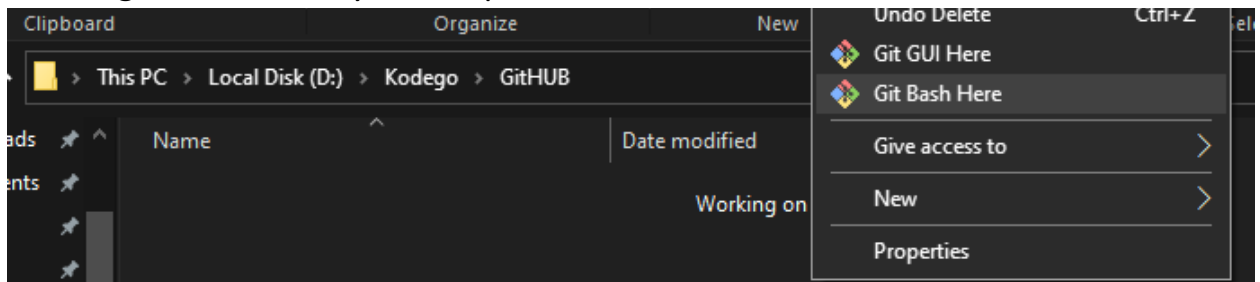


SSH Key Observation

1. Creating a New SSH Key. 1st is open Gitbash Terminal.



2. Use this code below.

```
$ ssh-keygen -t ed25519 -C "your_email@example.com"
```

This is the result you should have.

```
VaN@WIN-K6GKF4N49I2 MINGW64 /d/Kodego/GitHUB (master)
$ ssh-keygen -t ed25519 -C "jestceroma88@gmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/Administrator/.ssh/id_ed25519):
Created directory '/c/Users/Administrator/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/Administrator/.ssh/id_ed25519
Your public key has been saved in /c/Users/Administrator/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:qDMuu7sYzjQUn3cf1OfGRpAin2DA4xi8qDzrpX1Z7Xo jestceroma88@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
| . . . . . |
| o o + . . o |
| ..= o + oo o |
| .oo.. +. = |
| o. o ...S. = |
| oo .....+ |
| .oo.+ o .. |
| ++*. = E |
| o==Bo .o |
+----[SHA256]-----+
VaN@WIN-K6GKF4N49I2 MINGW64 /d/Kodego/GitHUB (master)
$
```

- Now that you need to know if the ssh-agent that you made is running
Open a New Gitbash then put this code: `$ eval "$(ssh-agent -s)"`

Result:

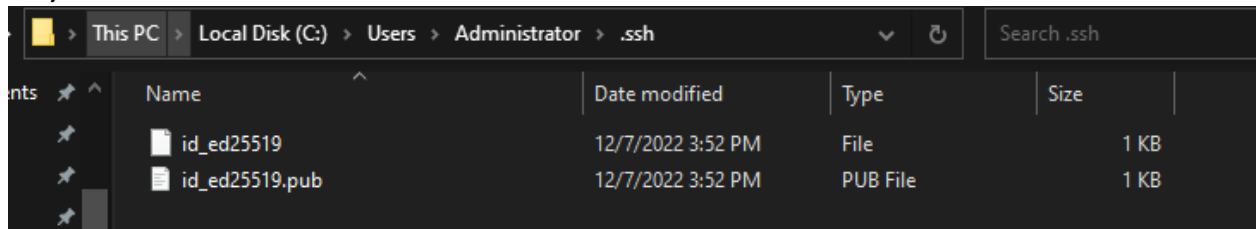
```
VaNeWIN-K6GKF4N49I2 MINGW64 /d/Kodego/GitHUB (master)
$ eval "$(ssh-agent -s)"
Agent pid 2101
```

- Now adding your created Key to you Ssh Agent put this code:
`$ ssh-add ~/.ssh/id_ed25519`

Result:

