Git & GitHub Cheat Sheet – Beginner to Advanced

---

1. Git vs GitHub

Feature	Git	GitHub
Type	Version Control System	Cloud Hosting for Git repos
Scope	Local & remote	Remote & collaboration
Usage	Track code history	Host repos, pull requests, issues
Commands	`git commit`, `git branch`	`git push`, `git pull`, PR creation
Install	Yes	Browser or CLI optional

---

2. Setup

```
git config --global user.name "Your Name"
git config --global user.email "you@example.com"
git config --list
git config --global core.editor "code --wait"  # Optional: VSCode as editor
```

---

3. Repository

```
git init                  # Initialize local repo
git clone <url>           # Clone remote repo
git remote -v             # List remote connections
git remote add origin <url>  # Add remote
```

---

4. Daily Workflow

```
git status                # Show changes
git add <file>            # Stage file
git add .                 # Stage all changes
git commit -m "message"   # Commit changes
```

```
git pull origin main      # Fetch + merge remote changes
git push                  # Push local changes
```

---

5. Branching & Merging

```
git branch                # List local branches
git branch -a             # List all branches
git checkout branchName   # Switch branch
git checkout -b featureX  # Create & switch branch
git merge branchName      # Merge into current branch
git branch -d branchName  # Delete local branch
git push origin --delete branchName # Delete remote branch
```

---

6. Undo / Reset

```
git reset --soft HEAD~1    # Undo last commit, keep staged
git reset --hard HEAD~1    # Undo last commit, discard changes
git checkout -- <file>     # Discard changes in file
git revert <commit-id>     # Undo commit via new commit
```

---

7. History & Inspection

```
git log                     # Commit history
git log --oneline           # Short commit list
git log --graph --all       # Visual graph
git diff                    # Unstaged changes
git diff --staged           # Staged changes
git show <commit-id>        # Show commit details
```

---

8. Stash & Recovery

```
git stash                   # Save uncommitted changes
git stash list              # List stashes
git stash apply             # Apply last stash
```

```
git stash pop               # Apply + remove stash
git reflog                  # Recover commits/actions
```

---

9. Tags

```
git tag                     # List tags
git tag v1.0                # Create tag
git push origin v1.0        # Push tag
```

---

10. Collaboration

```
git fetch origin            # Fetch remote changes
git pull origin main        # Fetch + merge
git push                    # Push changes
```

---

11. Advanced Commands

```
git rebase branchName        # Replay commits on top
git cherry-pick <commit>     # Apply a specific commit
git log --author="Name"      # Filter by author
git blame <file>             # See who changed each line
git bisect                   # Find bug commit
git merge --no-ff branchName # Merge with commit history
```

---

12. Best Practices

- Branch per feature → PR → review → merge
- Keep `main` stable
- Pull often → avoid conflicts
- Stash changes before switching branches
- Always set upstream branch when pushing new branch

Notes for Interviews: - Be ready to explain difference between local and remote branches, git pull vs fetch, and merge vs rebase. - Know how to undo mistakes safely. - Mention workflow for collaborative coding (branching, PR, merge).

End of Cheat Sheet