CS 235 Lab 2 report

Name : Rukun Aaron , Upi = raar518

Sections

- Notes
- Reflection
- Tasks

What is git?

- **Git the most popular version control system in the world**
  - It record the changes made to our code over type in a database, known as a repository.

- **Instead of manually saving version of a software, git allows us to verision code by doing the following:**
  - Creating a repository using following command (in our desired repository):

```
  git init
```

  - Or going on github (a repository hosting service) and creating a repository and cloning to your local machine using:

```
Cloning with HTTPS URLS :
git clone  https://github.com/Username/Your-Repository.git
```

```
Cloning with SSH over the HTTPS port
git clone git@github.com:Username/Your-Repository.git
```

  - Add a branch to make your changes:

```
Create a branch using:
git branch <branch_name>

Switch to the newly created branch:
git checkout <branch_name>
```

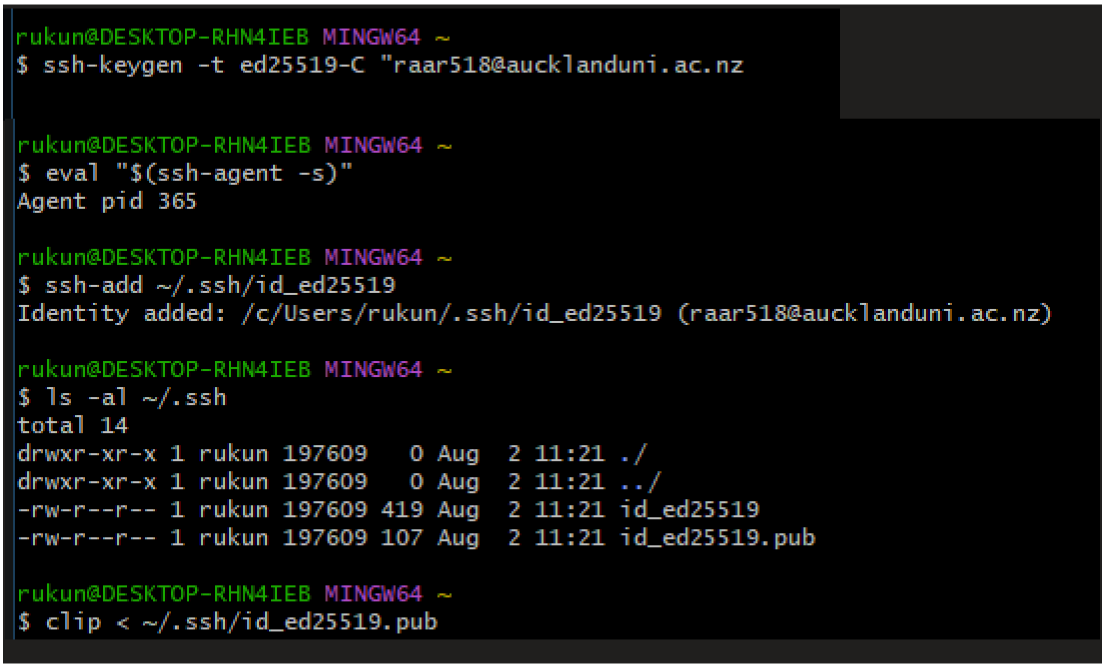
- After adding our changes we push the branch to the remote repository:

```
Stage all changes using:
git add .

Commit the changes and write a message using:
git comming -m <Message>

Update the remote repository on github using:
git push origin <branch_name>

Merge the new branch with the main/master branch:
git checkout main
git merge <branch_name>
```

What is SSH and why do we use it when using GitHub

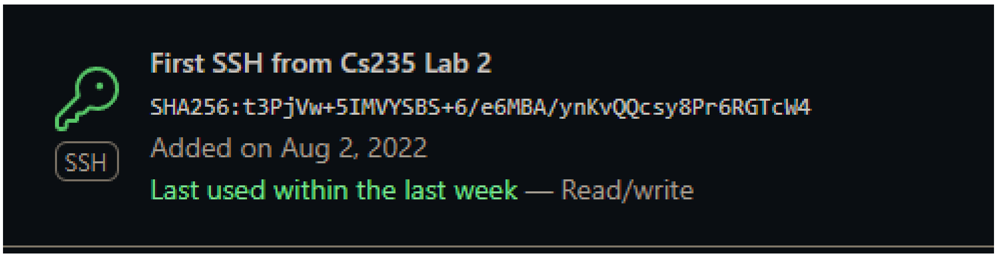
- **Github does not accept accept password to authenticate git commands; hence we use SSH to use to verify your identity**.