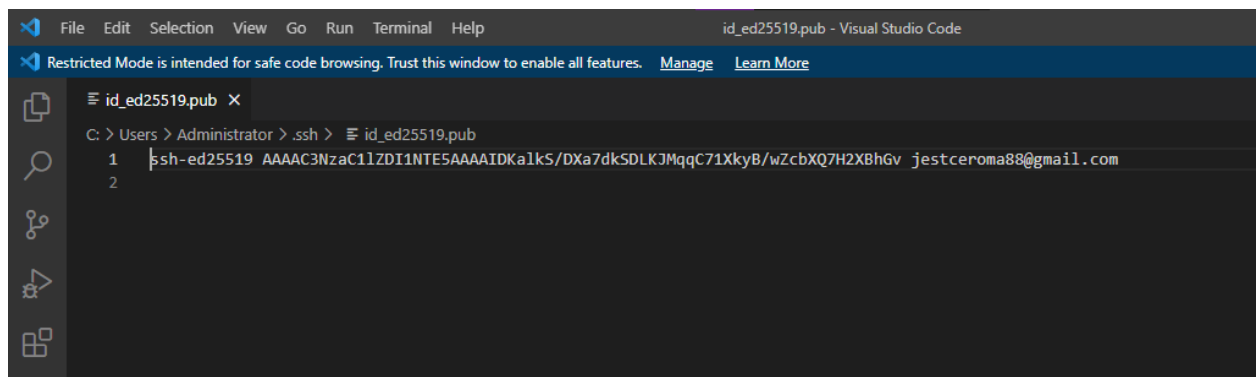
```
VaNeWIN-K6GKF4N49I2 MINGW64 /d/Kodego/GitHUB (master)
$ ssh-add ~/.ssh/id_ed25519
Identity added: /c/Users/Administrator/.ssh/id_ed25519 (jestceroma88@gmail.com)
```

- Now were Ready to add our SSh key to our github first go to where your Key is Located.

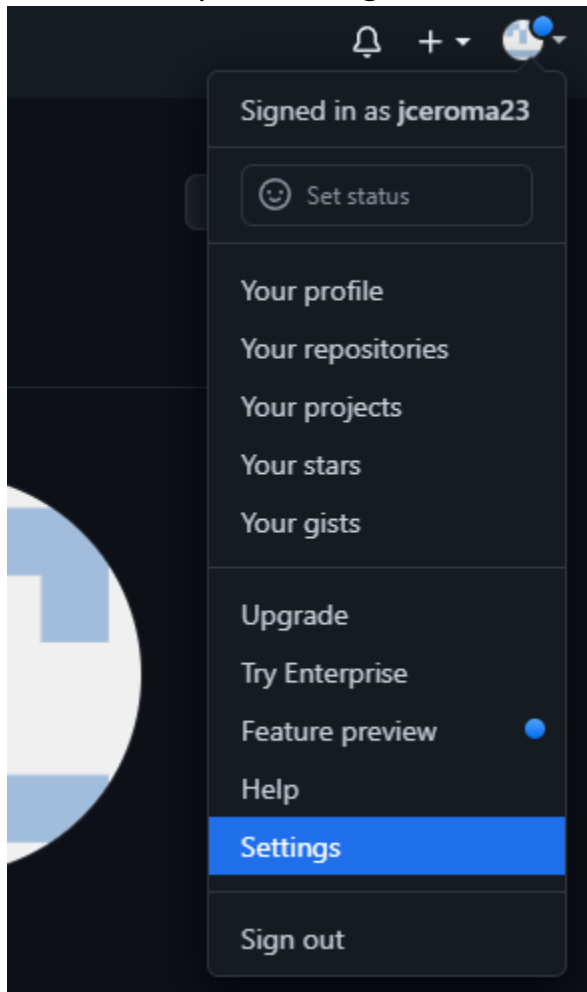


- Open File.Pub using VS.Code

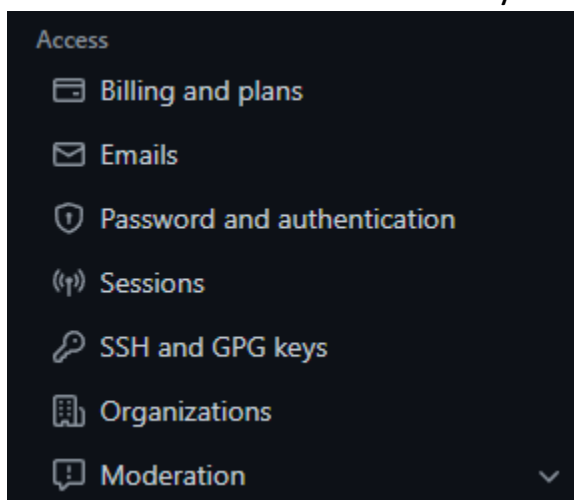
Result:



