

Git & GitHub — Cheat Sheet

FREE CHEAT SHEET

One page from the complete **Git & GitHub Companion Guide**. [Get the full guide →](#)

The Daily Workflow Loop

STEP	COMMAND
1. Pull latest	<code>git pull</code>
2. Create branch	<code>git checkout -b feature/name</code>
3. Make changes	<i>(edit files)</i>
4. Stage changes	<code>git add .</code>
5. Commit	<code>git commit -m "message"</code>
6. Push	<code>git push origin feature/name</code>
7. Open PR	<i>(via GitHub)</i>
8. Review → Merge	<i>(via GitHub)</i>

Essential Commands

TASK	COMMAND
Check status	<code>git status</code>
View history	<code>git log --oneline</code>
View changes	<code>git diff</code>
Switch branch	<code>git checkout branch-name</code>
List all branches	<code>git branch -a</code>
Delete local branch	<code>git branch -d branch-name</code>
Undo last commit (keep changes)	<code>git reset --soft HEAD~1</code>
Safely undo a pushed commit	<code>git revert <hash></code>
Shelve uncommitted changes	<code>git stash</code>
Restore shelved changes	<code>git stash pop</code>
Abort a merge in progress	<code>git merge --abort</code>
Find lost commits	<code>git reflog</code>

Branch Naming

TYPE	PATTERN	EXAMPLE
Feature	<code>feature/description</code>	<code>feature/user-auth</code>
Bug fix	<code>bugfix/description</code>	<code>bugfix/null-reference</code>
Hot fix	<code>hotfix/description</code>	<code>hotfix/security-patch</code>
Chore	<code>chore/description</code>	<code>chore/update-deps</code>
Docs	<code>docs/description</code>	<code>docs/update-readme</code>

Commit Message Format

type(scope): short description	
TYPE	WHEN TO USE
<code>feat</code>	New functionality
<code>fix</code>	Bug fix
<code>chore</code>	Maintenance, config
<code>docs</code>	Documentation only
<code>refactor</code>	Restructuring, no behavior change
<code>test</code>	Adding or updating tests
<code>style</code>	Formatting, whitespace only

Golden Rules

RULE
Always pull before you start working.
RULE
One branch = one purpose.
RULE
One commit = one logical change.
RULE
Never push directly to main.
RULE
A good commit message explains what changed and why.
RULE
When in doubt, run <code>git status</code> first.
RULE
Never guess when resolving a merge conflict. Ask.
RULE
Green CI before merge. No exceptions.