**Creating a workspace through Atom:**

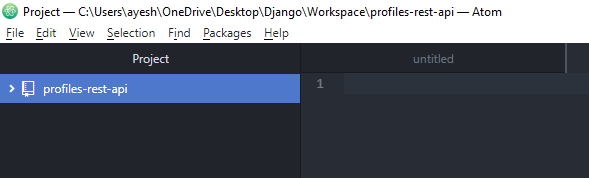
1. Create a folder “Workspace” and within it create another folder with name “profile-rest-api”

2. Go to Atom

- Go to Files

- Add Project Folder

- Search for the profile-rest-api folder and click open



3. Either go to the project folder “profiles-rest-api” and click on Git Bash here

Or

Go to Git Bash and perform the following commands

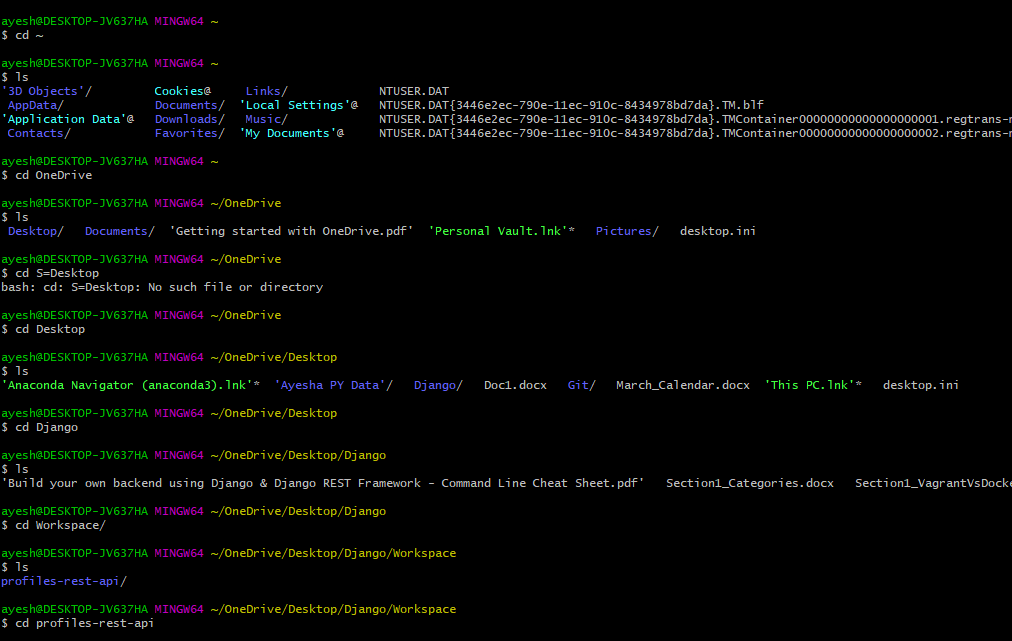
> cd ~

To check if you are in home directory

> ls

List all the folders in the current directory.

> cd directory\_name



> pwd

To check your current directory



**Creating a Git Project:**

In Git Bash

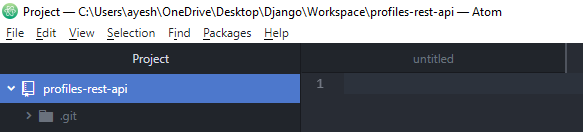
1. Run git init to initialize the git repository

(should be inside our project folder before running git init)



