

## Experiment -1.1

### Install Git and creating repository

**Student Name:** Adarsh Kumar Singh

**UID:** 22BDO10053

**Branch:** CSE(DEVOPS)

**Section/Group:** 22BCD-1/B

**Semester:** 4th

**Date of Performance:** 15-01-24

**Subject Name:** Git and GitHub

**Subject Code:** 22CSH-293

**1. Aim/Overview of the practical:** Install Git and creating repository.

**2. Task to be done:** Download Git for Windows. And, to make repositories.

**3. Steps for experiment:-**

- 1) Go to the official Git website: <https://git-scm.com/downloads>.
- 2) Click the download link for Windows and allow the download to complete.



3) Browse to the download location (or use the download shortcut in your browser). Double-click the file to extract and launch the installer.

4) Allow the app to make changes to your device by clicking **Yes** on the User Account Control dialog that opens.

- 
- 5) The installer will ask you for an installation location. Leave the default, unless you have reason to change it, and click **Next**.
  - 6) The installer will offer to create a start menu folder. Simply click **Next**.
  - 7) Select a text editor you'd like to use with Git. Use the drop-down menu to select Notepad++ (or whichever text editor you prefer) and click **Next**.
  - 8) The next step allows you to choose a different name for your initial branch. The default is 'master.' Unless you're working in a team that requires a different name, leave the default option and click **Next**.
  - 9) Choose the [terminal emulator](#) you want to use. The default MinTTY is recommended, for its features. Click **Next**.
  - 10) The installer now asks what the **git pull** command should do. The default option is recommended unless you specifically need to change its behavior. Click **Next** to continue with the installation.
  - 11) Next you should choose which credential helper to use. Git uses credential helpers to fetch or save credentials. Leave the default option as it is the most stable one, and click **Next**.
  - 12) Depending on the version of Git you're installing, it may offer to install experimental features. At the time this article was written, the options to include support for pseudo controls and a built-in file system monitor were offered. Unless you are feeling adventurous, leave them unchecked and click **Install**.
  - 13) Once the installation is complete, tick the boxes to view the Release Notes or Launch Git Bash, then click **Finish**.



## How to Launch Git in Windows

- Git has two modes of use – a **bash scripting shell** (or command line) and a **graphical user interface (GUI)**. To launch **Git GUI** open the **Windows Start** menu, type **git gui** and press **Enter** (or click the application icon). **Connecting to a Remote Repository**
- You need a GitHub username and password for this next step.

## Create a Test Directory

- Create a new test directory (folder) by entering the following:

**mkdir first\_git**

- Change your location to the newly

created directory: **cd first\_git**

**Note:** If you already have a GitHub repository, use the name of that project instead of **first\_git**

```
MINGW64:/c/Users/adars/first_git  
  
adars@ASUS MINGW64 ~  
$ mkdir first_git  
  
adars@ASUS MINGW64 ~  
$ cd first_git
```

## Configure GitHub Credentials

Configure your local Git installation to use your GitHub credentials by

entering the following: `git config --global user.name "github_username"`

`git config --global user.email "email_address"`

**Note:** Replace **github\_username** and **email\_address** with your GitHub credentials.

We can also see the list of configurations by using the command `git config – list`.

```
adars@ASUS MINGW64 ~/first_git  
$ git config --global user.name "adarshkrsingh07"  
  
adars@ASUS MINGW64 ~/first_git  
$ git config --global user.email "adarshkrdixit@gmail.com"
```

```
adars@ASUS MINGW64 ~/first_git
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
core.fsmonitor=true
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
core.editor="C:\Users\adars\AppData\Local\Programs\Microsoft VS Code\bin\code" -
-wait
user.name=adarshkrsingh07
user.email=adarshkrdixit@gmail.com
```

## Clone a GitHub Repository

Go to your repository on GitHub. In the top right above the list of files, open the **Clone or download** drop-down menu. Copy the **URL for cloning over HTTPS**.

Switch to your PowerShell window, and enter the following:

```
git clone repository_url
```

## List Remote Repositories

Your working directory should now have a copy of the repository from GitHub. It should contain a directory with the name of the project. Change to the directory:

```
cd first_git or 22BCD-1
```

Once you're in the sub-directory, list the remote

```
repositories: git remote -v
```

### Enter the details in the current file

You can enter the desired details into your current file by the following commands:

Cat > first\_git

Enter the details in  
the filePress

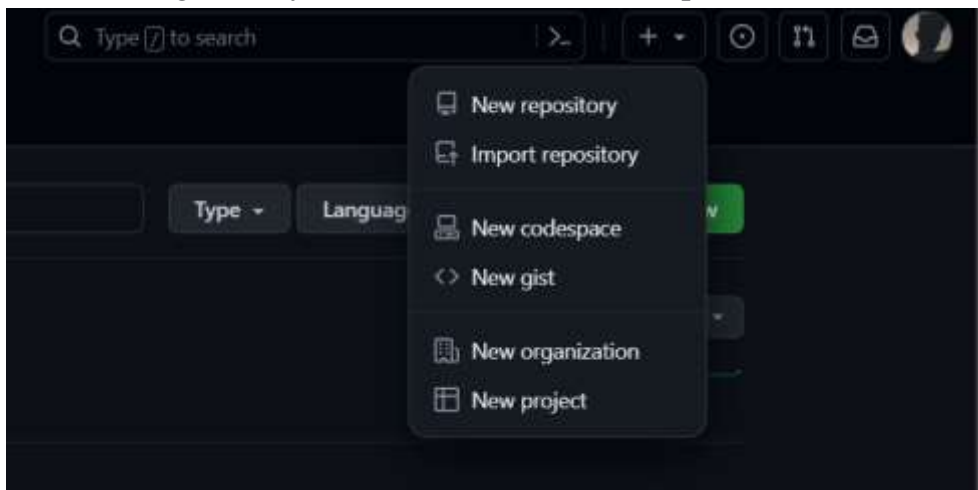
Cntrl+D

Then, you can see the entered details through the command :- Cat first\_git

```
adars@ASUS MINGW64 ~/first_git
$ cat > first_git
hello, this is my first git file
.
adars@ASUS MINGW64 ~/first_git
$ cat first_git
hello, this is my first git file
.
adars@ASUS MINGW64 ~/first_git
$
```

### Creating Repository on GitHub

1. After successful login into your account. Click on the option (+) to add new repository to your account.




2. After clicking **new repository** option, we will have to initialize some things like, **naming our project**, choosing the **visibility** etc. After performing these steps click **Create Repository** button.

