

# Git Commands and Their Uses - Practical Guide

## Setup Commands

- `git --version` — Check installed Git version
- `git config --global user.name "Your Name"` — Set username
- `git config --global user.email "email@example.com"` — Set email
- `git config --list` — View configuration settings

## Repository Creation

- `git init` — Create new repository in current folder
- `git clone` — Download existing repository from GitHub

## Daily Workflow Commands

- `git status` — Show file changes and staging info
- `git add .` — Add all files to staging
- `git add` — Add specific file to staging
- `git commit -m "message"` — Save changes to repository
- `git push` — Upload commits to GitHub
- `git pull` — Download latest changes from GitHub
- `git log` — View commit history

## Branching Commands

- `git branch` — List all branches
- `git branch` — Create new branch
- `git checkout` — Switch to branch
- `git checkout -b` — Create and switch to new branch
- `git merge` — Merge branch into current branch
- `git branch -d` — Delete a branch

## Remote (GitHub Connection)

- `git remote add origin` — Connect local repo to GitHub
- `git remote -v` — Show remote repository links
- `git push -u origin main` — First push to GitHub
- `git pull origin main` — Pull latest code from GitHub

## Undo Mistakes

- `git reset` — Unstage a file
- `git reset --soft HEAD~1` — Undo last commit (keep changes)
- `git reset --hard HEAD~1` — Undo last commit (delete changes)
- `git checkout --` — Discard file changes

## Inspection Commands

- `git diff` — Show differences in files
- `git blame` — See who changed each line
- `git show` — Show details of a commit

## Stash Commands

- `git stash` — Temporarily save changes without commit
- `git stash pop` — Restore stashed changes

## Tagging (Versions)

- `git tag v1.0` — Create version tag
- `git push origin v1.0` — Push tag to GitHub