In Atom

2. Go to Atom and check that the project has been initialized as a .git repository



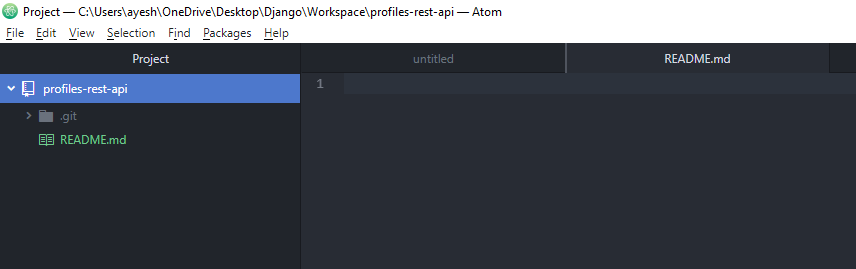
3. Add files to our git project

First file is a README.md file

Right click on profiles-rest-api folder and click on “New File”

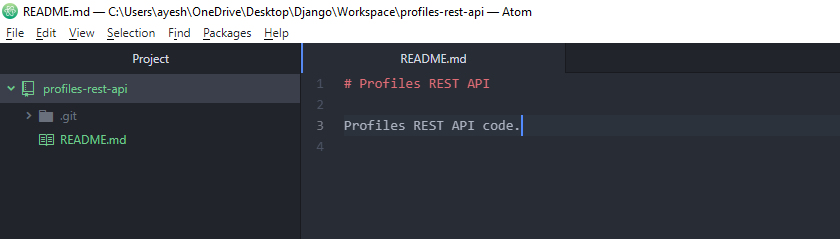
4. Provide the filename as README.md and click enter

It creates an empty readme file.



\*It is in green, because the files are not committed in git yet.

5. Go to the README.md file and add comments as below.



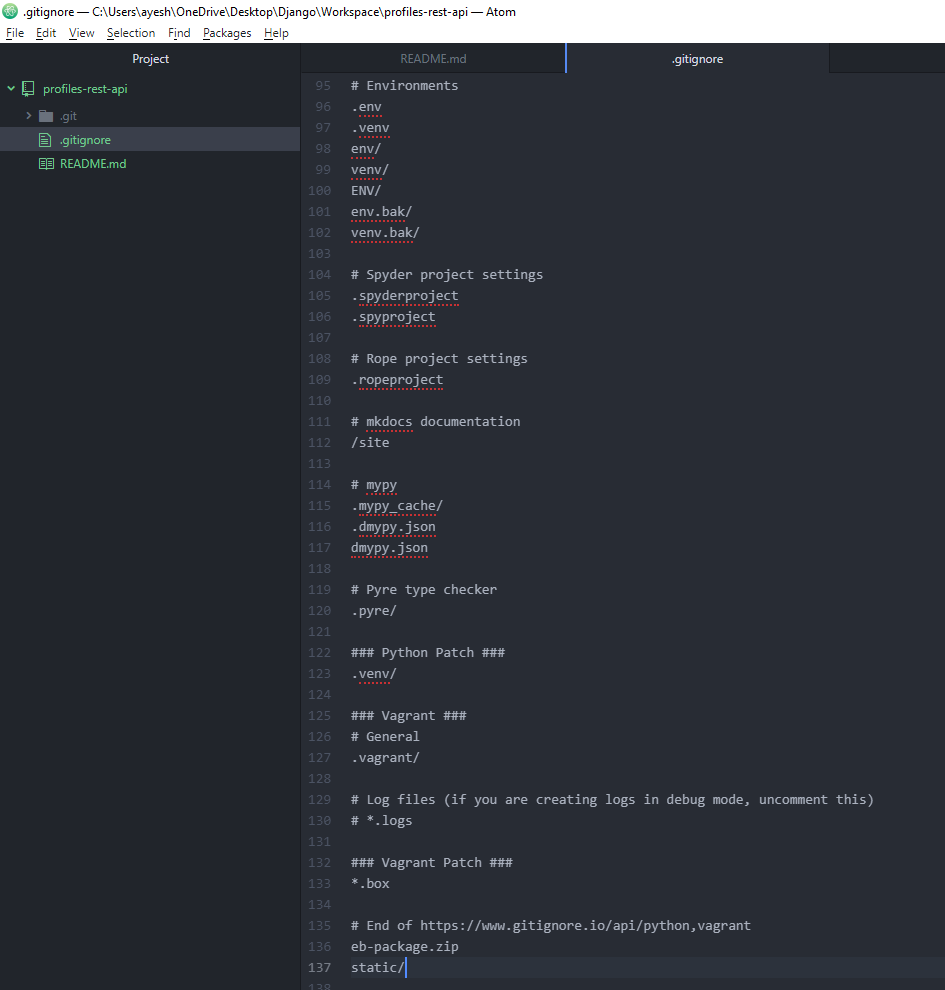
Save the changes.

