EXPERIMENT 1:
Demonstrate version control in Git and Github using simple html code.
PROCEDURE

Step1: Downloading and Installing Git (go to https://git-scm.com/downloads)
To configure Git, we need to specify the name, email address, and branch by using the **git config - -global** command.
Step2: Open a cmd window or terminal on your computer and execute all commands.

```
git --version
git config --global user.name "Your Name"
git config --global user.email <your email
address>
git config --list
git init
git status
git add .
git commit -m "first commit"
git log --oneline
git checkout <second commit's number>
index.html
git remote add origin <repository URL>
git push -u origin master
git clone <repository URL>
```

Syntaxes of some git commands:

git config --global user.name “ [username]”

git config --global user.email [email address]
git init = to start git in your Project.

git add (file name)= adds the name of the particular file you want to commit in the staging area.

git commit –m “first commit”

git clone (repository name)= to copy an existing repository in another location to the current location.

git push (remote storage name)= to upload/push files from the local repository/storage to another storage.

git rm (filename) = to remove a file from a working repository

git branch = to check the current branch you are working on, either main or master

- Get familiar with version control, git and GitHub
- Create your own repository and project folder structure

SOURCE CODE:

While working on Git, we actively use two repositories.

- (Git)Local repository: The local repository is present on our computer and consists of all the files and folders. This Repository is used to make changes locally, review history, and commit when offline.
- (GitHub)Remote repository: The remote repository refers to the server repository that may be present anywhere. This repository is used by all the team members to exchange the changes made.

Both repositories have their own set of commands. There are separate Git Commands that work on different types of repositories.

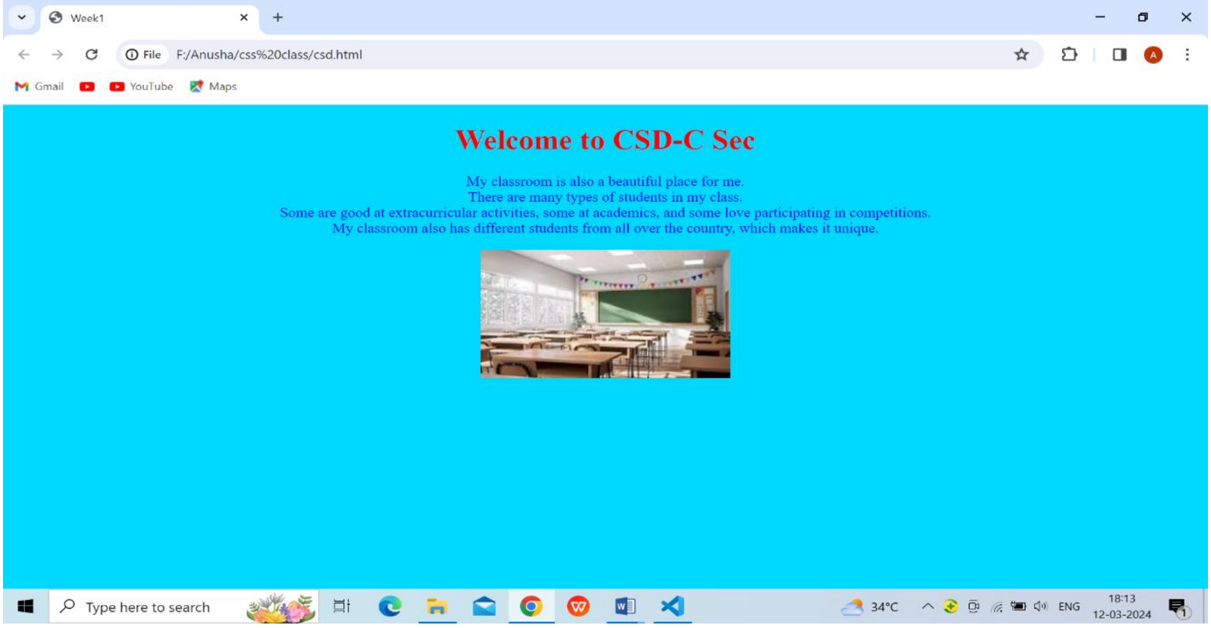
Step 1:-Create one Folder week1

Step2:-Create one HTML File index.html in week1 folder.

Code:-

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Week1</title>

</head>

<body style="background-color:rgb(0, 217, 255)">

   <center>

       <h1 style="color:red">Welcome to CSD-C Sec</h1>

       <p style="color:blue">My classroom is also a beautiful place for me.<br>
    There are many types of students in my class. <br>

    Some are good at extracurricular activities, some at academics, and some love participating in competitions.

    <br> My classroom also has different students from all over the country, which makes it unique.</p>

