If it’s a new repo that one is trying to setup:

1. Create the repo on github.com Say: <repo name>
2. Now on the local machine go to the directory that contains the content to be repo-ed:
   * In the Terminal type: git init
   * This will create hidden files that Git uses to manage its source code control. The idea is that you will have a single local directory which you will sync with the one on your GitHub account.
   * If you delete or replace the contents of this directory, it will mess up your configuration, so try to avoid anything like this.
   * You only need to do this once for each Git repo on your local drive.
3. We create an alias for the remote repo that allows for easy and simple commands to synchronise the local and remote repos.
   * In the terminal, type: git remote add origin https://github.com/<username>/<repo name>.git
     1. where <username> is your GitHub account username and <repo name> is your repository name that you just created.
   * To communicate with the outside world, Git uses *remotes*. These are repositories other than the one on your local disk which you can push your changes into (so that other people can see them) or pull from (so that you can get changes from others). This is essentially how this system works for Git and GitHub.
   * The command git remote add origin creates a new remote called origin located on GitHub.
   * Once you do this, you can push to origin instead of typing out the whole URL.
4. We now select files that we would like to add to the remote repo.
   * In the terminal, type: git add .
   * This will add all the files (and recursive directories) to what will be a new "commit" of your source code.
   * Note, this command won't echo any response in the Terminal window.
5. Now, you need to commit the files that you've just added, creating the snapshot of your source code.
   * In the terminal type: git commit -m "initial source commit"
   * The -m flag is used to add a message about your the commit, which is standard practice. This should be a description of the most recent changes you've made.
   * You'll see some details, which shows which files were included in the commit.
6. Pushing the code will upload your local changes to the github repo.
   * In the terminal, type: git push origin master
   * This will upload your local repository into the remote connection that we previously called *origin*and *master*is the branch name