository.