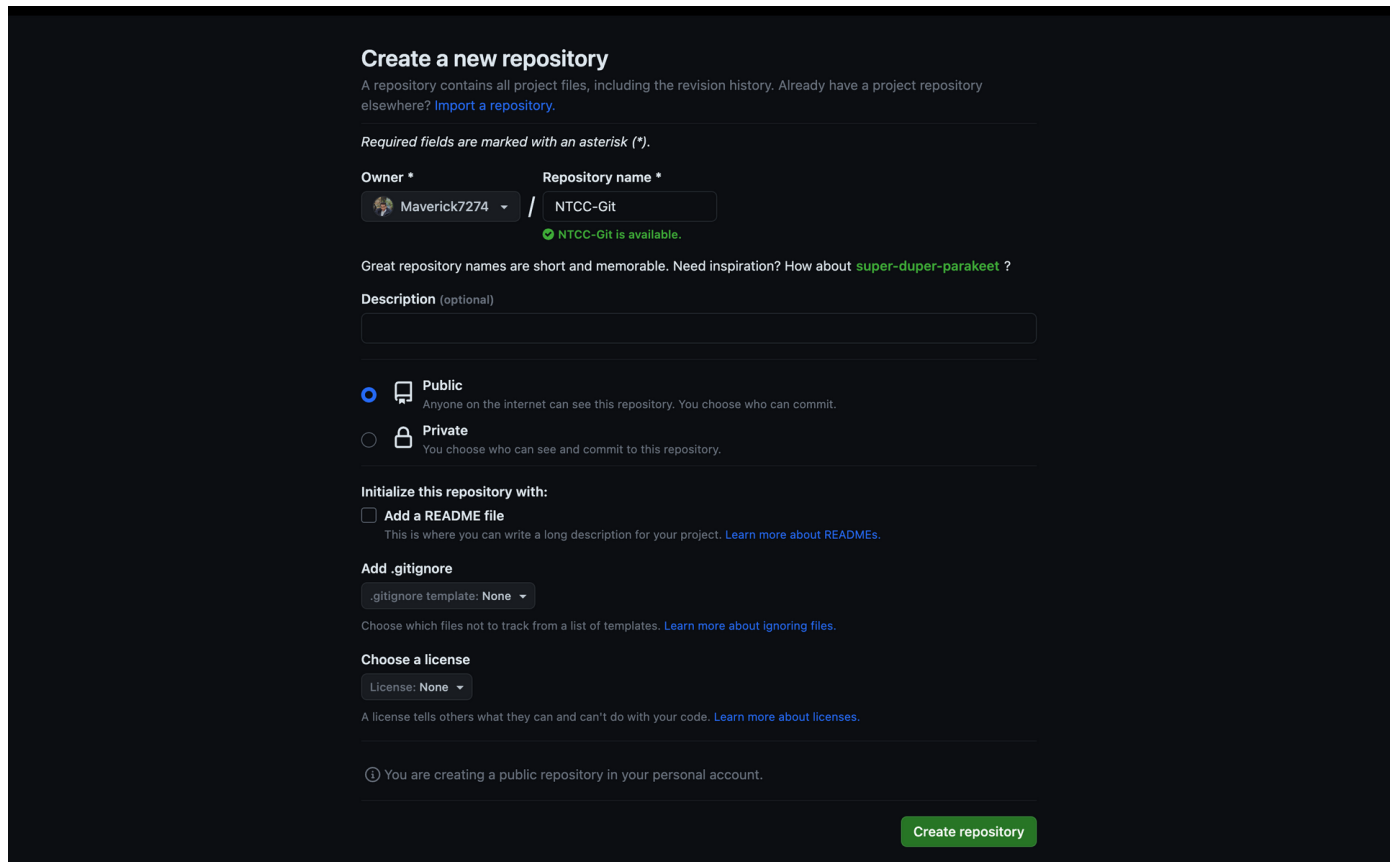


Practical Implementation of Git with GitHub

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).



The screenshot shows the GitHub 'Create a new repository' interface. At the top, it says 'Create a new repository' and provides a brief explanation of what a repository is. Below this, there's a note that required fields are marked with an asterisk. The form has two main sections: 'Owner' and 'Repository name'. The 'Owner' dropdown is set to 'Maverick7274'. The 'Repository name' text input contains 'NTCC-Git', and a green checkmark indicates it's available. Below these, there's a 'Description (optional)' text area. The 'Visibility' section has two radio buttons: 'Public' (selected) and 'Private'. Under 'Public', it says 'Anyone on the internet can see this repository. You choose who can commit.' Under 'Private', it says 'You choose who can see and commit to this repository.' The 'Initialize this repository with:' section has a checkbox for 'Add a README file'. Below that, there's a section for 'Add .gitignore' with a dropdown menu set to 'None'. The 'Choose a license' section has a dropdown menu set to 'None'. At the bottom, there's a green 'Create repository' button. A small informational note at the bottom left states: 'You are creating a public repository in your personal account.'

