Experiment-1

Objective: To learn how to create a Git repository, track changes in a project folder, and upload it to our own GitHub repository using basic Git commands.

Software/Tools used: Git, GitHub, Visual Studio Code

Theory overview: Git is a version control system used to track changes in files and manage code history during software development. It helps developers work efficiently, avoid losing code, and collaborate with others by maintaining different versions of a project.

GitHub is an online platform that hosts Git repositories. It allows users to upload their local Git projects to the cloud, making it easy to back up work, share code, and collaborate with teammates or the public.

Visual Studio Code (VS Code) is a powerful code editor that supports Git integration. It provides a user-friendly interface for managing Git operations like initializing a repository, staging changes, committing files, and pushing to GitHub.

In this experiment, we use:

- **VS Code** to write and manage code files.
- **Git** to track and commit changes locally.
- **GitHub** to store the repository online.

We create a folder, initialize it as a Git repository, write some code (like for BMI Calculator website) , and then push the folder to our own GitHub repository using Git commands in the VS Code terminal.

Procedure:

1. I installed **Git** on my laptop to manage version control locally.
2. I installed **Visual Studio Code (VS Code)** to write and manage our project files easily.
3. I created a **GitHub account** to store my repositories online.
4. I made a new folder on my device and opened it in VS Code.
5. I initialized Git in that folder to start tracking changes.
6. I added files (like project code) into the folder.
7. I saved and committed the changes using Git from the terminal in VS Code.
8. I created a new repository on my GitHub account.
9. I connected the local folder to my GitHub repository.
10. Finally, I pushed the project folder from my device to my **own GitHub repository**, where it is now safely stored and can be viewed online.

Code:

```
PS C:\Users\MITANAV\OneDrive\Desktop\md> git init
>> git add .
>> git commit -m "Initial commit"
>> echo "# OSTlabrepo" >> README.md
>> git init
>> git add README.md
>> git commit -m "first commit"
>> git branch -M main
>> git remote add origin https://github.com/mitashidua/OSTlabrepo.git
>> git push -u origin main
>> git config --global user.email "mitashidua@gmail.com"
>> git config --global user.name "Mitashi Dua"
```

```
PS C:\Users\MITANAV\OneDrive\Desktop\md> git commit -m "Initial commit for OSTlab"
>>
[main (root-commit) a966eb0] Initial commit for OSTlab
 2 files changed, 192 insertions(+)
 create mode 100644 README.md
 create mode 100644 index.html
PS C:\Users\MITANAV\OneDrive\Desktop\md> git push -u origin main
>>
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 16 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 2.19 KiB | 749.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/mitashidua/OSTlabrepo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Output: