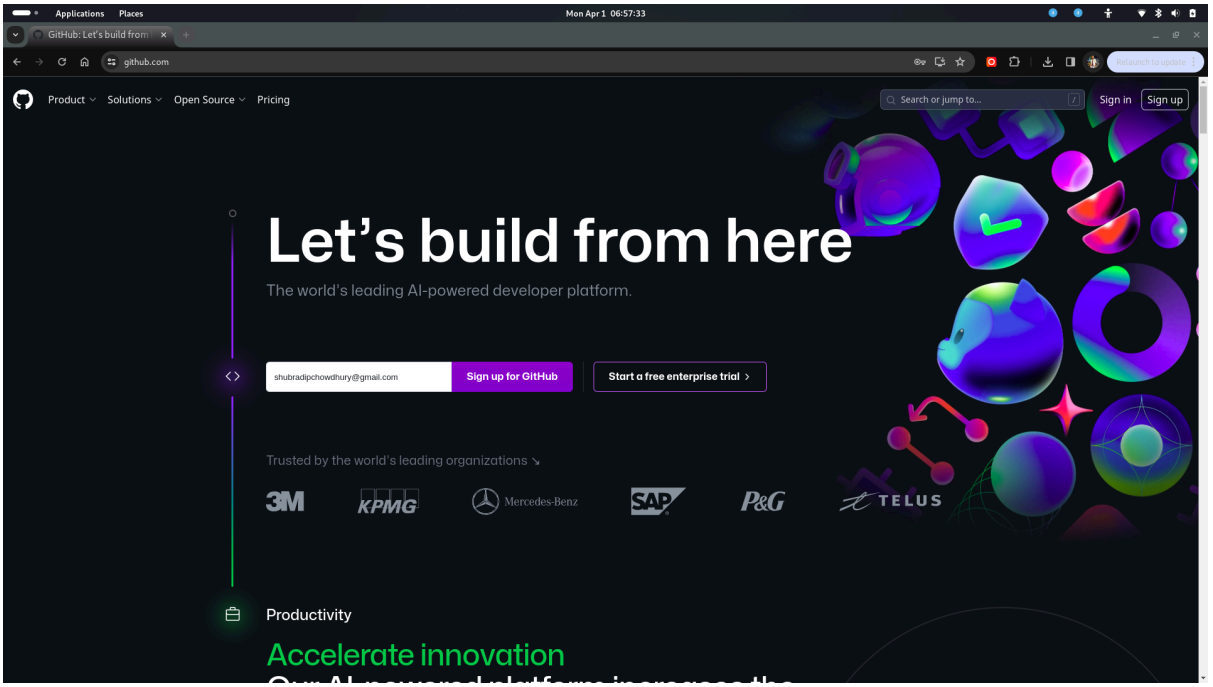
ASSIGNMENT-8

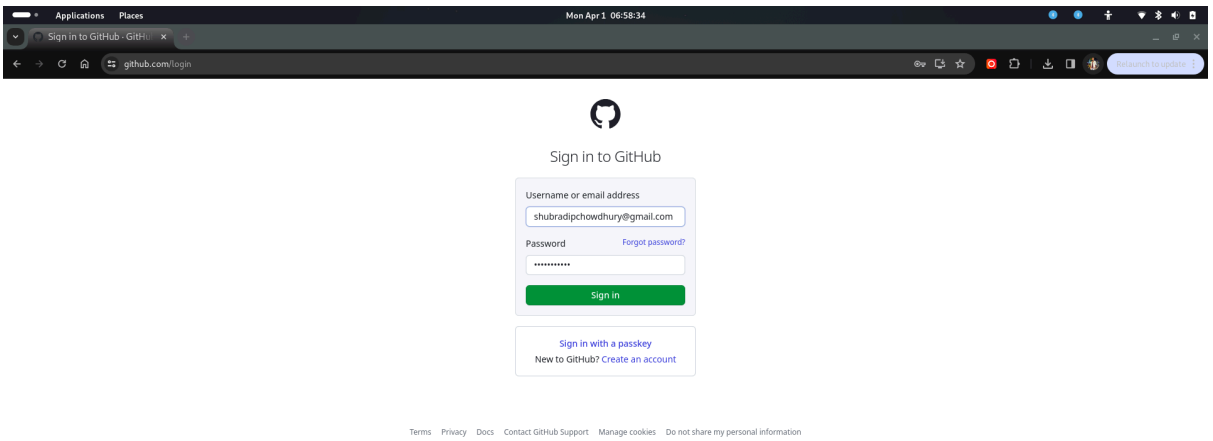
Problem Statement: Deploy a project from a local machine to GitHub and vice-versa.

The steps to deploy a project from local machine to GitHub are as follows: -

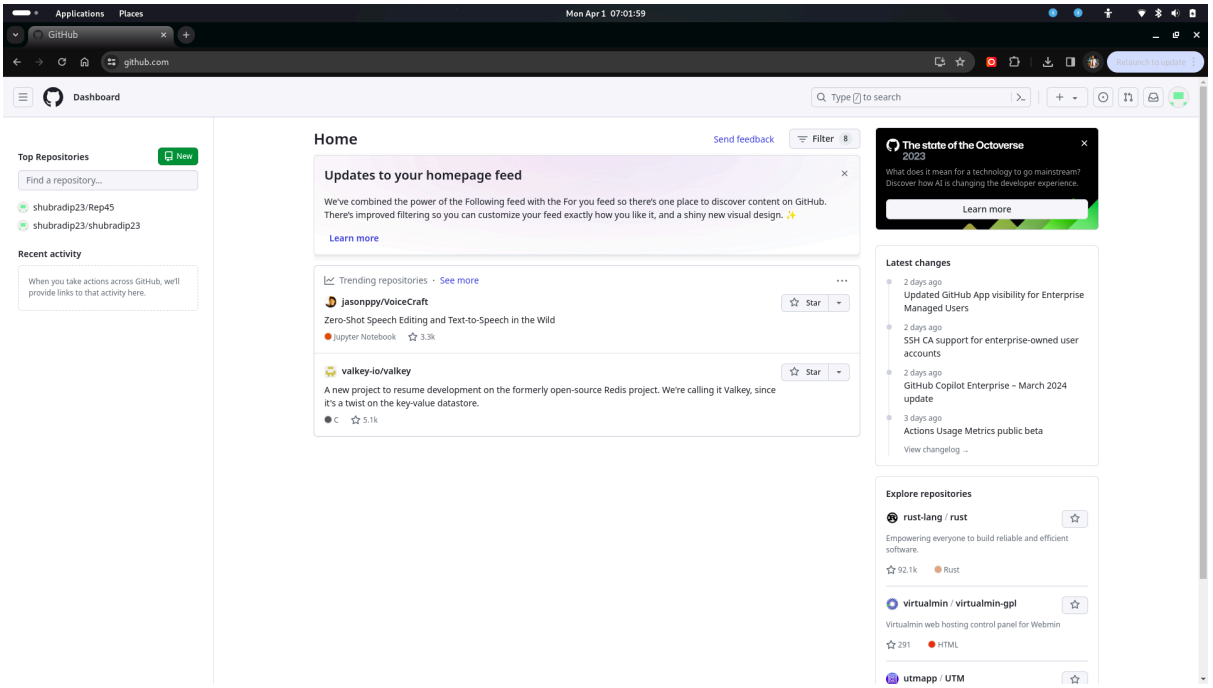
1. The initial and most important step involves downloading Git, a DevOps tool, and establishing a GitHub account.

