Git

22 Git Commands Every Engineer Must Know

by Gina Acosta

1. Basic Commands

`git init`	Initialize a new Git repository
`git clone <repo>`	Clone a repository to local
`git status`	Show the working directory status
`git log`	View commit history

2. Branching

`git branch`	List, create, or delete branches
`git branch <branch>`	Create a new branch
`git checkout <branch>`	Switch to a specific branch
`git merge <branch>`	Merge a branch into the current one

3. Staging & Committing

`git add <file>`	Add files to the staging area
`git add .`	Add all changes to staging
`git commit -m "message"`	Commit changes with a message
`git reset <file>`	Unstage a file
`git stash`	Save changes for later use

4. Collaboration

`git pull <remote> <branch>`	Fetch and merge changes from remote
`git push <remote> <branch>`	Push changes to a remote repository
`git fetch <remote>`	Download changes without merging
`git remote -v`	List remote repository connections
`git tag <tagname>`	Create a new tag

5. Advanced Commands

`git cherry-pick <commit>`	Apply specific commits elsewhere
`git bisect`	Find a commit causing an issue
`git diff <file>`	Show changes between commits/files
`git blame <file>`	Show who changed each line

1. Git commands
2. □□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□:

- Clone a Repository:
Command:
git clone <repository_url>

- Initialize a Repository:
Command:
git init

2. □□□□□□□□□□□□□□□ □□□□□□□□ □□□□□□□□□□□□□□□□□:
- Create a New Branch:
Command:
git branch <branch_name>

- Switch to a Branch:
Command:
git checkout <branch_name>

- Create and Switch to a New Branch:
Command:
git checkout -b <new_branch_name>

- List Branches:
Command:
git branch

3. □□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□:
- Stage Changes:
Command:
git add <file_name>

- Stage All Changes:
Command:
git add .

- Commit Changes:
Command:
git commit -m "Commit message"

4. □□□□□□□□□□□□□□□ □□□□□□ □□□□□□□□□□□□□□□□:
- Pull Changes from Remote:

Command:
git pull origin <branch_name>

- Push Changes to Remote:
Command:
git push origin <branch_name>

5. [illegible]:
- Merge Branch into Current Branch:
Command:
git merge <branch_name>

6. [illegible]:
- Add a Remote Repository:
Command:
git remote add <remote_name> <repository_url>

- List Remote Repositories:
Command:
git remote -v

7. [illegible]:
- Check for Conflicts:
Command:
git diff

- Resolve Conflicts and Continue Merge:
Command:
git add <file_name>
git merge --continue

8. [illegible]:
- Workflow Syntax Checking:
Command:
git pull origin <branch_name>
git push origin <branch_name>

9. [illegible]:
- Check Git Status:
Command:
git status

- View Commit History:
Command:
git log

- Ignore Files (Add to `.gitignore`):
Command:
echo "<file_name>" >> .gitignore

□□□□□□□□□□□□□□□□□ □□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□ : https://www.thealpha.dev/

1.Core:
- git init
- git clone
- git add
- git commit
- git status
- git diff
- git checkout
- git reset
- git log
- git show
- git tag
- git push
- git pull

2.Branching:
- git branch
- git checkout -b
- git merge
- git rebase
- git branch --set-upstream-to
- git branch --unset-upstream
- git cherry-pick

3.Merging:
- git merge
- git rebase

4.Stashing:
- git stash
- git stash pop
- git stash list
- git stash apply
- git stash drop

5.Remotes:

- git remote
- git remote add
- git remote remove
- git fetch
- git pull
- git push
- git clone --mirror

6.Configuration:
- git config
- git global config
- git reset config

7. Plumbing:
- git cat-file
- git checkout-index
- git commit-tree
- git diff-tree
- git for-each-ref
- git hash-object
- git ls-files
- git ls-remote
- git merge-tree
- git read-tree
- git rev-parse
- git show-branch
- git show-ref
- git symbolic-ref
- git tag --list
- git update-ref

8.Porcelain:
- git blame
- git bisect
- git checkout
- git commit
- git diff
- git fetch
- git grep
- git log
- git merge
- git push
- git rebase
- git reset
- git show
- git tag

9.Alias:

- git config --global alias.<alias> <command>

10.Hook:

- git config --local core.hooksPath <path>

11.Experimental: (May not be fully Supported)

- git annex
- git am
- git cherry-pick --upstream
- git describe
- git format-patch
- git fsck
- git gc
- git help
- git log --merges
- git log --oneline
- git log --pretty=
- git log --short-commit
- git log --stat
- git log --topo-order
- git merge-ours
- git merge-recursive
- git merge-subtree
- git mergetool
- git mktag
- git mv
- git patch-id
- git p4
- git prune
- git pull --rebase
- git push --mirror
- git push --tags
- git reflog
- git replace
- git reset --hard
- git reset --mixed
- git revert
- git rm
- git show-branch
- git show-ref
- git show-ref --heads
- git show-ref --tags
- git stash save
- git subtree
- git tag --delete
- git tag --force

- git tag --sign
- git tag -f
- git tag -l
- git tag --verify
- git unpack-file
- git update-index
- git verify-pack
- git worktree

---------- END ----------

Some good resources to Learn Git faster ☺

1.Git Official Documentation:

https://git-scm.com/doc

2.GitHub Learning Lab:

https://rb.gy/ksc45f

3.Codecademy Course

https://lnkd.in/g-87iUdn

4. Pro Git: by Scott Chacon [Book]

https://lnkd.in/gecriC8P

5. YouTube

• FreeCodeCampOrg- beginer

https://rb.gy/ljxt5s

• FreeCodeCampOrg- Intermediate

https://rb.gy/1x6mc

• Programming with mosh

https://rb.gy/vfkom

Revision #4
Created 23 April 2024 14:24:15 by sedawk
Updated 14 March 2025 12:26:13 by sedawk