Git Ignore File – A file which tells git not to include the specific files and directories to our git project when performing a commit.

6. Secondly, create a .gitignore file and include all the files to ignore.

Copy the details from the link below.

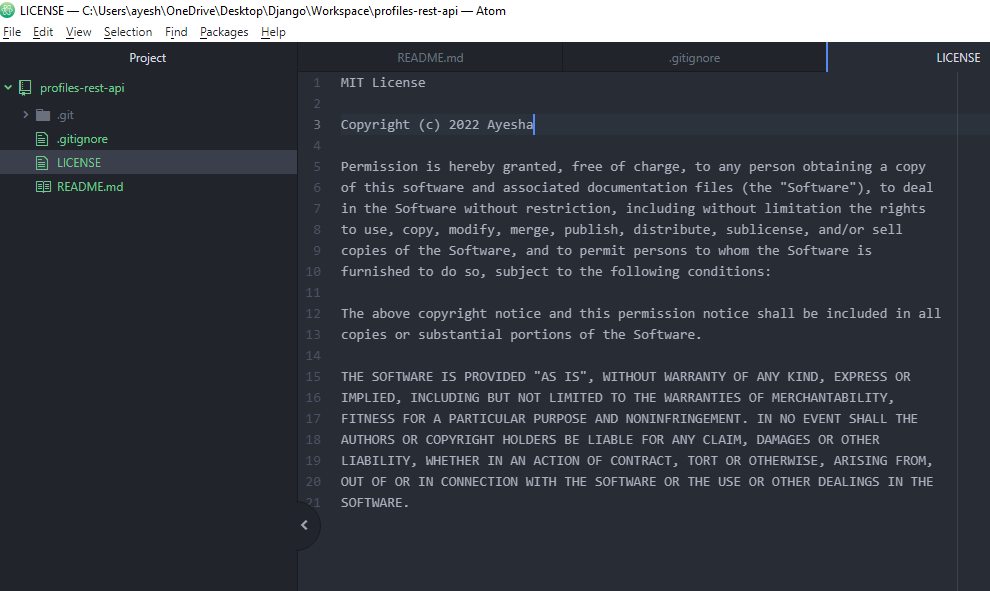
<https://gist.github.com/LondonAppDev/dd166e24f69db4404102161df02a63ff>



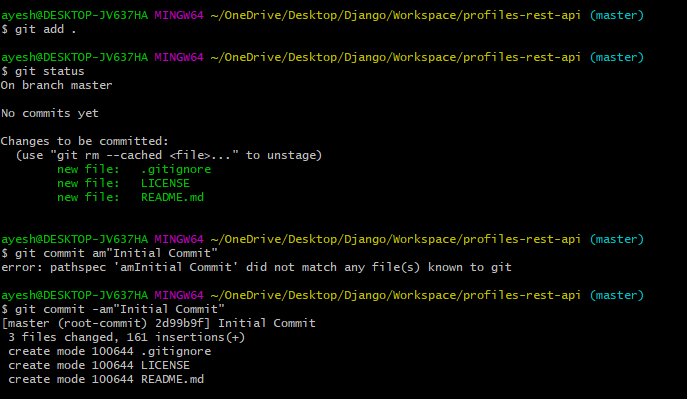
7. Create a third file named ‘LICENSE’ and include all the license details from the link below.

