

Complete GIT & GITHUB CheatSheet

- Developer Shaurya

Important Links

- **GIT & GITHUB Cheatsheet :**

<https://developershaurya.com/git-and-github-cheatsheet/>

- **Website:** <https://developershaurya.com/>

- **All Cheatsheets :**

<https://developershaurya.com/cheatsheets/>

- **Youtube:**

<https://www.youtube.com/@DeveloperShaurya>

- **(Developer Shaurya) Community :**

https://t.me/developer_shaurya (Telegram)

CheatSheet

GitHub

GitHub CheatSheet – A Quick Guide for Beginners & Developers

by Developer Shaurya • September 26, 2025 • 0



This is a **complete Git & GitHub Cheatsheet (2025 Edition)** — everything you need from **setup to collaboration, branching, merging, GitHub usage, and troubleshooting** — all in one place

Table of Contents

- [1. Setup & Configuration](#)
 - [Install Git](#)
 - [Configure Git \(First Time\)](#)
 - [Check Config](#)
- [2. Create or Clone Repository](#)
 - [Initialize New Repo](#)
 - [Clone Existing Repo](#)
- [3. Basic Workflow](#)
 - [Check Status](#)
 - [Add Changes](#)
 - [Commit Changes](#)
 - [View Commit History](#)
- [4. Branching & Merging](#)
 - [Create Branch](#)
 - [Switch Branch](#)
 - [Create + Switch](#)
 - [Merge Branch](#)
 - [Delete Branch](#)
- [5. Remote Repositories \(GitHub\)](#)
 - [Add Remote](#)
 - [View Remotes](#)
 - [Push to GitHub](#)
 - [Pull from GitHub](#)

- [6. Updating & Syncing](#)
 - [Fetch Latest Changes](#)
 - [Pull Updates](#)
 - [Push Updates](#)
- [7. Undo & Fix Mistakes](#)
 - [Unstage File](#)
 - [Undo Last Commit \(Keep Changes\)](#)
 - [Undo Last Commit \(Remove Changes\)](#)
 - [Revert Commit](#)
 - [Checkout Old Version](#)
- [8. Stashing \(Temporary Storage\)](#)
 - [Save Work](#)
 - [List Stashes](#)
 - [Apply Stash](#)
 - [Drop Stash](#)
- [9. Git Diff & Logs](#)
 - [Show Changes](#)
 - [Show Differences Between Commits](#)
- [10. Tags & Releases](#)
 - [Create Tag](#)
 - [Push Tags](#)
- [11. GitHub Collaboration](#)
 - [Fork Repo \(On GitHub UI\)](#)
 - [Create Pull Request \(PR\)](#)
 - [Review PR](#)
- [12. SSH Authentication \(Optional\)](#)
 - [Generate SSH Key](#)
 - [Copy Key](#)
 - [Test Connection](#)

- [13. Git Ignore](#)
- [14. GitHub Pages \(Deploy Site\)](#)
- [15. Useful Shortcuts](#)
- [16. Troubleshooting](#)
 - [Merge Conflicts](#)
 - [Remote Already Exists](#)
 - [Push Rejected \(Out of Date\)](#)
- [17. Advanced Commands](#)
 - [View Who Changed a Line](#)
 - [Show File at Commit](#)
 - [Clean Untracked Files](#)
- [18. GUI Tools](#)
- [19. GitHub CLI \(Optional\)](#)
 - [Install](#)
 - [Common Commands](#)
- [20. Best Practices](#)

1. Setup & Configuration

Install Git

```
# Windows
winget install --id Git.Git -e --source winget

# macOS
brew install git

# Linux (Debian/Ubuntu)
sudo apt install git
```

Configure Git (First Time)

```
git config --global user.name "Your Name"
git config --global user.email "your_email@example.com"
git config --global core.editor "code --wait"    # Use VS Code
git config --global color.ui auto
```

Check Config

```
git config --list
```

2. Create or Clone Repository

Initialize New Repo

```
git init
```

Clone Existing Repo

```
git clone <repo_url>  
# Example:  
git clone https://github.com/username/repo.git
```

3. Basic Workflow

Check Status

```
git status
```

Add Changes

```
git add <file>      # Add specific file  
git add .           # Add all changes
```

