

### Most Frequently Used Commands

`git clone <url>`

Downloads a copy of a remote repository. Remember to use the Kerberos (KB5) URL from Gitlab.

`git add [file name]`

Add a modified file to the staging area. You can also add an entire directory by using the directory name instead.

`git commit -m "[commit message]"`

Commits all staged files to the repository with a commit message. If you accidentally run "git commit" without a message, it will likely open Vim. Press "i" to enter insert mode, write a commit message, hit escape, then type ":wq" and hit enter to save the message.

`git branch [branch name]`

Create a new branch. *This will not switch the current branch to the newly created branch.* Use "git checkout [branch name]" to switch the current branch.

`git checkout [branch name]`

Switch the currently active branch to [branch name]. Note that "git checkout" does many different things depending on the exact arguments you pass to it. Switching branches is the primary use case.

`git push -u [remote repo] [branch name]`

Upload the current branch to the remote repo. Every time you make a new branch, you should run this command with the "-u" option. This sets up a default so git will understand which branch on the remote repo is supposed to track the current branch in the local repo. After running this the first time, you can simply run "git push" with no other arguments.

`git fetch`

Download all files from the remote repo. This does not actually change your working directory, it just stores a copy of the remote repository in your .git folder.

`git merge`

With no arguments, merge will update your local repository with all the changes that were most recently fetched from the remote repository. This will change your working directory.

`git pull`

In theory, this is the same thing as running "git fetch" and then running "git merge". Most of the time, you will be running "git pull", but occasionally you will need to fetch and merge manually.

`git merge [branch name]`

### Most Frequently Used Commands (cont)

Updates the current branch with all the changes committed in [branch name].

`git status`

Lists the current state of the working directory, staging area, and the repository. I would recommend running this very frequently. You should at least run this before you commit any modifications.

### Glossary

Working directory (A.K.A. "working tree")

The directory you are currently working in. This will have all the files you are working with, regardless of their status according to git. This is not a git specific concept.

Staging area

This is where git stores the changes you have added but have not yet committed. You can think of this as a space that's "inbetween" the working directory and the repository

Repository (A.K.A. "repo")

This is where git stores all of your commits. You can think of this as the hidden ".git" directory. Your local repository is what you will be working with directly. It's the repository on your computer.

Remote repository

A git repository that tracks the same project but exists in some other location. Typically, the Gitlab repo is the only remote repo you will be pushing to. But everyone else involved in the project will probably have their own repo that could be considered "remote" in some sense. Every remote repo is given a name stored in your local repo. By convention, the original repository you cloned from will be given the name "origin."

Commit

As a noun: A snapshot or version of your project. A point in .git history.

As a verb: to create a new commit by using the "git commit" command.

head (lowercase)

The most recent commit of a branch.

HEAD (uppercase)

The head of the current branch.



### .gitignore Syntax

Pattern	Explanation
# text comment	Comments in the .gitignore file that are ignored. Useful for documentation
name.file	A specific file to be ignored. You can specify a path to a subdirectory with "/"
*end_of_name.file	Ignore every file that ends in "end_of_name.file"
start_of_name*	Ignore every file that starts with "start_of_name"
directory_name/*	Ignore every file in the "directory_name" folder



By **ethan.rahman**

[cheatography.com/ethan-rahman/](https://cheatography.com/ethan-rahman/)

Not published yet.

Last updated 6th October, 2023.

Page 2 of 2.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>