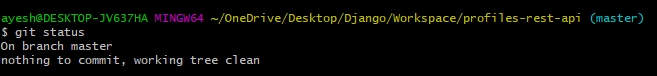
<https://choosealicense.com/licenses/mit/>

(Including MIT License details, since this course is a part of it.)



8. Commit the changes to git.





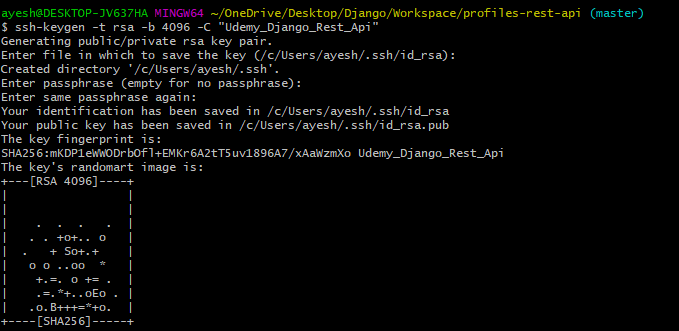
**Pushing to GitHub:**

1. In Git Hub, run command ls ~/.ssh

(to make sure we don’t have any existing keys in our ssh directory)

By default ssh directory is where all the private public keys are created in the system.

2. Create a public private key pair



\*filename – don’t enter anything, leave it as default

3. Run command ls –al ~/.ssh



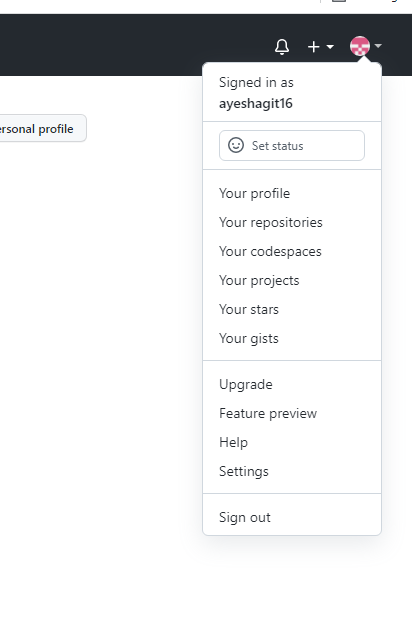
\*id\_rsa (private key- never share)

\*id\_rsa.pub (public key – can upload to github in order to authenticate)

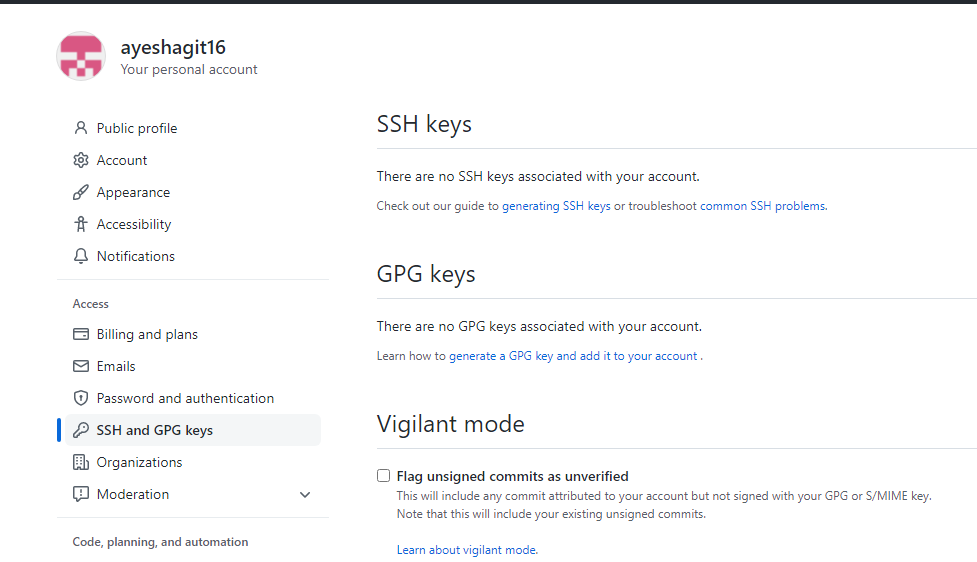
4. We need to add the public rsa key to our github account.

