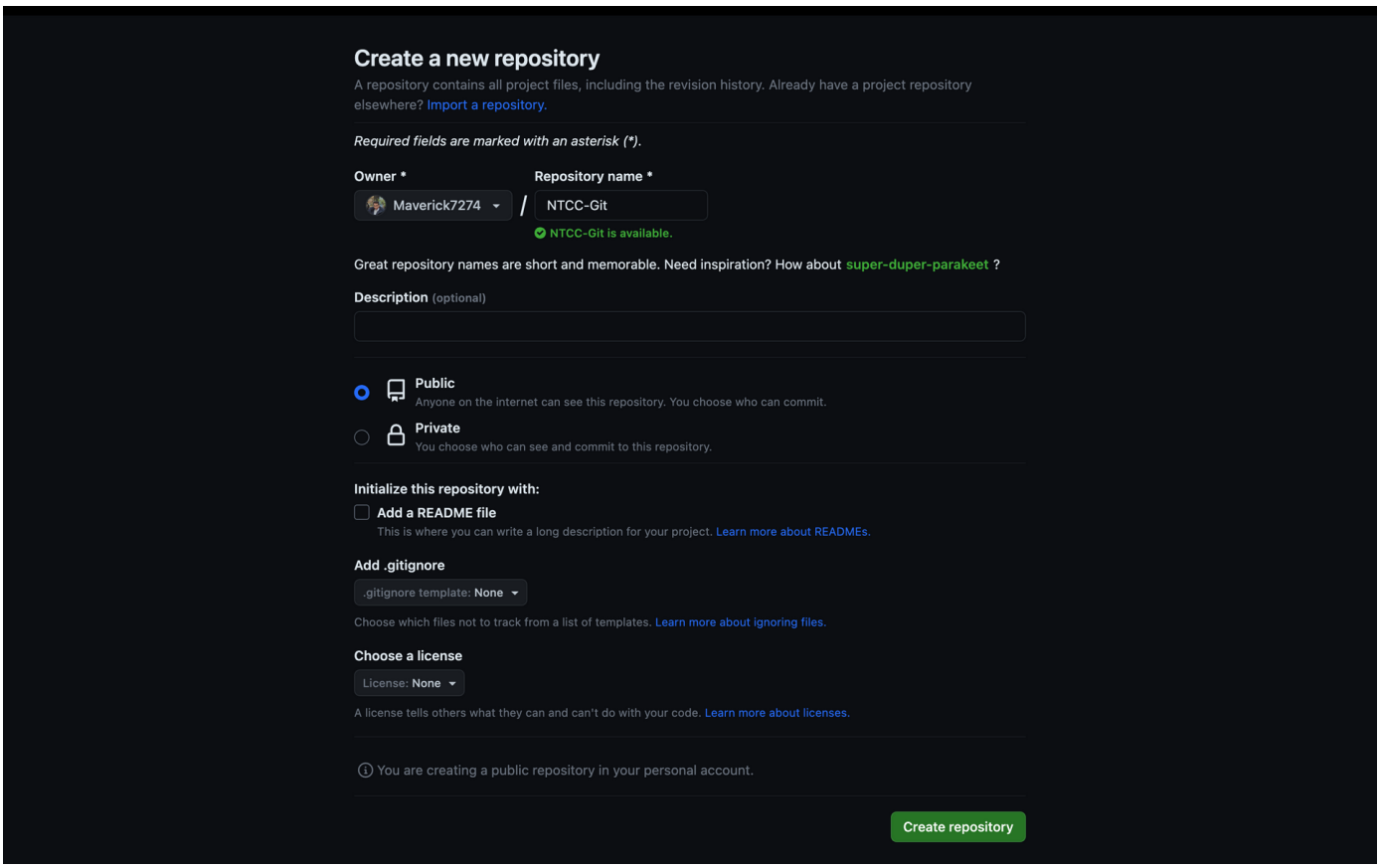
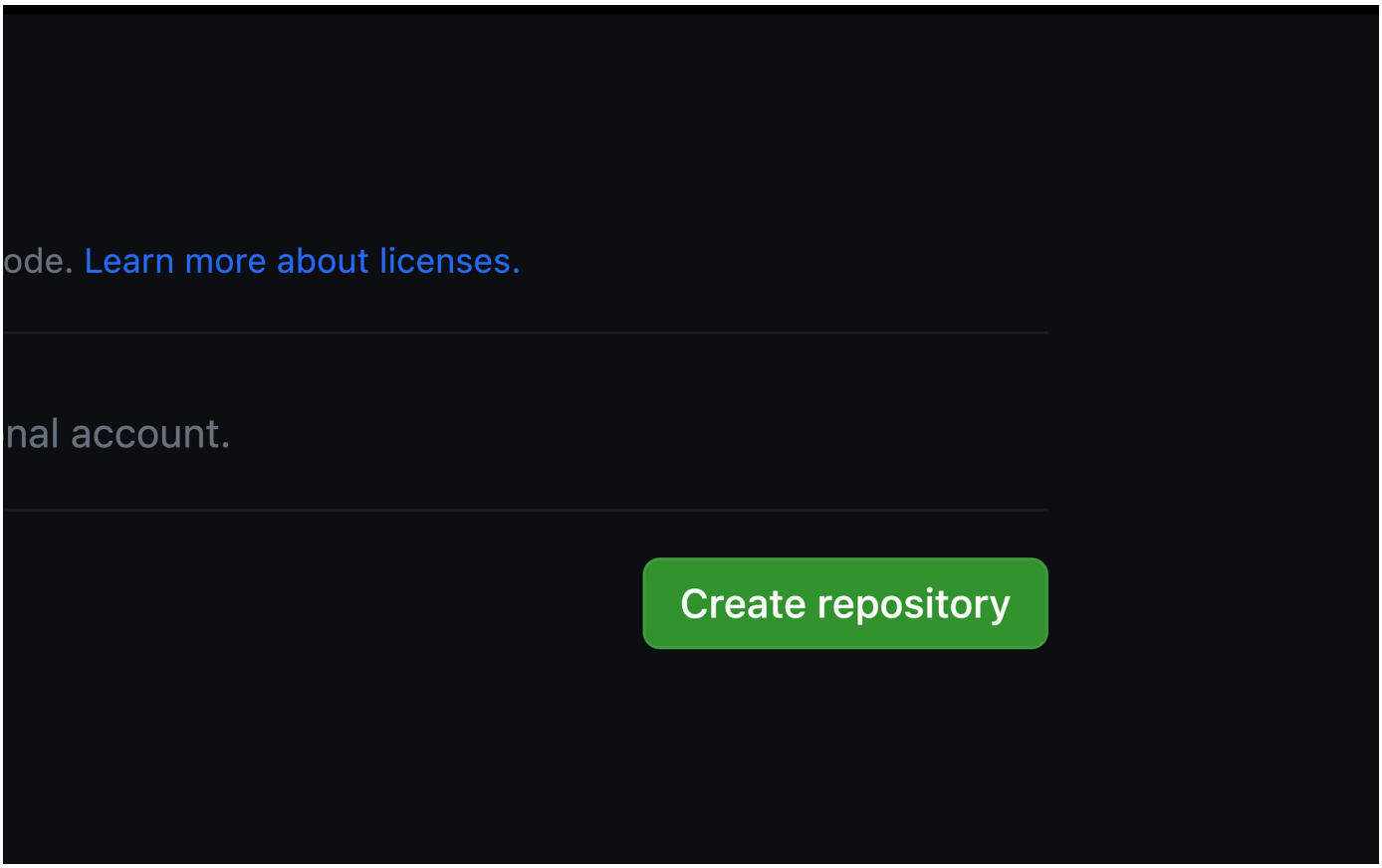
Practical Implementation of Git with GitHub

To demonstrate the practical use of Git in GitHub, assuming you already have a GitHub account, let's walk through the process of creating a new repository, making changes to a file, and pushing those changes to GitHub.