    <img src="class.jfif" alt="class room">

  </center>

</body>

</html>
```

Write above code in index.html. After completion of code save and run you get output like



Welcome to CSD-C Sec

My classroom is also a beautiful place for me.
There are many types of students in my class.
Some are good at extracurricular activities, some at academics, and some love participating in competitions.
My classroom also has different students from all over the country, which makes it unique.

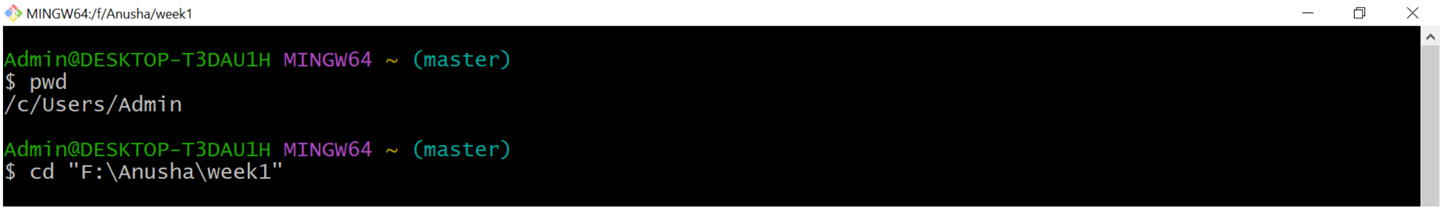
Step 3:-use cmd (or) terminal in visual studio (or) git bash

Open Git bash and check current directory use command **pwd**(print working directory)

If you are folder available in other directory use this command to change directory

Cd (directory location)

→Cd “F:\Anusha\week1”

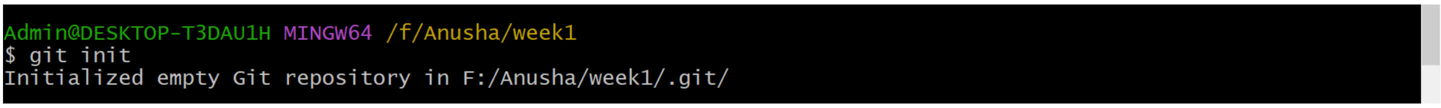


```
Admin@DESKTOP-T3DAU1H MINGW64 ~ (master)
$ pwd
/c/Users/Admin

Admin@DESKTOP-T3DAU1H MINGW64 ~ (master)
$ cd "F:\Anusha\week1"
```

Step 4:- **Git Commands: Working With Local Repositories(default branch master)**

1.git init

- The command git init is used to create an empty Git repository.
- After the git init command is used, a .git folder is created in the directory with some subdirectories. Once the repository is initialized, the process of creating other files begins.



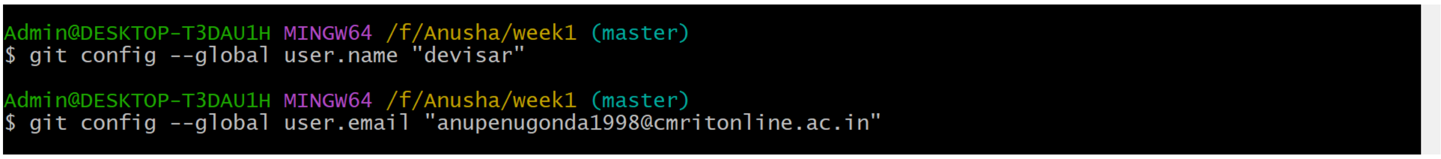
```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1
$ git init
Initialized empty Git repository in F:/Anusha/week1/.git/
```

2.git config

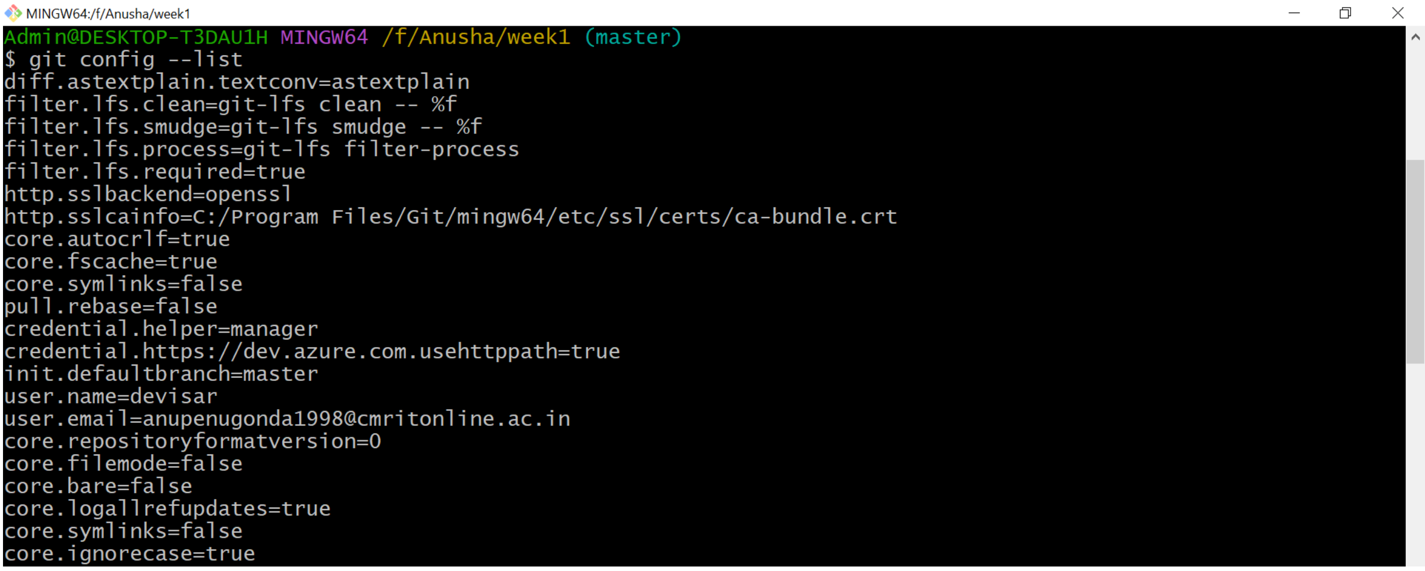
- The git config command is used initially to configure the user.name and user.email. This specifies what email id and username will be used from a local repository.
- When git config is used with --global flag, it writes the settings to all repositories on the computer.

git config –global user.name “user name”

git config –global user.email “email id”



```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git config --global user.name "devisar"

Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git config --global user.email "anupenugonda1998@cmritonline.ac.in"
```

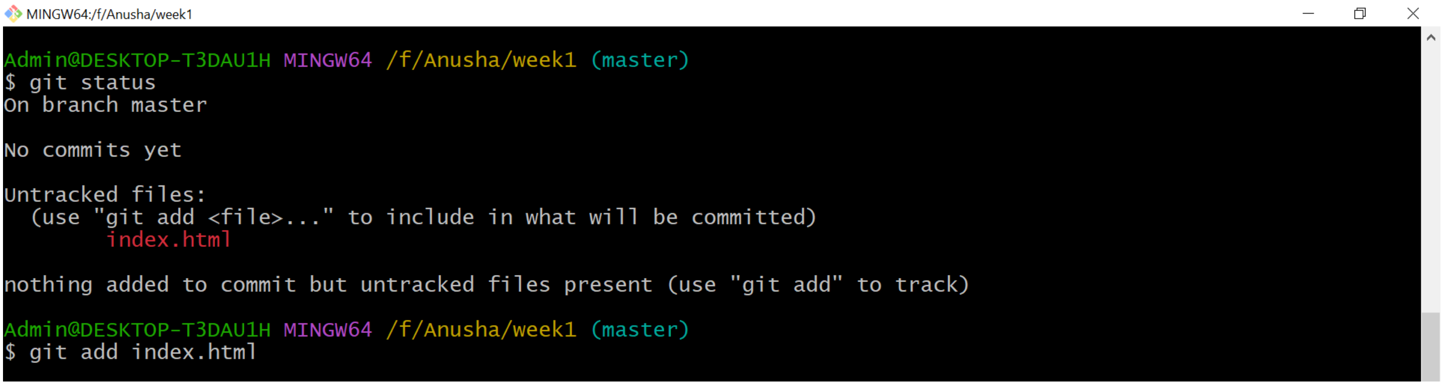


```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=devisar
user.email=anupenugonda1998@cmritonline.ac.in
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
```

3.git add

- Add command is used after checking the status of the files, to add those files to the staging area.
- Before running the commit command, "git add" is used to add any new or modified files.

git add . (or) git add file name

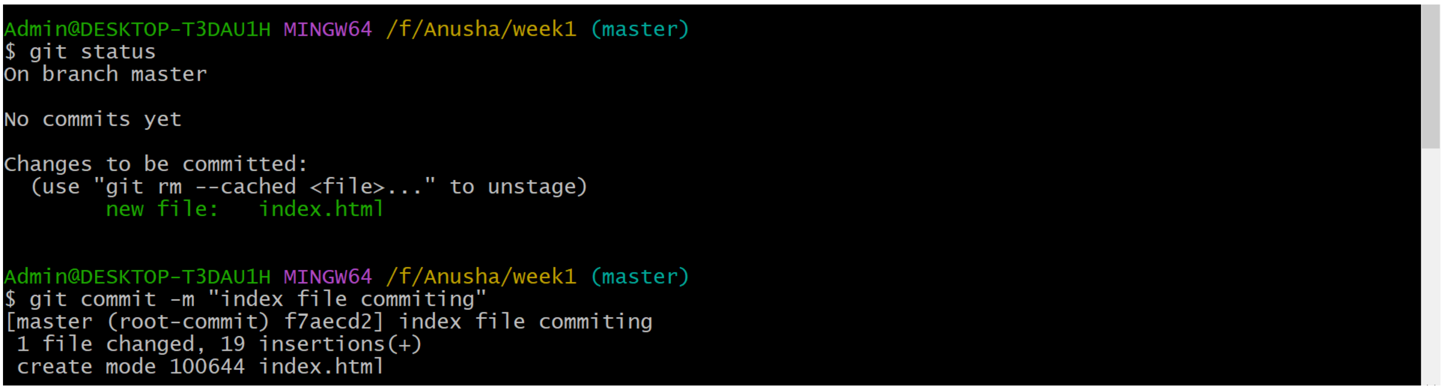


```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git status
On branch master

No commits yet

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

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

Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git add index.html
```

4.git commit

- The commit command makes sure that the changes are saved to the local repository.
- The command "git commit –m <message>" allows you to describe everyone and help them understand what has happened.

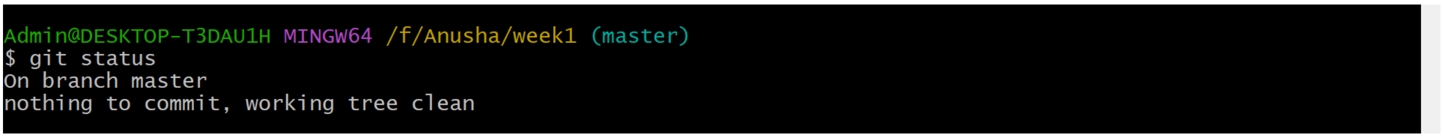
git commit -m “commit message”



```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git status
On branch master

No commits yet

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


Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git commit -m "index file commiting"
[master (root-commit) f7aecd2] index file commiting
 1 file changed, 19 insertions(+)
 create mode 100644 index.html
```

5.git status

- The git status command tells the current state of the repository.
- The command provides the current working branch. If the files are in the staging area, but not committed, it will be shown by the git status. Also, if there are no changes, it will show the message no changes to commit, working directory clean.

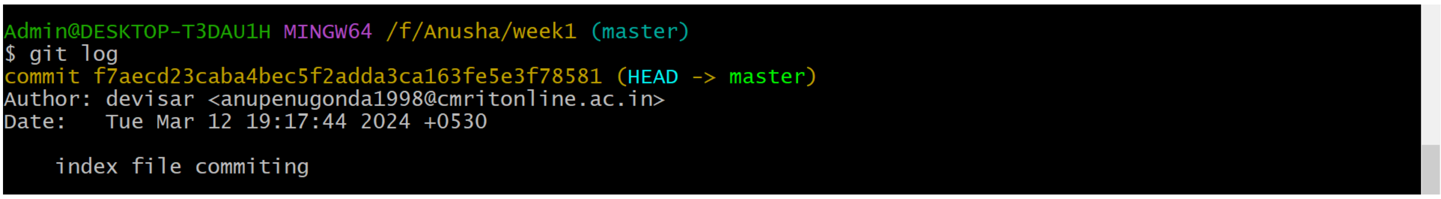