2. Log in to your GitHub account using your login details.

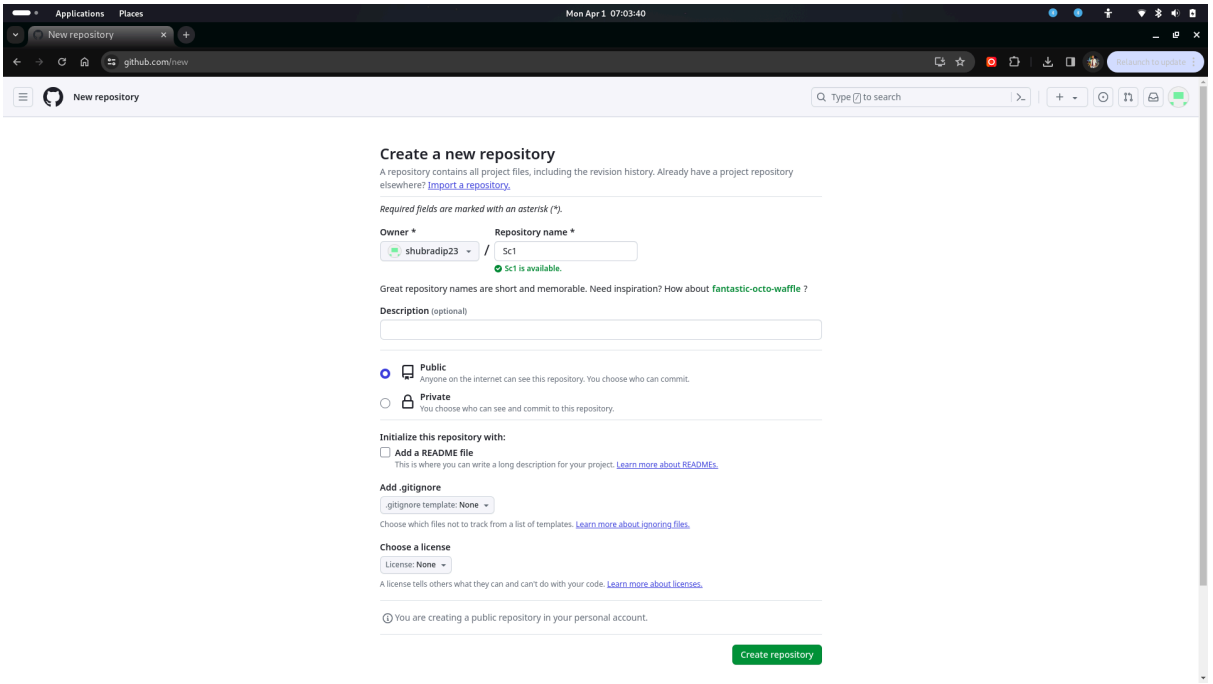




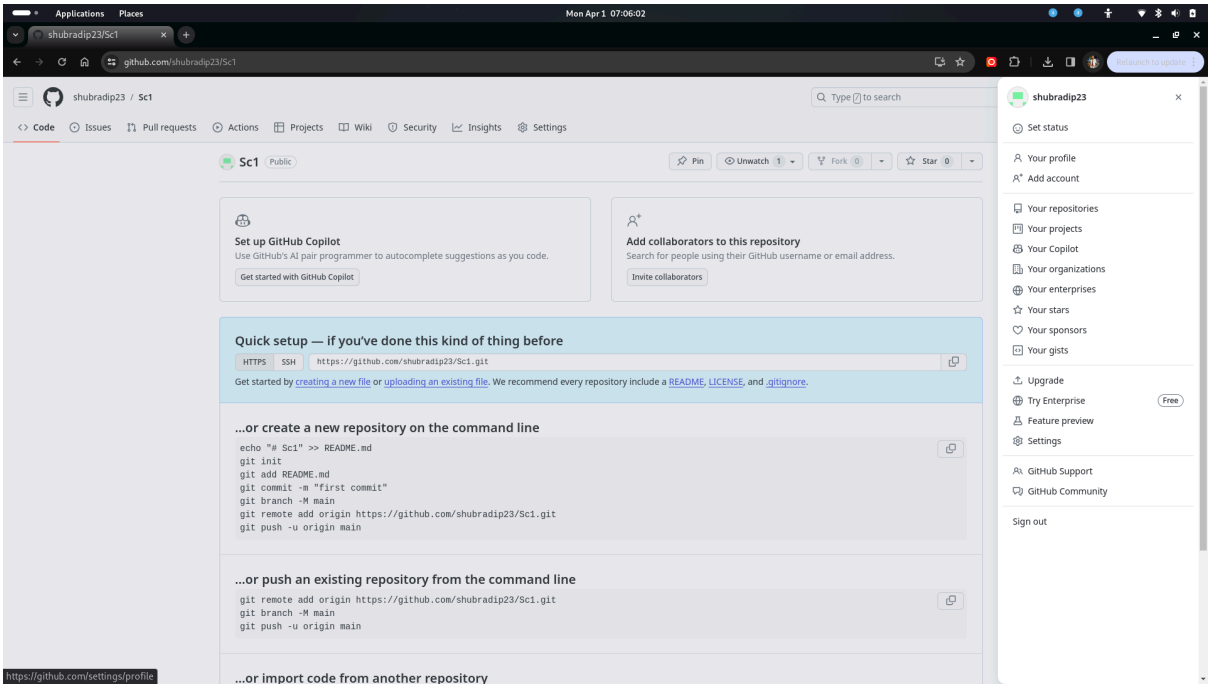
3. Upon successfully logging in, initiate the creation of a new repository by selecting the appropriate option on the left side of the window panel.



