Git Commands for Beginners – Cheat Sheet

Basic Setup Commands

Command	Purpose
git config --global user.name "Your Name"	Set your global username for commits.
git config --global user.email "you@example.com"	Set your global email address for commits.
git config --list	View your Git configuration details.

Repository Initialization & Cloning

Command	Purpose
git init	Initialize a new Git repository in the current folder.
git clone <repo-url>	Clone (download) a remote repository to your local machine.

Tracking and Managing Changes

Command	Purpose
git status	Show the current state of your working directory and staging area.
git add <file>	Add a specific file to the staging area.
git add .	Add all changed files to the staging area.
git reset <file>	Unstage a file without losing changes.
git diff	Show changes made in files that are not yet staged.
git diff --staged	Show changes between staged files and the last commit.

Committing Changes

Command	Purpose
git commit -m "commit message"	Commit staged changes with a message.
git commit -am "commit message"	Add and commit all tracked changes in one step.

Viewing History

Command	Purpose
git log	View the commit history.
git log --oneline	View concise commit history.
git show <commit-id>	Show details of a specific commit.

Branching and Merging

Command	Purpose
git branch	List all branches.
git branch <branch-name>	Create a new branch.
git checkout <branch-name>	Switch to another branch.
git checkout -b <branch-name>	Create and switch to a new branch.
git merge <branch-name>	Merge a branch into the current branch.
git branch -d <branch-name>	Delete a branch.

Working with Remote Repositories

Command	Purpose
git remote -v	Show remote repositories linked to your project.
git remote add origin <repo-url>	Link your local repo to a remote repository.
git push -u origin <branch-name>	Push a branch to the remote repository for the first time.
git push	Push committed changes to the remote repository.
git pull	Fetch and merge changes from the remote repository.
git fetch	Fetch changes from the remote (without merging).

Undoing Mistakes

Command	Purpose
git restore <file>	Discard changes in a file (revert to last commit).
git reset --hard <commit-id>	Reset your repository to a specific commit (warning: destructive).
git revert <commit-id>	Create a new commit that undoes a specific commit.

Bonus Tips

Command	Purpose
git stash	Temporarily save uncommitted changes.
git stash pop	Reapply stashed changes.
git tag <tag-name>	Tag a specific commit (often for releases).