```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git status
On branch master
nothing to commit, working tree clean
```

6.git log

- The git log command shows the order of the commit history for a repository.

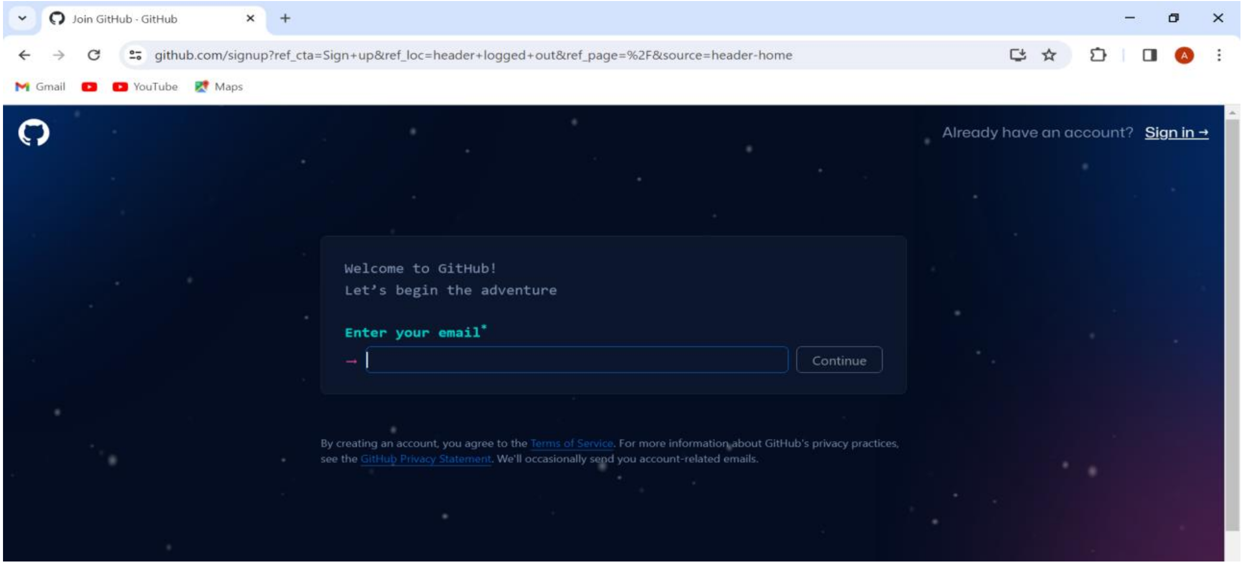
- The command helps in understanding the state of the current branch by showing the commits that lead to this state.



```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git log
commit f7aecd23caba4bec5f2adda3ca163fe5e3f78581 (HEAD -> master)
Author: devisar <anupenugonda1998@cmritonline.ac.in>
Date:   Tue Mar 12 19:17:44 2024 +0530

