

Configuration

```
git config list
```

Display all settings

```
git config --global user.name $user
```

Set user name

```
git config get user.email
```

Display user name

```
git config --global user.email $email
```

Set email address

```
git config get user.name
```

Display email address

```
git config --global core.editor $editor
```

Set text editor

```
git config --global alias.$alias $cmd
```

Set alias for a command

```
git config --global init.defaultBranch $branch
```

Set default name for *main* branch

```
git config --global core.autocrlf true
```

Set end of line character (Windows). Default value

```
git config --global core.autocrlf input
```

Set end of line character (Linux/MacOs)

Initializing, Staging & Committing

```
git help --all
```

Print all commands available

```
git $cmd -help
```

Print command options

```
git $cmd --help
```

Open Git documentation in browser

```
git --version
```

Display Git version

```
git init
```

Initialize working tree

Initializing, Staging & Committing (cont)

```
git status
```

Display state of working tree and index

```
git status -s
```

Display summarized version

```
git add $file
```

Stage file

```
git add .
```

Stage all changes recursively

```
git add -A
```

Stage changes to match working tree

```
git commit -m $msg
```

Commit changes with message

```
git commit -am $msg
```

Commit no staged changes

```
git commit --amend
```

Edit last commit message (CLEAN index)

```
git checkout $hash
```

Move to previous commit

```
git mv $old $new
```

Move/rename and stage

```
git rm -f $file
```

Remove file from tree and index

```
git rm --cached $file
```

Remove file from index

```
git rm --cached -r $dir
```

Remove directory recursively from index

Logs

```
git log
```

Display commit history

```
git log --oneline
```

Display shortened commit history



Logs (cont)

```
git log -$n
```

Display **n** commits

```
git log --stat
```

Display changes, insertions & deletions

```
git log -p
```

Show committed changes file by file

```
git log --all --decorate --oneline --graphic
```

Display commit history tree

```
git reflog --relativ e-date
```

Display command history

```
git blame $file
```

Display commit, author & date per line

```
git diff
```

Compare working tree and index

```
git diff $file
```

Compare file in working tree and index

```
git diff --cached
```

Compare index and last commit

```
git diff HEAD
```

Compare working tree and last commit

```
git show $hash
```

Display changes in commit

```
git show $hash:$file
```

Display file content in commit

```
git ls-files
```

Display files in index

```
git ls-tree --name -only HEAD
```

Display files in local repository

Branching

```
git branch
```

List local branches

```
git branch -r
```

List remote branches

```
git branch -a
```

List local and remote branches

```
git branch $branch
```

Create branch

```
git branch $branch $src
```

Create branch rooting from other

```
git branch -m $old $new
```

Rename branch

```
git branch -d $branch
```

Delete branch

```
git branch -D $branch
```

Delete NOT merged branch

```
git checkout $branch
```

Move to branch

```
git checkout -b $branch
```

Create branch and move

```
git switch $branch
```

Move to branch

```
git switch -
```

Move to previous branch

```
git switch -c $branch
```

Create branch and move

```
git merge $branch
```

Merge branch into current one



Remote Repositories

```
git remote add $alias $url
```

Add remote repository

```
git remote -v
```

List remote repositories

```
git remote rename $old $new
```

Rename remote repository

```
git clone $url
```

Clone remote repository

```
git clone $url $path
```

Clone remote repository to specific folder

```
git diff $alias/$branch
```

Compare index and remote repository

```
git log $alias/$branch
```

Display commit history from remote repository

```
git fetch $alias
```

Download information from remote repository

```
git merge $alias/$branch
```

Merge remote repository into current branch

```
git pull $alias
```

Pull (fetch+merge) remote repository into current branch

```
git pull $alias $branch
```

Pull remote repository into specific branch

```
git push $alias
```

Push current branch into remote repository

```
git push $alias $branch
```

Push specific branch into remote repository

```
git push --set-upstream $alias $branch
```

Push to remote repository and set it as default target for that branch

```
git ls-remote
```

Display remote repository references

Undoing Changes

```
git checkout HEAD $file
```

Retrieve file from last commit

```
git checkout $hash $file
```

Retrieve file from other commit

```
git restore $file
```

Retrieve file from index

```
git restore --staged $file
```

Retrieve index file from last commit

```
git revert
```

Create new commit that undoes last commit's changes

```
git revert $hash
```

Revert specific commit

```
git revert HEAD --no-edit
```

Revert last commit skipping message editor

```
git revert HEAD~$n --no-edit
```

Revert n commits back skipping editor

```
git reset --soft $hash
```

Delete commit. Undo repo. Leave wd and index unchanged

```
git reset $hash
```

Delete commit. Undo repo and index. Leave wd unchanged

```
git reset --hard $hash
```

Delete commit. Undo repo, index and wd

Advanced Topics

```
git rebase $hash
```

Move branch ancestor to other commit

```
git rebase -i HEAD~$n
```

Squash. Combine n commits into one

```
git cherry -pick $hash
```

Append commit to current branch

```
git stash
```

Save wd and index to stash. Revert to last commit

Advanced Topics (cont)

```
git stash list
```

List available stashes

```
git stash pop
```

Apply latest stash. Remove it from stack

```
git stash pop stash@{$n}
```

Apply specific stash. Remove it from stack

```
git stash apply
```

Apply latest stash. Keep it in stack

```
git stash drop
```

Remove latest stash from stack

```
git tag
```

List tags

```
git tag $name
```

Create tag to current commit

```
git tag $name -m $msg
```

Create tag with message

```
git tag $name $hash
```

Create tag to specific commit

```
git push $remote --tags
```

Push tags to remote repository

```
git worktree list
```

List working trees

```
git worktree add $path
```

Create linked tree

```
git worktree remove $tree
```

Delete tree

Debugging with bisect



By Jesús Arellano Sánchez
(ludovico14)
cheatography.com/ludovico14/

Not published yet.
Last updated 25th January, 2025.
Page 4 of 4.

Sponsored by [Readable.com](https://readable.com)
Measure your website readability!
<https://readable.com>