Lab 1.3 GitHub

Clone https (you can use this to get sample code) read-only

1. Get the url you want:

either my

```
https://github.com/andrewbeattycourseware/pands-course-material.git
```

Or from github

   a. Go to GitHub
   b. Select the repository you want to clone (create one if required)
   c. In code select the https
   d. Copy the url to the clipboard by clicking the clipboard symbol on right

2. Go to CMDER (or terminal in VSCode)
3. Navigate to where you want to make the repository
4. Clone the repository (you can paste the name in right clicking)

```
git clone PASTED.URL
```

5. Set the pull mode to merge

```
git config pull.rebase false
```

6. Pull the content of the remote repository

```
git pull
```

7. If this was your repository you could add files and directories to it, commit and push, but you would have to enter your username and password each time you push…. Also I think there is some set up you need to do. This can be a pain so it would be easier if you could tell GitHub that the machine that you are on has permission to push to your repository…. That is where SSH comes in….

Clone using SSH (handy for your repository) (for your machine)

1. More information on this

[About SSH - GitHub Docs]

ssh-keygen creates a key pair
ed25519 used for encryption
-C email, this is probably not required

2. From your home directory (in my case /home/abeatty
3. Check if there are any keys already on your machine

```
ls .ssh
```

4. If none create one key pair

```
ssh-keygen -t ed25519 -C "your.email@gmit.ie"
```

5. Copy the public key to clipboard

```
clip < .ssh\id_ed25519.pub
```

clip	is the clipboard
<	redirect input

6. Go to GitHub
7. Click the icon in the top right and select settings
8. Select SSH and GPG keys
9. Click new SSH key and paste in the public key

You are now ready to clone your repository on GitHub

10. Go to the repository that you want; you may need to make one in your GitHub eg create one called pands-weekly-tasks (and you may want to do this with mywork)
11. Select code
12. Select SSH and copy the URL by clicking the clipboard

This is the repository you want to create

```
git@github.com:yourAccount/theRepository.git
```

In the terminal on your machine (CMDER).

13. Navigate to where you want to put this repository on your machine.
14. Clone the repository like you did with HTTPS

```
git clone PASTED.URL

git config pull.rebase false

git pull
```

So, assuming that you cloned your repository on GitHub, you can add files to it.

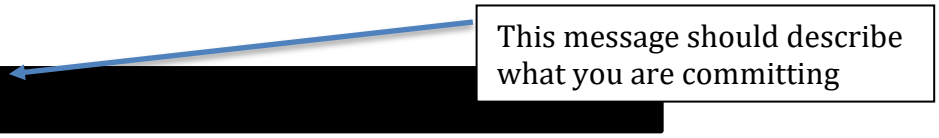
15. Create the file (and/or directory)

```
mkdir nameOfDirectory
cd nameOfDirectory
echo "test content" > test.txt
cd ..
```

16. Add these to the repository on your machine

```
git add .
```

17. Commit the changes




```
git commit -m "test"
```

18. Push these changes to GitHub

```
git push
```

19. Go to GitHub and see if the changes were made to your repository on it.

Extra Creating a repository on local and linking with GitHub (for example you want to link mywork the hard way, are you really sure you want to do this)

1. Create a repository on GitHub
2. Go into the repository, click code and copy the URL (SSH if you have set up your keys already)
3. Navigate to the directory that you wish to make a repository out of.
4. Create the repository

```
git init -b main
```

5. If you try to pull now you will get an error like

```
git pull
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>
```

6. to fix this Set the remote url

```
git remote add origin git@github.com:yourrepo
```

7. Set the remote branch

```
git fetch
git checkoutmain
```

```
git pull
```

8. You should now be able to pull

```
git push
```

9. and push your code
10. Hold on nothing happened, ahhh we need to add the files to the repository

```
git add .
git commit -m “initial”
git push
```

Danger if you need to remove a repository from your machine just delete the .git directory

12. Check If the .git directory exists

```
ls -a
```

13. You can delete the repository (and not the files in it) Danger

```
rm -rf .git
```

14. Or you can delete the directory containing the repositoryDanger, cd to the parent directory of directory that contains the repository

```
rm -rf nameOfDirectoryContainingTheRepository
```