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

ode. [Learn more about licenses.](#)

nal account.

Create repository

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

The screenshot shows the GitHub interface for creating a new repository. At the top, the repository name 'NTCC-Git' is displayed as 'Public'. Navigation buttons include 'Pin', 'Unwatch 1', 'Fork', and 'Star 0'. Below these are two main options: 'Create a codespace' (with a description: 'Add a README file and start coding in a secure, configurable, and dedicated development environment.') and 'Invite collaborators' (with a description: 'Find people using their GitHub username or email address.').

The 'Quick setup — if you've done this kind of thing before' section provides a 'Set up in Desktop' button, an 'or' separator, and 'HTTPS' and 'SSH' options. The HTTPS URL is 'https://github.com/Maverick7274/NTCC-Git.git'. Below this, it says: 'Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).'

The '...or create a new repository on the command line' section shows a terminal window with the following commands:

```
echo "# NTCC-Git" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Maverick7274/NTCC-Git.git
git push -u origin main
```

The '...or push an existing repository from the command line' section shows the following commands:

```
git remote add origin https://github.com/Maverick7274/NTCC-Git.git
git branch -M main
git push -u origin main
```

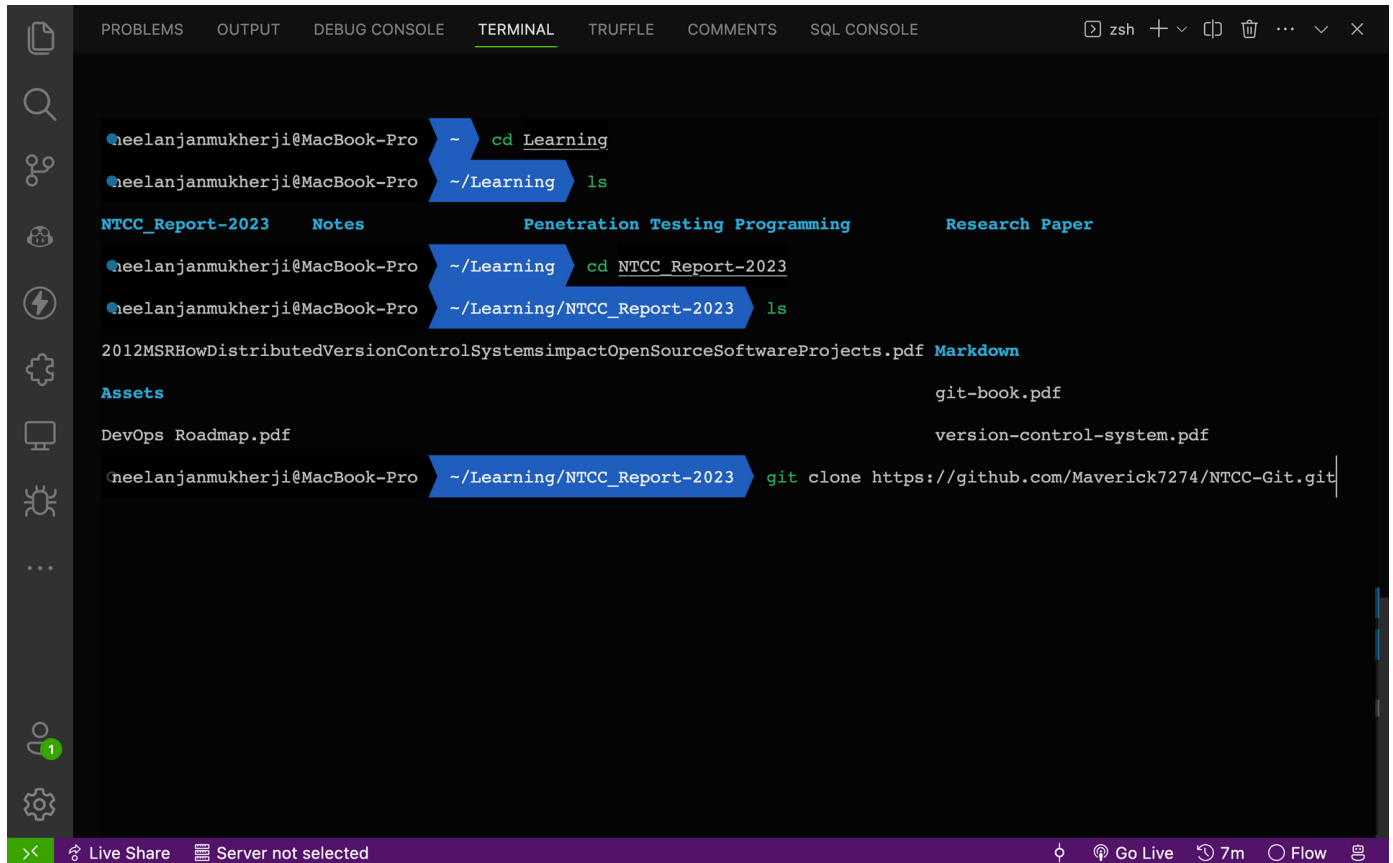
The '...or import code from another repository' section states: 'You can initialize this repository with code from a Subversion, Mercurial, or TFS project.' and includes an 'Import code' button.