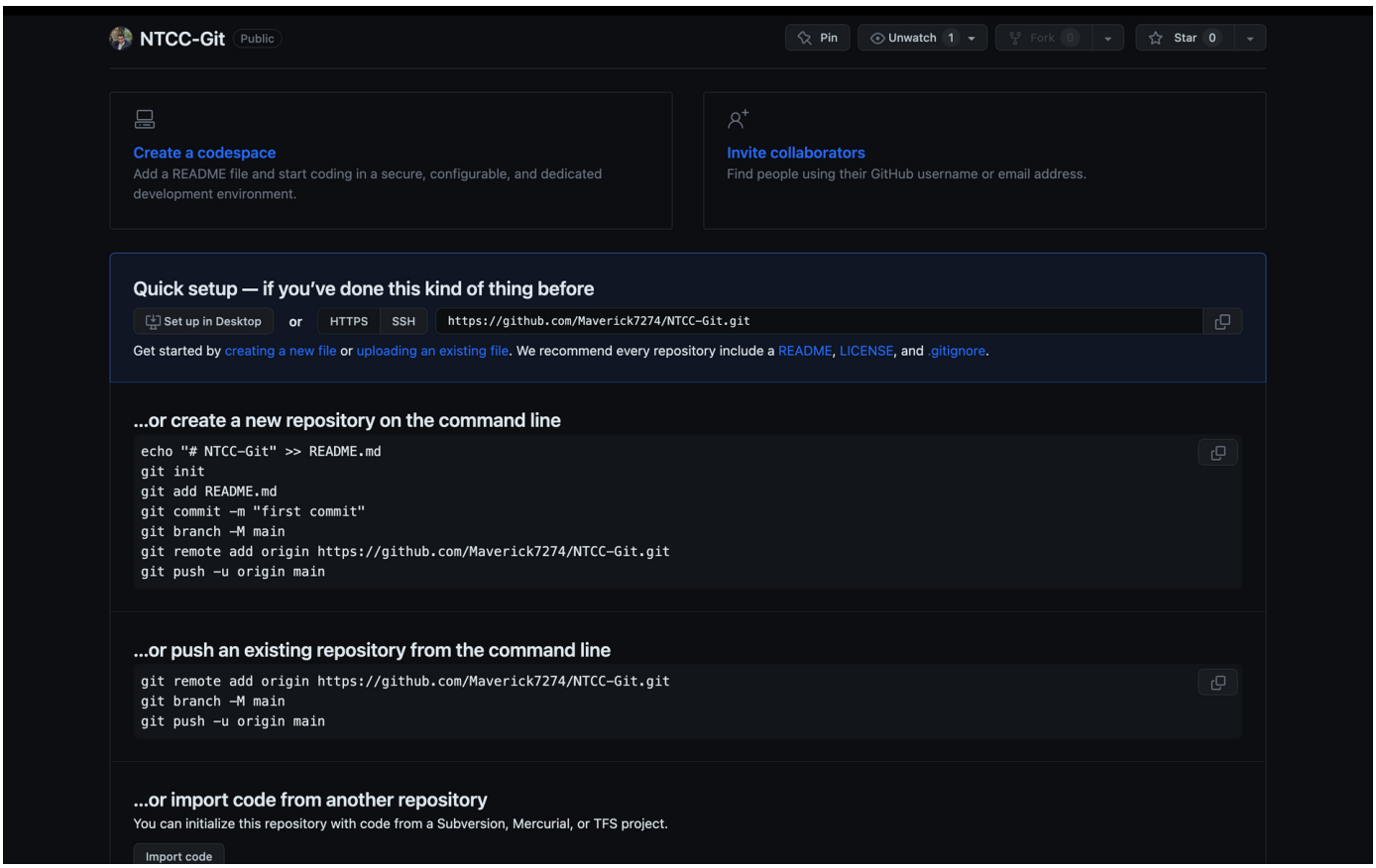
- First, log in to your GitHub account then use this URL : https://github.com/new to create a new repository. Give it a name, add an optional description, and choose the visibility (public or private).



- Next, click the green button to create the repository.



- Now, you'll see a page with instructions for creating a new repository on the command line.

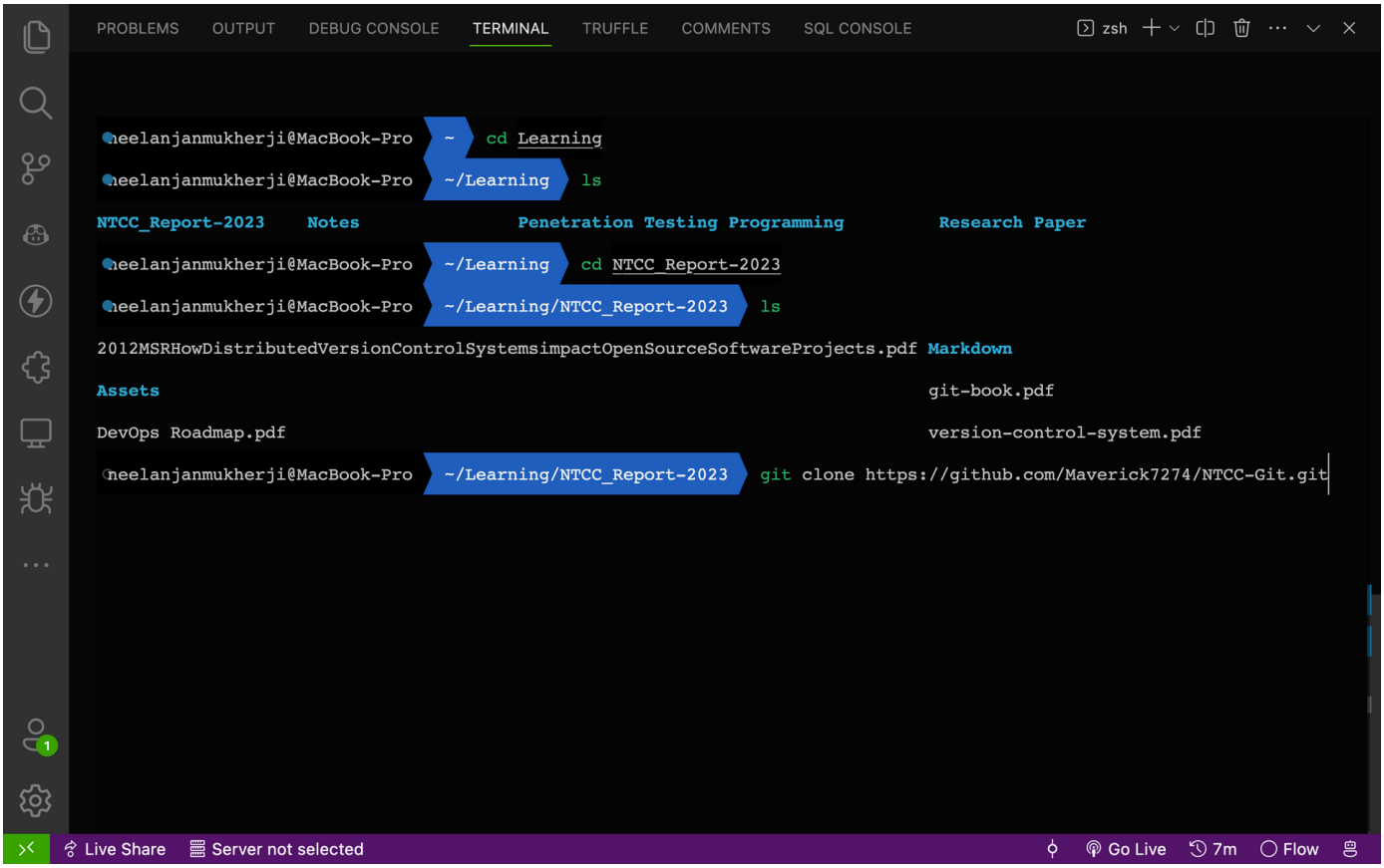


> NOTE : In these next step we will clone the repository to our local machine. If you are using a UNIX based system, you can use the terminal to clone the repository. If you are using a Windows system, you can use the Git Bash terminal to clone the repository.

