Git Cheatsheet

Meedos

Introduction

This is a short cheatsheet on how I use Git daily. I don't use the more complex commands yet as I haven't learned them. Hope this is useful to some of you. I'll try to break it up in use-cases or scenarios with examples as I believe it is easier to understand.

Initializing a new git repo

On an existing codebase/project or on a empty directory type :

```
$ git init
```

This command creates a hidden folder named .git where all the git magic happens (hashes, working dir, etc...).
By default this creates the default branch "master" which you are on now, more on that later.

Most basic usage

The following commands are the most used commands by far, you'll do them a couple of times a day.

Add work to the index

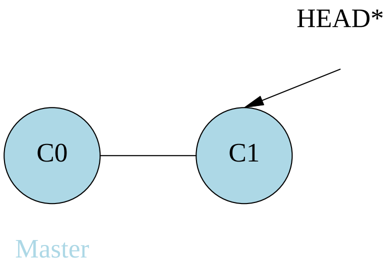
```
$ git add .
```

This command adds your files to the index waiting to be committed. The "." means all that is within the current directory ("*" works fine too). As an image, "git add" takes a snapshot of the content of the working tree, that snapshot will then be used for the next commit.

Committing To save your changes to the repository

```
$ git commit -m 'This is a commit message'
```

This creates a new commit with the current index's content.



HEAD* is where you are currently in the tree. Doing another commit will continue the tree to the right with "C2" so on and so forth.

Pushing to a remote

Let’s say you are working with a team and the code base is meant to be stored on a remote server such as gitlab.com, gitea, or even github.com. You first need to create a repository on one of those services and push you local changes to that remote. The next commands will assume you already created a repo at https://gitlab.com/meedos/test_repo.git

adding a remote

You can add a remote with a given name and a url with.

```
$ git remote add origin https://gitlab.com/meedos/test_repo.git
```

A local repo can have multiple remotes.