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.

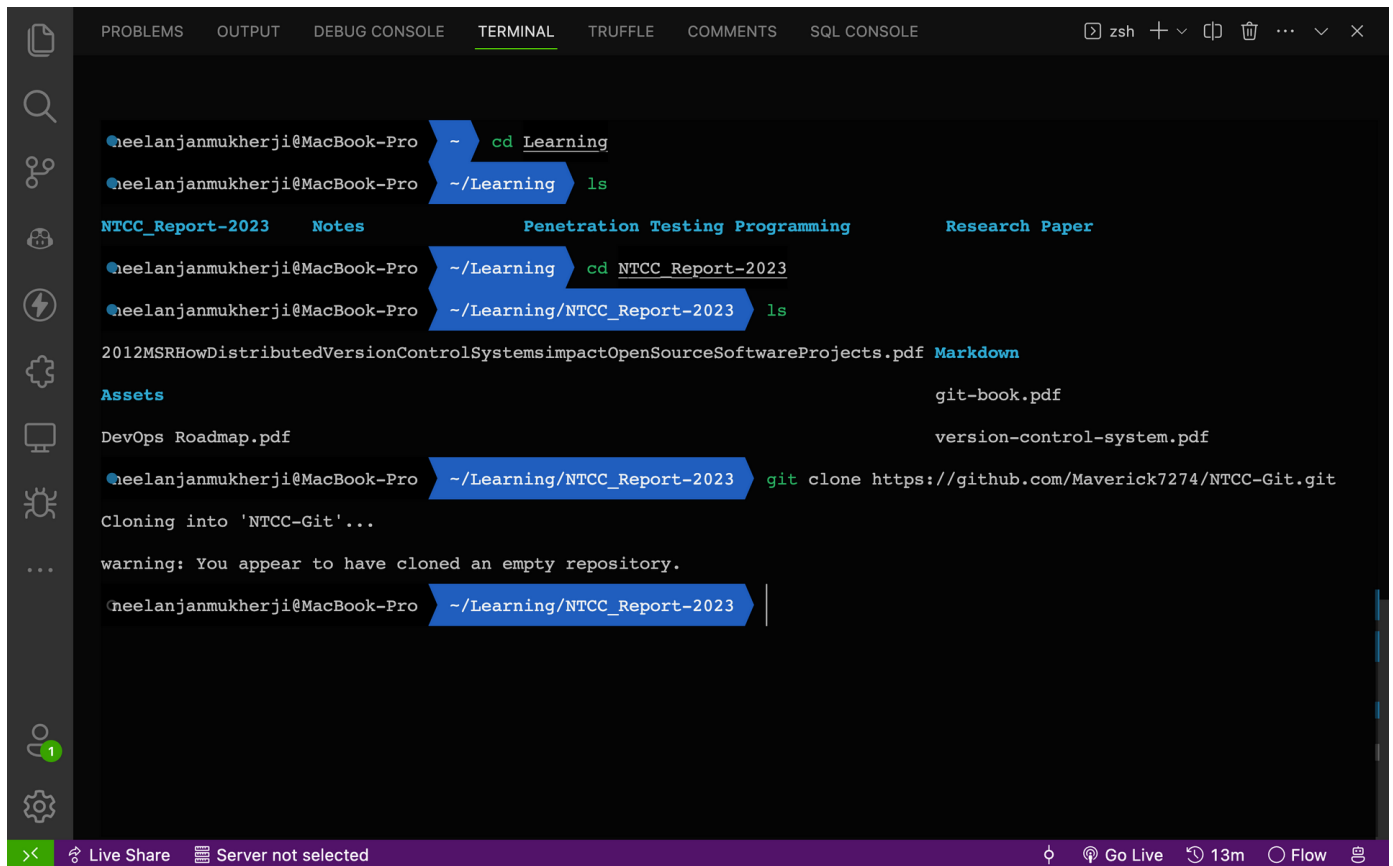


```
neelanjanmukherji@MacBook-Pro ~ % cd Learning
neelanjanmukherji@MacBook-Pro ~/Learning % ls
NTCC_Report-2023  Notes  Penetration Testing Programming  Research Paper
neelanjanmukherji@MacBook-Pro ~/Learning % cd NTCC_Report-2023
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023 % ls
2012MSRHowDistributedVersionControlSystemsimpactOpenSourceSoftwareProjects.pdf  Markdown
Assets  git-book.pdf
DevOps Roadmap.pdf  version-control-system.pdf
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023 % git clone https://github.com/Maverick7274/NTCC-Git.git
```