1. We created a pair of SSH key doing the following:



```
rukun@DESKTOP-RHN4IEB MINGW64 ~
$ ssh-keygen -t ed25519-C "raar518@aucklanduni.ac.nz


rukun@DESKTOP-RHN4IEB MINGW64 ~
$ eval "$(ssh-agent -s)"
Agent pid 365

rukun@DESKTOP-RHN4IEB MINGW64 ~
$ ssh-add ~/.ssh/id_ed25519
Identity added: /c/Users/rukun/.ssh/id_ed25519 (raar518@aucklanduni.ac.nz)

rukun@DESKTOP-RHN4IEB MINGW64 ~
$ ls -al ~/.ssh
total 14
drwxr-xr-x 1 rukun 197609   0 Aug  2 11:21 ./
drwxr-xr-x 1 rukun 197609   0 Aug  2 11:21 ../
-rw-r--r-- 1 rukun 197609 419 Aug  2 11:21 id_ed25519
-rw-r--r-- 1 rukun 197609 107 Aug  2 11:21 id_ed25519.pub

rukun@DESKTOP-RHN4IEB MINGW64 ~
$ clip < ~/.ssh/id_ed25519.pub
```

2. Then I uploaded my public key to my GitHub:



First SSH from Cs235 Lab 2

SHA256:t3PjVw+5IMVYSBS+6/e6MBA/ynKvQQcsy8Pr6RGTcW4

Added on Aug 2, 2022

Last used within the last week — Read/write

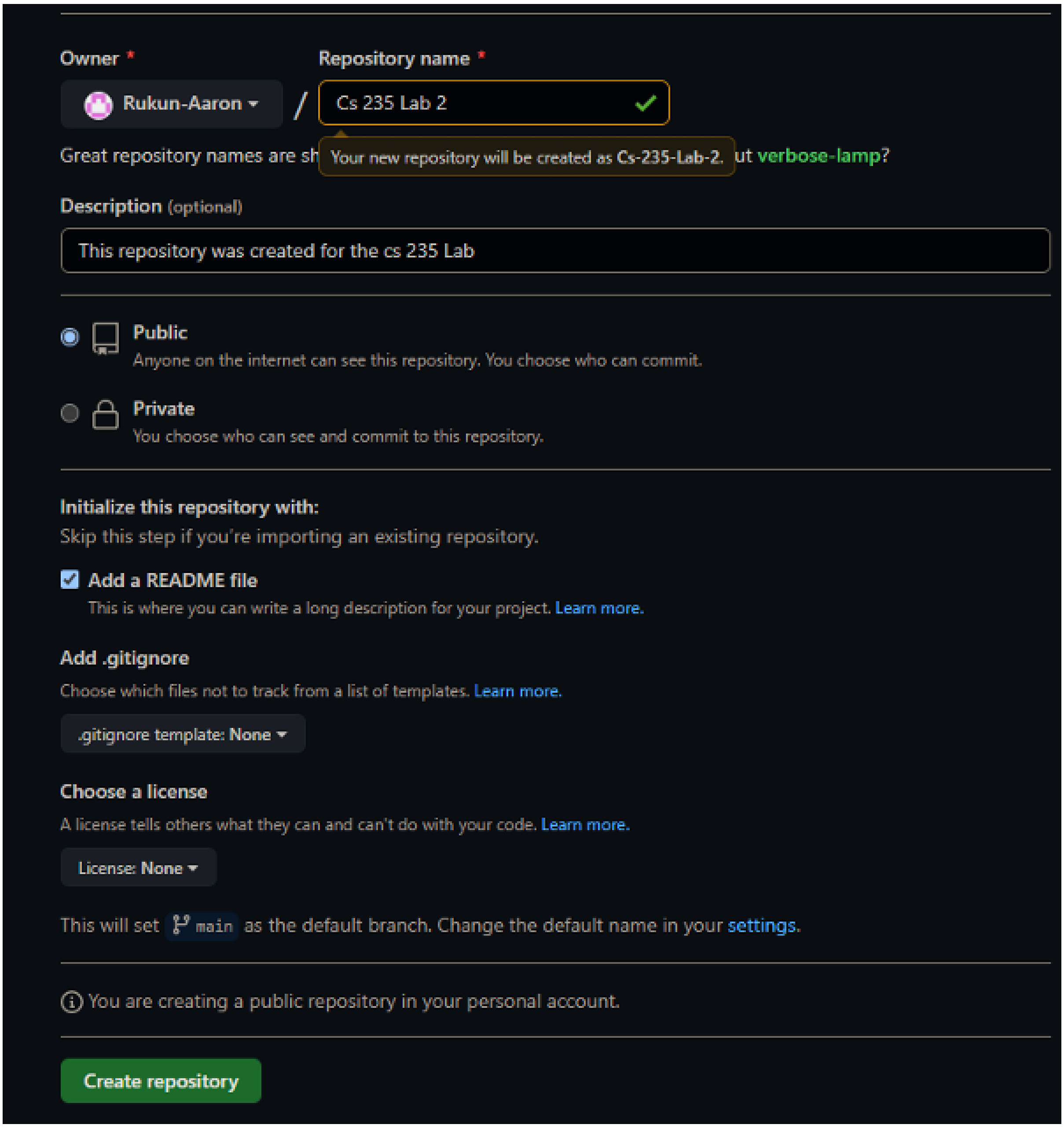
Reflection

What I learnt:

- I learnt various about various features of Git and Github which include:
  - Creating a repository
  - Cloning a repository
  - How to create a branch
  - How to push changes and merge branch
  - What forking is and how to create pull requests.
  - How to generate and add SSH keys to GitHub

Student A Tasks:

1. **Create a repository on GitHub and clone it:**



Owner * Repository name *

Rukun-Aaron / Cs 235 Lab 2 ✓

Great repository names are sh Your new repository will be created as Cs-235-Lab-2. ut verbose-lamp?

Description (optional)

This repository was created for the cs 235 Lab

● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☑ Add a README file
This is where you can write a long description for your project. Learn more.

Add .gitignore
Choose which files not to track from a list of templates. Learn more.

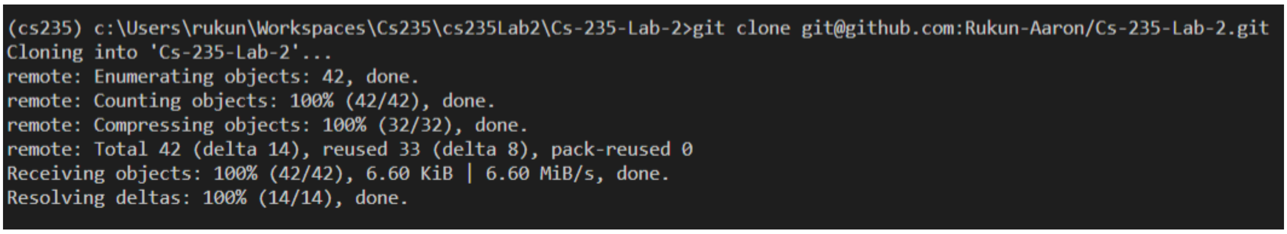
.gitignore template: None

Choose a license
A license tells others what they can and can't do with your code. Learn more.