There are two ways to clone the repository to your local machine.

Using `git clone` command

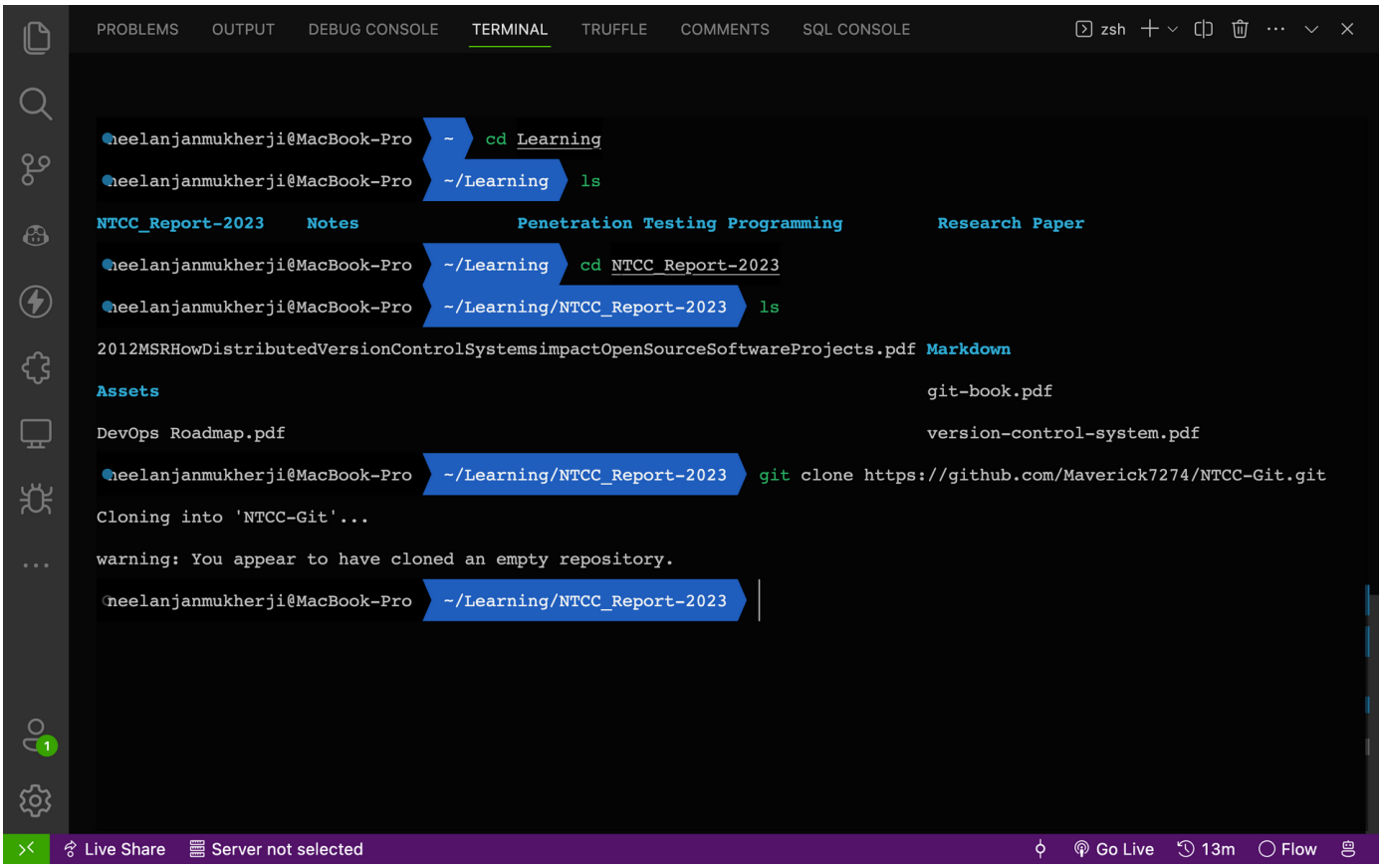
- Open a terminal window and navigate to the directory where you want to create the repository.



- Here for demonstration we are using a UNIX based system. I would recommend using a UNIX based system for almost every professional server runs in a Linux or a UNIX based Operating System.
- To clone the repository, copy the URL from the Quick setup box, then use the `git clone` command with the copied URL.

```
git clone https://github.com/Maverick7274/NTCC-Git.git
```

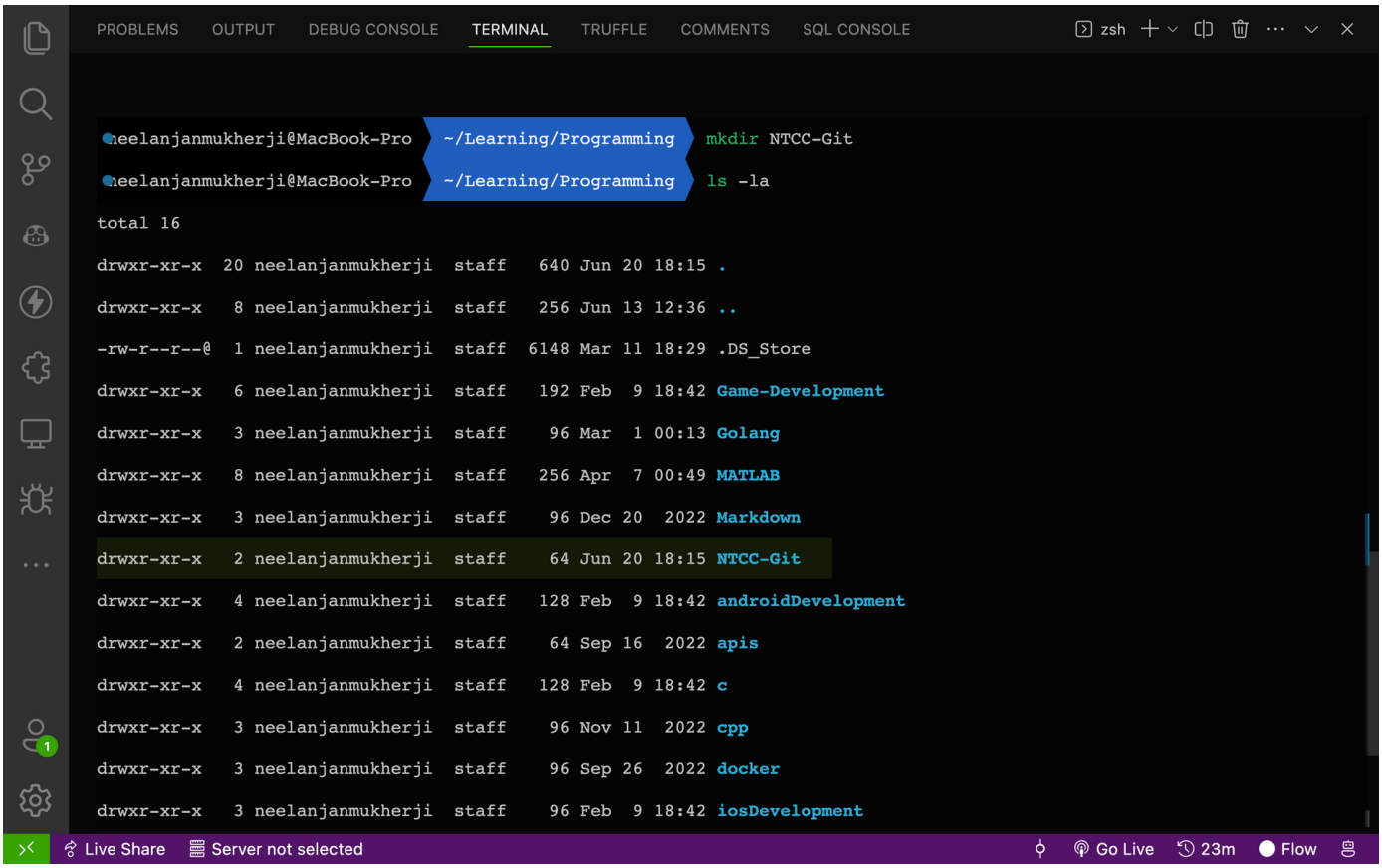
- Now, you have a local copy of the repository on your machine.




```
Cloning into 'NTCC-Git'...
warning: You appear to have cloned an empty repository.
```

