

# ASSIGNMENT 8

**Problem Statement:** Deploy a project from local machine to Github and vice versa.

## CREATE NEW REPOSITORY

1. Sign in to your Github account.



Sign in to GitHub

Username or email address



Password

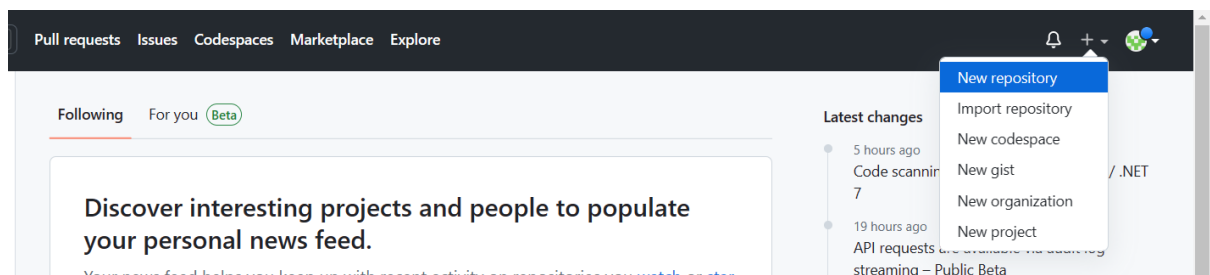
[Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

[Terms](#) [Privacy](#) [Security](#) [Contact GitHub](#)

2. Click on the '+' sign in the top right then click on **New repository**.

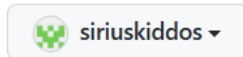


3. Give a **Repository name**. Next, select **Private** and click on **Create repository**.

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner \*



Repository name \*




Great repository names are short and memorable. Need inspiration? How about [animated-octo-adventure?](#)

Description (optional)

☐  **Public**

Anyone on the internet can see this repository. You choose who can commit.

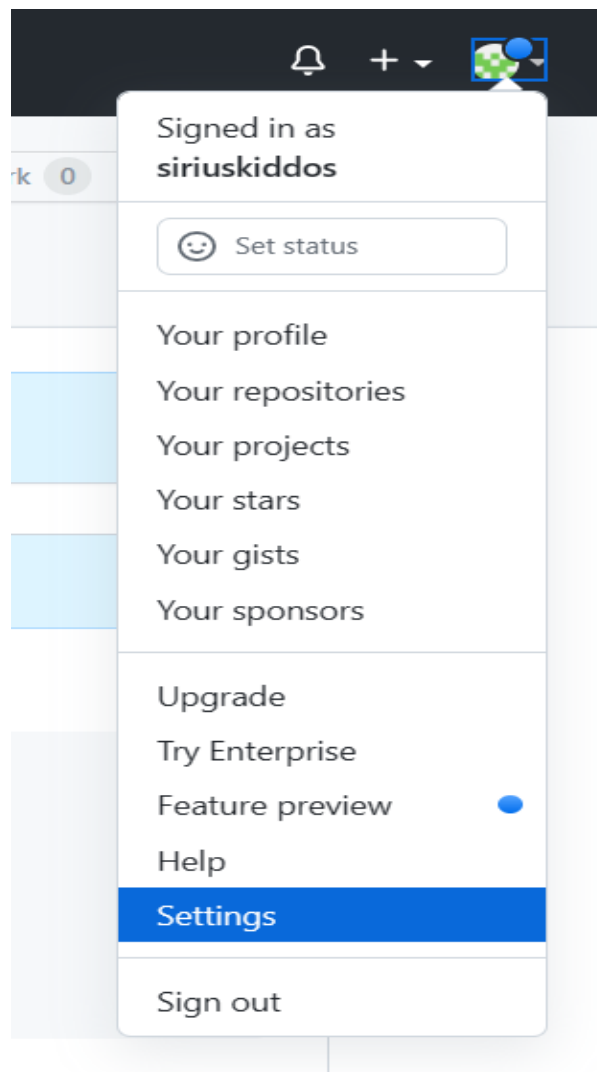
☒  **Private**

You choose who can see and commit to this repository.

