Garratt's Git & GitHub Cheatsheet

A practical Git and GitHub reference for daily development. Focused on safe workflows, collaboration, and commands you'll actually use.

Repository Basics

Goal	Command	Example
Init repo	git init	git init
Clone	Copy remote repo	git clone https://github.com/user/repo.git
Status	See changes	git status
Stage files	Prepare commit	git add .
Commit	Save snapshot	git commit -m "Add homepage"

Branches

Task	Command	Example
List branches	View local branches	git branch
Create branch	New feature	git checkout -b feature/seo
Switch	Change branch	git switch main
Merge	Bring changes in	git merge feature/seo
Delete	Clean up	git branch -d feature/seo

Viewing History

Need	Command	Example
Log	Commit history	git log --oneline --graph
File history	Per-file log	git log -- path/to/file
Diff staged	Review staged	git diff --staged
Diff branches	Compare	git diff main..feature

Undoing Changes (Safely)

Scenario	Command	Example
Unstage file	Remove from index	git restore --staged file.js
Discard changes	Reset file	git restore file.js
Amend commit	Fix last commit	git commit --amend
Revert commit	Safe undo	git revert abc123
Reset hard	Danger zone	git reset --hard HEAD~1

Remotes

Task	Command	Example
List remotes	Check origin	git remote -v
Add remote	Link repo	git remote add origin https://github.com/user/repo.git
Fetch	Get updates	git fetch origin
Pull	Fetch + merge	git pull
Push	Send commits	git push origin main

Stashing

Task	Command	Example
Stash work	Save dirty state	git stash
List stashes	View saved	git stash list
Apply stash	Restore	git stash apply
Pop stash	Apply + remove	git stash pop

GitHub Workflow

Step	What happens	Notes
Fork	Copy repo	Used when you don’t have write access
Branch	Isolate work	One feature per branch
Pull Request	Request merge	Describe what/why
Review	Feedback	Comments + approvals
Merge	Land changes	Squash or merge commit

.gitignore Essentials

Ignore	Why	Example
node_modules	Dependencies	node_modules/
Env files	Secrets	*.env
Build output	Generated files	dist/ build/
OS files	Noise	.DS_Store

Good Habits

Practice	Why	Tip
Small commits	Easy reviews	One logical change per commit
Clear messages	Readable history	Use imperative tone
Pull often	Avoid conflicts	Especially on shared branches
Review diffs	Catch mistakes	git diff before commit
Backup remotes	Safety net	Push before risky ops