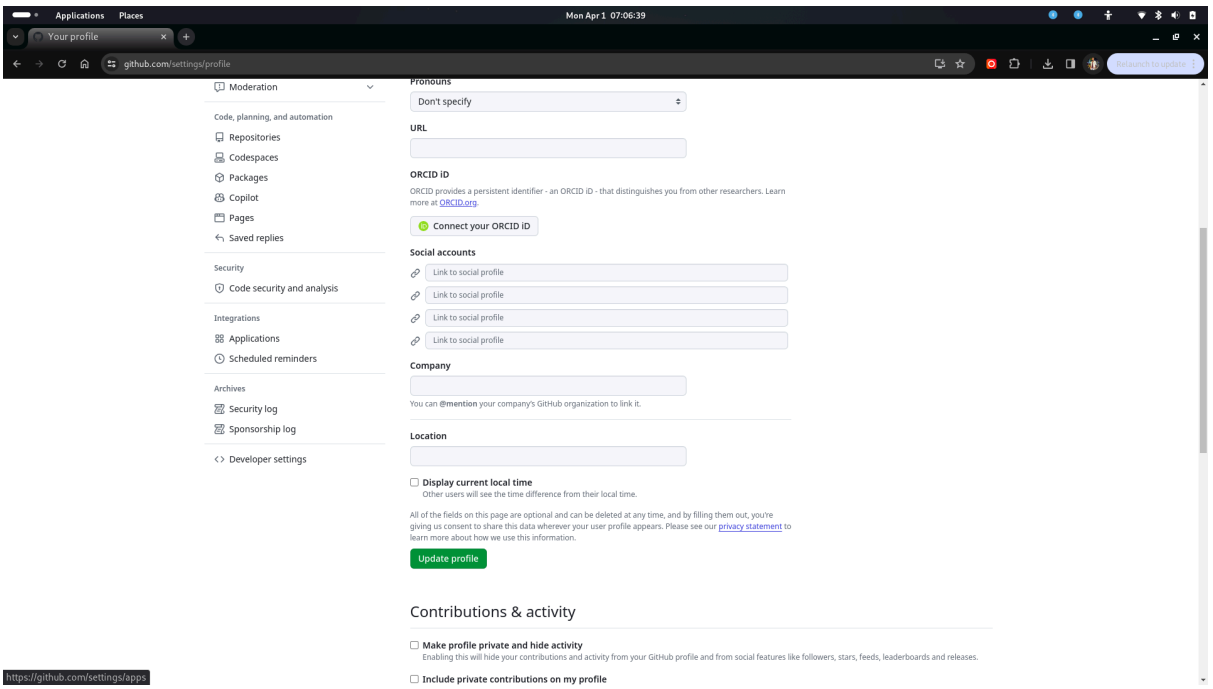
4. Choose an appropriate name for your repository (for example, we'll use "**sc1**") and ensure that the repository is set to public.Simply click on "**create repository**" without making any additional modifications.



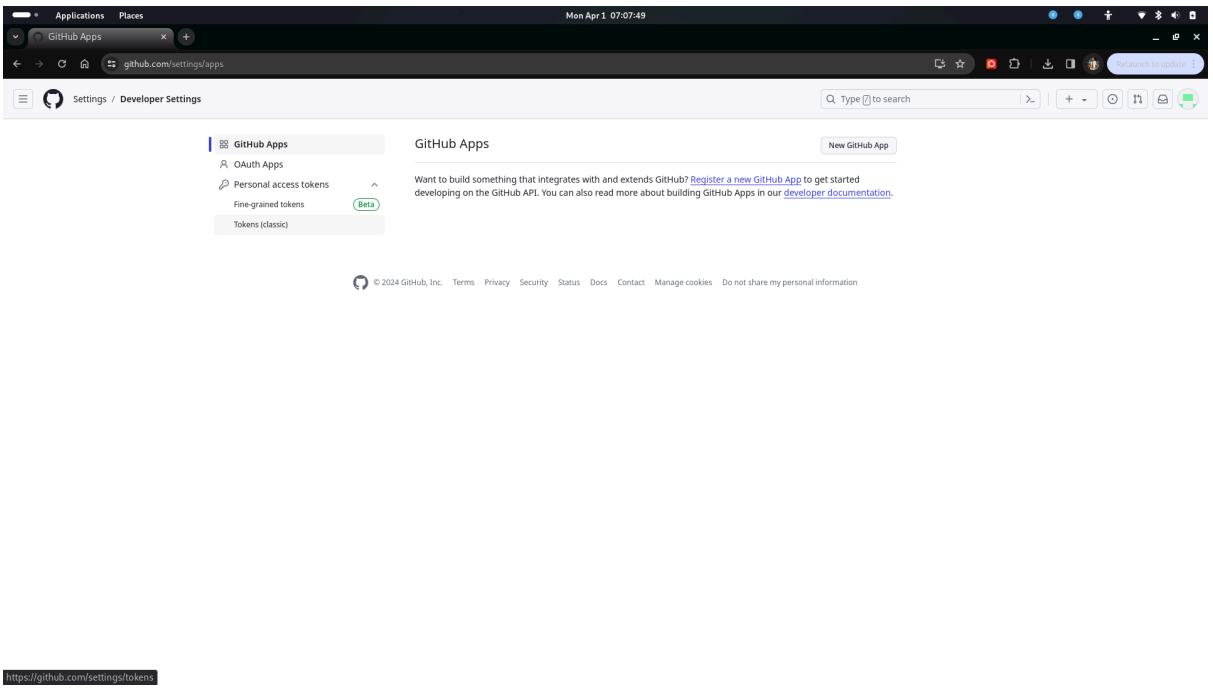
5. After creating a repository, navigate to your account on the right-hand side, and then locate and click on "**Settings**."



