Git Commands

```
$ git init
```

→ To initialize a hidden folder of git in the current folder/ directory

```
$ git config --global user.name "Your Name"
```

```
$ git config --global user.email "you@exp.com"
```

Note :-

Above two (immediate two) commands are for first time user after downloading git to set up your github username and email locally on to your system

To check whether credentials are added Navigate to

1. → Control panel on your system.

2. → In User Accounts > Manage Network passwords

3. → In Windows Credentials

Find | Generic Credentials

git https://github.com

{ - - -

details

- - - -

}

$ git add <filename.filetype>

$ git add . ← to upload all files in directory/folder to github

To add single file to github

$ git remote add origin <url>

Here Url is the url in your repository → <> Code ▾ → local

HTTPS

<- url ->

```
$ git commit -m "first commit"
```

→ -m is used to type commit msg here first commit is the commit msg.

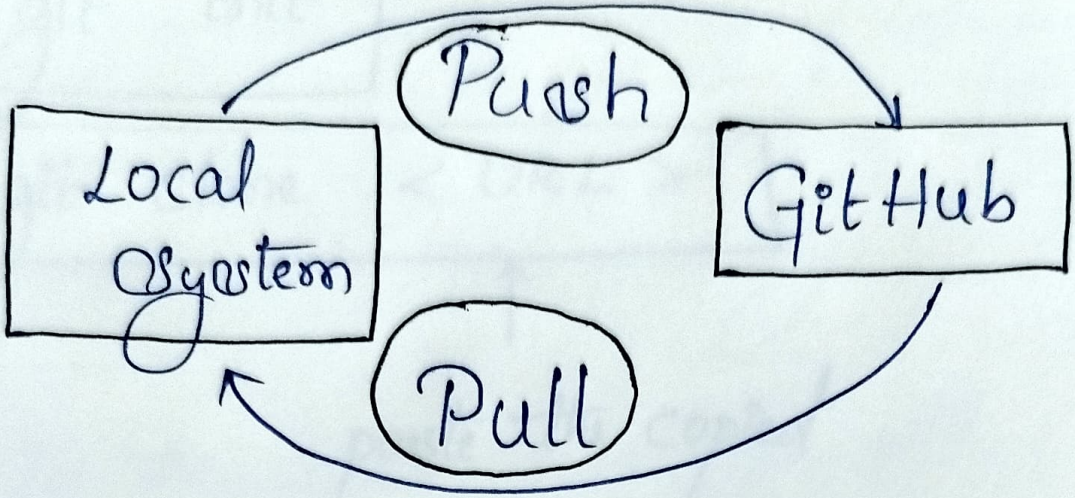
```
$ git push -u origin master
```

→ This command is for initial push request to master branch in github.

Note :-

When files or folders are moved

From		To
Local / Laptop	Push →	GitHub
GitHub	Pull →	Local / Laptop



To clone a repository locally

→ First copy the repository URL from Code option in repository you want to clone from GitHub.

→ Open Git Bash in the folder where you want to clone your GitHub repo locally.

Execute the following steps :-

```
$ git init
```

```
$ git clone <URL>
```

↑

paste the copied URL Here

⇒ Now you have cloned the repo locally in your system.

⇒ If you want you can download from the zip folder in Code option, but it is not prefered.

Forking / Fork

→ To create a copy of another github repo (of someone [stranger]) in your github repository.

Forking → Creating a copy of stranger repository, as a part of our repository / in our repository.

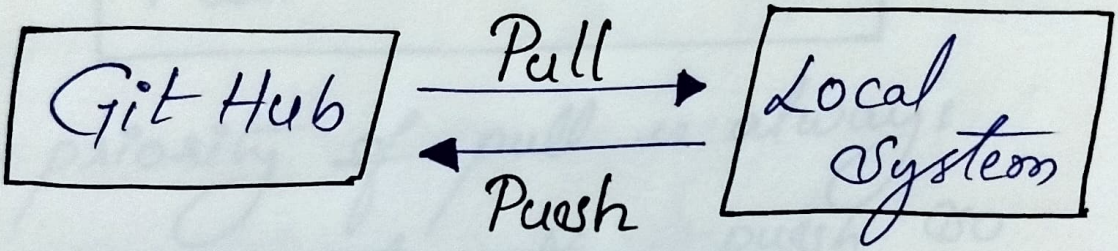
⇒ In the copied repo you will have an option to contribute i.e after making changes you can push the changes back to the stranger's repo and those changes will be reflecting in his repo if accepts our push request.

Pull Request

⇒ When your GitHub repo files and files stored in your system locally are not synced in (i.e

local system is missing files from comparing with files in GitHub repo).

⇒ To create copy of those files locally from GitHub we initiate a Pull Request.



$ git status

→ This command allows you to check the status of files

*** from local system to GitHub not from GitHub to local system

Meaning logically this command checks the push status not the pull status.

Priority Order

Pull >>> Push

⇒ priority of pull is always so much greater than push so we must always check the status properly (i.e whether the GitHub files and local files are in sync or not).

⇒ Always perform pull first then push changes in order to reduce chances of faults & errors.

Imp Note

→ When you made changes in your local files and saved locally but made a pull request from GitHub that has the same files but has other modifications, then due to the pull your modifications will no longer exists in the files so, we must be very careful while doing a pull request.

→ This is the reason why we must always pull first and then check the sync and only then do our contribution or modification to the code.

`$ git push` → rather than first time rest all times we use this command to push.

Branch

→ Initially Branch is a copy of main

Example :- A Tree having several branch but a single trunk (main).

- To switch to a certain branch (existing) we use

```
$ git checkout branchname
```

- To create a new branch (non existing branch) we use

```
$ git branch home
```

- To merge a branch we use

```
$ git merge branchname
```