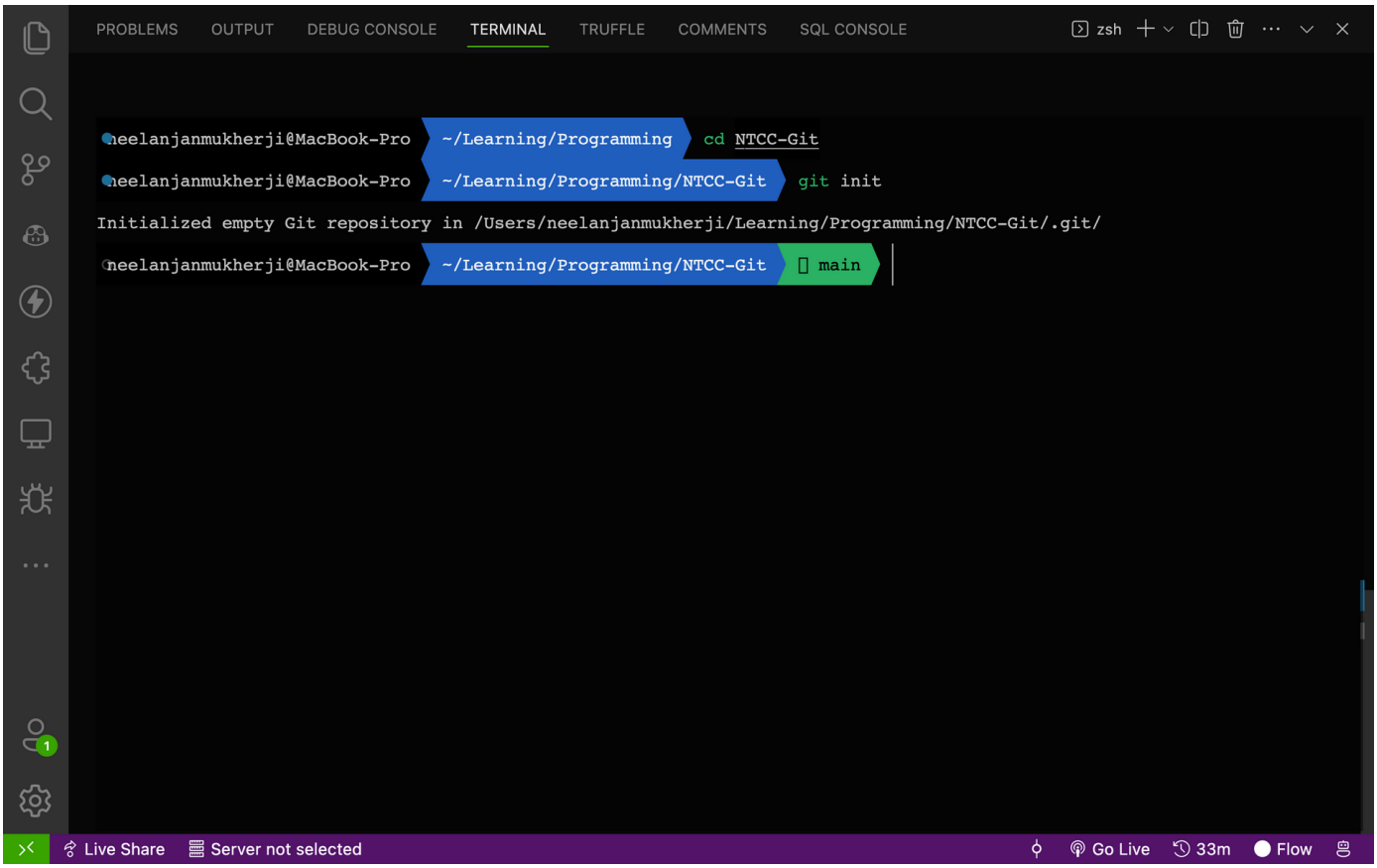
Using `remote add` command

- Open a terminal window and navigate to the directory where you want to create the repository.
- use `mkdir` command to create a new directory.




```
mkdir NTCC-Git && cd NTCC-Git
```

- Now, use `git init` command to initialize the repository.

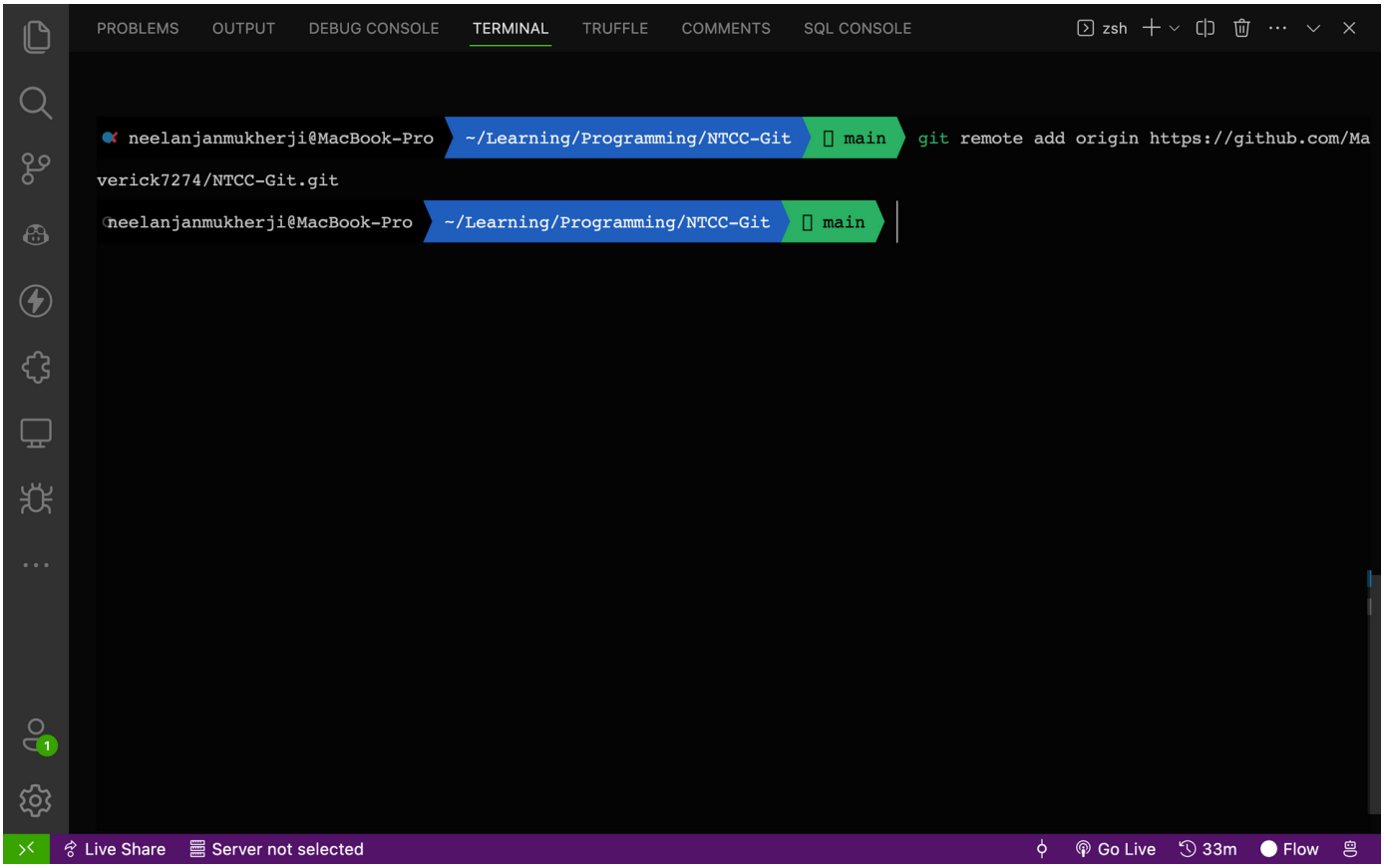



```
git init
```

Output :