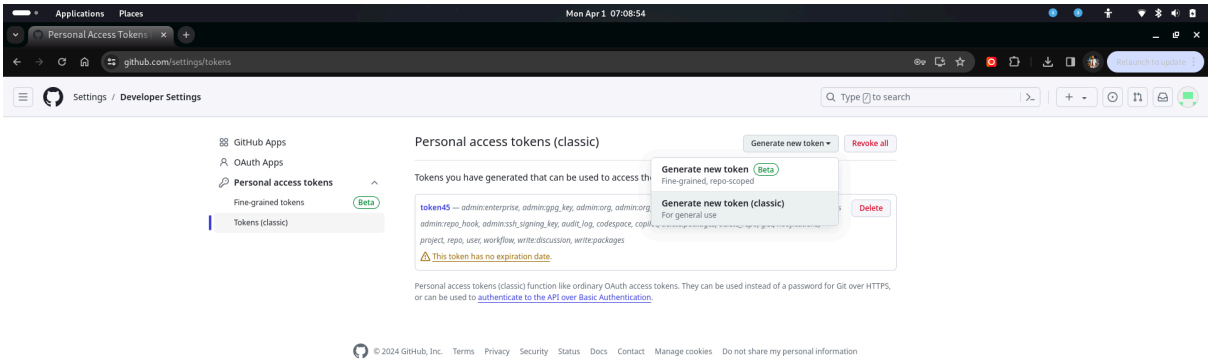
6. Scroll down and locate "**Developer settings**" on the left-hand side, then click on it.



7.Select "**Personal Access Tokens**" and choose "**Tokens (classic)**" from the dropdown list.