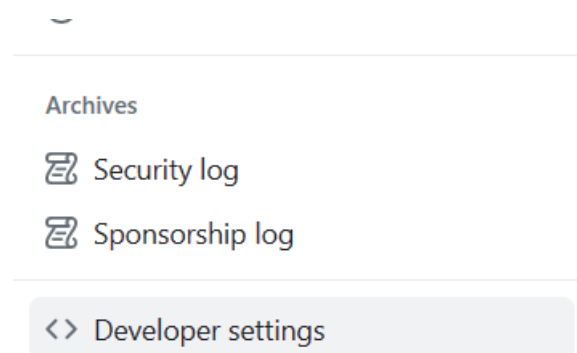
 You are creating a private repository in your personal account.

Create repository

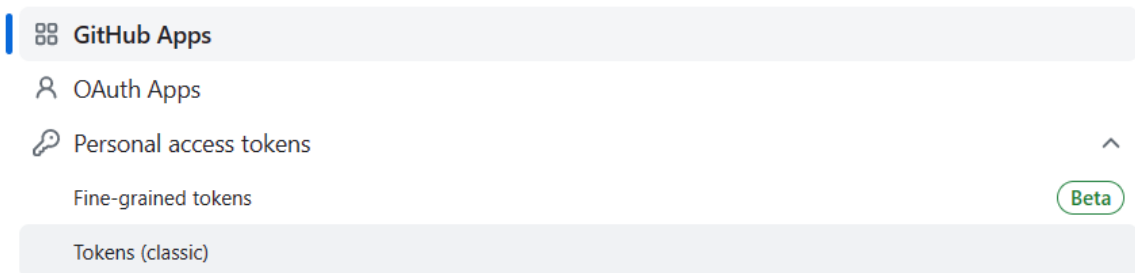
4. Click on your account at the top right, go to **Settings**.



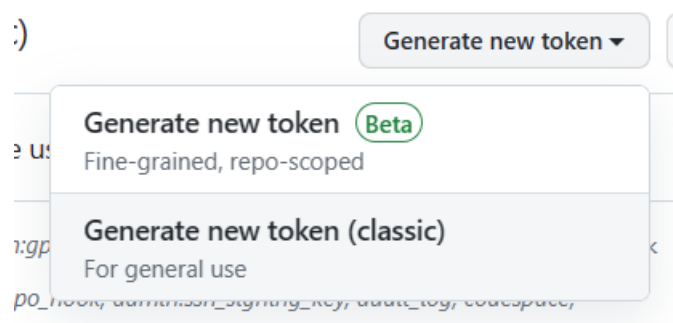
**Then, go to Developer settings.**



5. Select the drop-down menu of Personal access tokens and click **Tokens (classic)**.



6. Click **Generate new token**, then **Generate new token (classic)**.



7. Give the Token name, set **Expiration** days as 90 and select all the checkboxes. Click Generate token. Save your token.

#### Note

Token2

What's this token for?

#### Expiration \*

90 days

The token will expire on Mon, Jul 3 2023

### Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

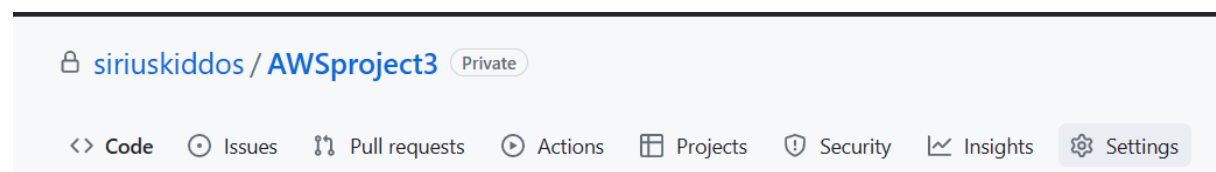
<input checked="" type="checkbox"/> <b>repo</b>	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input checked="" type="checkbox"/> <b>workflow</b>	Update GitHub Action workflows
<input checked="" type="checkbox"/> <b>write:packages</b>	Upload packages to GitHub Package Registry
<input checked="" type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input checked="" type="checkbox"/> <b>delete:packages</b>	Delete packages from GitHub Package Registry
<input checked="" type="checkbox"/> <b>admin:org</b>	Full control of orgs and teams, read and write org projects
<input checked="" type="checkbox"/> write:org	Read and write org and team membership, read and write org projects
<input checked="" type="checkbox"/> read:org	Read org and team membership, read org projects
<input checked="" type="checkbox"/> manage_runners:org	Manage org runners and runner groups
<input checked="" type="checkbox"/> <b>admin:public_key</b>	Full control of user public keys
<input checked="" type="checkbox"/> write:public_key	Write user public keys
<input checked="" type="checkbox"/> read:public_key	Read user public keys
<input checked="" type="checkbox"/> <b>admin:repo_hook</b>	Full control of repository hooks


