**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

Set Up a Local Git Repository: Initialize a Git repository locally and version control your static website

Name: Bhavadharani S Department: CSE



**Introduction**

Version control is a fundamental practice in software development that allows you to manage changes to your code over time. It provides a systematic way to track updates, collaborate with others, and revert to previous versions if needed. Git is one of the most widely used version control systems, known for its efficiency, flexibility, and distributed nature.

In this POC, we’ll initialize a local Git repository to version control your static website. By doing so, you’ll be able to track changes to your project files, experiment with new features in a controlled way, and easily share your project with others if needed. Setting up a Git repository is a critical step towards maintaining a structured and reliable workflow, especially for developers and teams working on collaborative projects.

**Overview**

Here’s what we will cover in this setup:

**1. Installing Git**: Ensure Git is installed on your system and properly configured.

**2. Creating a Local Repository**: Initialize a Git repository in the root folder of your static website

**3. Staging and Committing Files**: Add your project files to the staging area and commit them to the repository to save a snapshot of your work.

**4. Reviewing the** evolves. **Repository State**: Use Git commands to check the status of your repository and verify that everything is tracked properly.

**Objectives**

By the end of this POC, you will:

**1. Understand the Basics of Version Control**: Gain insight into the importance of Git for managing and tracking changes in your projects.

**2. Set Up a Git Repository**: Learn how to initialize a Git repository to version control your static website locally.

**3. Track Changes Effectively**: Understand how to stage and commit files to ensure every change is logged.

4. **Organize Your Project**: Maintain a clean and structured workflow for your static website, with the ability to roll back changes when needed.

5. **Prepare for Collaboration**: Lay the groundwork to share your repository and collaborate with others using Git when required

**Importance of Setting Up a Local Git Repository**

**Track Changes**: Git records all modifications, ensuring a clear history of your project.

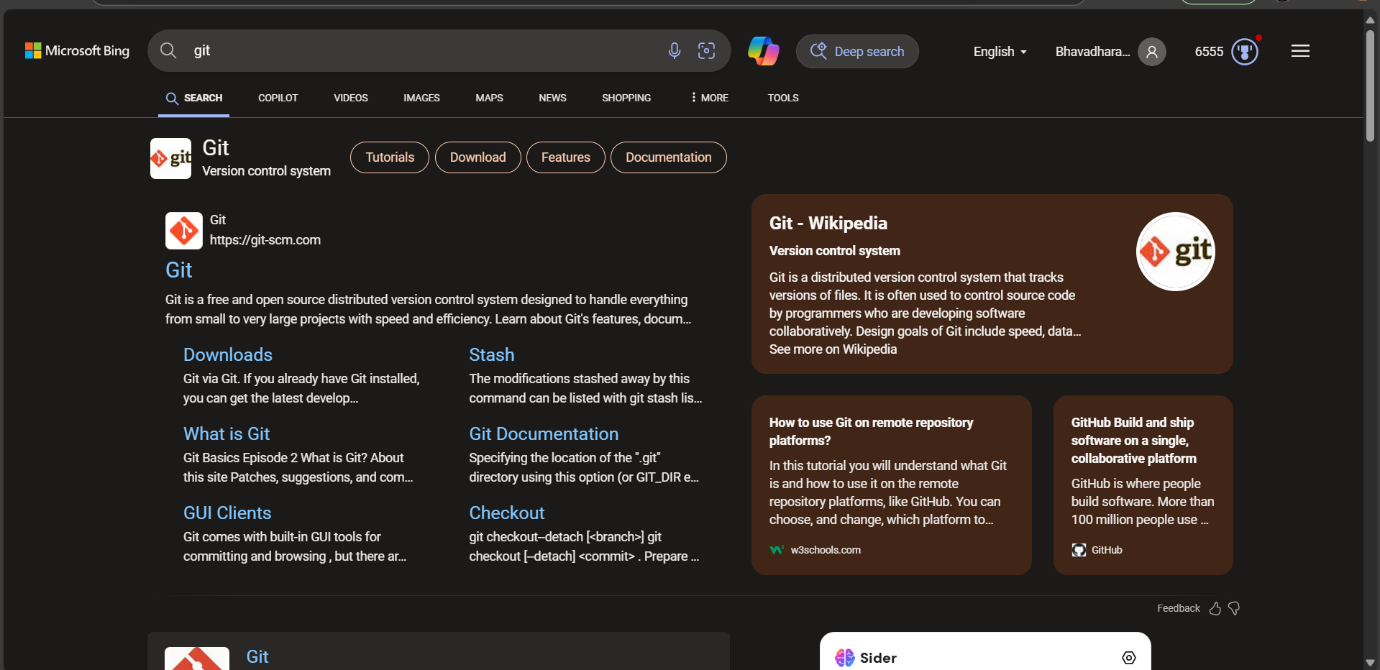
**Rollback**: Easily revert to previous versions to recover from mistakes.