License: None

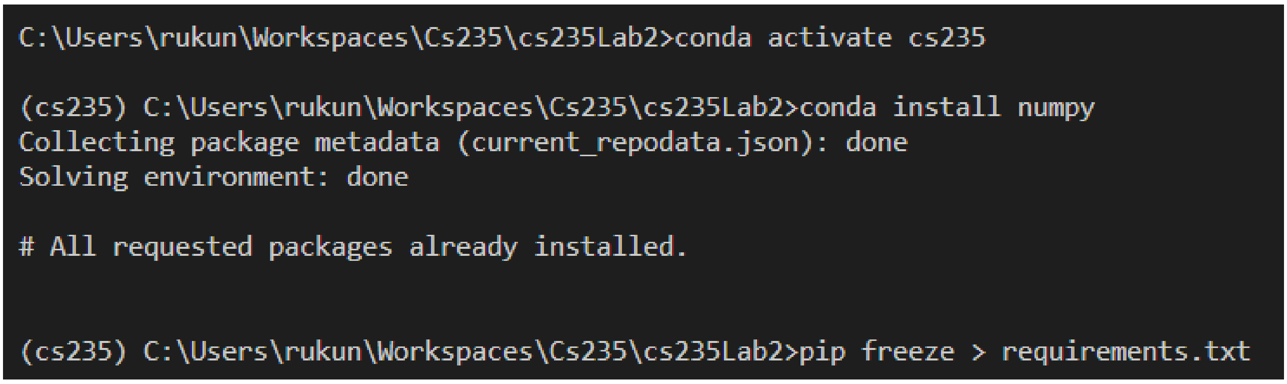
This will set main as the default branch. Change the default name in your settings.

ⓘ You are creating a public repository in your personal account.



```
(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git clone git@github.com:Rukun-Aaron/Cs-235-Lab-2.git
Cloning into 'Cs-235-Lab-2'...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (42/42), done.
remote: Compressing objects: 100% (32/32), done.
remote: Total 42 (delta 14), reused 33 (delta 8), pack-reused 0
Receiving objects: 100% (42/42), 6.60 KiB | 6.60 MiB/s, done.
Resolving deltas: 100% (14/14), done.
```

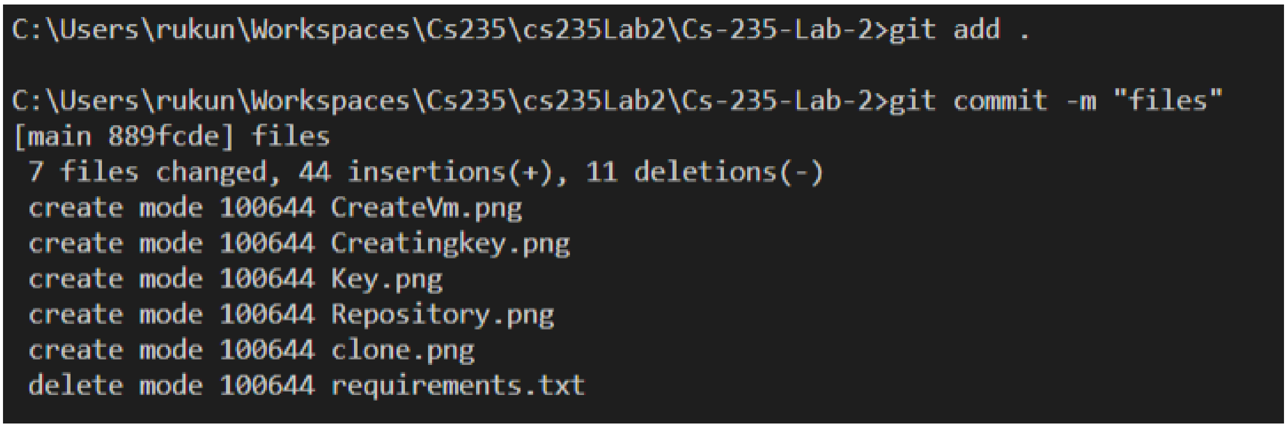
2. **Activating a virtual environment, generating a requirements file and adding the startup project to the repository:**



```
C:\Users\rukun\Workspaces\Cs235\cs235Lab2>conda activate cs235

(cs235) C:\Users\rukun\Workspaces\Cs235\cs235Lab2>conda install numpy
Collecting package metadata (current_repodata.json): done
Solving environment: done

# All requested packages already installed.


(cs235) C:\Users\rukun\Workspaces\Cs235\cs235Lab2>pip freeze > requirements.txt
```