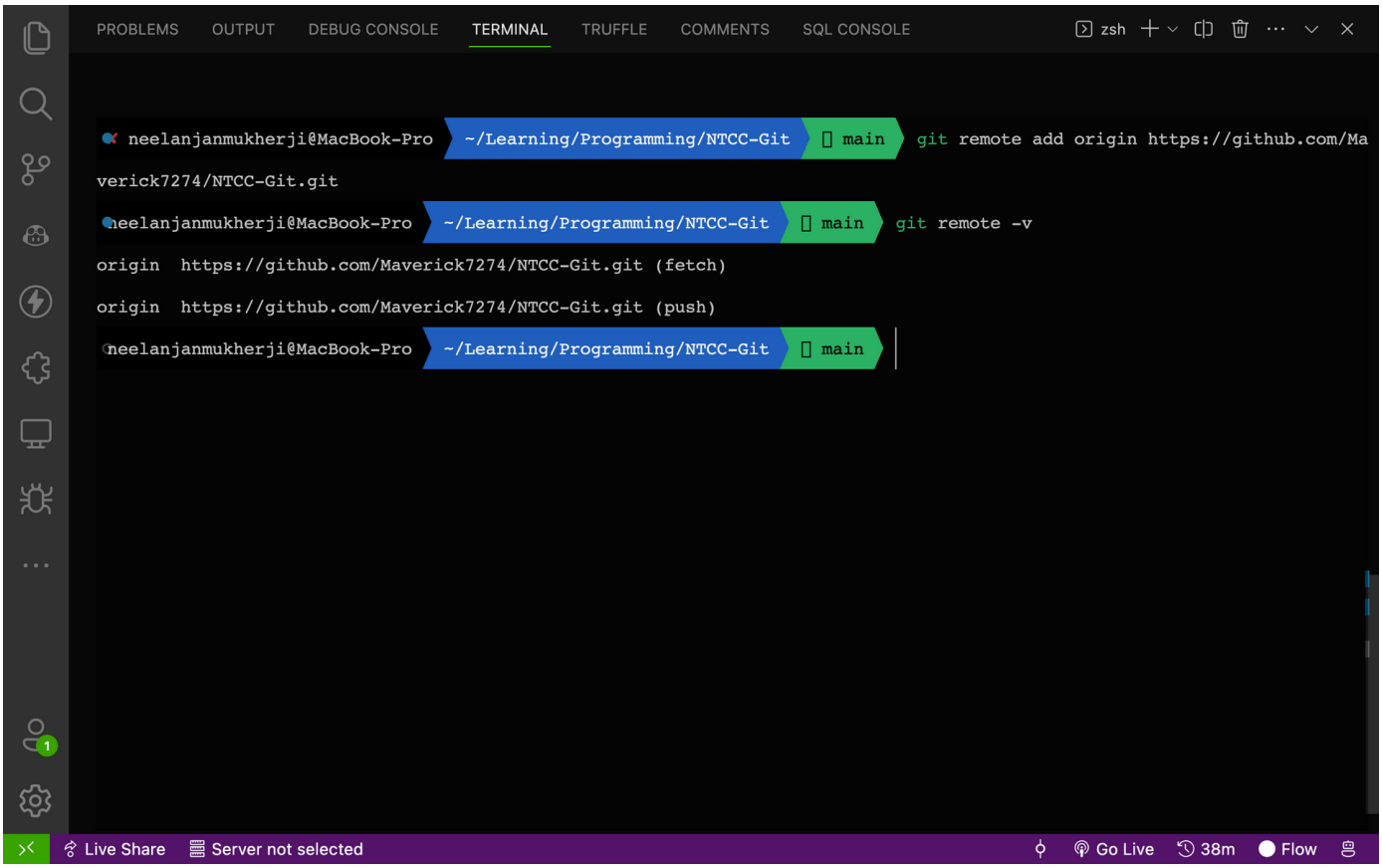
```
Initialized empty Git repository in
/Users/neelanjanmukherji/Learning/Programming/NTCC-Git/.git/
```

- Now, use `git remote add` command to add the remote repository(again copy the url from the Quick setup box).




```
git remote add origin https://github.com/Maverick7274/NTCC-Git.git
```

- To verify the remote repository, use `git remote -v` command.




```
git remote -v
```

Output :

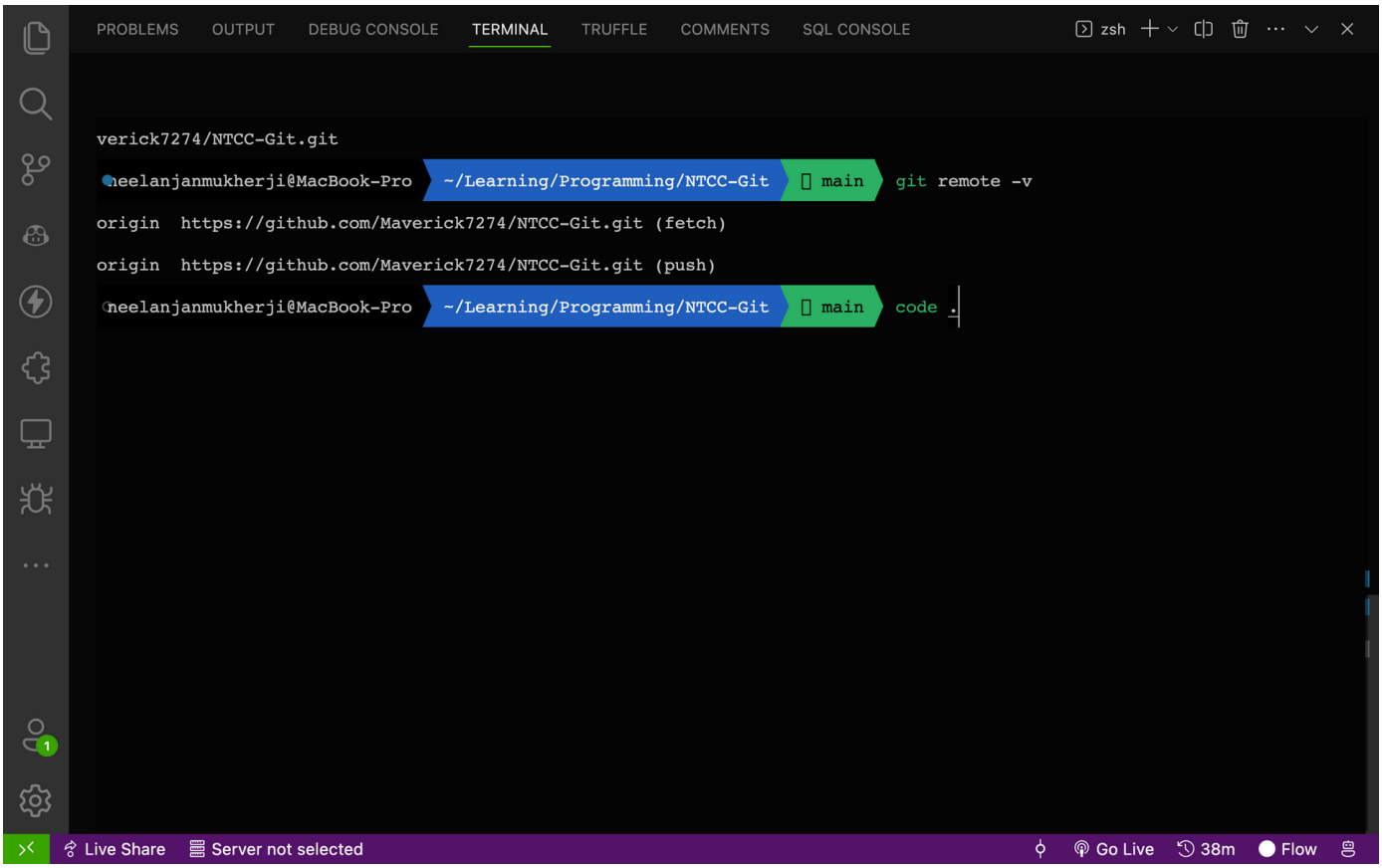
```
origin  https://github.com/Maverick7274/NTCC-Git.git (fetch)
origin  https://github.com/Maverick7274/NTCC-Git.git (push)
```

- Now, you have a local copy of the repository on your machine.