    index file commiting
```

Step5:-Git Commands: Working With Remote Repositories

a. First Create GitHub Account and Login



github.com/signup?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home

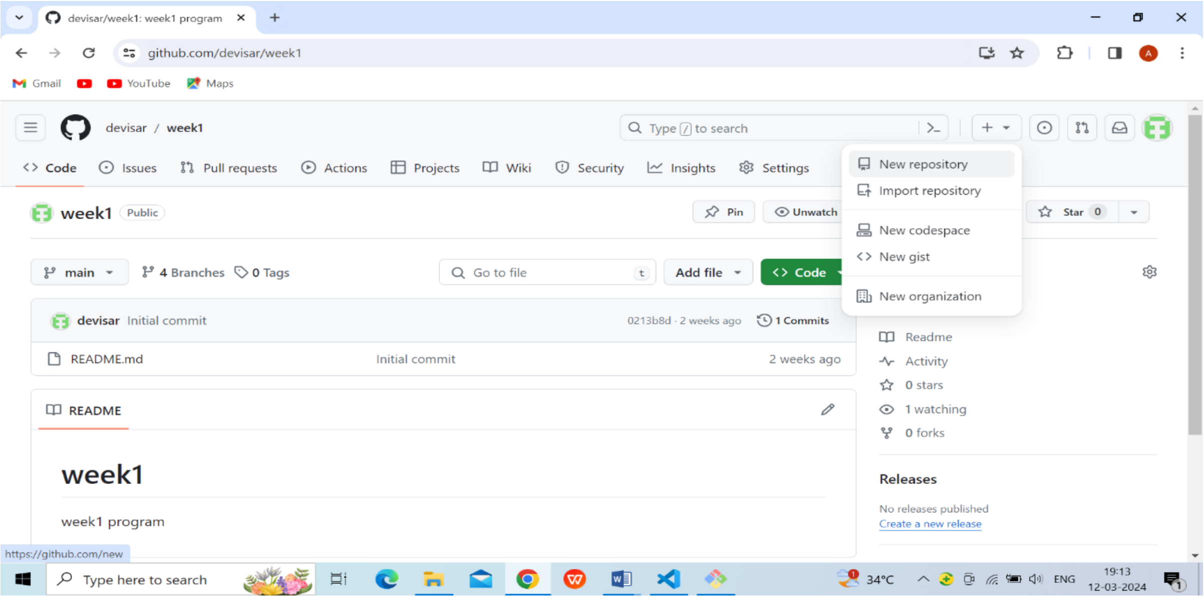
Welcome to GitHub!
Let's begin the adventure

Enter your email*

By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement. We'll occasionally send you account-related emails.

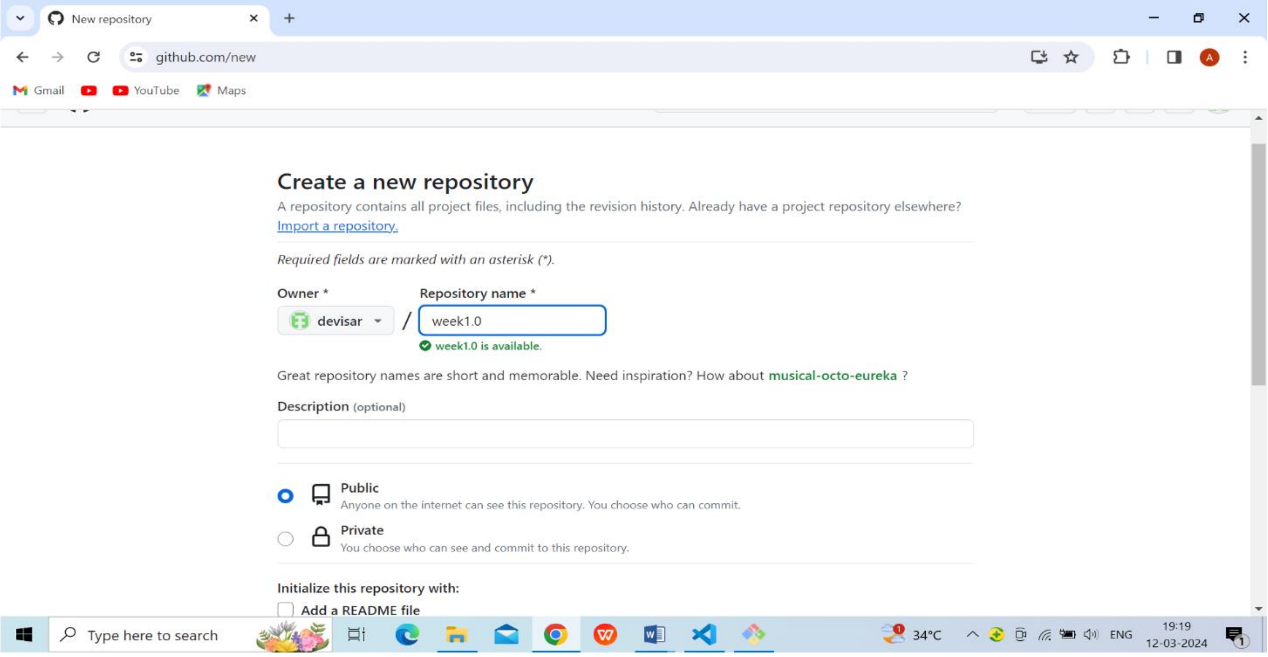
Fig.1.Github sign in and sign up page

b. Next create new Repository



github.com/devisar/week1

Fig.2.creating new repository page



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 * devisar / Repository name * week1.0

week1.0 is available.

Great repository names are short and memorable. Need inspiration? How about musical-octo-eureka ?

Description (optional)

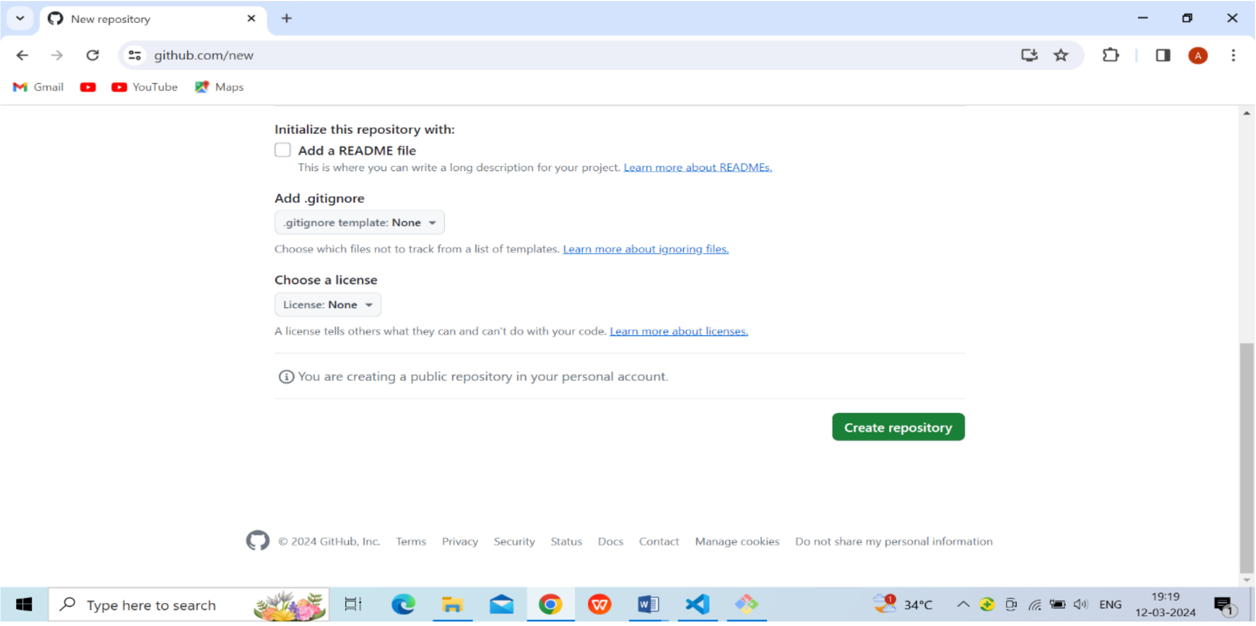
● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:

□ Add a README file

Fig.3.Enter repository name page



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

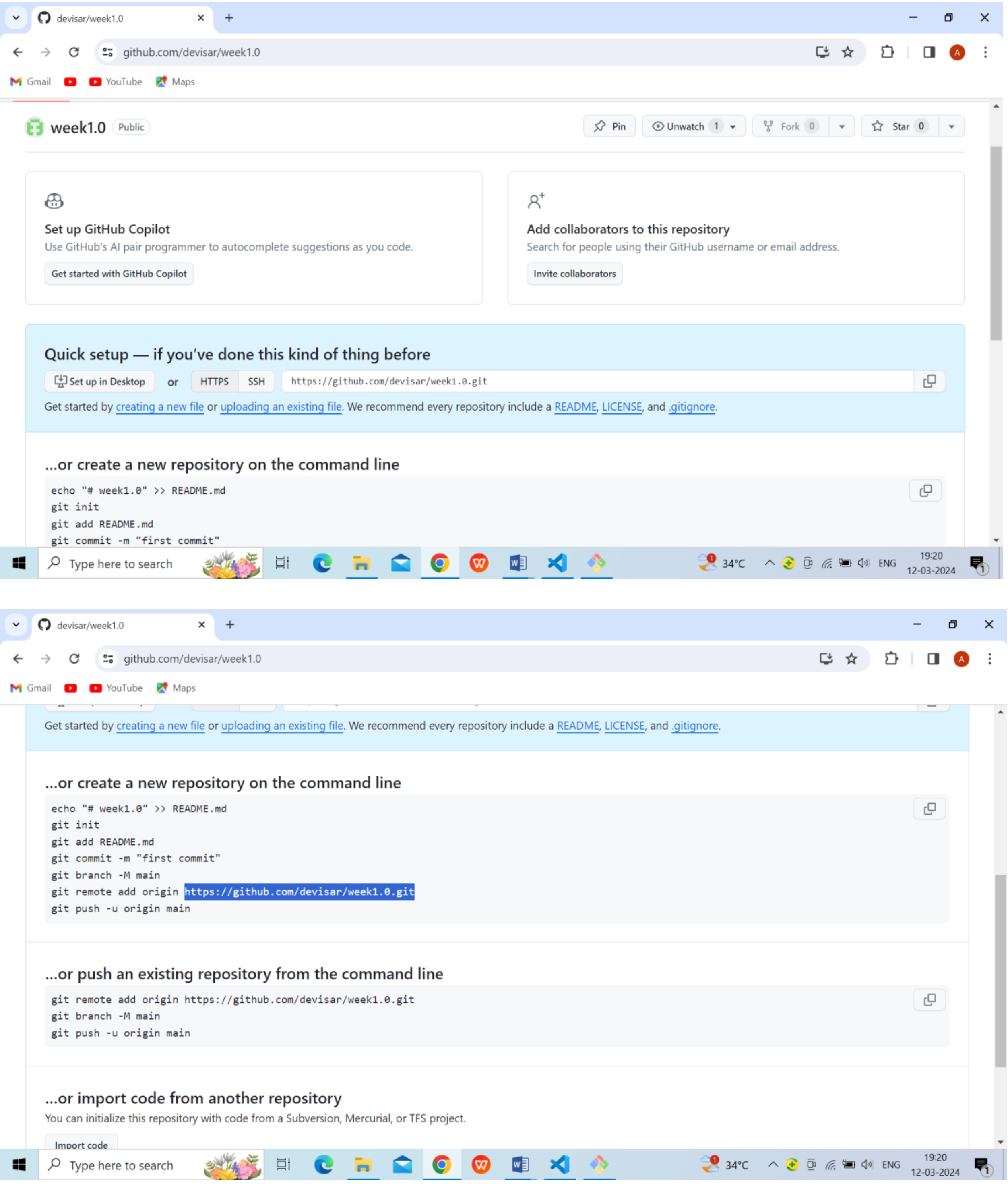
A license tells others what they can and can't do with your code. Learn more about licenses.

