



# GIT AND GITHUB

IMPORTANT COMMANDS  
USE IN INDUSTRIES



-----  
Command: git init

-- Purpose:

-- Initializes a new Git repository in the current directory.

-- Example:

git init

-----  
Command: git clone

-- Purpose:

-- Creates a copy of a remote repository on your local machine.

-- Example:

git clone <repository\_url>

-----  
Command: git add

-- Purpose:

-- Stages changes for commit in the current directory.

-- Example:

git add <file\_name>

-----  
Command: git commit

-- Purpose:

-- Commits the staged changes with a descriptive message.

-- Example:

git commit -m "Add new feature"

-----  
Command: git pull

-- Purpose:

-- Fetches and merges changes from a remote repository.

-- Example:

git pull origin <branch\_name>

-----  
Command: git push

-- Purpose:

-- Pushes committed changes to a remote repository.

-- Example:

git push origin <branch\_name>

-----  
Command: git status

-- Purpose:

-- Shows the current status of files in the repository.

-- Example:

git status

-----  
Command: git log

-- Purpose:

-- Displays the commit history.

-- Example:

git log

-----  
Command: git branch

-- Purpose:

-- Lists existing branches in the repository.

-- Example:

git branch

-----  
Command: git checkout

-- Purpose:

-- Switches to a different branch.

-- Example:

git checkout <branch\_name>

-----  
Command: git merge

-- Purpose:

-- Merges changes from one branch into another.

-- Example:

git merge <source\_branch>

-----  
Command: git remote

-- Purpose:

-- Lists remote repositories connected to your local repository.

-- Example:

git remote -v

-----  
Command: git fetch

-- Purpose:

-- Retrieves changes from a remote repository without merging.

-- Example:

git fetch origin

-----  
Command: git reset

-- Purpose:

-- Resets changes in the working directory or staging area.

-- Example:

git reset <file\_name>

-----  
Command: git rebase

-- Purpose:

-- Reapplies commits from one branch onto another.

-- Example:

git rebase <target\_branch>

-----  
Command: git tag

-- Purpose:

-- Creates a tag for a specific commit.

-- Example:

git tag v1.0

-----  
Command: git cherry-pick

-- Purpose:

-- Applies a specific commit from one branch to another.

-- Example:

git cherry-pick <commit\_hash>

-----  
Command: git diff

-- Purpose:

-- Shows the differences between working directory and repository.

-- Example:

git diff

-----  
Command: git stash

-- Purpose:

-- Temporarily saves changes that are not ready for commit.

-- Example:

git stash

-----  
Command: git show

-- Purpose:

-- Displays detailed information about a commit.

-- Example:

git show <commit\_hash>

-----  
Command: git remote add

-- Purpose:

-- Adds a remote repository as a new connection.

-- Example:

git remote add origin <repository\_url>

-----  
Command: git remote remove

-- Purpose:

-- Removes a remote repository from connections.

-- Example:

git remote remove origin

-----  
Command: git log --oneline

-- Purpose:

-- Shows commit history in one-line format.

-- Example:

git log --oneline

-----  
Command: git log -p

-- Purpose:

-- Shows commit history with patch changes.

-- Example:

git log -p

-----  
Command: git blame

-- Purpose:

-- Displays who made changes to a file and when.

-- Example:

git blame <file\_name>

-----  
Command: git config

-- Purpose:

-- Sets or displays Git configuration.

-- Example:

git config --global user.name "Your Name"

-----  
Command: git remote show

-- Purpose:

-- Displays detailed information about a remote repository.

-- Example:

git remote show origin

-----  
Command: git grep

-- Purpose:

-- Searches for a pattern in files.

-- Example:

git grep "search\_term"

-----  
Command: git log --author

-- Purpose:

-- Filters commit history by author.

-- Example:

git log --author="John Doe"

-----  
Command: git log --since

-- Purpose:

-- Filters commit history by date.