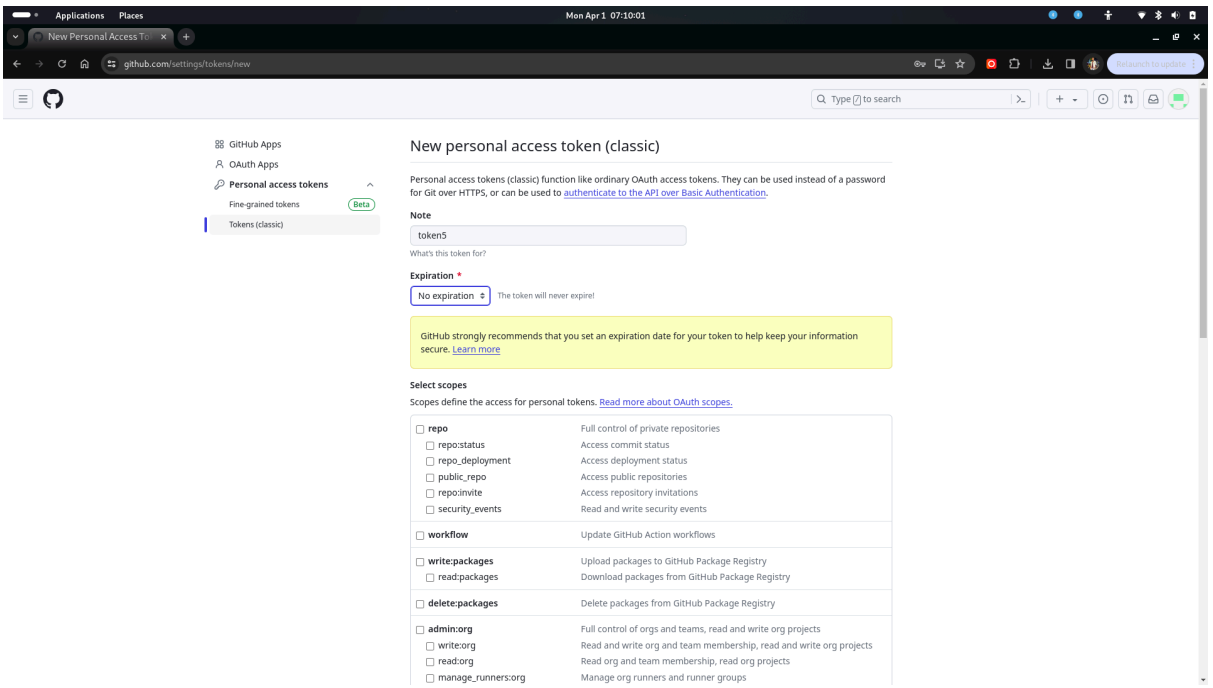


8. Click on "**Generate new token**" above, and then choose "**Generate new token (classic)**" from the dropdown list.

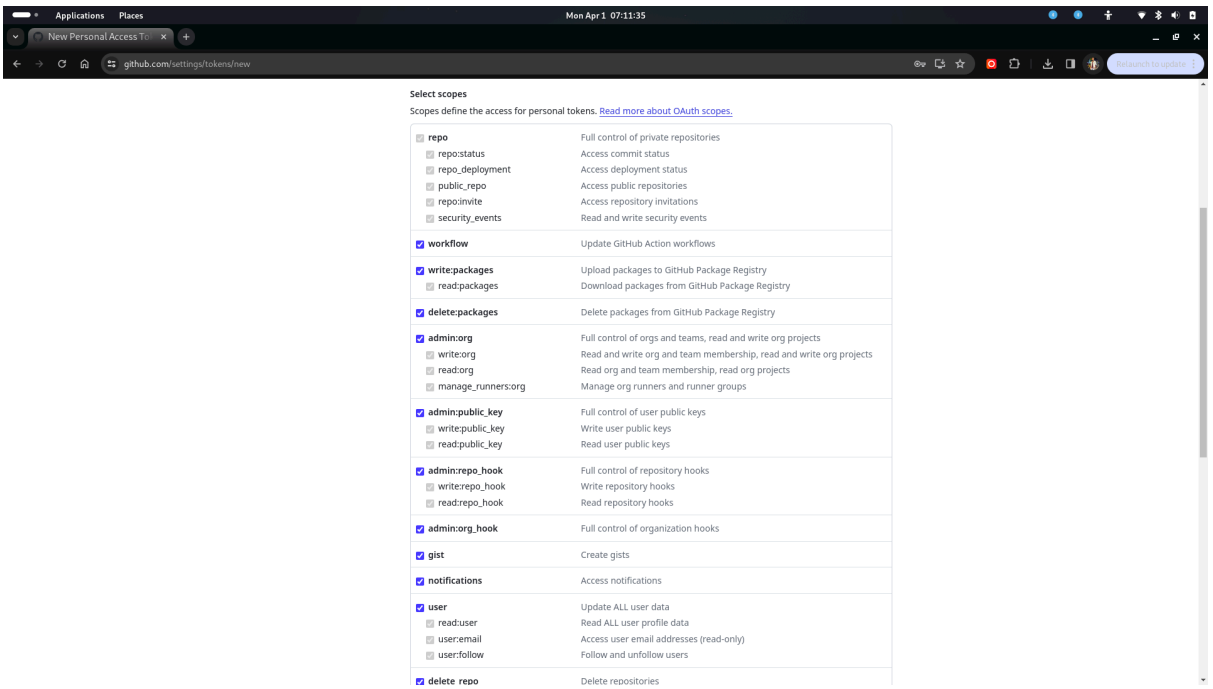


https://github.com/settings/tokens/new

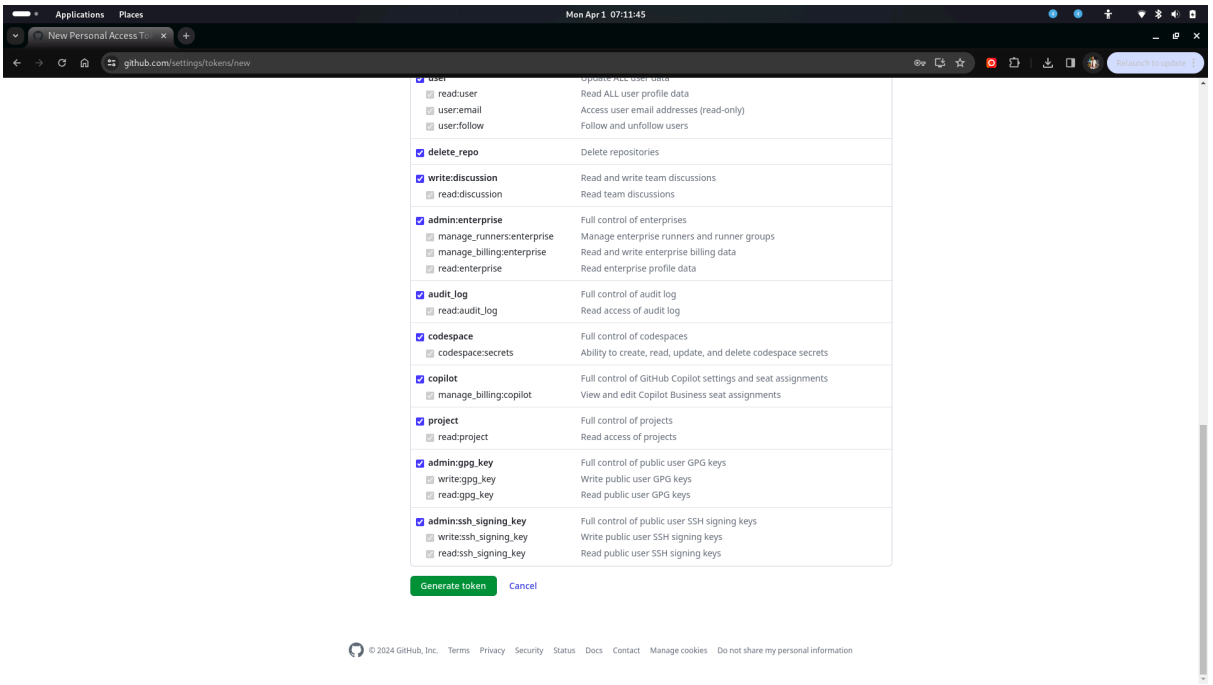
9. Enter an appropriate token name (for example, "**Token5**"), and select the option "**No expiration**" instead of choosing any time slot.



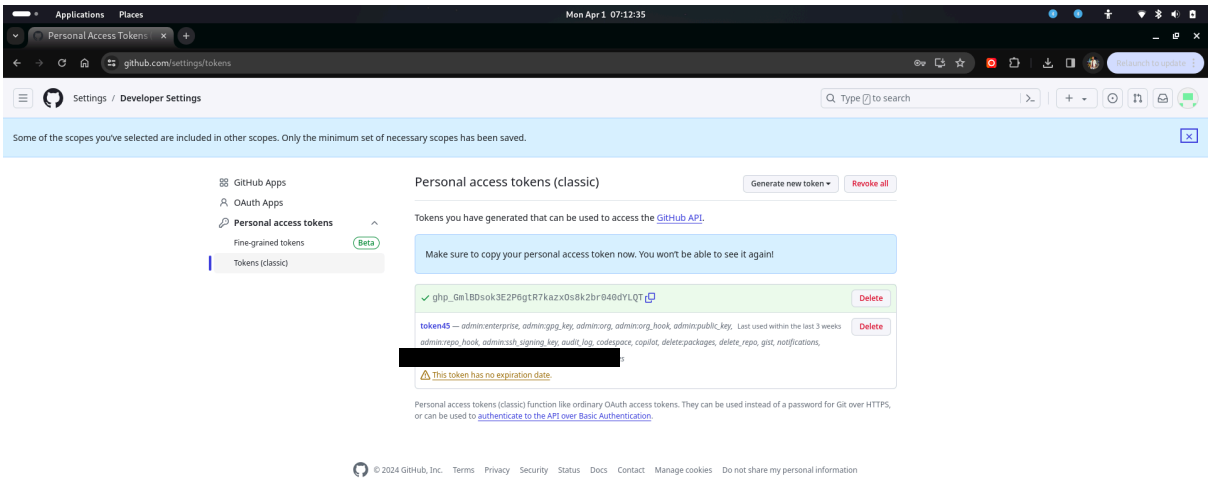
10. Ensure to check all the outer checkboxes provided below, ensuring that none is left unchecked.