ⓘ You are creating a public repository in your personal account.

Create repository

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

Fig.4.Click button to create repository page



Set up GitHub Copilot

Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Get started with GitHub Copilot

Add collaborators to this repository

Search for people using their GitHub username or email address.

Invite collaborators

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/devisar/week1.0.git`

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# week1.0" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/devisar/week1.0.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/devisar/week1.0.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

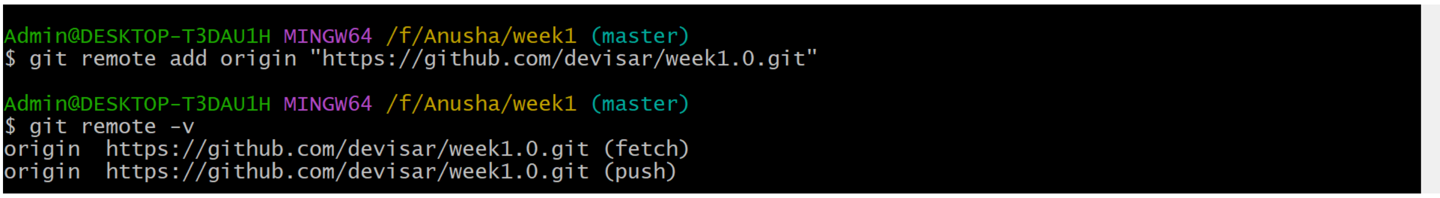
Import code

Fig.5.After opening page

7.git remote

- The git remote command is used to create, view, and delete connections to other repositories.
- The connections here are not like direct links into other repositories, but as bookmarks that serve as convenient names to be used as a reference.

Git remote add origin <address>

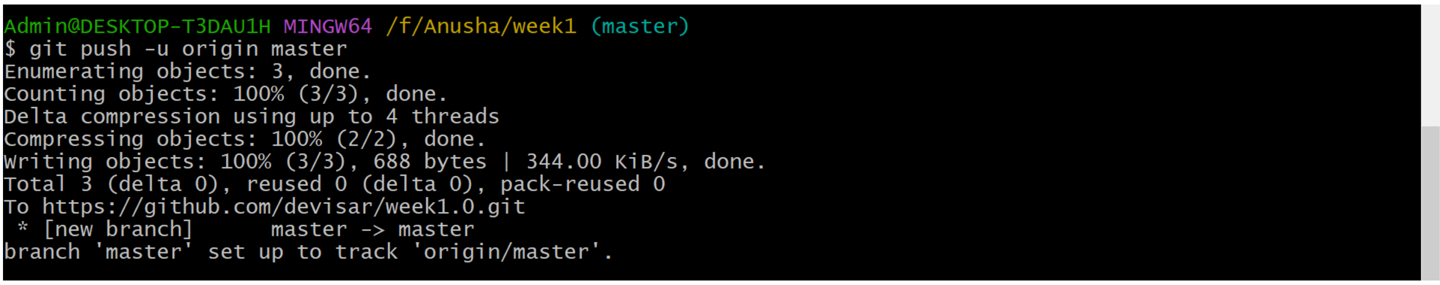


