Professional Git Command Guide

Basic Terminal Commands:

mkdir → Create a new directory (folder). Example: mkdir myproject

cd → Change directory. Example: cd myproject

ls / dir → List files and folders in current directory.

touch filename → Create an empty file.

echo "text" > filename → Create a file and write text into it.

Git Initialization:

git init → Initialize a local Git repository.

git status → Show current branch, staged files, unstaged changes, and untracked files.

Staging Area Commands:

git add filename → Stage a specific file for commit.

git add . → Stage all files in current directory.

git add -A → Stage all changes (new, modified, deleted).

git add -u → Stage modified and deleted files (not new files).

Commit Commands:

git commit → Open editor to write commit message and save staged changes.

git commit -m "message" → Commit with inline message description.

Commit = permanent snapshot in project history.

Viewing History:

git log → Show full commit history.

git log --oneline → Compact summary of commits.

git log --graph → Visual branch and merge graph.

git log --all → Show commits from all branches.

git log --stat → Show changed files per commit.

Branch Management:

git branch → List branches.

git branch new-feature → Create new branch.

git checkout new-feature → Switch branch (old method).

git switch new-feature → Switch branch (modern method).

git branch -v → Show branches with latest commit info.

Remote & GitHub Connection:

git remote add origin URL → Connect local repo to remote (GitHub).

git remote -v → Verify remote connections.

git clone URL → Download full repository with history from GitHub.

git push → Upload local commits to remote repository.

git pull → Download and merge latest changes from remote.

Merge & Collaboration:

git merge branch-name → Merge specified branch into current branch.

If conflict occurs → Edit file → git add . → git commit