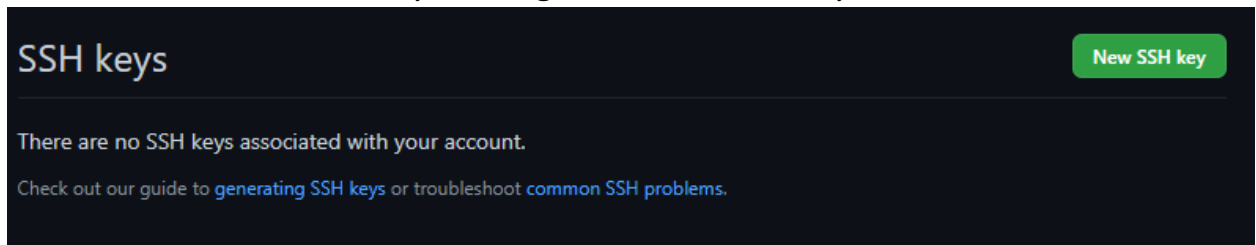
7. Copy it and paste it to your GitHub Account
Go to your Settings



Next is Look for SSH and GPG Keys



8. Now from SSH and GPG keys settings click New SSH Key



9. Add a Title for your Key

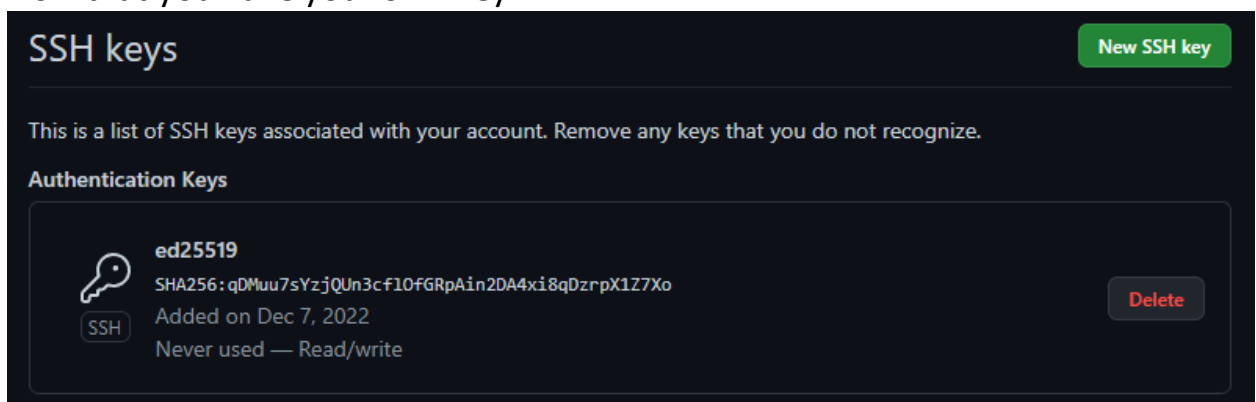
Paste the SSH Key to the text box for Keys

