

Github Cheatsheet

Local ↔ Cloud Flow

Working Directory → Staging → Local Repo → Remote (GitHub)
(files) git add git commit git push

(files) ←————— ←————— ←— git pull/fetch

LOCAL (Your Computer)	REMOTE (GitHub)
— .git/ (history)	— origin/main
— Working files	— Pull Requests
— Branches	— Issues
	— Actions

git push →
← git pull

Setup & Clone

```
# Connect local repo to GitHub
git remote add origin https://github.com/username/repo.git
git push -u origin main
```

```
# Clone existing
git clone https://github.com/username/repo.git
```

Basic Operations

git push	# Upload commits
git push -u origin branch-name	# Push new branch
git pull	# Download + merge

git fetch	# Download only
git remote -v	# View remotes

Pull Request Workflow

1. Create branch

git switch -c feature-name

2. Commit changes

git add .

git commit -m "Add feature"

3. Push to GitHub

git push -u origin feature-name

4. GitHub web: Create PR → Review → Merge

5. After merge

git switch main

git pull

git branch -d feature-name

Forking Workflow

1. Fork on GitHub (Fork button)

2. Clone your fork

git clone https://github.com/YOUR-USERNAME/repo.git

3. Add original as upstream

git remote add upstream https://github.com/ORIGINAL/repo.git

4. Sync fork

git fetch upstream

git merge upstream/main

```
git push origin main
```

```
# 5. Create PR from fork to original
```

Repository Types

Public: Anyone can see/clone (portfolios, open source)

Private: Only you + collaborators (work, experiments)

GitHub Features

Issues: Bug tracking, feature requests

Projects: Kanban boards

Actions: CI/CD automation

Pages: Free static hosting (username.github.io/repo)

Releases: Version tagging

Wiki: Documentation

Best Practices

Repo Structure

```
repo/  
├── README.md  
├── .gitignore  
├── LICENSE  
├── requirements.txt (or pyproject.toml)  
├── src/  
├── tests/  
└── .github/workflows/
```

README Template

```
# Project Name
Brief description

## Installation
pip install -r requirements.txt

## Usage
python main.py

## Features
- Feature 1
- Feature 2
```

Good PR

Title: "Add user authentication with JWT"
Description: What/Why/How to test
Keep < 400 lines
One feature per PR
Link issues: "Fixes #123"

GitHub CLI

```
gh repo clone user/repo      # Clone
gh repo create name --public  # Create repo
gh pr create                  # Create PR
gh pr list                    # List PRs
gh pr checkout 123            # Checkout PR
gh issue list                  # List issues
```

Web Shortcuts

? → Show all shortcuts
t → File finder
. → Open web editor
Ctrl+K → Command palette
g c → Code tab
g i → Issues tab
g p → PRs tab

Common Fixes

Push rejected (remote ahead)
git pull --rebase
git push

Wrong remote URL
git remote set-url origin <new-url>

Merge conflict in PR
git fetch origin
git merge origin/main
Resolve conflicts
git push

Security

Never commit:
.env
*.key
secrets.json
credentials/

If secrets leaked:

1. Rotate immediately
2. Remove from history
3. Force push

Platform Comparison

Feature	GitHub	GitLab	Bitbucket	Forgejo
Hosting	Cloud	Cloud/Self	Cloud/Self	Self only
Free private	Unlimited	Unlimited	Limited	Unlimited
CI/CD	Actions	Built-in	Pipelines	Actions
Share	90%	10%	5%	Niche
Type	Corporate	Corporate	Corporate	FOSS

Quick Reference

Task	How
Create repo	GitHub web → New
Push branch	<code>git push -u origin branch</code>
Create PR	Push → GitHub → Compare & PR
Fork	Fork button → Clone your fork
Sync fork	<code>git fetch upstream</code> + merge
Enable Pages	Settings → Pages
Protect branch	Settings → Branches
Add collaborator	Settings → Collaborators