```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git remote add origin "https://github.com/devisar/week1.0.git"

Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git remote -v
origin  https://github.com/devisar/week1.0.git (fetch)
origin  https://github.com/devisar/week1.0.git (push)
```

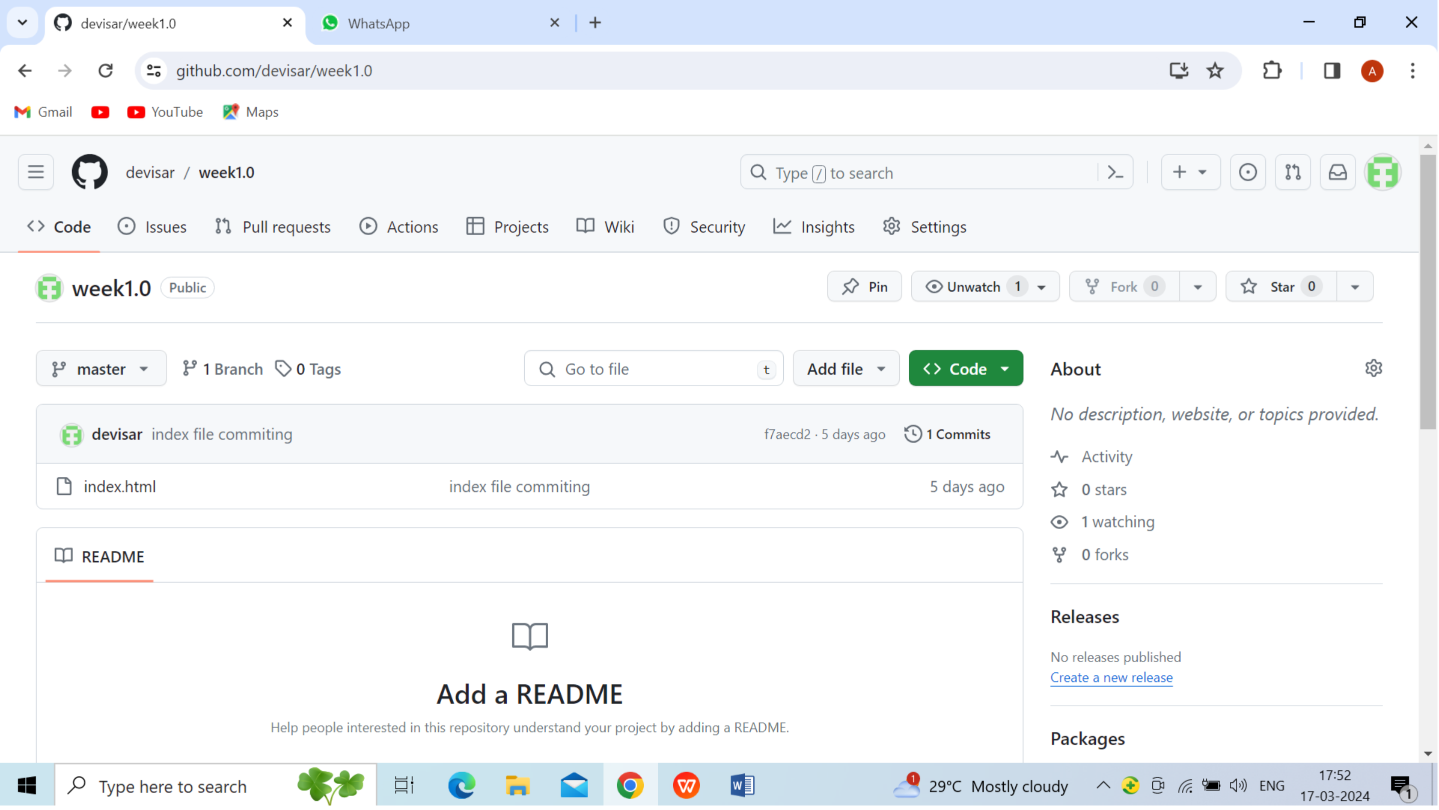
8.git push

- The command git push is used to transfer the commits or pushing the content from the local repository to the remote repository.
- The command is used after a local repository has been modified, and the modifications are to be shared with the remote team members.

git push –u origin master