For key Type must be Authentication Key

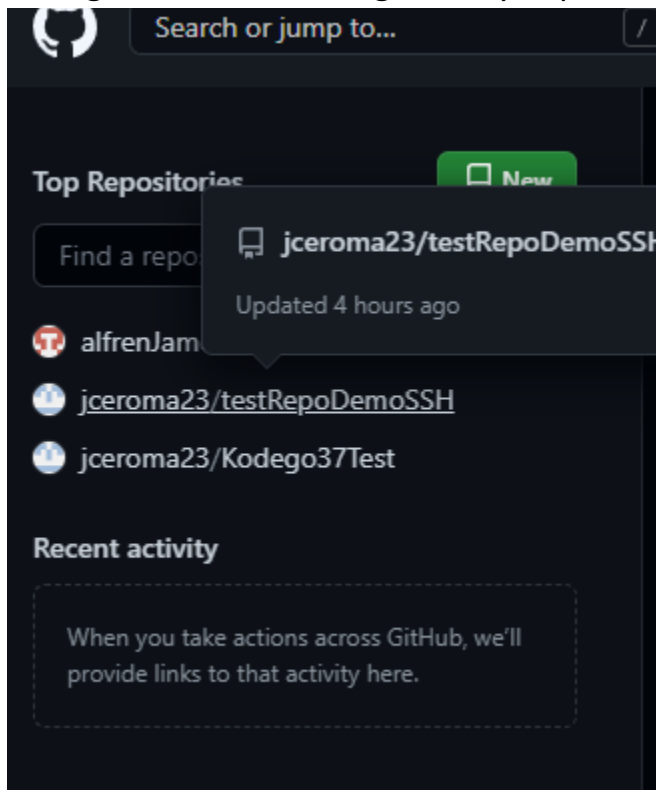
A screenshot of the 'SSH keys / Add new' form. It has a title field, a 'Key type' dropdown menu set to 'Authentication Key', and a large text area for the 'Key'. Below the text area is a green button labeled 'Add SSH key'. The key text area contains a hint: 'Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com''.

Then Add SSH Key

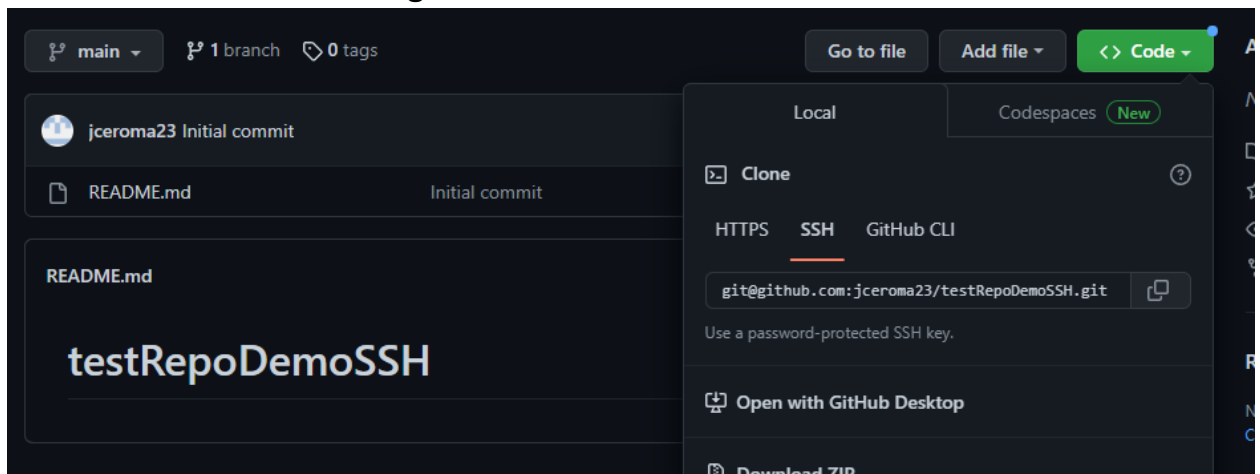
10. Now that you have your own key



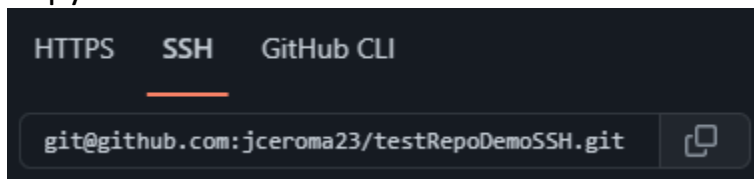
11. Now go back home and go to any Repository



12. From there click code and go to SSH tab



13. Copy The SSH



14. Then we're checking now the Connection of our SSH Connection. Open new Git Bash and enter this code and paste the SSH that you copied.

\$ Git clone git@github.com:jceroma23/testRepoDemoSSH.git

```
VaN@WIN-K6GKF4N49I2 MINGW64 ~/.ssh
$ git clone git@github.com:jceroma23/testRepoDemoSSH.git
Cloning into 'testRepoDemoSSH'...
The authenticity of host 'github.com (20.205.243.166)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

VaN@WIN-K6GKF4N49I2 MINGW64 ~/.ssh
```

Finish