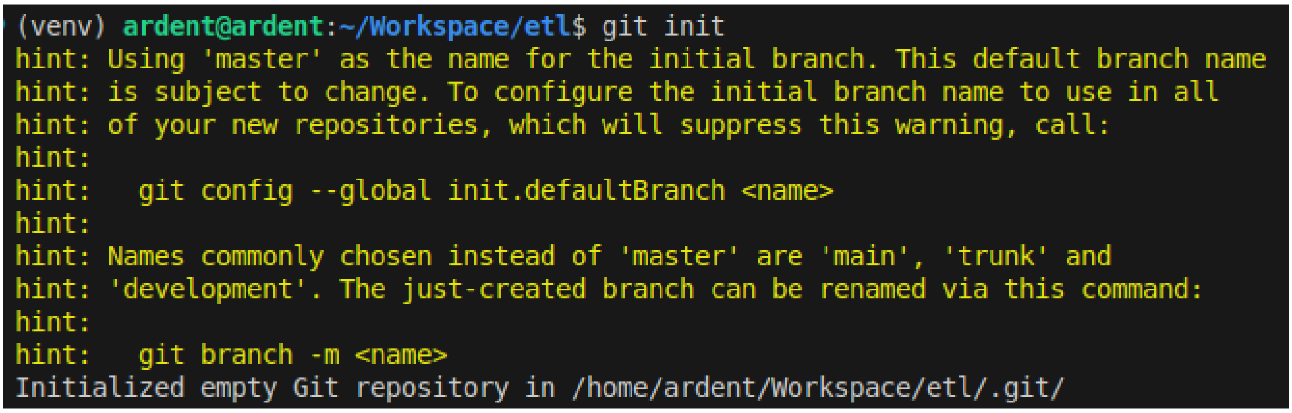
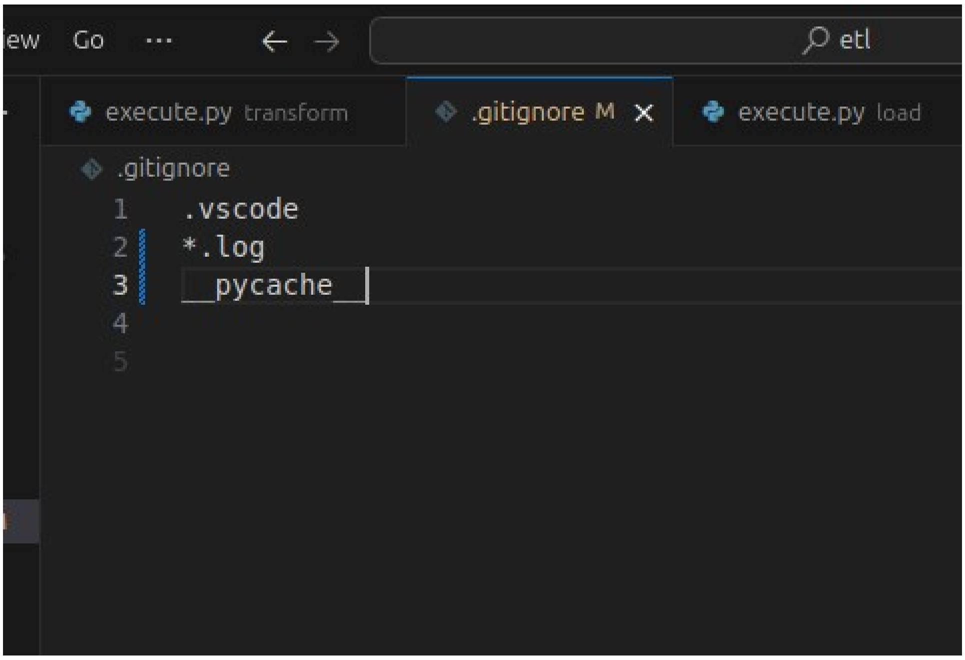
1. Initialize git repository in your working directory



```
(venv) ardent@ardent:~/Workspace/etl$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ardent/Workspace/etl/.git/
```