github.com

a. Go to your github account -> Click on Settings

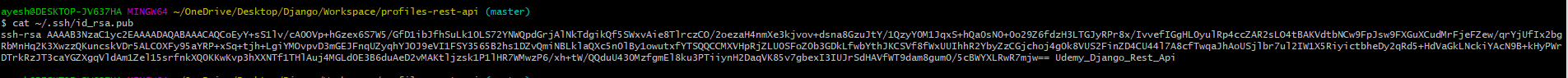


b. Click on SSH and GPG keys.



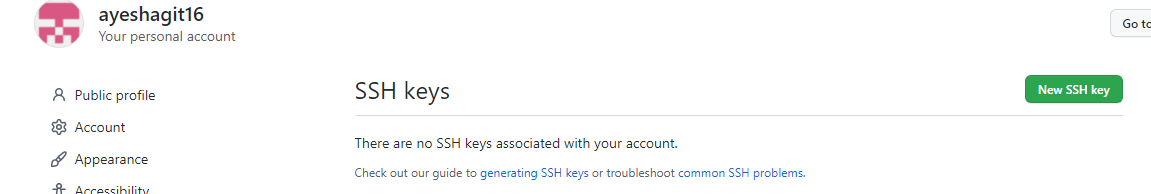
c. In Git Bash

Output the public key using command cat/.ssh/id\_rsa.pub



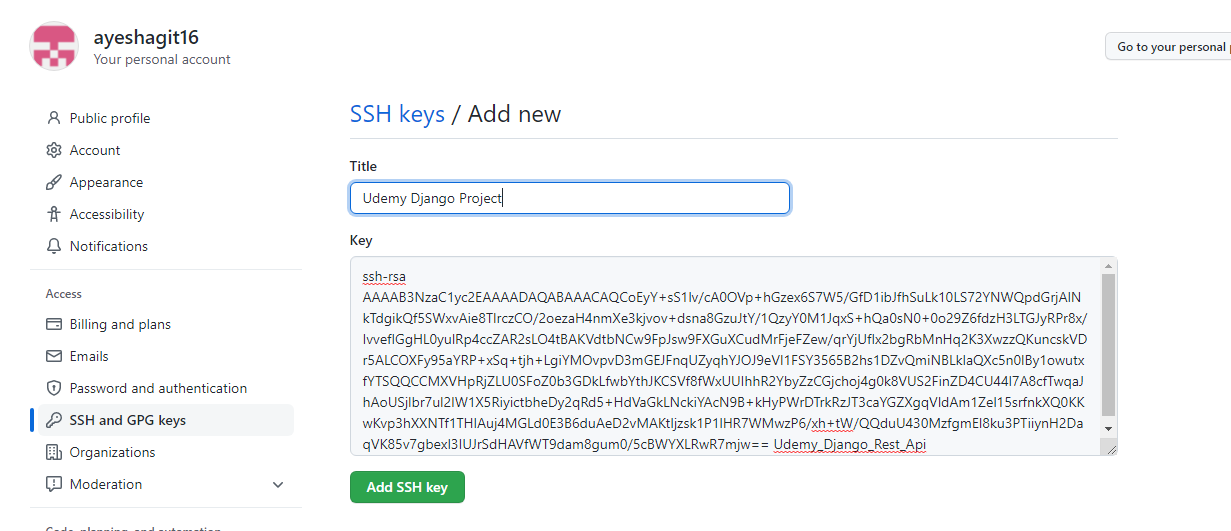
d. Copy the whole key and go to your git hub account