Continue...

Generate token


Cancel

- Go to the newly created repository and then to repository **Settings**. Click **Collaborators**. Then click **Add people** and invite people by searching.





 General

Access

 Collaborators

Code and automation

 Actions 

## Who has access

PRIVATE

Only those with write access can push to the repository

[Manage access](#)

## Manage access



You haven't invited any collaborators yet


Add people

## Manage access

Add people

☐ Select all Type ▾

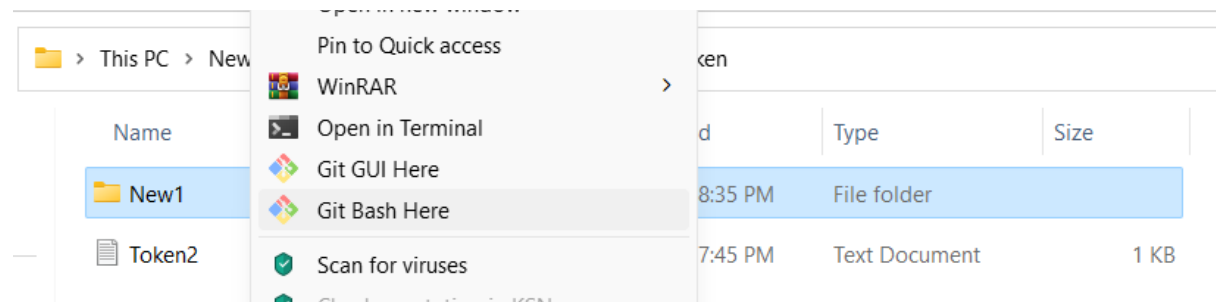
☐

 **Soham-2002**  
Collaborator

Remove

## DEPLOYING A PROJECT FROM LOCAL MACHINE TO GITHUB.

1. Open the required HTML folder with Git Bash.



2. Type and execute the following commands one by one.

- a. git init

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1
$ git init
Initialized empty Git repository in D:/6th Sem/AWS/Ass8_Token/New1/.git/
```

- b. ls

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ ls
Home.html  Next.html
```

- c. git status

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git status
On branch master

No commits yet

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

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

- d. git add .

- e. git status

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git add .

user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git status
On branch master

No commits yet

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

user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ |
```

f. `git config --global user.email sohailhaque1234@gmail.com`

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git config --global user.email "sohailhaque1234@gmail.com"
```

g. `git config --global user.name "siriuskiddos"`

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git config --global user.name "siriuskiddos"
```

h. `git commit -m "Html files committed"`

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git commit -m "done 1"
[master (root-commit) aea43d6] done 1
2 files changed, 12 insertions(+)
create mode 100644 Home.html
create mode 100644 Next.html
```