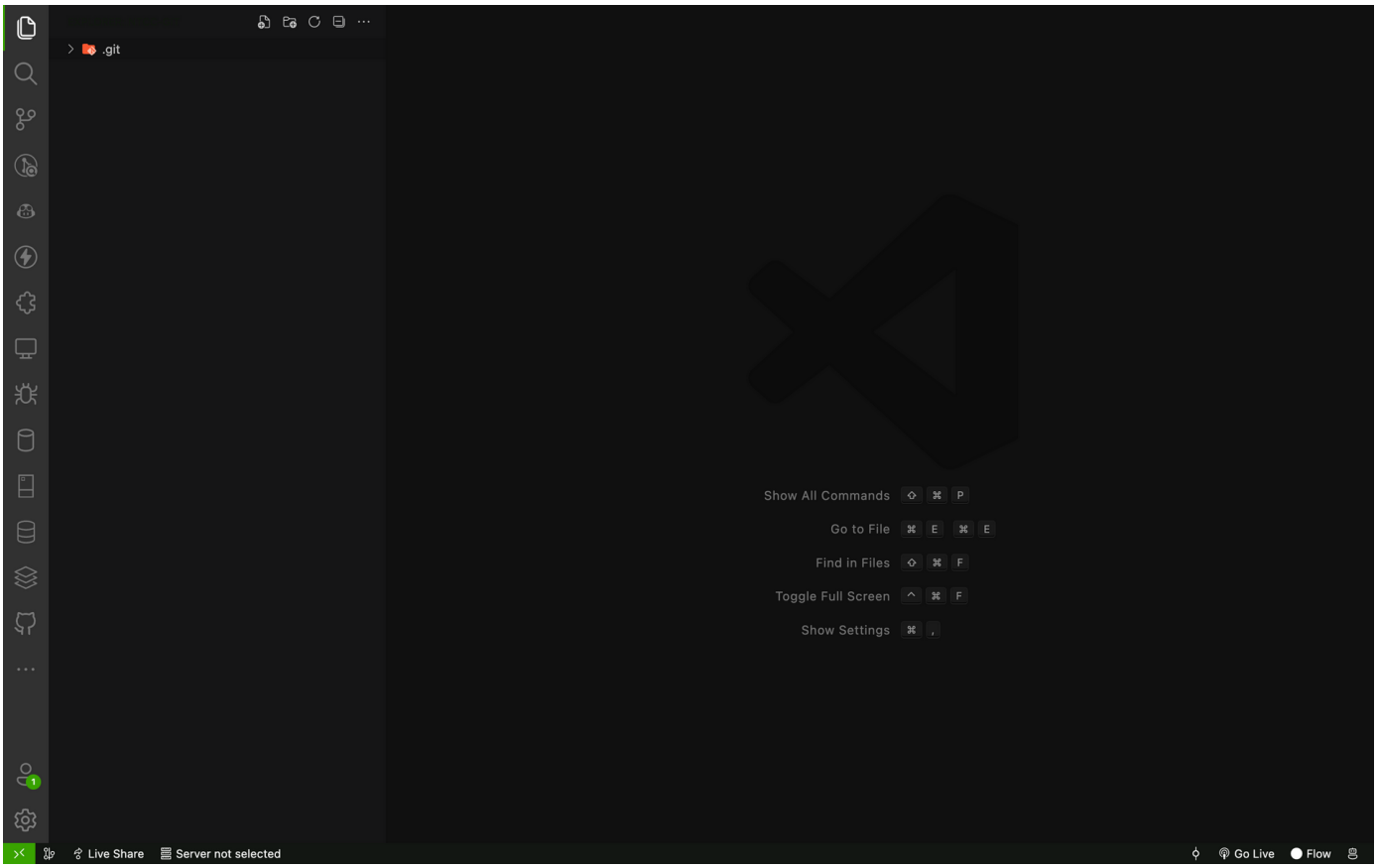
Making Changes to the Repository

- Now, open the repository in your favorite code editor.
- Here, we are using VS Code.

> PRO TIP : You can open the repository in VS Code by using the command `code .` in the terminal.