Commit Changes

```
git commit -m "Your commit message"
```

View Commit History

```
git log  
git log --oneline    # Short version  
git log --graph --decorate --oneline
```

4. Branching & Merging

Create Branch

```
git branch <branch_name>
```

Switch Branch

```
git checkout <branch_name>  
# or  
git switch <branch_name>
```

Create + Switch

```
git checkout -b <branch_name>  
# or  
git switch -c <branch_name>
```

Merge Branch

```
git checkout main  
git merge <branch_name>
```

Delete Branch

```
git branch -d <branch_name>    # Safe delete  
git branch -D <branch_name>    # Force delete
```

5. Remote Repositories (GitHub)

Add Remote

```
git remote add origin <repo_url>
```

View Remotes

```
git remote -v
```

Push to GitHub

```
git push -u origin main    # First time  
git push                   # Next times
```

Pull from GitHub

```
git pull origin main
```

6. Updating & Syncing

Fetch Latest Changes

```
git fetch
```

Pull Updates

```
git pull
```

Push Updates

```
git push
```

7. Undo & Fix Mistakes

Unstage File

```
git reset <file>
```

Undo Last Commit (Keep Changes)

```
git reset --soft HEAD~1
```

Undo Last Commit (Remove Changes)

```
git reset --hard HEAD~1
```

Revert Commit

```
git revert <commit_hash>
```

Checkout Old Version

```
git checkout <commit_hash> -- <file>
```

8. Stashing (Temporary Storage)

Save Work

```
git stash
```

List Stashes

```
git stash list
```

Apply Stash

```
git stash apply
```

Drop Stash

```
git stash drop
```

9. Git Diff & Logs

Show Changes

```
git diff
```

Show Differences Between Commits

```
git diff <commit1> <commit2>
```

10. Tags & Releases

Create Tag

```
git tag <tag_name>  
git tag -a <tag_name> -m "Message"
```

Push Tags

```
git push origin <tag_name>  
git push origin --tags
```

11. GitHub Collaboration

Fork Repo (On GitHub UI)

Create Pull Request (PR)

1. Push branch to GitHub.
2. Go to repo → "Pull Requests" → "New Pull Request".

Review PR

- Comment, approve, or request changes.

12. SSH Authentication (Optional)

Generate SSH Key

```
ssh-keygen -t ed25519 -C "your_email@example.com"
```

Copy Key

```
cat ~/.ssh/id_ed25519.pub
```

Add it to **GitHub** → **Settings** → **SSH and GPG keys** → **New SSH Key**

Test Connection

```
ssh -T git@github.com
```

13. Git Ignore

Create `.gitignore` file:

```
node_modules/  
.env  
.DS_Store  
*.log
```

Add and commit:

```
git add .gitignore  
git commit -m "Add .gitignore"
```

14. GitHub Pages (Deploy Site)

```
git branch -M main
git push -u origin main
```

Then go to:

Repo Settings → **Pages** → **Source: main /root** → **Save**

Your site:

```
https://<username>.github.io/<repo-name>/
```

15. Useful Shortcuts

Command	Description
git init	Create repo
git clone URL	Clone repo
git status	Check status
git add .	Stage all
git commit -m	Commit
git push	Push changes
git pull	Pull updates
git branch	List branches
git checkout	Switch branch
git merge	Merge
git log --oneline	Compact history

16. Troubleshooting

Merge Conflicts

Open files → fix conflicts →

```
git add .  
git commit
```

Remote Already Exists

```
git remote remove origin  
git remote add origin <new_url>
```

Push Rejected (Out of Date)

```
git pull origin main --rebase  
git push origin main
```

17. Advanced Commands

View Who Changed a Line

```
git blame <file>
```

Show File at Commit

```
git show <commit_hash>:<file_path>
```

Clean Untracked Files

```
git clean -fd
```

18. GUI Tools

- **GitHub Desktop** (Beginner-friendly)
- **Sourcetree**
- **GitKraken**
- **VS Code Git Integration**

19. GitHub CLI (Optional)

Install

```
gh auth login
```

Common Commands

```
gh repo create  
gh pr create  
gh pr view  
gh pr merge
```

20. Best Practices

Commit often with clear messages

Use feature branches

Pull before push

Write `.gitignore` early

Use `git stash` before switching branches

Never commit credentials (`.env`)