**Essential git commands every developer should know**

[[dhruv profile image](https://dev.to/dhruv) Dhruv](https://dev.to/dhruv)Sep 28 '17 ・3 min read

[**#git**](https://dev.to/t/git) [**#commandline**](https://dev.to/t/commandline) [**#versioncontrol**](https://dev.to/t/versioncontrol) [**#bestpractices**](https://dev.to/t/bestpractices)

If you’re one of those developers who still don’t use any version control system, I don’t know how you’re still managing to get work done.  
In this post, I’m focusing on important git commands that gets all (almost) your work done ( I know you’re a GUI person ).

**1) git config**

**Utility** : To set your user name and email in the main configuration file.  
**How to** : To check your name and email type in ***git configâ€Š --â€Šglobal user.name*** and ***git config â€Š--â€Šglobal user.email***. And to set your new email or name ***git configâ€Š --â€Šglobal user.name = “Dhruv Nenwani”*** and ***git configâ€Š --â€Šglobal user.email = “***[***nendhruv@gmail.com***](mailto:nendhruv@gmail.com)***”***

**2) git init**

**Utility** : To initialise a git repository for a new or existing project.  
**How to** : ***git init*** in the root of your project directory.

**3) git clone**

**Utility** : To copy a git repository from remote source, also sets the remote to original source so that you can pull again.  
**How to** : ***git clone <:clone git url:>***

**4) git status**

**Utility** : To check the status of files you’ve changed in your working directory, i.e, what all has changed since your last commit.  
**How to** : ***git status*** in your working directory. lists out all the files that have been changed.

**5) git add**

**Utility** : adds changes to stage/index in your working directory.  
**How to** : ***git add .***

**6) git commit**

**Utility** : commits your changes and sets it to new commit object for your remote.  
**How to** : ***git commit -m”sweet little commit message”***

**7) git push/git pull**

**Utility** : Push or Pull your changes to remote. If you have added and committed your changes and you want to push them. Or if your remote has updated and you want those latest changes.  
**How to** : ***git pull <:remote:> <:branch:>*** and ***git push <:remote:> <:branch:>***

**8) git branch**

**Utility** : Lists out all the branches.  
**How to** : ***git branch*** or ***git branch -a*** to list all the remote branches as well.

**9) git checkout**

**Utility** : Switch to different branches  
**How to** : ***git checkout <:branch:> or \*\*\_git checkout -b <:branch:>*** if you want to create and switch to a new branch.

**10) git stash**

**Utility** : Save changes that you don’t want to commit immediately.  
**How to** : ***git stash*** in your working directory. ***git stash*** apply if you want to bring your saved changes back.

**11) git merge**

**Utility** : Merge two branches you were working on.  
**How to** : Switch to branch you want to merge everything in. git merge ***<:branch\_you\_want\_to\_merge:>***

**12) git reset**

**Utility** : You know when you commit changes that are not complete, this sets your index to the latest commit that you want to work on with.  
**How to** : ***git reset <:mode:> <:COMMIT:>***

**13) git remote**

**Utility** : To check what remote/source you have or add a new remote.  
**How to** : ***git remote*** to check and list. And ***git remote add <:remote\_url:>***

These are the commands that I feel are essential and get things done, at least for me. Comment here if you think I’ve missed something important or if something can be done differently.

This was originally published on [Medium](https://medium.com/@nendhruv/essential-git-commands-every-developer-should-know-1249d4d597b5)