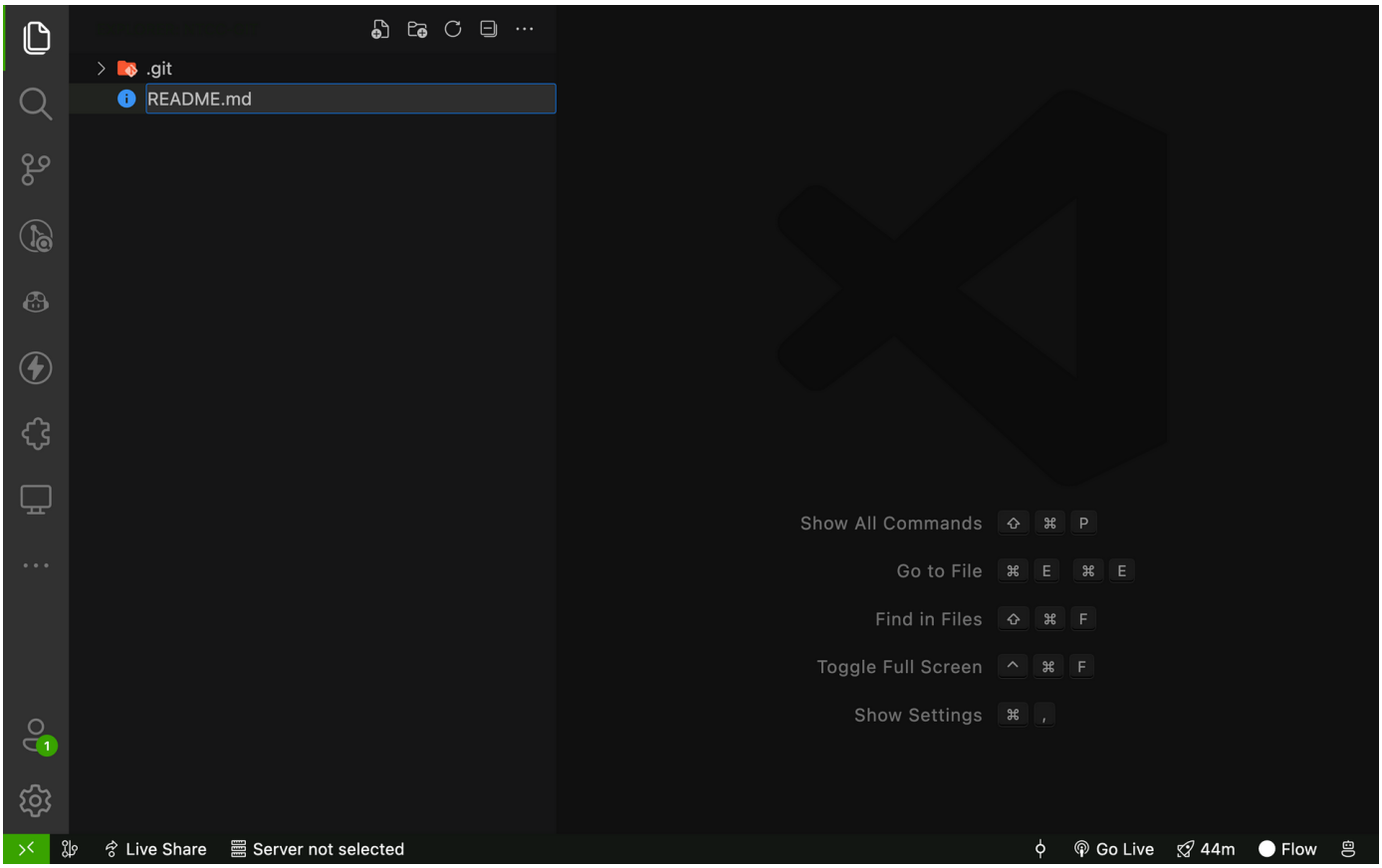


- After opening VS Code or any other code editor, navigate to the directory where you have cloned the repository.

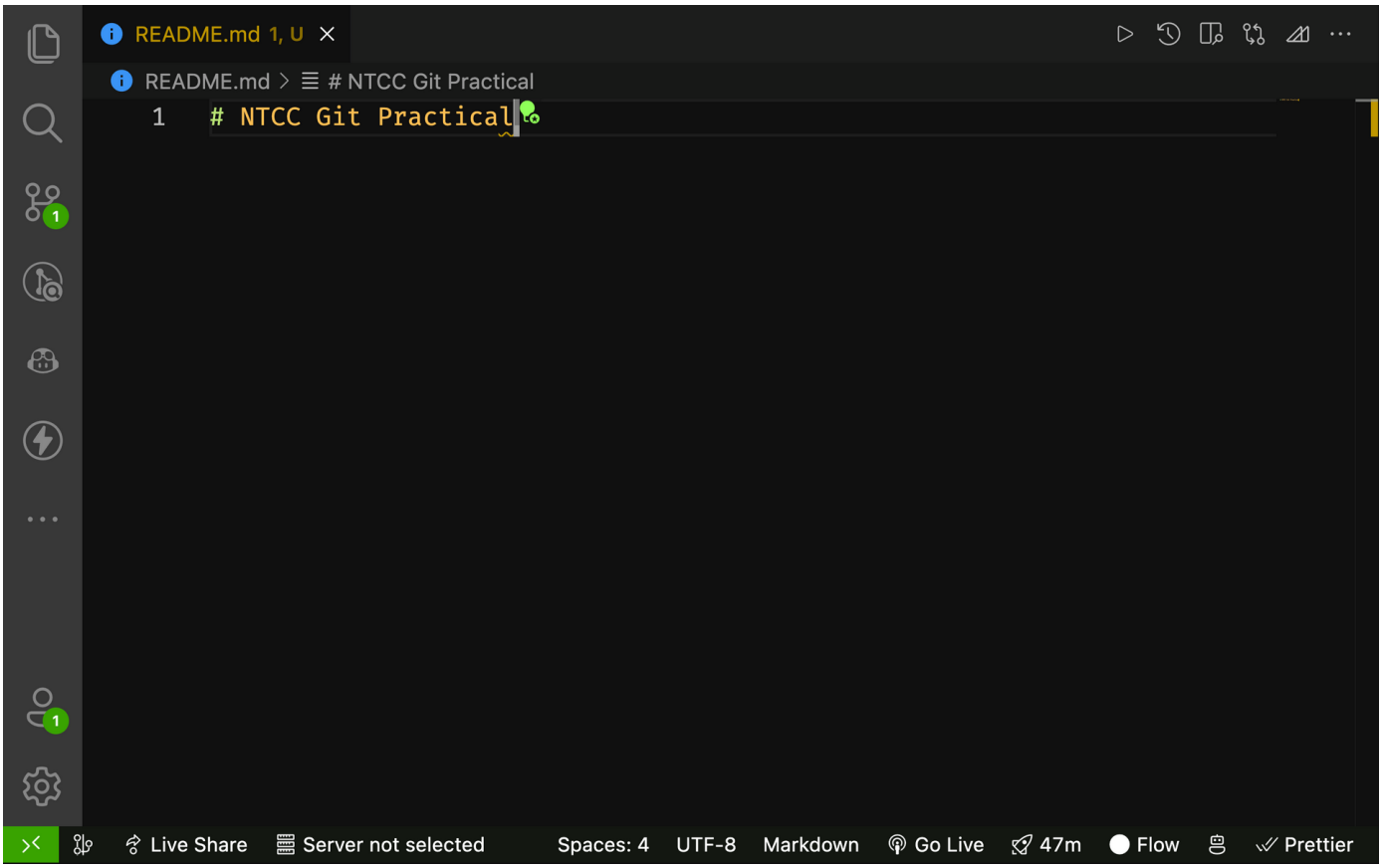


- Now here we have no files in the repository. It's a good practice to create a `README.md` file in the repository. It will help others to understand the repository. So, let's create a `README.md` file.
- To create a new file, click on the `New File` button.

- Now, give the file a name `README.md`.



- Markdown is a lightweight markup language for creating formatted text using a plain-text editor. John Gruber and Aaron Swartz created Markdown in 2004 as a markup language that is appealing to human readers in its source code form. Markdown is widely used in blogging, instant messaging, online forums, collaborative software, documentation pages, and readme files.
- To learn more about Markdown, visit https://www.markdownguide.org/.
- Now, let's add some text to the `README.md` file.



- Here in the demonstration we have used VS Code's auto-save feature. If you are using any other code editor, you have to save the file manually.
- Now, let's check the status of the repository.
- To check the status of the repository, use `git status` command.

```
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git  main git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        README.md

nothing added to commit but untracked files present (use "git add" to track)
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git  main
```

```
git status
```

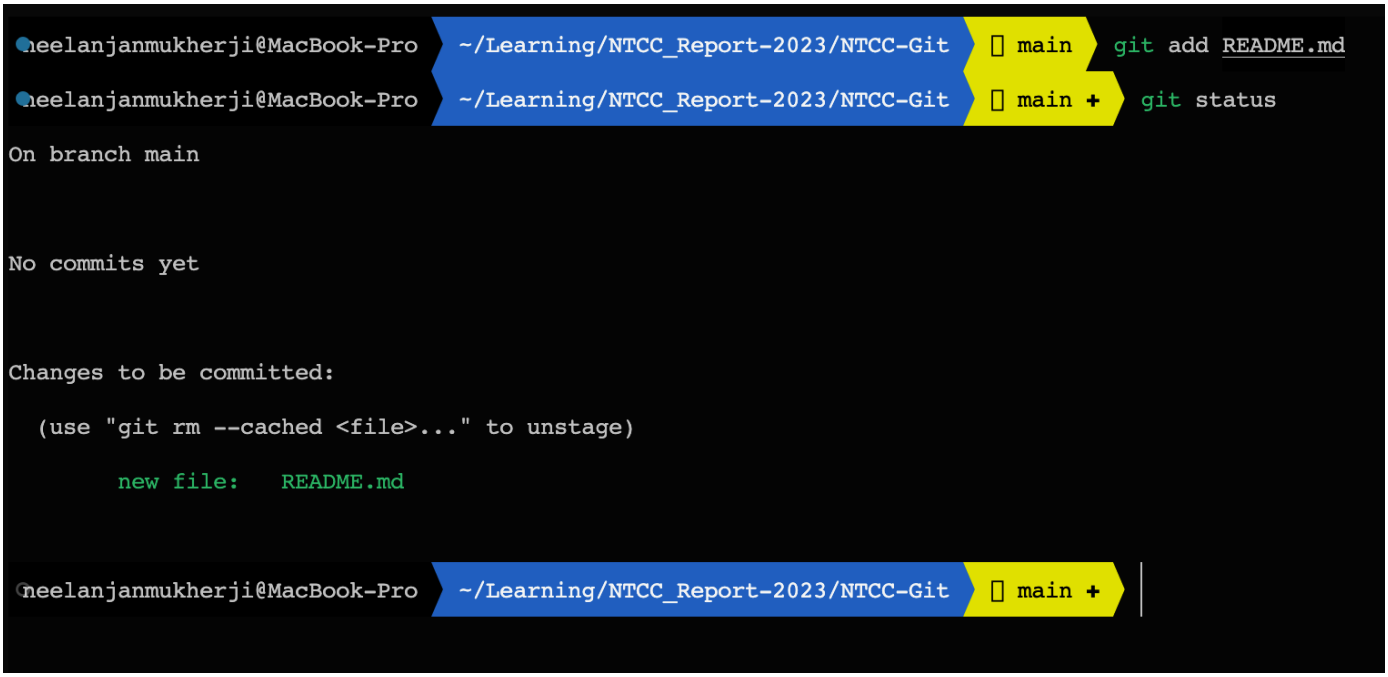
Output :

```
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        README.md

nothing added to commit but untracked files present (use "git add" to track)
```

