Ultimate Git Command Cheat Sheet (With Purpose)

LEVEL 1 DAILY ESSENTIALS

git init:

Start a new Git repo in your project folder

git status:

See current changes, staged files, untracked files

git add filename / .:

Track file(s) for the next commit

git commit -m "msg":

Save your changes with a message

git log:

View history of commits

git log --oneline:

Compact view of commits

git mv old new:

Rename a file and track the rename

git rm filename:

Delete a file and stage the deletion

git restore filename:

Undo changes in working directory

git restore --staged file:

Unstage a file before commit

git diff:

See unstaged changes

git diff --cached:

See staged but uncommitted changes

git config --global user.name/email:

Set user identity for commits

git config --list:

View current Git config settings

LEVEL 2 BRANCHING & MERGING

git branch:

List all local branches

git branch branch-name:

Create a new branch

Ultimate Git Command Cheat Sheet (With Purpose)

git checkout branch-name:

Switch to a branch

git checkout -b branch-name:

Create and switch to a new branch

git branch -m new-name:

Rename current branch

git branch -d branch-name:

Delete a branch after merging

git merge branch-name:

Merge branch into current branch

git merge-base b1 b2:

Find common ancestor of two branches

LEVEL 3 RESET, REVERT, STASH

git reset --soft HEAD~1:

Move commit pointer, keep staged files

git reset --mixed HEAD~1:

Move pointer, unstage changes (default)

git reset --hard HEAD~1:

Discard all changes + commits (careful!)

git revert <commit>:

Undo commit by creating a new reverse commit

git stash:

Temporarily save changes without committing

git stash list:

See list of saved stashes

git stash apply:

Reapply last (or specific) stash

git stash drop:

Delete stash entry

REMOTE & GITHUB

git clone <url>:

Download remote repo to local machine

git remote add origin <url>:

Link local repo to GitHub

Ultimate Git Command Cheat Sheet (With Purpose)

git push origin branch:

Upload changes to GitHub

git pull origin branch:

Download + merge changes from GitHub

git push -u origin branch:

Push and set upstream tracking

git remote -v:

View current remote URLs

git remote set-url origin <url>:

Update remote GitHub URL

LOGS & TRACKING

git blame file:

See who last edited each line

git show:

Display details about commit or tag

git log --graph:

Visualize branch structure

git log file:

See history of a specific file

git shortlog:

Group commits by author

git reflog:

View all recent HEAD changes (undo magic!)

ADVANCED COMMANDS

git rebase branch:

Replay commits onto another branch (clean history)

git rebase --abort:

Cancel a rebase in progress

git cherry-pick <id>:

Copy one commit into another branch

git tag v1.0:

Mark a specific commit as a release/version

git tag:

List all tags

git push origin v1.0:

Push tag to GitHub

git clean -f:

Delete untracked files

git clean -fd:

Delete untracked files and folders

git diff branch1..branch2:

Compare two branches changes

LOW-LEVEL / PLUMBING COMMANDS

git cat-file -p <hash>:

View internal content of Git objects

git rev-list HEAD:

List commit IDs in current branch

git rev-parse HEAD:

Get hash of current commit

git ls-tree HEAD:

View file structure at a commit

git show-ref:

Show all references (tags, branches)

git archive:

Export repo snapshot as zip/tar

.git/index:

Git's internal file tracking database

GIT TOOLS & UTILITY

git gui:

Launch Git GUI (if installed)

git instaweb:

Preview local repo in a web browser

git daemon:

Start local Git server

.git/hooks/:

Automate tasks (e.g., auto lint before commit)

TROUBLESHOOTING / RECOVERY

Ultimate Git Command Cheat Sheet (With Purpose)

git credential-cache exit:

Clear saved login

git switch -c branch:

Create and switch (modern `checkout`)

git merge --abort:

Cancel a failed merge

git rebase --abort:

Abort a rebase

<<<<<<< HEAD:

Conflict markers to help fix code manually