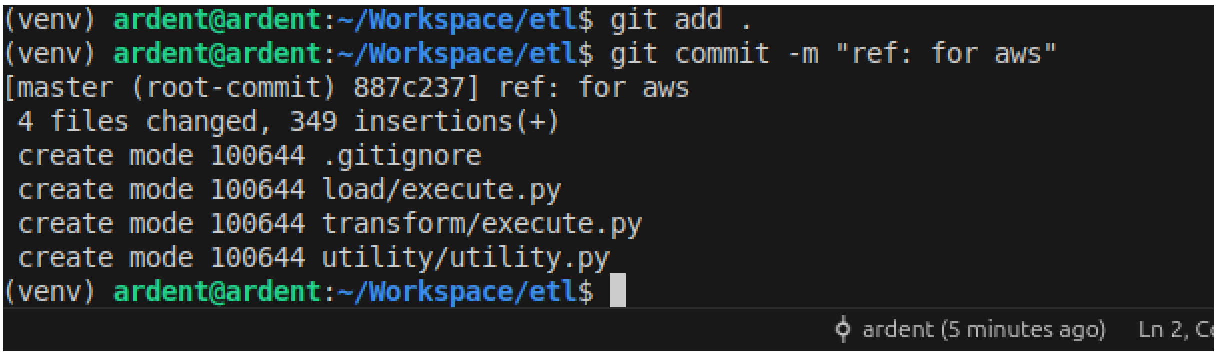
2. Create a .gitignore file and add the files and folders you do not want to track/push



```
.vscode
*.log
__pycache__
```

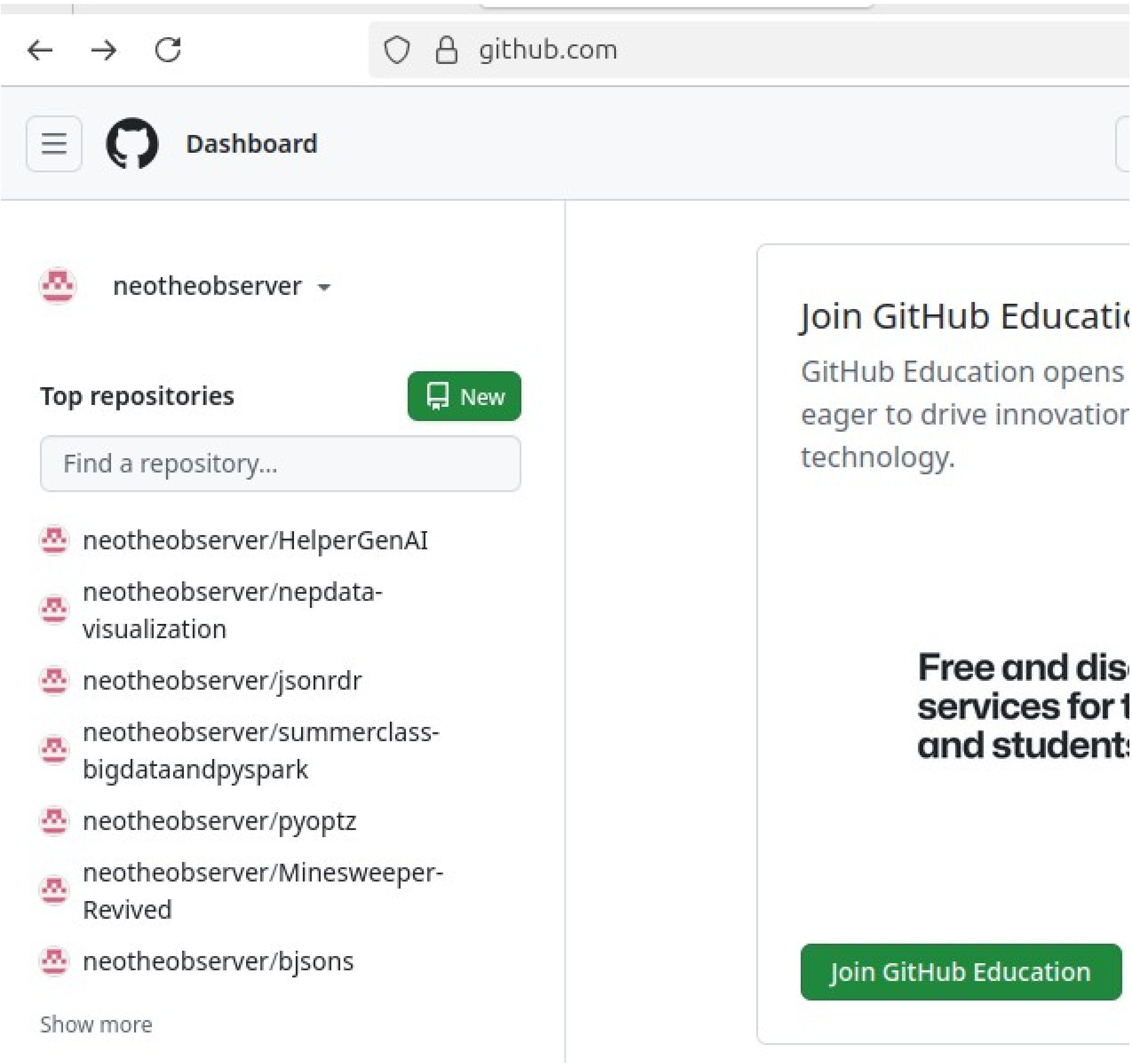
I have only added .vscode and all the files with .log extension for now as I want to push/track all others except for the launch.json file which contains my passwords and configs.

3. Then start tracking the file using git add and commit the changes using git commit. Use . With add command to include all files and use -m flag with commit command to write as message for that commit. Follow the standard for writing messages. Do not copy mine. Do your own



```
(venv) ardent@ardent:~/Workspace/etl$ git add .
(venv) ardent@ardent:~/Workspace/etl$ git commit -m "ref: for aws"
[master (root-commit) 887c237] ref: for aws
 4 files changed, 349 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 load/execute.py
 create mode 100644 transform/execute.py
 create mode 100644 utility/utility.py
(venv) ardent@ardent:~/Workspace/etl$ 
```

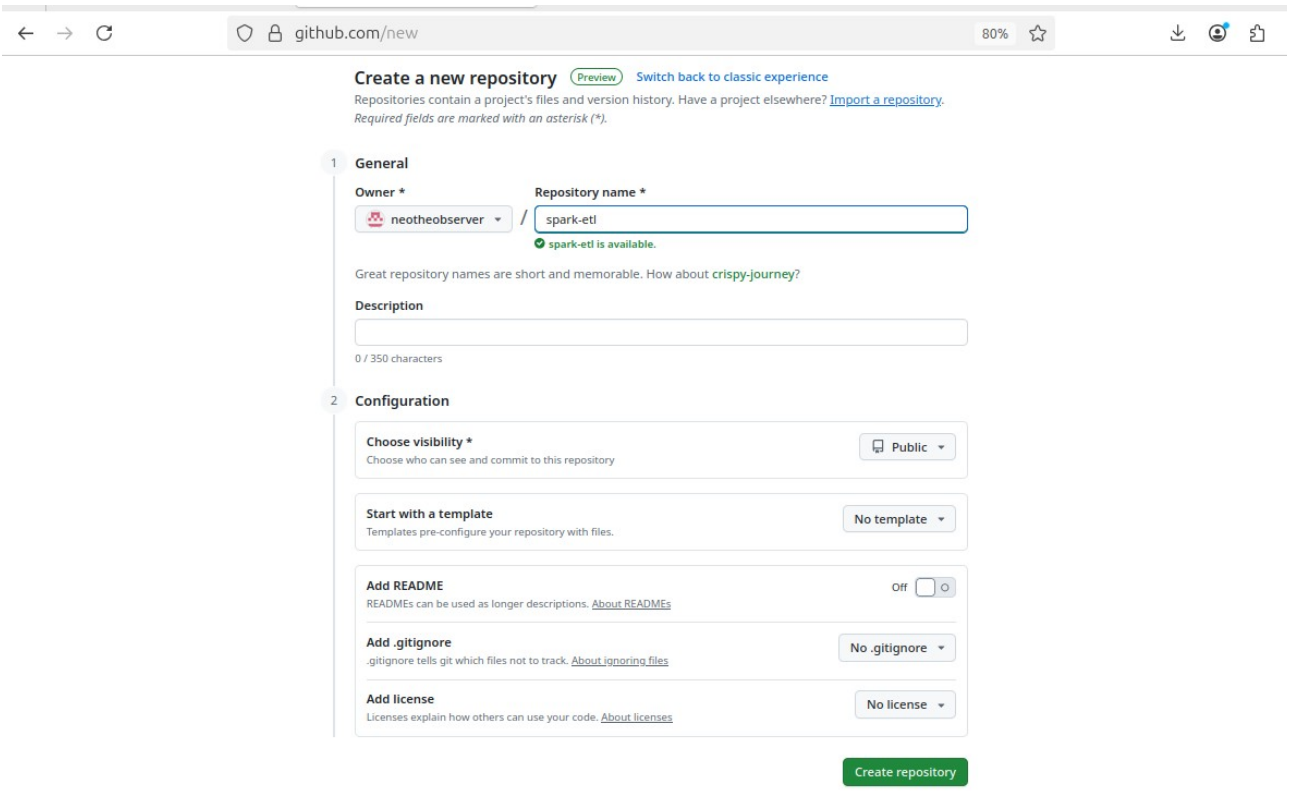
4. Login to Github.com and Go to your dashboard



5. Click the New Button to create new repository

Give a name to the repository. Leave everything else to default unless you know what you are doing. Make sure the visibility is set to public if you want others to see this repo.

Then click on Create repository



Create a new repository (Preview) Switch back to classic experience

Repositories contain a project's files and version history. Have a project elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

1 General

Owner * neotheobserver / Repository name * spark-etl

spark-etl is available.

Great repository names are short and memorable. How about crispy-journey?

Description

0 / 350 characters

2 Configuration

Choose visibility * — Public

Choose who can see and commit to this repository

Start with a template — No template

Templates pre-configure your repository with files.

Add README — Off

READMEs can be used as longer descriptions. About READMEs

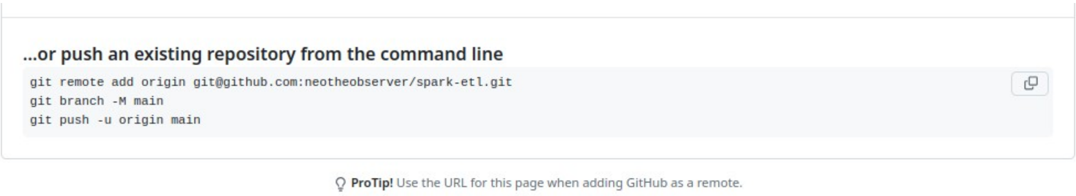
Add .gitignore — No .gitignore

.gitignore tells git which files not to track. About ignoring files

Add license — No license

Licenses explain how others can use your code. About licenses

6. In the next page follow the instruction at the below. To push an existing repository from command line as this is our case. We have already created added and committed an local repo.



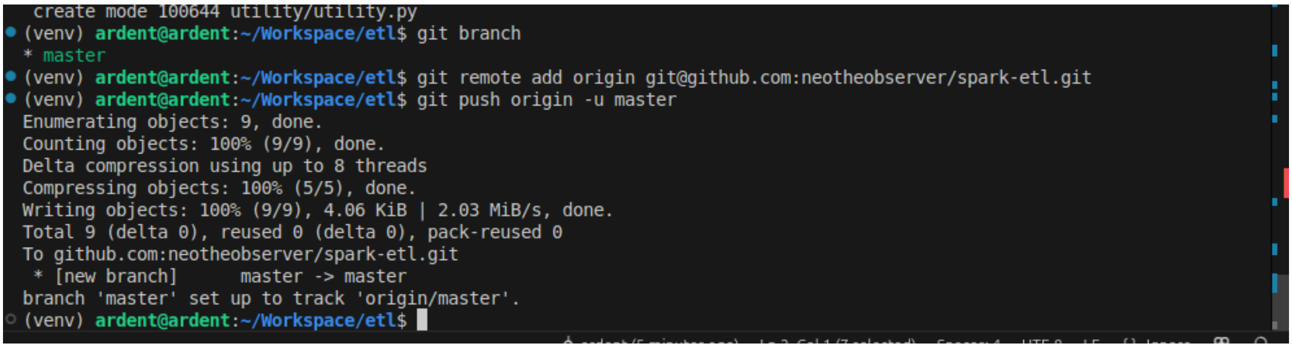
...or push an existing repository from the command line

```
git remote add origin git@github.com:neotheobserver/spark-etl.git
git branch -M main
git push -u origin main
```

ProTip! Use the URL for this page when adding GitHub as a remote.