- Here, we can see that the file `README.md` is untracked.
- Now, let's add the file to the staging area.
- To add the file to the staging area, use `git add` command.

```
git add README.md
```

- Now, let's check the status of the repository.
- To check the status of the repository, use `git status` command.

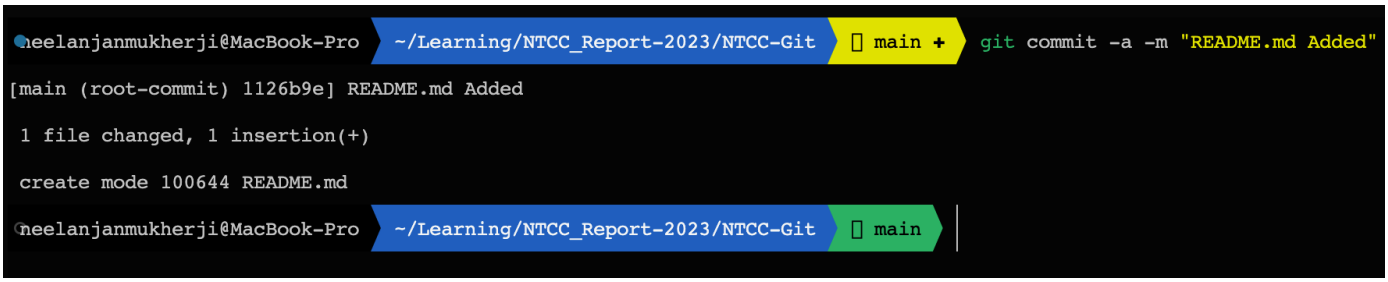



```
git status
```

Output :

```
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   README.md
```

- Now, let's commit the changes.
- To commit the changes, use `git commit` command.

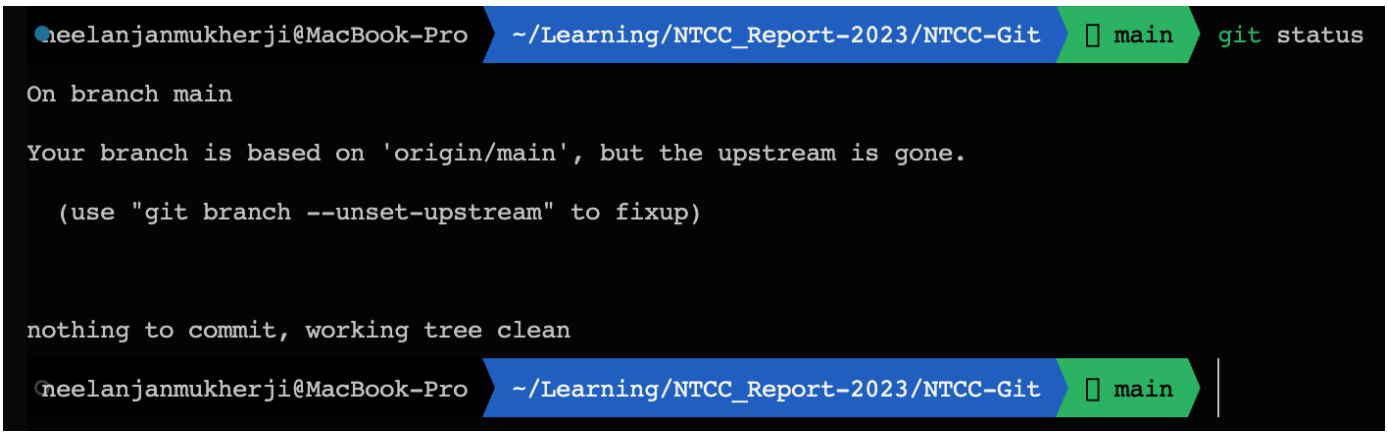



```
git commit -a -m "Added README.md file"
```

Output :

```
[main (root-commit) 1126b9e] README.md Added
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
```

- Now, let's check the status of the repository.
- To check the status of the repository, use `git status` command.

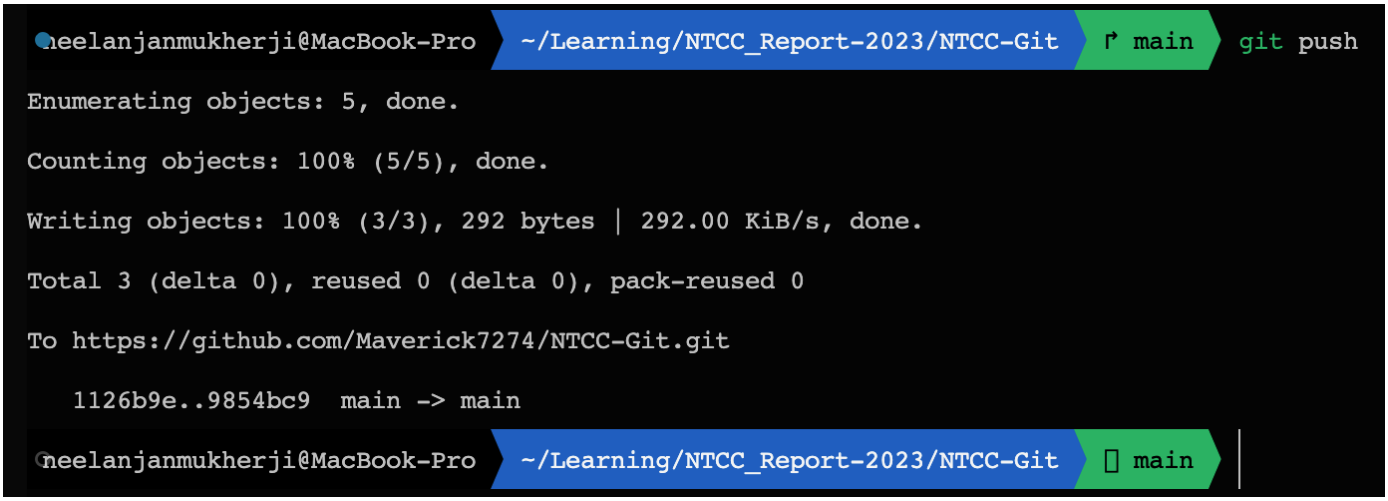



```
git status
```

Output :

```
On branch main
Your branch is based on 'origin/main', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

nothing to commit, working tree clean
```

- Now, let's push the changes to the remote repository.
- To push the changes to the remote repository, use `git push` command.




```
git push origin main
```

Output :

```
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 292 bytes | 292.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Maverick7274/NTCC-Git.git
   1126b9e..9854bc9  main -> main
```

- Now, you can see the changes in the remote repository.