-- Example:

git log --since="2023-01-01"

-----  
Command: git log --graph

-- Purpose:

-- Displays commit history in a graph format.

-- Example:

git log --graph

-----  
Command: git stash pop

-- Purpose:

-- Applies the last stashed changes and removes it.

-- Example:

git stash pop

-----  
Command: git stash list

-- Purpose:

-- Lists all stashed changes.

-- Example:

git stash list

-----  
Command: git rm

-- Purpose:

-- Removes files from the working directory and stages the deletion.

-- Example:

git rm <file\_name>

-----  
Command: git mv

-- Purpose:

-- Renames or moves files and stages the changes.

-- Example:

git mv <old\_file\_name> <new\_file\_name>



-----  
Command: git log --grep

-- Purpose:

-- Filters commit history by commit message.

-- Example:

git log --grep="fix"

-----  
Command: git log --all

-- Purpose:

-- Displays commit history for all branches.

-- Example:

git log --all

-----  
Command: git checkout -b

-- Purpose:

-- Creates a new branch and switches to it.

-- Example:

git checkout -b <new\_branch\_name>

-----  
Command: git log --stat

-- Purpose:

-- Shows summary of file changes in commit history.

-- Example:

git log --stat

-----  
Command: git branch -d

-- Purpose:

-- Deletes a local branch.

-- Example:

git branch -d <branch\_name>

-----  
Command: git clean

-- Purpose:

-- Removes untracked files and directories.

-- Example:

git clean -f

-----  
Command: git reset --hard

-- Purpose:

-- Discards all changes and resets to a specific commit.

-- Example:

git reset --hard <commit\_hash>

-----  
Command: git log -n

-- Purpose:

-- Limits the number of displayed commits.

-- Example:

git log -n 5

-----  
Command: git merge --no-ff

-- Purpose:

-- Performs a merge without fast-forwarding.

-- Example:

git merge --no-ff <branch\_name>

-----  
Command: git commit --amend

-- Purpose:

-- Changes the author's email of the last commit.

-- Example:

git commit --amend --author="new\_email@example.com"

-----  
Command: git log --author=

-- Purpose:

-- Filters commit history by author's email.

-- Example:

git log --author="john@example.com"

-----  
Command: git clean -f

-- Purpose:

-- Removes untracked files and directories forcefully.

-- Example:

git clean -f

-----  
Command: git init --bare

-- Purpose:

-- Initializes a new bare Git repository.

-- Example:

git init --bare

-----  
Command: git blame -L

-- Purpose:

-- Blames changes in specific lines of a file.

-- Example:

git blame -L 10,20 <file\_name>

-----  
Command: git remote -v

-- Purpose:

-- Lists remote repositories with their URLs.

-- Example:

git remote -v

-----  
Command: git log ..

-- Purpose:

-- Displays commit history between two ranges.

-- Example:

git log <start\_commit>..<end\_commit>

-----  
Command: git clean -f

-- Purpose:

-- Removes untracked files and directories forcefully.

-- Example:

git clean -f

-----  
Command: git clean -f

-- Purpose:

-- Removes untracked files and directories forcefully.

-- Example:

git clean -f

-----  
Command: git init --bare

-- Purpose:

-- Initializes a new bare Git repository.

-- Example:

git init --bare

-----  
Command: git blame -L

-- Purpose:

-- Blames changes in specific lines of a file.

-- Example:

git blame -L 10,20 <file\_name>

-----  
Command: git remote -v

-- Purpose:

-- Lists remote repositories with their URLs.

-- Example:

git remote -v

-----  
Command: git log ..

-- Purpose:

-- Displays commit history between two ranges.

-- Example:

git log <start\_commit>..<end\_commit>

-----  
Command: git log ..

-- Purpose:

-- Displays commit history between two ranges.

-- Example:

git log <start\_commit>..<end\_commit>

End of Git and GitHub Commands.