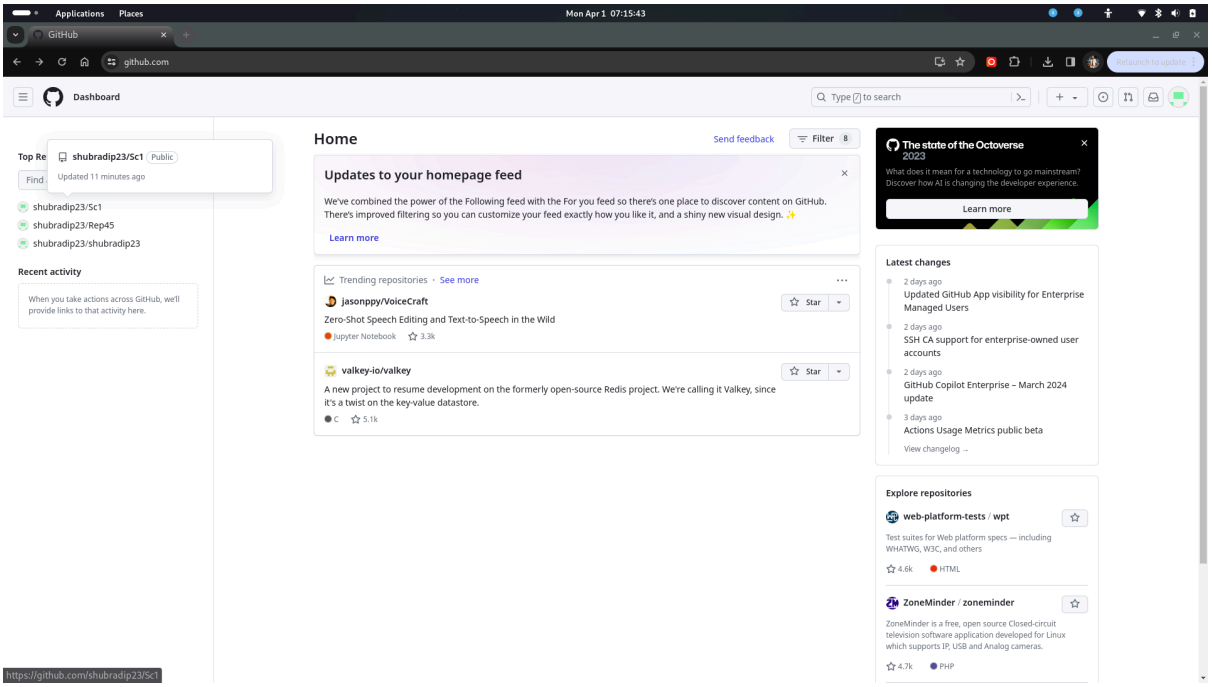
11. Continue checking all the outer checkboxes until the end, ensuring thorough coverage. Finally, click on "**Generate token**". If there is any redundancy in the token name, an error will be displayed at the top.



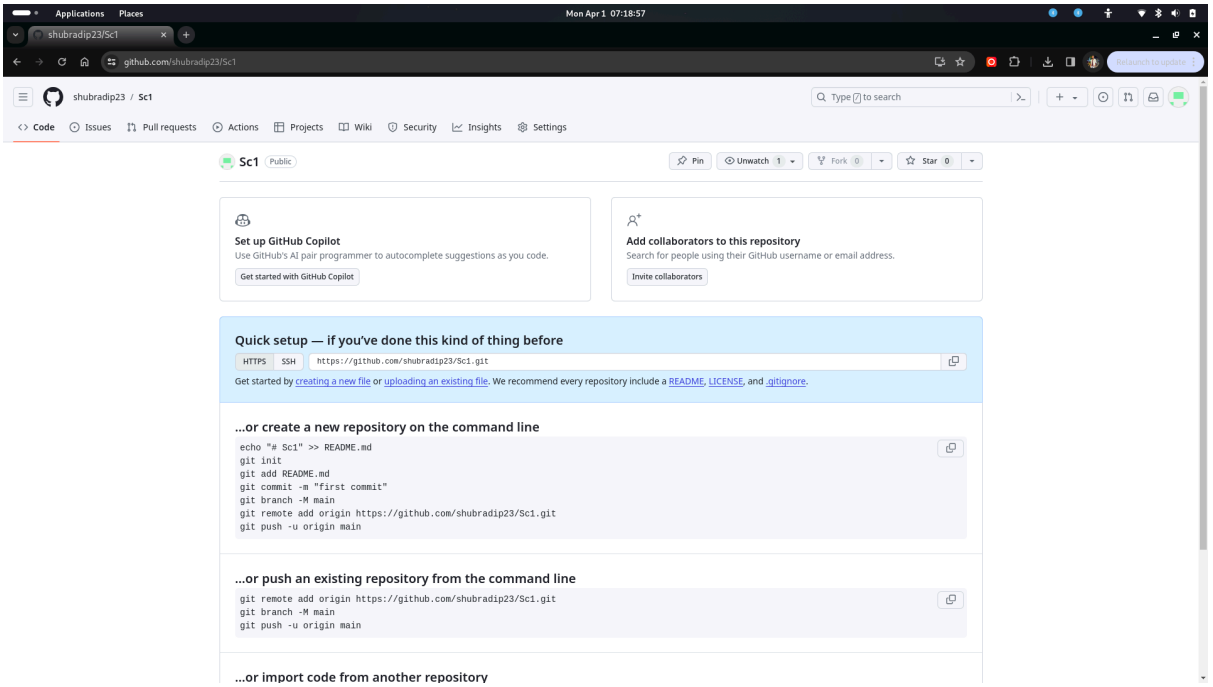
12. Now, copy the personal access token immediately as it won't be accessible again. It's advisable to save this token in a text document for easy access later on.



13. Return to the dashboard and select the repository, namely "**Repo1**."



14. In the subsequent window that opens up, choose "**HTTPS**" and copy the pathname for future reference.



15. Prior to proceeding, navigate to the "**Control Panel**". Then, select "**User Account**" and proceed to "**Manage Windows Credentials**". If there's any existing GitHub account opened, remove it.

16. Right-click on the "HTML" folder, which contains the project files to be uploaded to GitHub, and select "Open with Git Bash here."

17. In the terminal, enter the following commands:

->git init

(This command creates a new Git repository)

->dir

(Displays directories and files and directories stored on disk.)

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/html]
└─$ dir
About.html  Home.html  index.html
```

->git add .

(It adds the current content of existing paths as a whole.)

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/html]
└─$ git add .
```

->git status

(Displays the state of working directory.)