# Create a new repository

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

*Required fields are marked with an asterisk (\*).*

Owner \*

 adarshkrsingh07 ▾

Repository name \*

/ first\_git

✔ first\_git is available.

Great repository names are short and memorable. Need inspiration? How about **miniature-parakeet** ?

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.

Initialize this repository with:



**Add a README file**

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

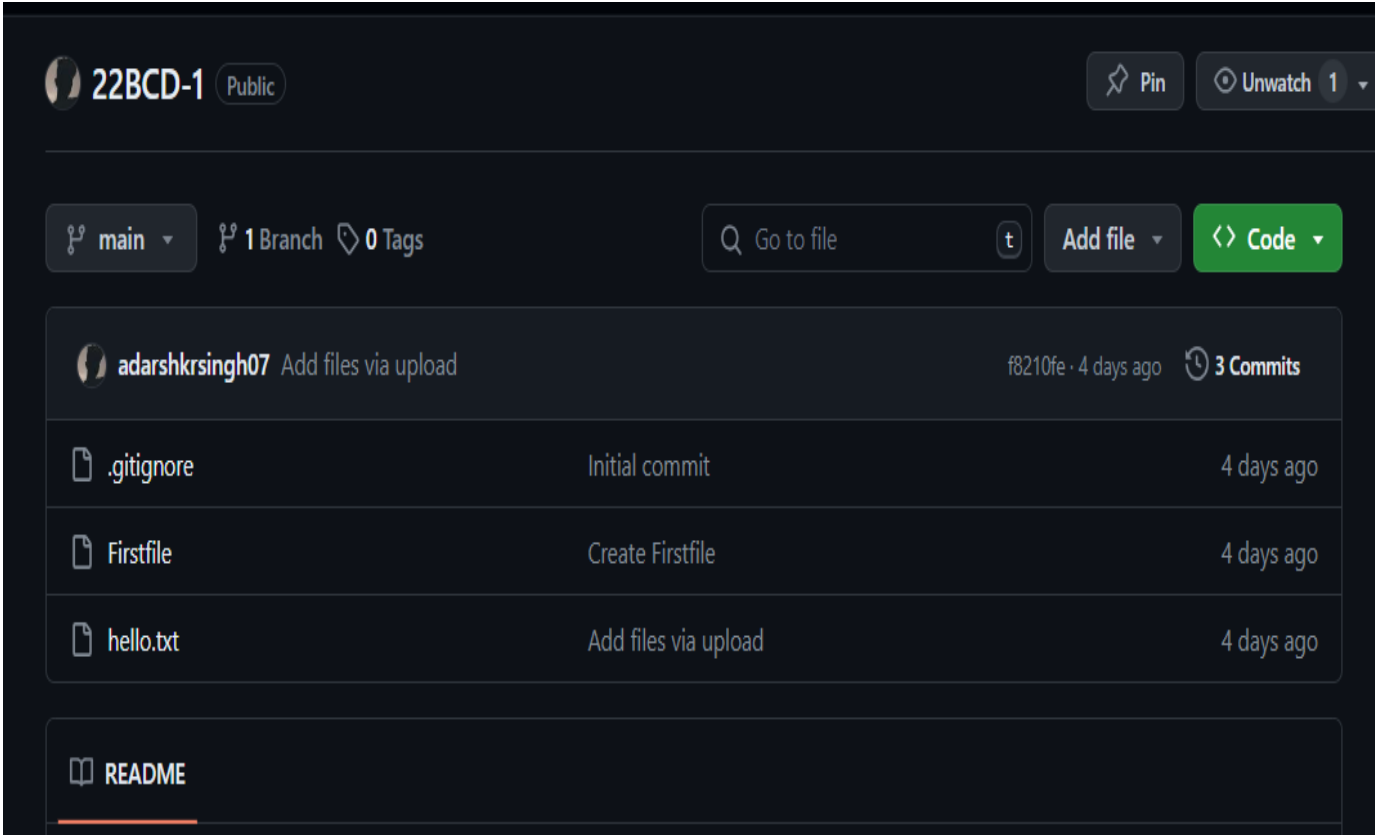
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾



3. After clicking the button, we will be directed to below page. Right now the only file we have is a readme file.
4. Now click on the “Upload files” button. Do some needed steps. Now we can see all the files in our github.



22BCD-1 Public

Pin Unwatch 1

main 1 Branch 0 Tags

Go to file t Add file <> Code

adarshkrsingh07 Add files via upload f8210fe · 4 days ago 3 Commits

.gitignore	Initial commit	4 days ago
Firstfile	Create Firstfile	4 days ago
hello.txt	Add files via upload	4 days ago

README



We have successfully created a repository and applied some commands on that.

**Learning outcomes (What I have learnt):**

1. Learnt about GitHub.
2. Learnt about Git.
3. Learnt about various git commands that can be applied on Git Bash.
4. Learnt about how to create repositories.
5. Learnt about how to pull request and push source code /files etc.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			