- 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.



```
neelanjanmukherji@MacBook-Pro ~ cd Learning
neelanjanmukherji@MacBook-Pro ~/Learning ls
NTCC_Report-2023  Notes  Penetration Testing Programming  Research Paper
neelanjanmukherji@MacBook-Pro ~/Learning cd NTCC_Report-2023
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023 ls
2012MSRHowDistributedVersionControlSystemsimpactOpenSourceSoftwareProjects.pdf Markdown
Assets  git-book.pdf
DevOps Roadmap.pdf  version-control-system.pdf
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023 git clone https://github.com/Maverick7274/NTCC-Git.git
Cloning into 'NTCC-Git'...
warning: You appear to have cloned an empty repository.
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023
```

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.

```
neelanjanmukherji@MacBook-Pro ~/Learning/Programming mkdir NTCC-Git
neelanjanmukherji@MacBook-Pro ~/Learning/Programming ls -la

total 16
drwxr-xr-x 20 neelanjanmukherji staff 640 Jun 20 18:15 .
drwxr-xr-x  8 neelanjanmukherji staff 256 Jun 13 12:36 ..
-rw-r--r--@ 1 neelanjanmukherji staff 6148 Mar 11 18:29 .DS_Store
drwxr-xr-x  6 neelanjanmukherji staff 192 Feb  9 18:42 Game-Development
drwxr-xr-x  3 neelanjanmukherji staff  96 Mar  1 00:13 Golang
drwxr-xr-x  8 neelanjanmukherji staff 256 Apr  7 00:49 MATLAB
drwxr-xr-x  3 neelanjanmukherji staff  96 Dec 20 2022 Markdown
drwxr-xr-x  2 neelanjanmukherji staff  64 Jun 20 18:15 NTCC-Git
drwxr-xr-x  4 neelanjanmukherji staff 128 Feb  9 18:42 androidDevelopment
drwxr-xr-x  2 neelanjanmukherji staff  64 Sep 16 2022 apis
drwxr-xr-x  4 neelanjanmukherji staff 128 Feb  9 18:42 c
drwxr-xr-x  3 neelanjanmukherji staff  96 Nov 11 2022 cpp
drwxr-xr-x  3 neelanjanmukherji staff  96 Sep 26 2022 docker
drwxr-xr-x  3 neelanjanmukherji staff  96 Feb  9 18:42 iosDevelopment
```

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

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

```
neelanjanmukherji@MacBook-Pro ~/Learning/Programming cd NTCC-Git
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git git init

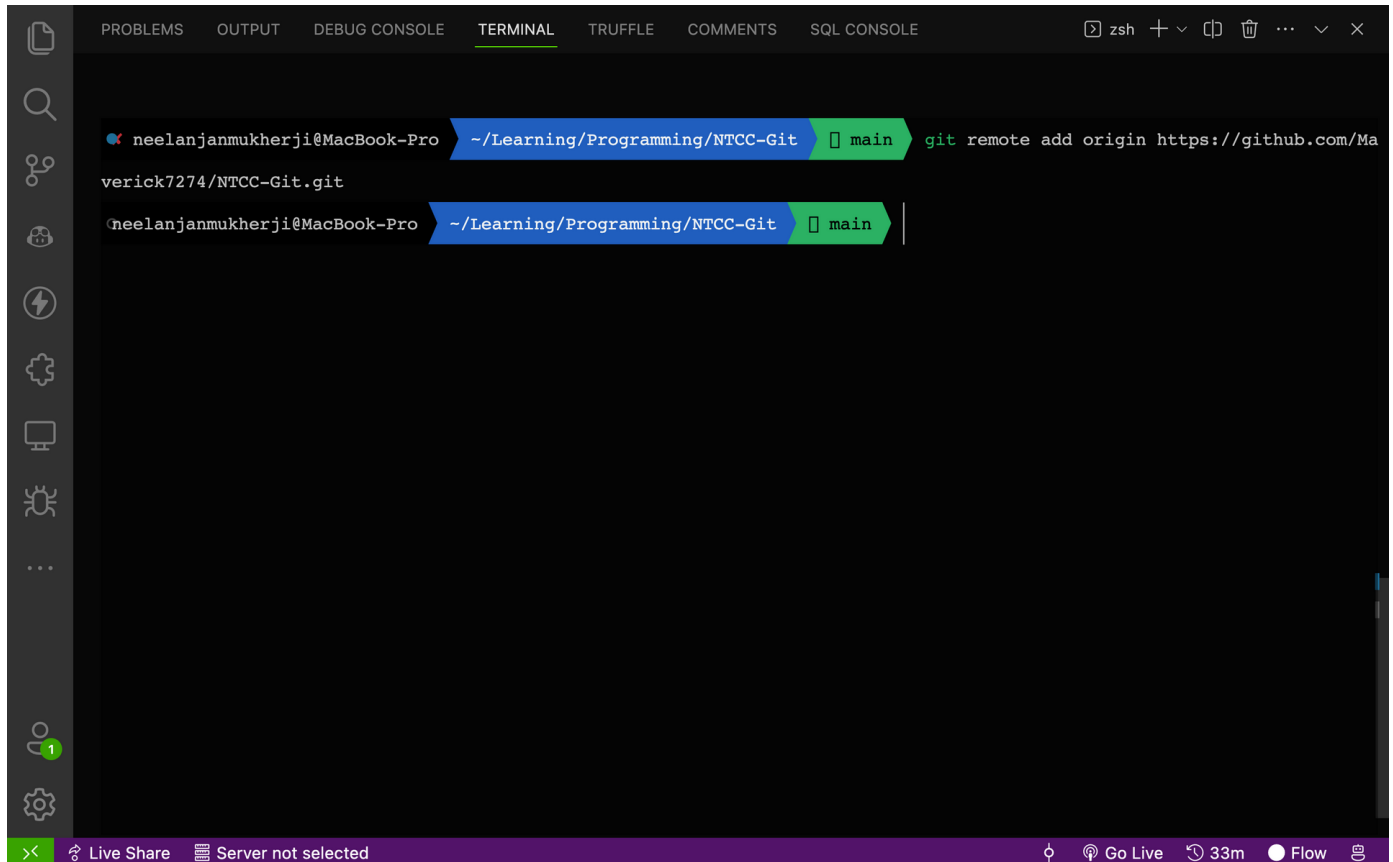
Initialized empty Git repository in /Users/neelanjanmukherji/Learning/Programming/NTCC-Git/.git/
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git main
```