Click on New SSH key

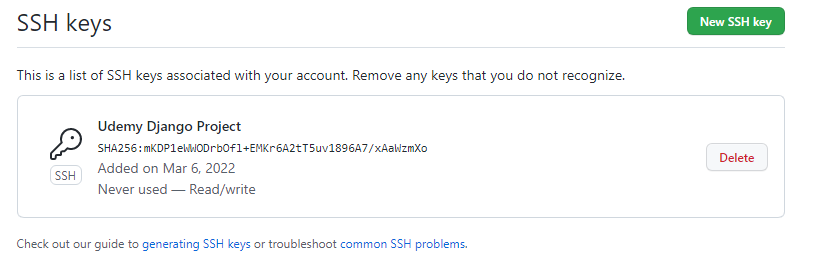


e. Paste the whole key and provide the Title.

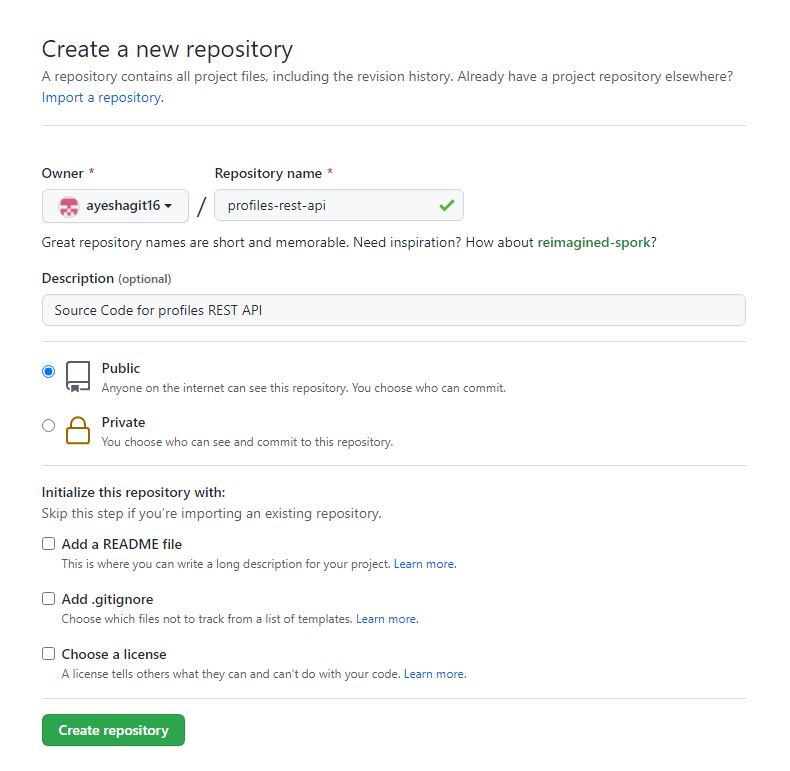
Click on Add SSH key



f. You can see the key in the list.



5. Create a new repository in git hub. Provide the repository name and description.

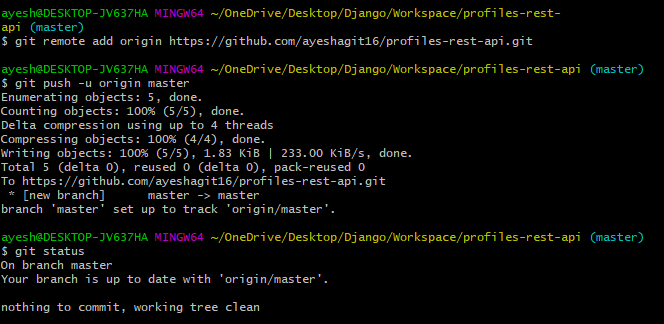


6. Push an existing repository from command line

Run command

a. git remote add origin https://github.com/ayeshagit16/profiles-rest-api.git

b. git push -u origin master



7. Go to your GitHub account and refresh.

