Fundamental git commands.

1. git clone → Git clone creates a copy of the project in your local working environment. You just need to provide a path for the project. This path can be copied from the project main on the hosting services such as GitLab and GitHub.

```
# clone with HTTPS
git clone https://github.com/*****
```

```
# clone with SSH
git clone git@gitlab.com:*****
```

2. git branch → Once you clone the project to your local machine, you only have the master branch. You should make all the changes on a new branch that can be created using git branch command.

```
git branch mybranch
```

Your branch is the copy of the master branch until you make any changes.

3. git switch → Creating a new branch does not mean that you are working on the new branch. You need to switch to that branch.

```
git switch mybranch
```

You are now on "mybranch" branch, & you can start making changes.

4. git status → Git provides a brief summary of the current status. you will see what branch you are working on. It also shows if you have make any changes or anything to commit.

```
git status
On branch mybranch
nothing to commit, working tree clean.
```

5. git add → When you made changes in the code, the branch you work on becomes different from the master branch. These changes are not visible in the master branch unless you take a series of actions.
The first action is the git add command. This command adds the changes to what is called the staging area

git add

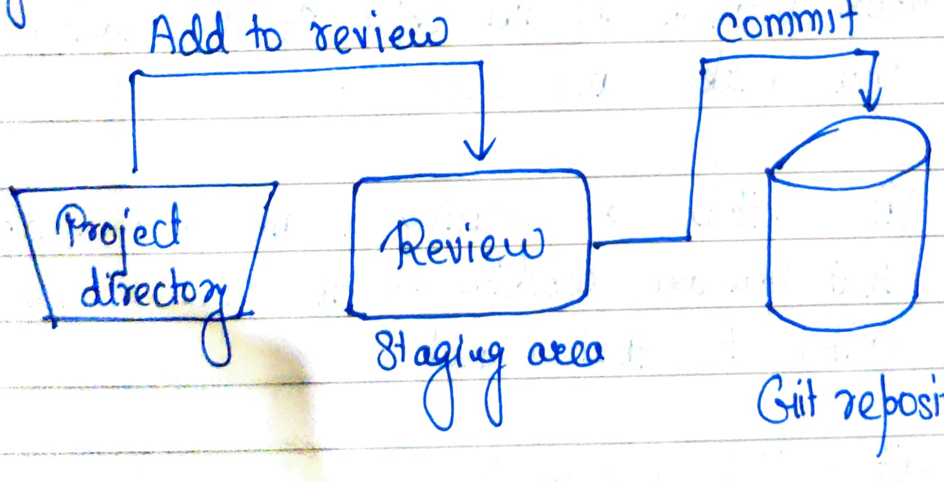

fig: Basic git workflow.

```
→ git commit -m "your message"
```

6. git commit → It is not enough to add your updated files or scripts to staging area. You also need to "commit" these changes using git commit commad. The imp. part of this cmd is the message part. It briefly explains what has been changed or the purpose of this change. There is not a strict set of rules to write commit msg. Msg should not be lengthy, but should explain what the change is about.

7. git push → The add & commit method make the changes in your local git repository. In order to store these changes in a remote branch (ie; master branch), you first need to push your code. It is worth mentioning that some IDEs like PyCharm allow for committing & pushing from the user interface. However, you still need to know ~~that~~ what each cmd does. After your branch is pushed, you will see a link in terminal that will take you to the hosting service website (Github, Gitlab). The link will open a page where you can create a merge req. A merge request is asking the maintainer of the project to "merge" your code to the master branch. Maintainer will first review your code. If the changes are ok, your code will be merged. Maintainer might also abort your branch & restore the master branch

8. git pull → The purpose of using a version control system is to maintain a project with many contributers. Thus, which you are working on a task in your local branch, there might be some changes in the remote branch. Git pull cmd is used for making your local branch up to date. Should use to update your local working directory with the latest files in the remote branch.