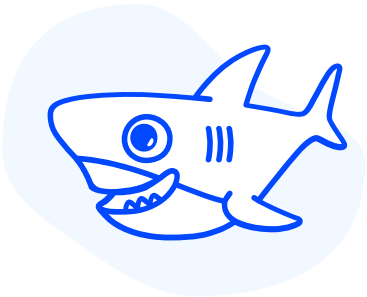
Git Cheat Sheet

Git is an open source version control system that works locally to help developers work together on software projects that matter. This cheat sheet provides a quick reference to commands that are useful for working and collaborating in a Git repository (repo).



Initializing

Starting up Git within a project and getting it connected.

`git init`
Initializes (or starts) your current working directory (folder) as a Git repository (repo).

`git clone https://www.github.com/username/repo-name`
Copies an existing Git repo hosted remotely.

`git remote` or `git remote -v`
Shows your current Git directory's remote repo. Use the -v flag for more info.

`git remote add upstream https://www.github.com/username/repo-name`
Adds the Git upstream to a URL.

Staging

Creating files staged after modifying a file and marking it ready to go in the next commit.

`git status`
Checks the status of your Git repo, including files added that are not staged.

`git add .` or `git add my_script.js`
Stages modified files. If you make changes that you want included in the next commit, you can run add again. Use "git add ." for all files to be staged, or specify specific files by name.

`git reset my_script.js`
Removes a file from staging while retaining changes within your working directory.

Committing

Recording changes made to the repo.

`git commit -m "Commit message"`
Commits staged files with a meaningful commit message so that you and others can track commits.

`git commit -am "Commit message"`
Condenses all tracked files by committing them in one step.

`git commit --amend -m "New commit message"`
Modifies your commit message.

Branching

Isolating work and managing feature development in one place.

`git branch`
Lists all current branches. An asterisk (*) will appear next to your currently active branch.

`git branch new-branch`
Creates a new branch. You will remain on your currently active branch until you switch to the new one.

`git checkout another-branch`
Switches to any existing branch and checks it out into your current working directory.

`git checkout -b new-branch`
Consolidates the creation and checkout of a new branch.

`git branch -d branch-name`
Deletes a branch.

Collaborating and Sharing

Downloading changes from another repository or sharing changes with the larger codebase.

`git push origin main`
Pushes or sends your local branch commits to the remote repo.
Note: some repos use master instead of main in their commands.

`git pull`
Fetches and merges any commits from the tracking remote branch.

`git merge upstream/main`
Merges the fetched commits.

Showing Changes

See changes between commits, branches, and more.

`git diff --staged`
Compares modified files that are in the staging area.

`git diff a-branch..b-branch`
Displays the diff of what is in `a-branch` but is not in `b-branch`.

`git diff 61ce3e6..e221d9c`
Uses commit id to show the diff between two specific commits.