Make sure you use your own url. Do not copy mine. Also in our case we will be using branch master instead of main so no need to execute the second command in the third command write master instead of main. As that is what was created by default

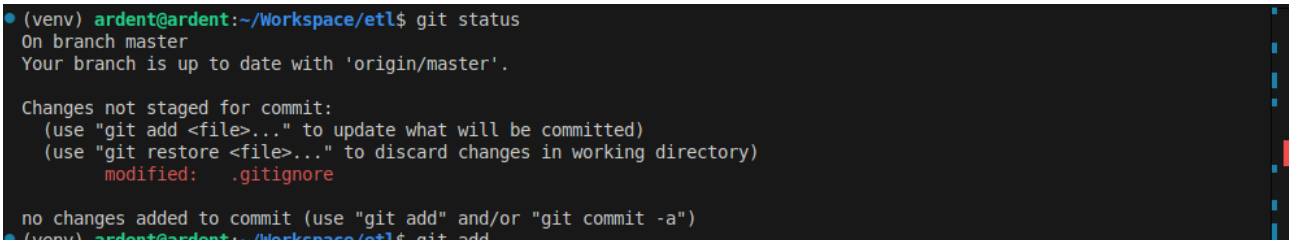
Use below image as reference but use your own unique url



```
 create mode 100644 utility/utility.py
(venv) ardent@ardent:~/Workspace/etl$ git branch
* master
(venv) ardent@ardent:~/Workspace/etl$ git remote add origin git@github.com:neotheobserver/spark-etl.git
(venv) ardent@ardent:~/Workspace/etl$ git push origin -u master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (9/9), 4.06 KiB | 2.03 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:neotheobserver/spark-etl.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
(venv) ardent@ardent:~/Workspace/etl$
```

Congratulation now you have synced your local repository with Github/remote repository. Any time you make change make sure you add commit and then push

First view the changes using git status

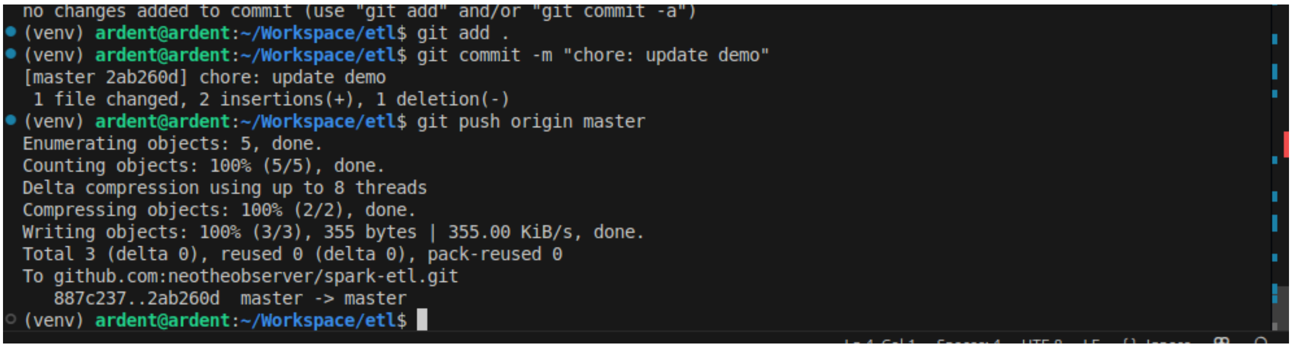


```
(venv) ardent@ardent:~/Workspace/etl$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   .gitignore

no changes added to commit (use "git add" and/or "git commit -a")
(venv) ardent@ardent:~/Workspace/etl$ git add
```

Update them if required or add to .gitignore if you don't want something that has been added or changed to be reflected in remote. Finally use add, commit and push to update the repo



```
no changes added to commit (use "git add" and/or "git commit -a")
(venv) ardent@ardent:~/Workspace/etl$ git add .
(venv) ardent@ardent:~/Workspace/etl$ git commit -m "chore: update demo"
[master 2ab260d] chore: update demo
 1 file changed, 2 insertions(+), 1 deletion(-)
(venv) ardent@ardent:~/Workspace/etl$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 355 bytes | 355.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:neotheobserver/spark-etl.git
   887c237..2ab260d  master -> master
(venv) ardent@ardent:~/Workspace/etl$
```

This is the most basics of git commands there are many more things to be done but this should suffice to get started and working with git.