Introduction to Version Control with Git

What is Git?

Git is a distributed version control system that allows multiple people to collaborate on a project while keeping track of changes.Git is a type of Source Code Management (SCM) tool. It helps in managing and tracking changes to source code files. Git provides features like version control, branching, merging, and collaboration, making it a popular choice among developers.

What is the difference between Git and GitHub?

Git is a version control software that helps developers track the changes they make in their codes during a project's lifetime. It runs on the user's local machine. **GitHub** is a remote repository hosted on the cloud that provides a graphic interface for Git version control to facilitate version control, code sharing, and collaborations.

Branching and merging are fundamental concepts in Git that allow developers to work on different features or fixes simultaneously and merge their changes back into the main codebase.

Basic Git commands

Setup & Configuration

git –version - checks git is present in system or not

git init -This method initialises git in the current directory.

git config --global user.name "username or email" - Configure git on your Local machine

git remote add origin https://github.com/user_name/repository_name.git - Connecting to a remote repository from the local machine.

git clone https://github.com/username/repo-name.git - Clone a remote repository to your local machine.

git help - Display information about git.

git status -Check the current status of your repository

git add “filename” -Stage a specific file for commit

git add . -To add all files at a time

git commit -m " message" -To commit file from stage to local repo

git commit -a -m "commit message" -Git is already tracking files and you made some changes to it and want to commit then use this command.

git push - saves changes in local repository to remote repository

git ls - check files in git

git rm -Remove files from the working directory.

git mv - Move or rename file or directory.

git log -check the log info

git reset --hard commit id -changes will be removed from everywhere like working directory , staging area and local repo

git reset --soft commit id -changes made in that commit will be removed from local repo

git reset --mixed commit id -changes made in that particular commit id will be removed from stage area and local repo. and only present in working directory

git revert commit id -the changes done in that particular commit it will be undo (re reverse)

Git branch commands

git branch “branch name” -To create feature branch from main/master branch

git checkout “branch name” -To switch branch from one to another

git checkout -b “branch name” -To create new branch and switch to it at a time

git branch -m “old branch” “new branch name” -To rename the branch name

git branch --merged -To check branch has been merged or not

git branch --no-merged -To check branch has been not merged

git branch -D “branch name” -To delete branch

git merge “branch name” -To merge the change from one branch another branch

git rebase “branch name” -for linear log history/ similar to merge but rewrite the log history.

@Madhuri kuruguntla