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/html]
└─$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   About.html
        new file:   Home.html
        new file:   index.html
```

->git commit -m “done”

(Take all of the changes that have been made locally and push them up to a remote repository.)

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/html]
└─$ git commit -m "done"
[master (root-commit) a2e4426] done
 Committer: Shubradip Chowdhury <shubradip@Shubradip.net>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 3 files changed, 37 insertions(+)
 create mode 100644 About.html
 create mode 100644 Home.html
 create mode 100644 index.html
```

->git remote add origin <the_repository_path_name>

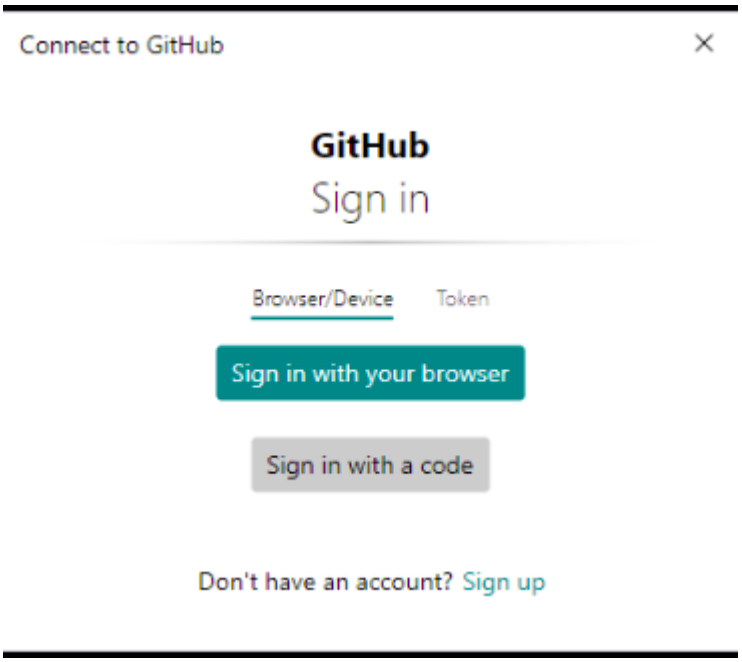
(**git remote add** is to add a short name, such as origin, which is like an alias to a URL. And **origin** is the usual path of where the remote repository points to.