`git init`

Output :

Initialized empty Git repository in
/Users/neelanjankmukherji/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).

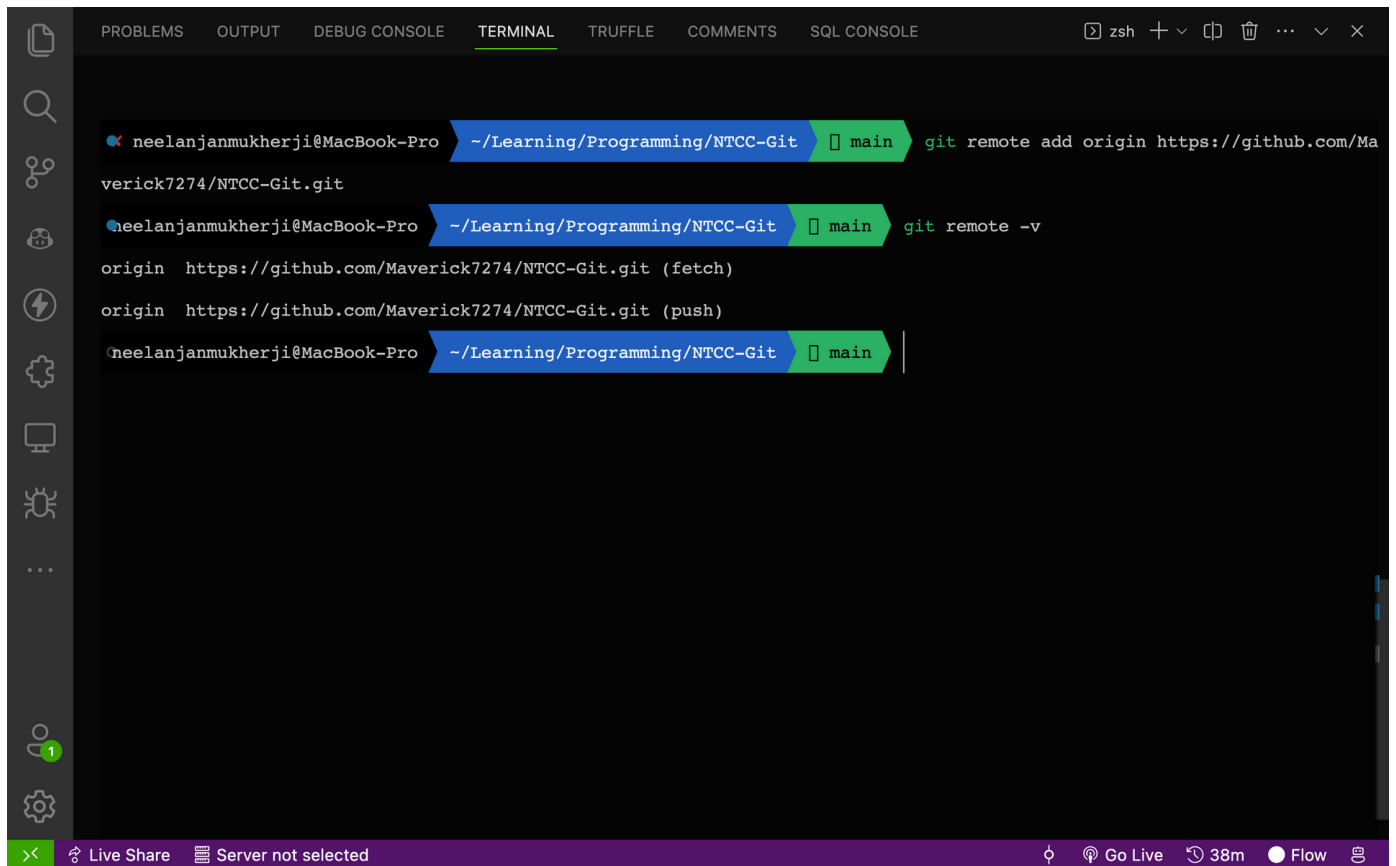


The screenshot shows a VS Code interface with a terminal window open. The terminal title bar indicates the user is 'neelanjankmukherji@MacBook-Pro' in the directory '~/Learning/Programming/NTCC-Git' on the 'main' branch. The terminal content shows the command `git remote add origin https://github.com/Maverick7274/NTCC-Git.git` being entered. The VS Code interface includes a sidebar on the left with icons for Explorer, Search, Source Control, Run and Debug, Extensions, Remote Explorer, Test Explorer, and Settings. The bottom status bar shows 'Live Share' and 'Server not selected'.

```
neelanjankmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git [main] git remote add origin https://github.com/Maverick7274/NTCC-Git.git
```

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