```
Admin@DESKTOP-T3DAU1H MINGW64 /f/Anusha/week1 (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 688 bytes | 344.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/devisar/week1.0.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```



github.com/devisar/week1.0

devisar / week1.0

week1.0 Public

master 1 Branch 0 Tags

devisar index file commiting f7aecd2 · 5 days ago 1 Commits

index.html index file commiting 5 days ago

README

Add a README

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

About

No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

Fig.6.Now File is uploaded

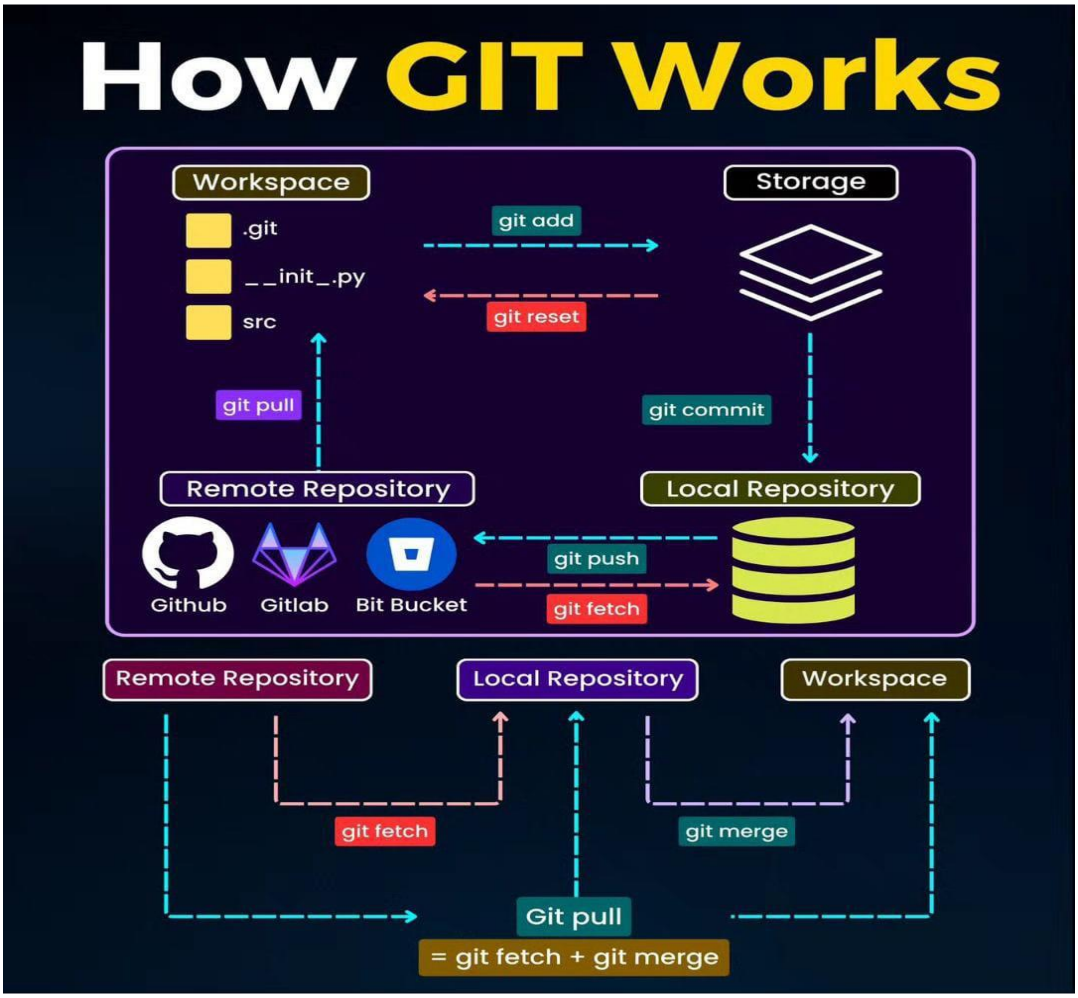


Fig.7.Git Working Process

9.git clone

The git clone command is used to create a local working copy of an existing remote repository.

- The command downloads the remote repository to the computer. It is equivalent to the Git init command when working with a remote repository.

Git clone <remote URL>

10.git pull

- The git pull command is used to fetch and merge changes from the remote repository to the local repository.
- The command "git pull origin master" copies all the files from the master branch of the remote repository to the local repository.

Git pull <branch name> <remote URL>

VIVA QUESTIONS:

1. What is Git?

   Git is an **open-source distributed version control system**. It is designed to handle minor to major projects with high speed and efficiency. It is developed to co-ordinate the work among the developers. The version control allows us to track and work together with our team members at the same workspace. Git was created by **Linus Torvalds** in **2005** to develop Linux Kernel. It is also used as an important distributed version-control tool for **the DevOps**.

2. What is the difference between Git and Github?

   Git is a version control system that allows developers to track changes in their code. GitHub is a web-based hosting service for git repositories. In simple terms, you can use git without Github, but you cannot use GitHub without Git.

3. What are the benefits of version control system?

   - Enhances the project development speed by providing efficient collaboration,
   - Reduce possibilities of errors and conflicts meanwhile project development through traceability to every small change,
   - Employees or contributors of the project can contribute from anywhere irrespective of the different geographical locations through this **VCS.**

4. What is a repository in Git?

   A Git repository is a central storage location for managing and tracking changes in files and directories. It is a crucial component of the Git version control system, which enables collaborative development and allows multiple developers to work on a project simultaneously.

5. How to create a repository in Git?

   - git init: Initializes a new Git repository.
   - git clone <repository_url>: Clones a remote repository to your local machine.
   - git add <file_name>: Stages a specific file for the next commit.
   - git commit -m "Commit message": Creates a commit with the staged changes.

6. What is the difference between git pull and git fetch?

   Git Fetch command fetches all the changes **from the remote repository to the local repository** without merging the changes into the working directory. Git Pull on the other hand brings the copy of the **remote directory changes into the working directory**.