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/html]
└─$ git remote add origin https://github.com/shubradip23/Sc1.git
```

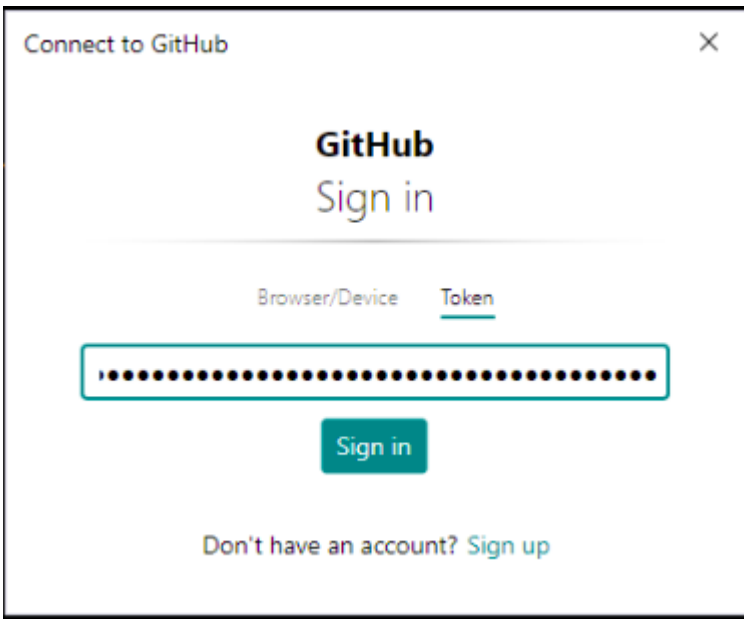
->git push -u origin master

(Push the commits in the local branch named **master** to the remote named **origin**. Once this is executed, all the documents that is last synchronized with **origin** will be sent to the remote repository and other people will be able to see them there.)Next, a "**Connect to GitHub**" window will appear.

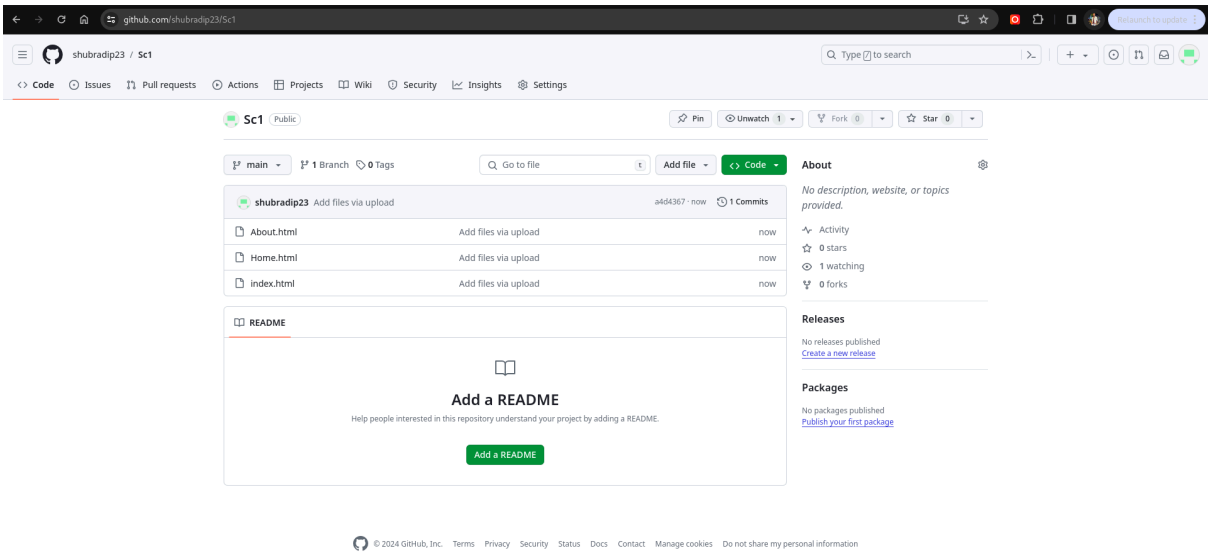
```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/html]
└─$ git push -u origin master
```



18. Now, utilize the generated token and paste it into the designated field to sign in to GitHub.



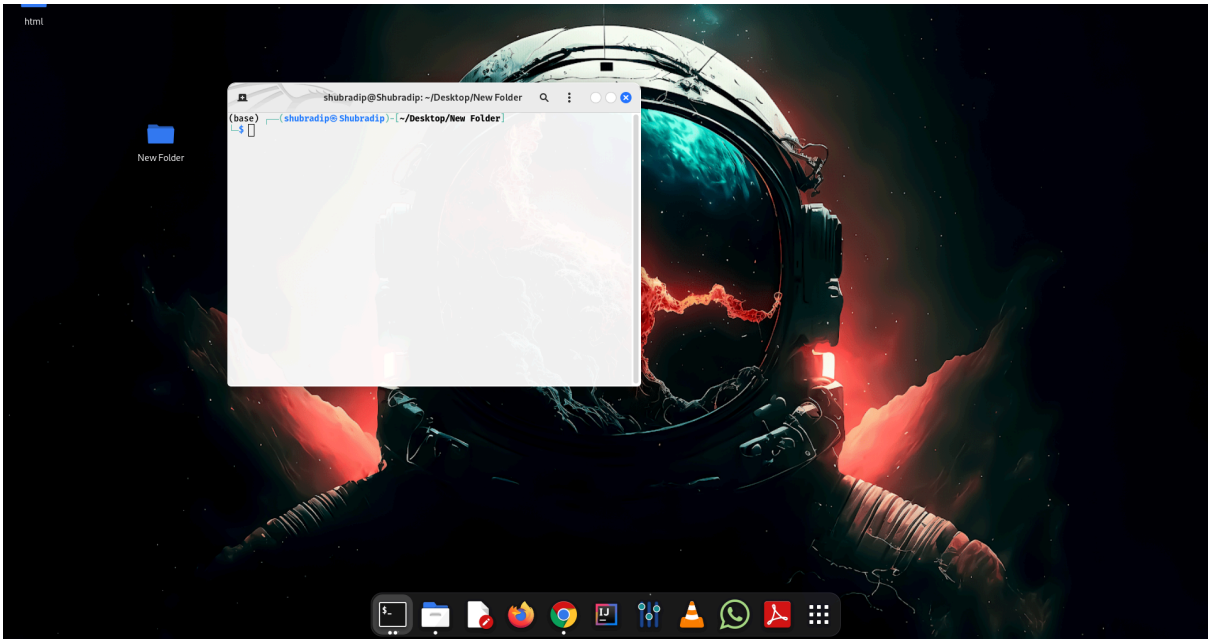
19. GitHub will open up. Now, click on "**Code**" in the bar located below the repository name.Now, you can observe that the files have been successfully deployed.



Now, deploying the project from GitHub to local machine, the steps are as follows:-

1. Create a new folder on your desktop.

2. Right-click on the folder and select "**Open with Git Bash**".



3. In the terminal start with the command,

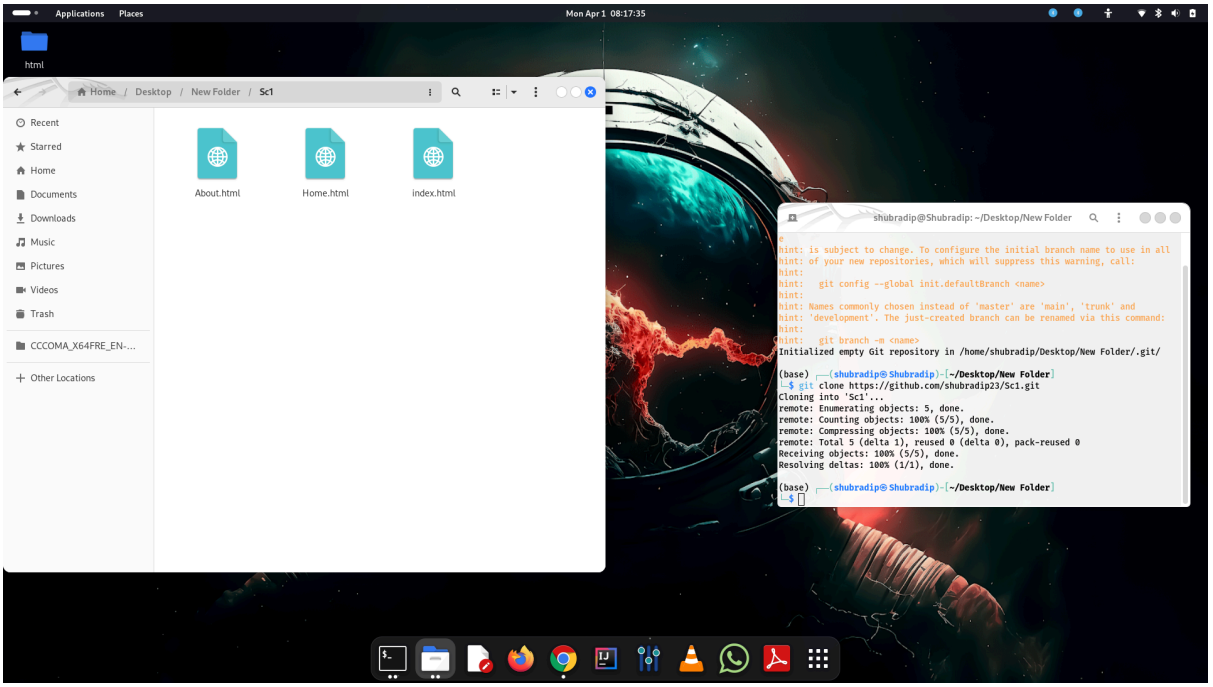
->git init

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/New Folder]
└─$ git init
hint: Using 'master' as the name for the initial branch. This default branch nam
e
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/shubradip/Desktop/New Folder/.git/
```

->git clone <Repository_Path>

```
(base) ┌──(shubradip㉿Shubradip)-[~/Desktop/New Folder]
└─$ git clone https://github.com/shubradip23/Sc1.git
Cloning into 'Sc1'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
Resolving deltas: 100% (1/1), done.
```

4. The files have been successfully cloned. Hence, the deployment of the project from GitHub to the local machine has been completed.



OBJ