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



The screenshot shows a VS Code terminal window with the following content:

```
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git [main] git remote add origin https://github.com/Ma
verick7274/NTCC-Git.git
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git [main] git remote -v
origin https://github.com/Maverick7274/NTCC-Git.git (fetch)
origin https://github.com/Maverick7274/NTCC-Git.git (push)
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git [main]
```

The terminal window has a dark theme and a sidebar on the left with various icons. The bottom status bar shows 'Live Share', 'Server not selected', 'Go Live', '38m', 'Flow', and a document icon.

```
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.

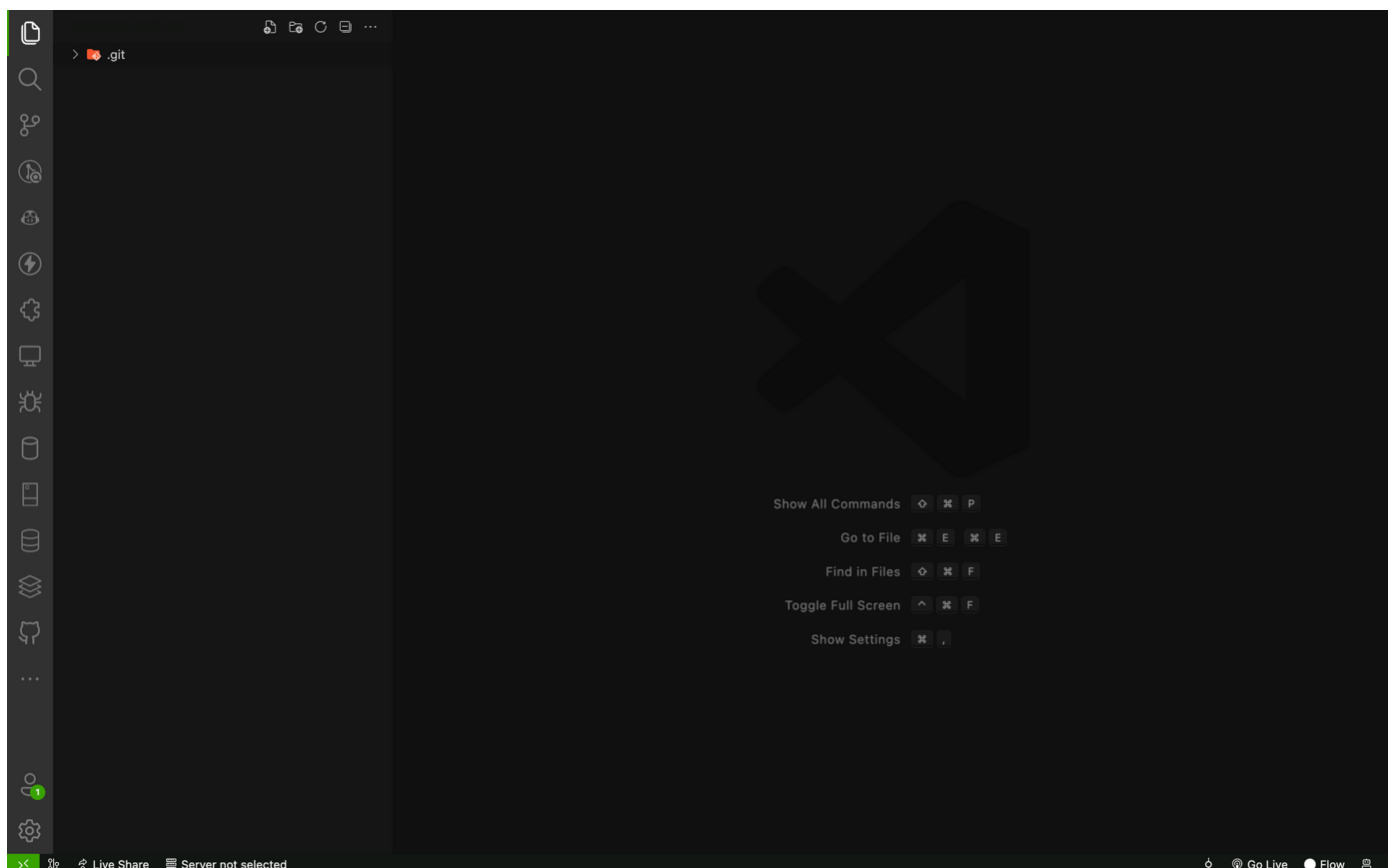
PRO TIP : You can open the repository in VS Code by using the command `code .` in the terminal.

The screenshot shows a VS Code terminal window with the following content:

```
verick7274/NTCC-Git.git
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git [main] git remote -v
origin https://github.com/Maverick7274/NTCC-Git.git (fetch)
origin https://github.com/Maverick7274/NTCC-Git.git (push)
neelanjanmukherji@MacBook-Pro ~/Learning/Programming/NTCC-Git [main] code .
```

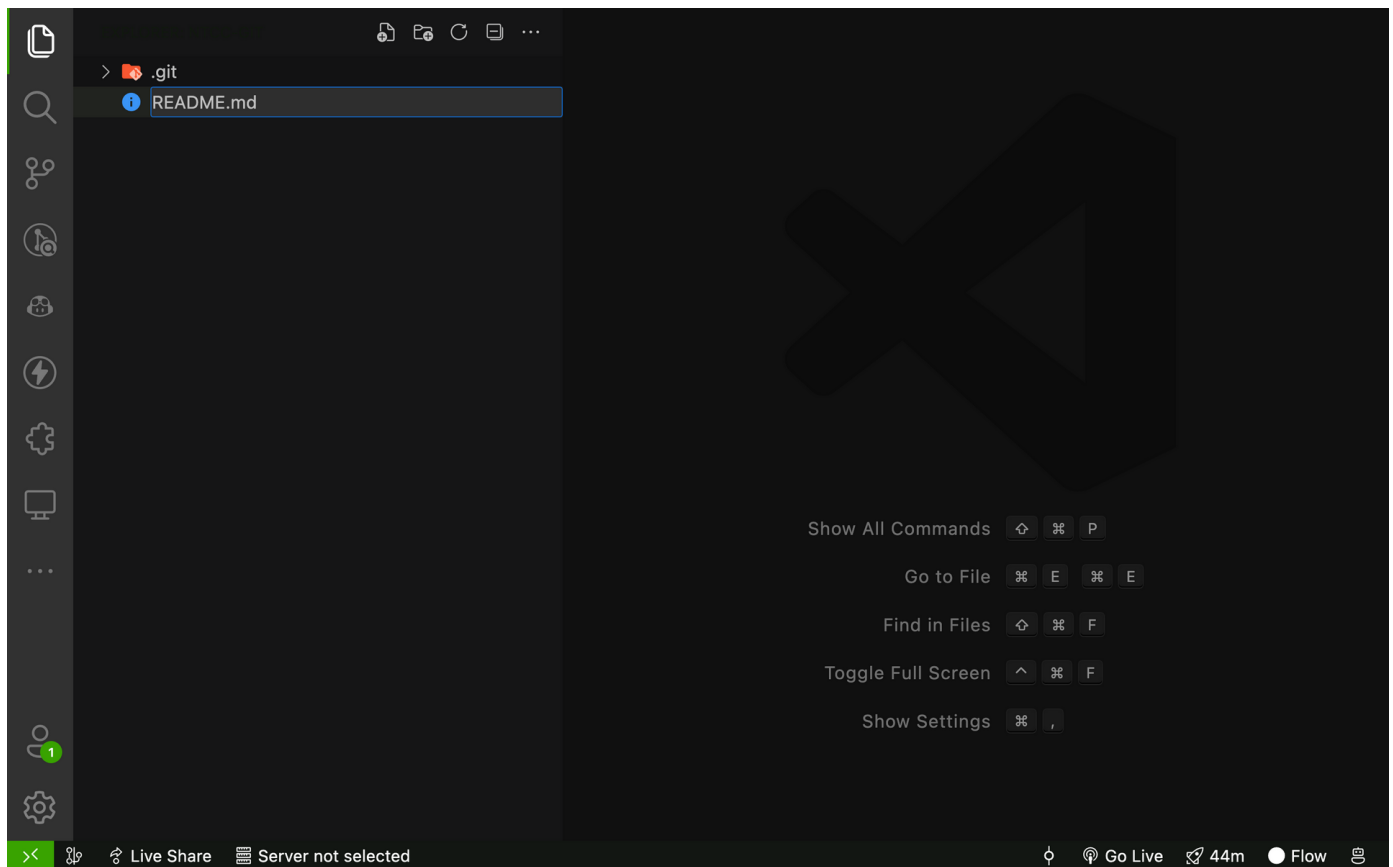
The terminal window is titled "verick7274/NTCC-Git.git". The prompt shows the user is on a MacBook-Pro in the directory ~/Learning/Programming/NTCC-Git, on the main branch. The output of the `git remote -v` command shows the origin remote pointing to https://github.com/Maverick7274/NTCC-Git.git for both fetch and push operations. The prompt then shows the user typing `code .`.

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

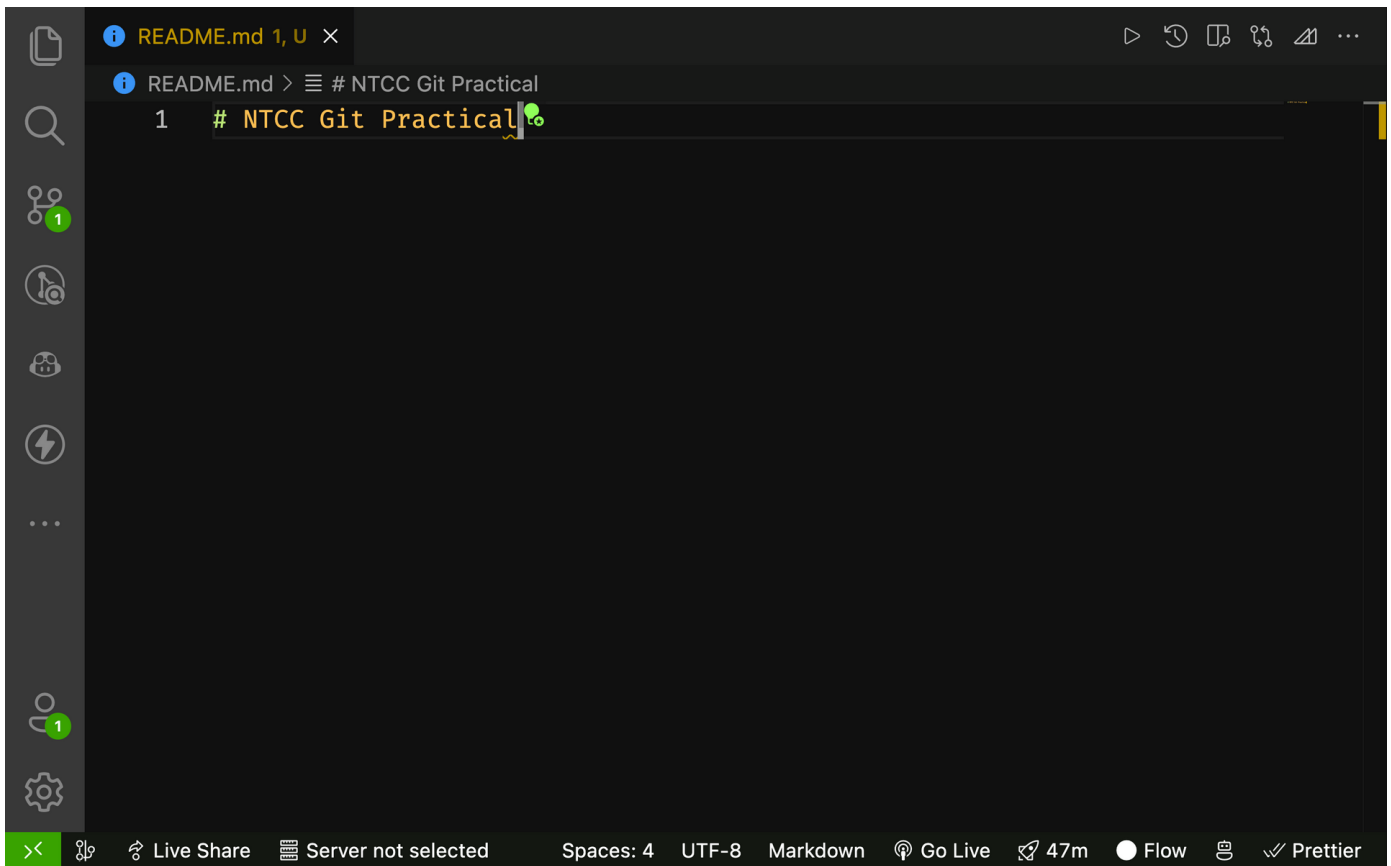


- Now here we have no files in the repository. It's a good practice to create a `README.md` file in the repository. It will help others to understand the repository. So, let's create a `README.md` file.
- To create a new file, click on the `New File` button.

- 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.

```
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git [main] git add README.md
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git [main +] git status

On branch main

No commits yet

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

neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git [main +]
```

```
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.

```
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git [main +] git commit -a -m "README.md Added"
[main (root-commit) 1126b9e] README.md Added
1 file changed, 1 insertion(+)
create mode 100644 README.md
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git [main]
```

```
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.

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

```
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.

```
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git main git push
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
neelanjanmukherji@MacBook-Pro ~/Learning/NTCC_Report-2023/NTCC-Git main
```

```
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.