i. `git remote add origin https://github.com/siriuskiddos/AWSproject3.git`

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git remote add origin https://github.com/siriuskiddos/AWSproject3.git
```

j. `git push -u origin master`

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New1 (master)
$ git push -u origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 470 bytes | 470.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/siriuskiddos/AWSproject3.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Go to your repository in Github and if it is already open, then refresh.

The screenshot shows the GitHub interface for a repository named 'AWSproject3' by user 'siriuskiddos'. The repository is marked as 'Private'. The navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. Below the navigation bar, there are buttons for 'Go to file', 'Add file', and a 'Code' dropdown menu. The main content area shows the commit history for the 'master' branch. The latest commit is titled 'siriuskiddos done 1' with commit hash 'aea43d6', made 2 minutes ago. It lists two files: 'Home.html' and 'Next.html', both committed 'done 1' 2 minutes ago. At the bottom, there is a prompt to 'Add a README' to help people understand the project.

Repository: [siriuskiddos / AWSproject3](#) (Private) Unwatch

Navigation: [Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)

Branch: [master](#) 1 branch 0 tags Go to file Add file Code

Commit	Files	Time
<a href="#">siriuskiddos done 1</a> (aea43d6) 2 minutes ago 1 commit	<a href="#">Home.html</a> done 1 2 minutes ago <a href="#">Next.html</a> done 1 2 minutes ago	

Help people interested in this repository understand your project by adding a README. Add a README

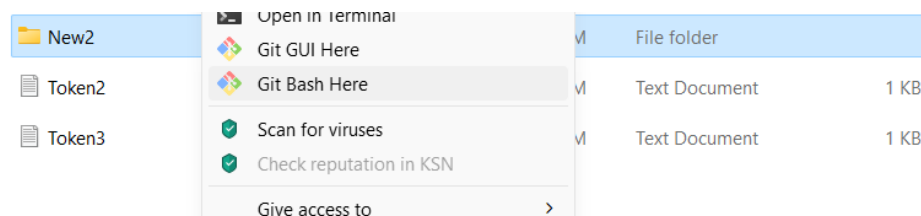
Releases: No releases yet Create



## CLONING A GIVEN REPOSITORY FROM GITHUB TO LOCAL MACHINE AND UPLOADING THAT TO A NEWLY CREATED REPOSITORY

### *Cloning from Github to local machine*

1. Create new directory and open Gitbash in this folder.



2. Type and execute the following commands

- a. git init

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2
$ git init
Initialized empty Git repository in D:/6th Sem/AWS/Ass8_Token/New2/.git/
```

- b. git clone <https://github.com/sudip7407/New-Repo1.git>

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2 (master)
$ git clone https://github.com/sudip7407/New-Repo1.git
Cloning into 'New-Repo1'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 15 (delta 6), reused 4 (delta 0), pack-reused 0
Receiving objects: 100% (15/15), done.
Resolving deltas: 100% (6/6), done.
```

### *Uploading the cloned project from local machine to my repository*

1. Create a new repository in Github
2. Open Gitbash in the cloned folder (remove the already existing .git hidden folder in the folder containing the cloned project.)

Type and execute the following commands

- a. git init

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2/New-Repo1 (master)
$ git init
Initialized empty Git repository in D:/6th Sem/AWS/Ass8_Token/New2/New-Repo1/.git/
```

- b. git add .

```
user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2/New-Repo1 (master)
$ git add .
```

- c. git commit -m "Committed"

```

user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2/New-Repo1 (master)
$ git commit -m "done 2"
[master (root-commit) 3f01179] done 2
4 files changed, 53 insertions(+)
create mode 100644 .gitignore
create mode 100644 New Text Document.txt
create mode 100644 index.js
create mode 100644 package.json

```

- d. `git remote add origin` <https://github.com/siriuskiddos/AWSproject4.git>

```

user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2/New-Repo1 (master)
$ git remote add origin https://github.com/siriuskiddos/AWSproject4.git

```

- e. `git push -u origin master`

```

user@LAPTOP-EN4MDAG1 MINGW64 /d/6th Sem/AWS/Ass8_Token/New2/New-Repo1 (master)
$ git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 825 bytes | 825.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/siriuskiddos/AWSproject4.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

```

Open the repository and refresh it if it is already opened.

The screenshot shows the GitHub repository page for 'siriuskiddos / AWSproject4'. The repository is private. The main branch is 'master', which has 1 branch and 0 tags. The commit history shows a single commit 'siriuskiddos done 2' with the hash '3f01179' and a timestamp of '6 minutes ago'. The commit message is 'done 2'. The files listed in the commit are: '.gitignore', 'New Text Document.txt', 'index.js', and 'package.json'. A button 'Add a README' is visible at the bottom of the commit list.

*HENCE THE CLONED PROJECT IS SUCCESSFULLY DEPLOYED*