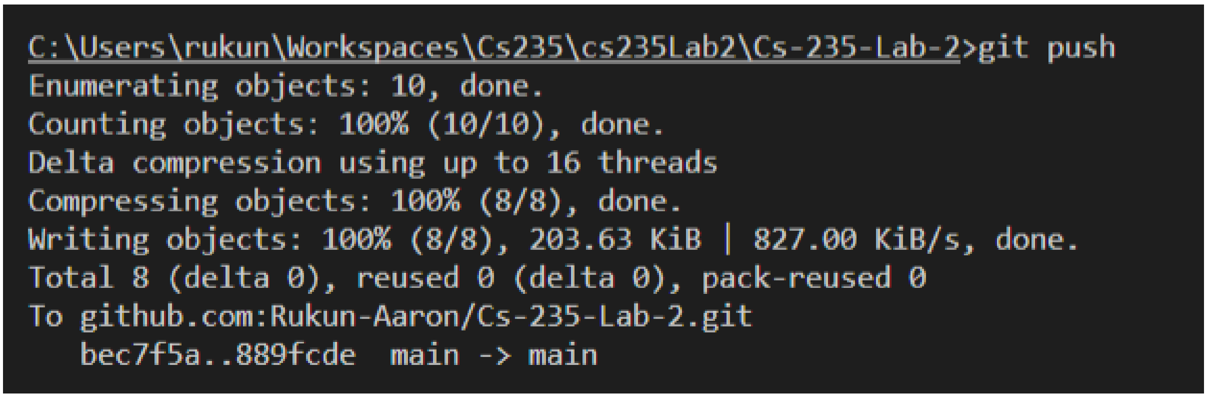
3. **Stage and commit changes to the main branch of the repository**

   - **Staged and commited files :**



```
C:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git add .

C:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git commit -m "files"
[main 889fcde] files
 7 files changed, 44 insertions(+), 11 deletions(-)
 create mode 100644 CreateVm.png
 create mode 100644 Creatingkey.png
 create mode 100644 Key.png
 create mode 100644 Repository.png
 create mode 100644 clone.png
 delete mode 100644 requirements.txt
```

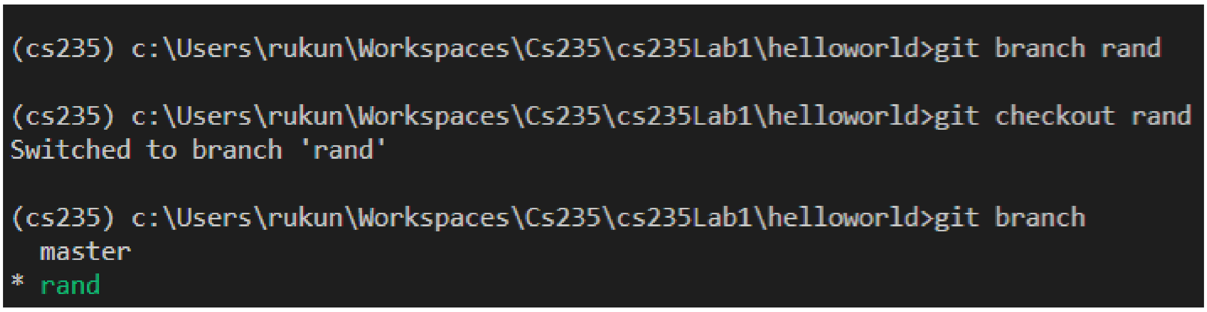
   - **Pushed changes to the remote repo**



```
C:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 16 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 203.63 KiB | 827.00 KiB/s, done.
Total 8 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Rukun-Aaron/Cs-235-Lab-2.git
   bec7f5a..889fcde  main -> main
```

4. **Create a new branch, modify the code on the branch and merge changes:**

   - **Creating and switching to branch rand:**

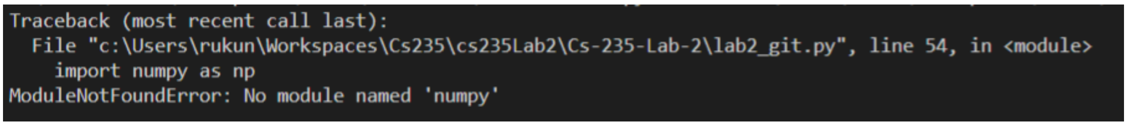


```
(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab1\helloworld>git branch rand

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab1\helloworld>git checkout rand
Switched to branch 'rand'

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab1\helloworld>git branch
  master
* rand
```

