Git bash command to work with git hub

Start a repository from scratch

First you need to create a repo in your profile at the github website

In order to connect it with a folder in your computer you will make a copy if

First make a directory

$ mkdir ~/git-repo

Go to it

$ cd git-repo

Initialized empty Git repository in c:/Users/user/git-repo/.git/ (your local copy)

$ git init

Point your local repository at the remote repository you created on GitHub server type:

$ git remote add origin <https://github.com/Udlan/hello-world>

(the command is "git remote add origin" and then the URL of your new repo)

[in order to paste into the git bash terminal press the Insert button]

Fork another Users Repository

You can make a copy of another user repo in order to have it and work with it on your own.

Go to the repo web page and click on the "Fork" button.

This will make a copy of this repo on your GitHub account.

You can download it to your local directory of GitHub ("clone" it)

$ git clone "the URL of the forked depo"

Basic git commands

The way connect between files in your computer to the GitHub is first you index them and then commit them to your GitHub repo as a intermediate changes

Add or update file in your working directory to the index

* git add .

add all new files

* git add -u

Will update changes

* git add -A

add new and update files

Committing the index

* git commit -m "massage"

Where message is a useful description of what you did

This only updates your local repo, not the remote repo on GitHub.

Pushing

Update your local commits to the remote repo (on GitHub)

* git push

Pulling

pull down changes from the origin repository

* git pull origin master

this will take all the files in your master root branch into your local rep.

branch

you can create a branch of some repo to work on without changing the original repo

from the the repo you want to wark on type:

* git checkout -b brsnchname

(branchname = the name of the new branch)

To see what branch you are on

* git branch

to switch back to the master branch

* git checkout master