**Day 6 Assignment 2**

**------------------------------------------------------------------------------------------------------------------------------------------**

**Problem**: Explain all the git commands

**Solution:**

here's an explanation of each Git command you mentioned:

1. git init: Initializes a new Git repository in the current directory or a specified directory. This command creates a new .git subdirectory with all necessary files for Git to operate within that directory.
2. git add: Adds changes in the current working directory to the staging area for the next commit. It prepares the changes to be included in the next snapshot of the repository
3. git commit: Records the changes made to the files in the repository. It takes a snapshot of the changes that have been staged and permanently stores those changes in the Git history.
4. git clone: Creates a local copy of a remote repository. It copies all versions of all files for a project, as well as the entire history of changes, from a remote repository to the local machine.
5. git fork: This is more commonly associated with platforms like GitHub. It creates a copy of a repository under your GitHub account. Forking is typically used when you want to propose changes to someone else's project or use someone else's project as a starting point for your own idea
6. git diff: Shows the differences between the changes made on the working directory and the staging area (git diff --staged) or between the staging area and the repository (git diff HEAD).
7. git branch: Lists, creates, or deletes branches. A branch in Git is a lightweight movable pointer to a commit. Creating a new branch allows you to work on new features or fixes without affecting the main branch.
8. git status: Displays the state of the working directory and the staging area. It shows which files are being tracked by Git, which changes are staged, and which changes haven't been tracked yet or are untracked.
9. git push: Uploads local repository content to a remote repository. This command transfers commits from your local repository to a remote repository, typically updating branches.
10. git pull: Fetches and integrates changes from a remote repository into the current branch. It is essentially a combination of git fetch followed by git merge.
11. git merge: Combines the changes from one branch into another. This command is used to integrate the changes from different branches into the current branch.
12. git remote: Manages the set of tracked repositories. You can use this command to add, remove, and manage remote repositories.
13. git log: Shows the commit history for the repository. It displays the list of commits along with details like the author, date, and commit message.
14. git reset: Resets the current branch to a specified state. It can modify the staging area and/or the working directory to match a particular commit.
15. git checkout: Switches branches or restores working tree files. You can use it to navigate between different branches or revert to previous versions of files.