   - **I implemented the getRand() function in the rand branch:**

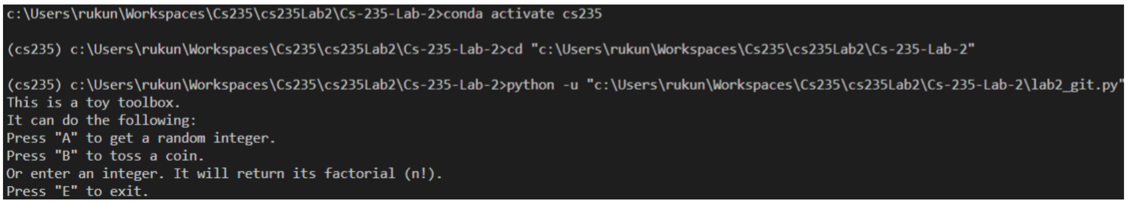
```
def get_rand():

my_randint = np.random.randint(0, 100)
return my_randint
```

     - Initally I wasn't able to run my code because the interpreter count find a module calle numpy.



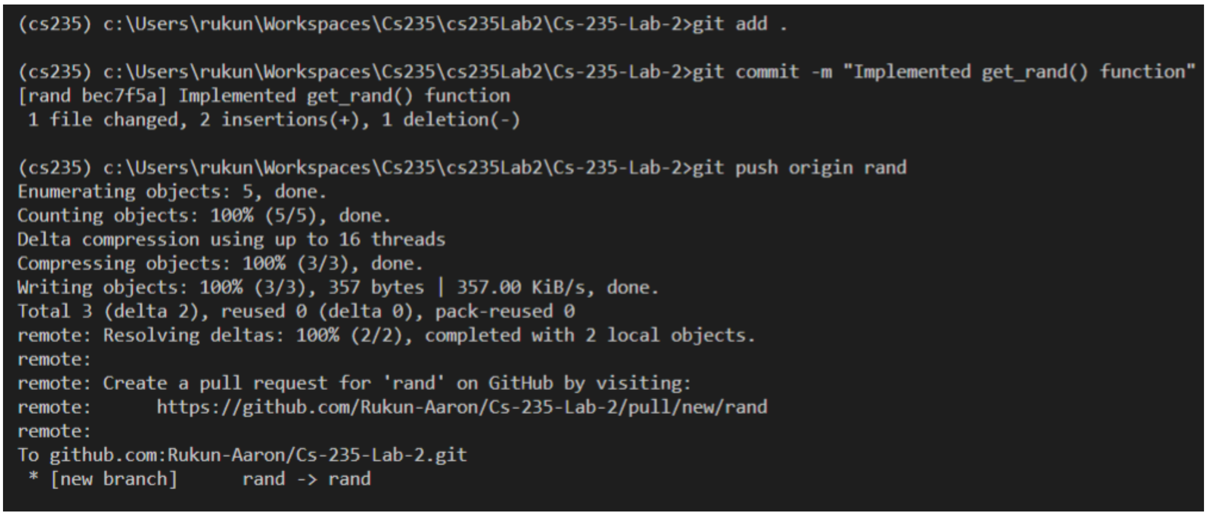
```
Traceback (most recent call last):
  File "c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2\lab2_git.py", line 54, in <module>
    import numpy as np
ModuleNotFoundError: No module named 'numpy'
```

- The reason for this is because I wasn't working in the virtual environement where I had previously downloaded numpy. So after activating the environment I was able to run my code:



```
c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>conda activate cs235

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>cd "c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2"

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>python -u "c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2\lab2_git.py"
This is a toy toolbox.
It can do the following:
Press "A" to get a random integer.
Press "B" to toss a coin.
Or enter an integer. It will return its factorial (n!).
Press "E" to exit.
```

