

Config

```
git config --global user.name "[name]"
git config --global user.email "[email address]"
git config --global color.ui auto
```

Init

```
git init [dir-name]
git clone [url]
git remote add [origin] [url]
git remove -v (list of remote)
git remote remove [origin]
```

Stage

```
git add [File|.|-A]
git status
git diff --staged
git commit [-m|--amend|-am] "texto" (Dont amend published commits)
```

Tag

```
git tag -a 1.1 -m "Descripcion" [HASH]
git tag -l
git tag -d [HASH]
git push --tags
```

Log

```
git log --graph|--oneline
git log -p file
git blame archivo
git reflog
```

Reset

```
git reset --soft [HASH] (solo commit)
git reset --mixed [HASH] (solo commit y staging)
git reset --hard [HASH] (commit, staging y worktree)
git reset --hard HEAD (undone current work)
git reset HEAD (undone git add)
```

Reset (cont)

```
git checkout HEAD "file" (undone current work of file)
```

Branch

```
git branch "name" | git checkout -b "name"
git branch [-av] (local and remote)
git branch -m "old-name" "new-name"
git checkout "HASH" (temp view of hash)
git merge "branch2" (merge current-branch with branch2)
git branch -d
```

Stash

```
git stash
git stash list
git stash apply
git stash drop
```

Cherry Pick

```
git cherry pick [HASH]
```

Fetch, pull, push

```
git fetch origin          git pull origin
[branch]                 [branch]
git push -u origin branch
```

Search, rename

```
git grep "hola mundo" [v2.x]
git mv [old-name] [new-name] (move o change name case-sensitive)
```

Remove wrong files committed before

```
gitignore
git rm -r --cached .
git add .
git commit -m "Remover archivos accidentalmente agregados al commit antes de ser añadidos al .gitignore"
```

Keep folders

```
touch .gitkeep
folder_with_tmp_files/*
!folder_with_tmp_files/.gitkeep
```

git-flow

```
[apt|brew] install git-flow
git flow init
```

Feature

git flow	start	[feature-name]
feature		
	finish	[feature-name]
	publish	[feature-name] (send to remote)
	track	[feature-name] (get remote feature)

Release

git flow	start	[release-name]
release		[optional base hash in dev]
	publish	[release-name]
	track	[release-name]
	finish	[release-name]

```
git push --tags
```

Hotfix

git flow	start	[hotfix-name] [optional base hash in dev]
hotfix		
	finish	[hotfix-name]