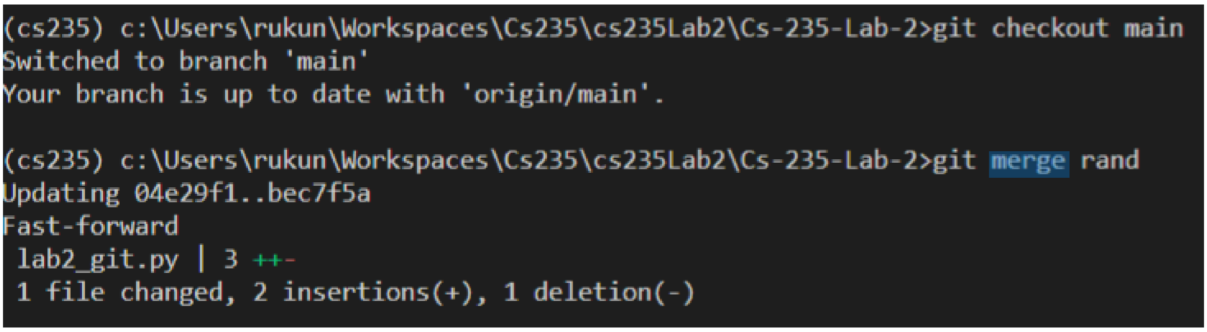
- **Then I staged, commit and pushed it to the remote repository**:



```
(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git add .

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git commit -m "Implemented get_rand() function"
[rand bec7f5a] Implemented get_rand() function
 1 file changed, 2 insertions(+), 1 deletion(-)

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git push origin rand
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 357 bytes | 357.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'rand' on GitHub by visiting:
remote:      https://github.com/Rukun-Aaron/Cs-235-Lab-2/pull/new/rand
remote:
To github.com:Rukun-Aaron/Cs-235-Lab-2.git
 * [new branch]      rand -> rand
```

- **Then I merged it with my main branch**

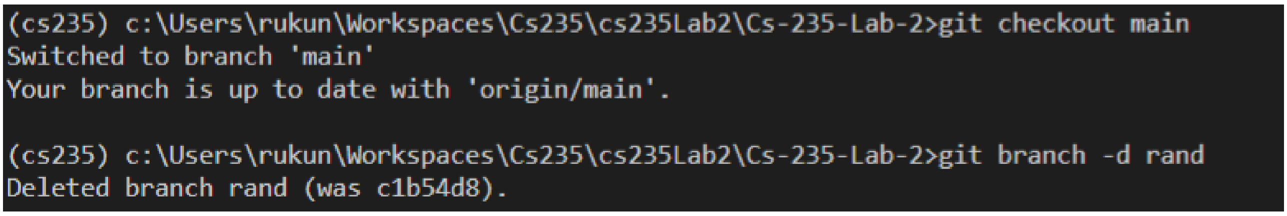
  I resolved a merge conflict I encountered but managed to resolve it and push my changes.



```
(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git merge rand
Updating 04e29f1..bec7f5a
Fast-forward
 lab2_git.py | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)
```

5. **Delete Branch**

   I delete the branch using:



```
(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

(cs235) c:\Users\rukun\Workspaces\Cs235\cs235Lab2\Cs-235-Lab-2>git branch -d rand
Deleted branch rand (was c1b54d8).
```

6. **Answer the following questions**

```
Question 1:

In a scenario, where you work with others as a team. You wrote some code
using the lab machine, but the code isn't working. You decide to continue
working on the code from home. What's the best way to commit your unfinished code
to GitHub without interrupt others? What commands do you need to achieve that?

Answer:
The best way to commit the unfinished code to GitHub is by creating and publishing
your changes to new temporary branch that is seperate from the main / branch
development was occuring on.

Question 2:

Explain the difference between merge and fork. Give an example for each use case.

Answer:

Merging is the way that Git & GitHub merges two seperate deveopment histories into
one. Ussually Usually when developing software, there are multiple branches that
team members work on to fix bugs, add new features etc; when one of these features
is ready to be added to the main branch, the feature branch and the main branch
are merged together. Merging is mostly done between branches, but can also be done
between repositories.


Forking is the act of "forking" a copy of someones repository onto your account,
this allows you to, after cloning it,  make any changes to the repository without
effecting the original repository. After making your changes, you can create a
pull request  to the owner to merge your changes with the original repo.
```