**Collaboration**: Prepares your project for team work, enabling smooth integration of changes.

**Step-by-Step Overview**

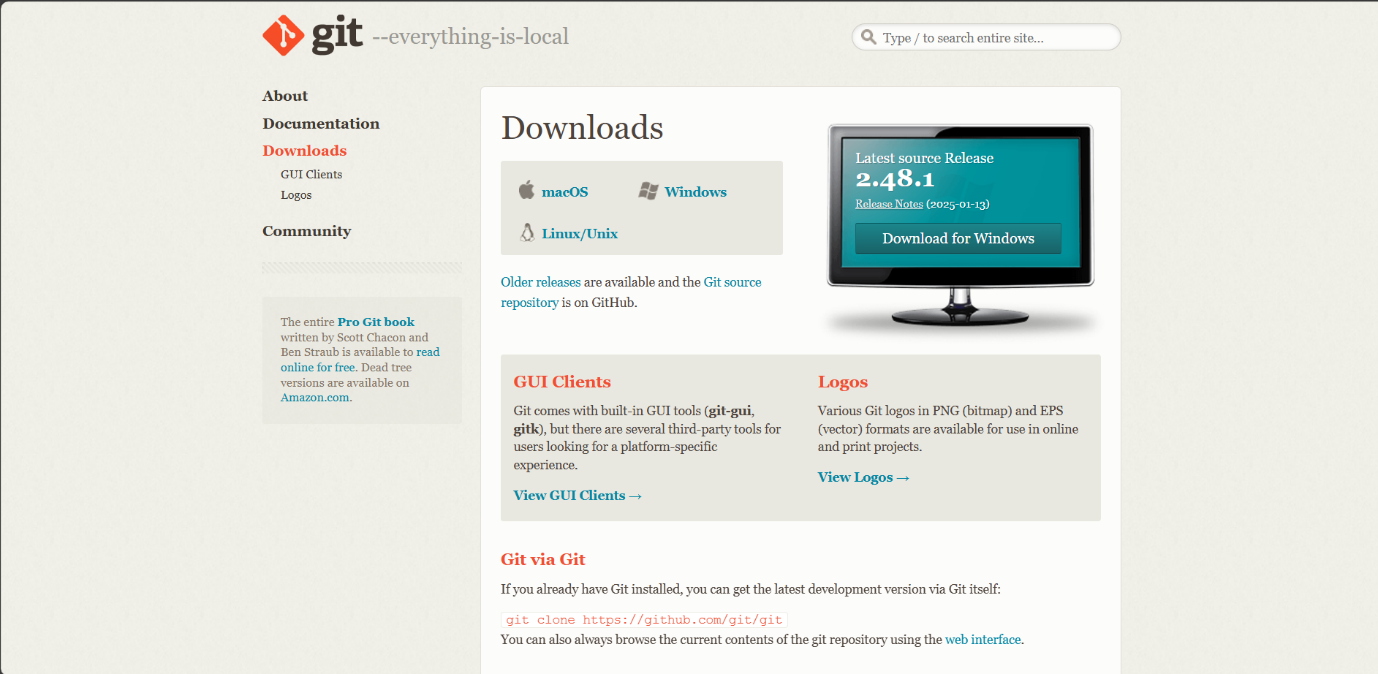
Step 1:

Search for "Git" download it, and click the "Downloads" option on the website.



Step 2 :

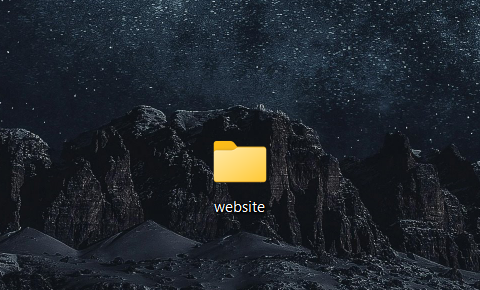
Click the **Windows** option on the download page and follow the installation wizard



Step 3:

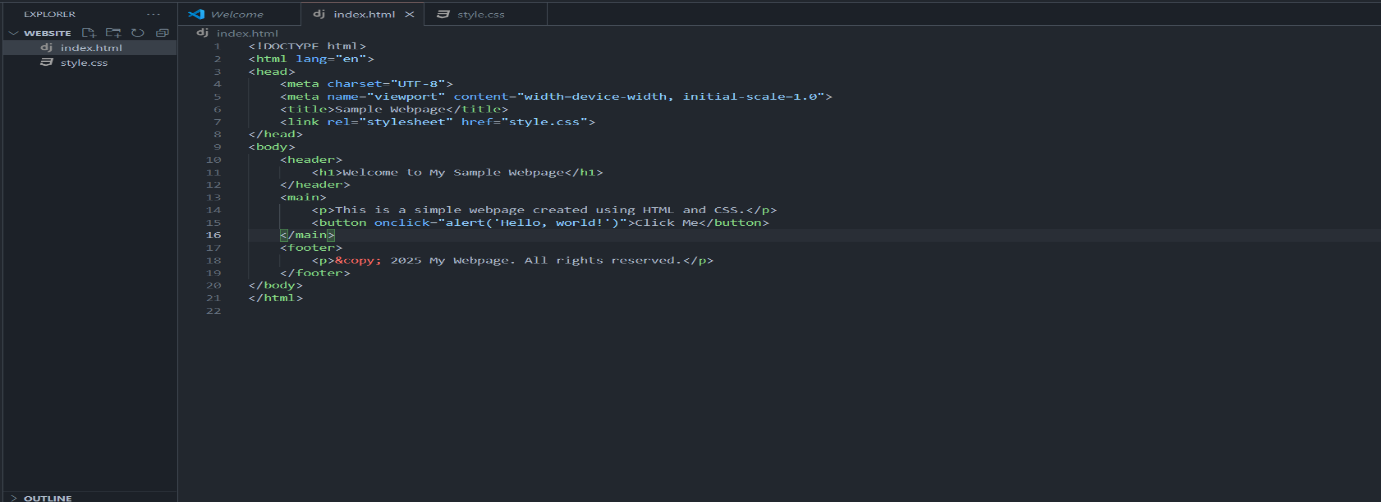
In your Desktop Create a folder named website for your static website

Inside that folder, create a simple HTML file named index.html. You can write some basic HTML



Step 4 :

1. Open the terminal in VS Code (or Command Prompt if you prefer).
2. Navigate to the folder named website that you created earlier

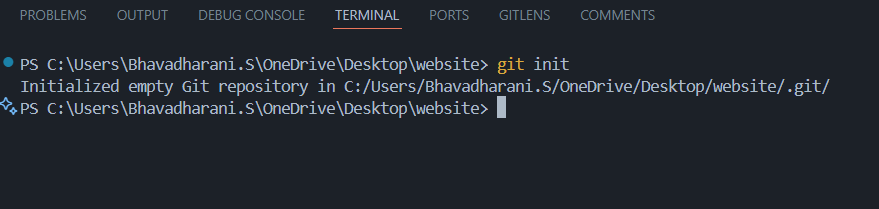


Step 5 :

1. Initialize Git in your project folder by typing

***git init***

This command will create a .git folder in your project directory, enabling Git to start tracking changes.

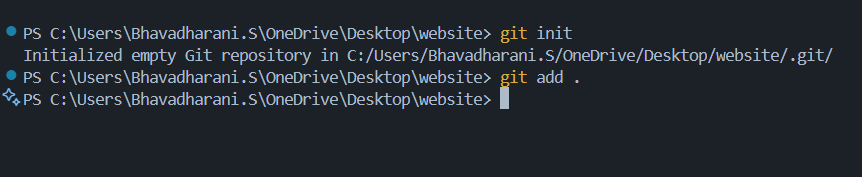


Step 6:

Use the git add command to tell Git which files to track. To track all files in the folder, type:

***git add .***

This adds all files in the current directory to Git’s tracking system

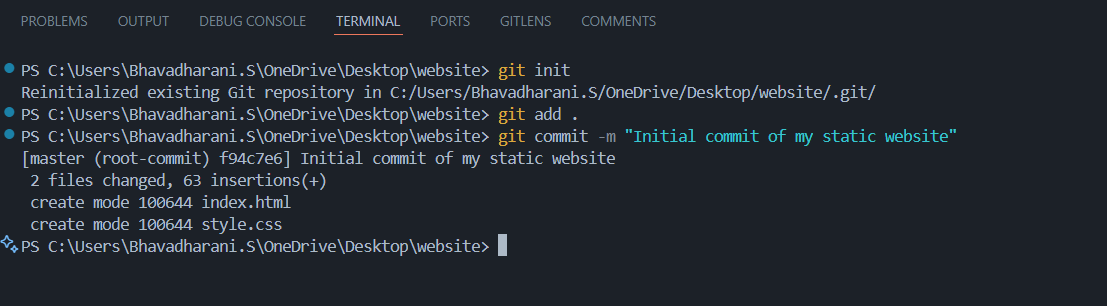


Step 7:

Save your changes in Git by committing them with a descriptive message:

***git commit -m "Initial commit of my static website"***

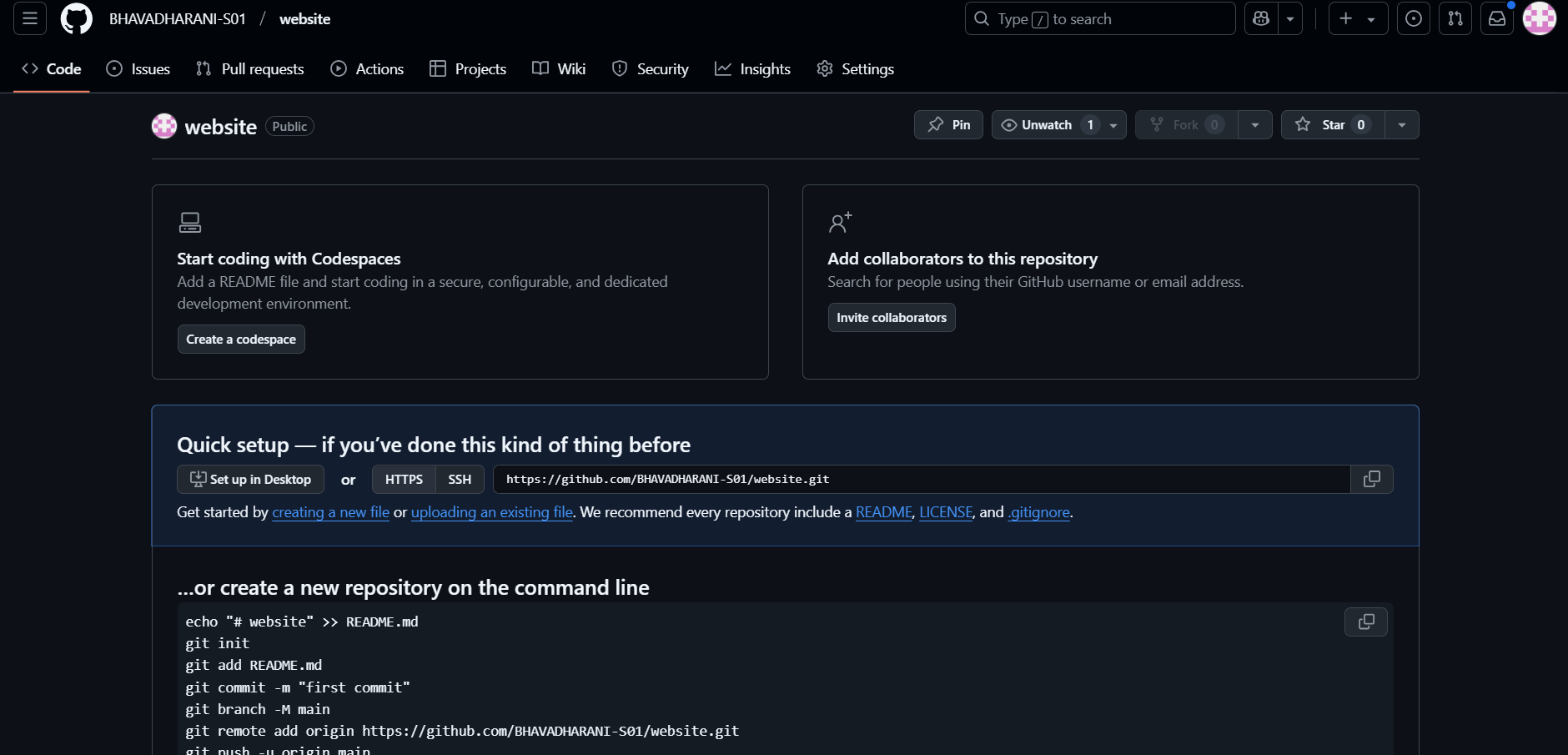
The -m flag allows you to add a message about your changes. In this case, we’re indicating that this is the "initial commit."



Step 8 :

Log in to GitHub and click the green **"New"** button at the top-right corner of the homepage.

1. Enter a repository name, such as my-website.
2. Leave the other settings as default and click **"Create repository"**.



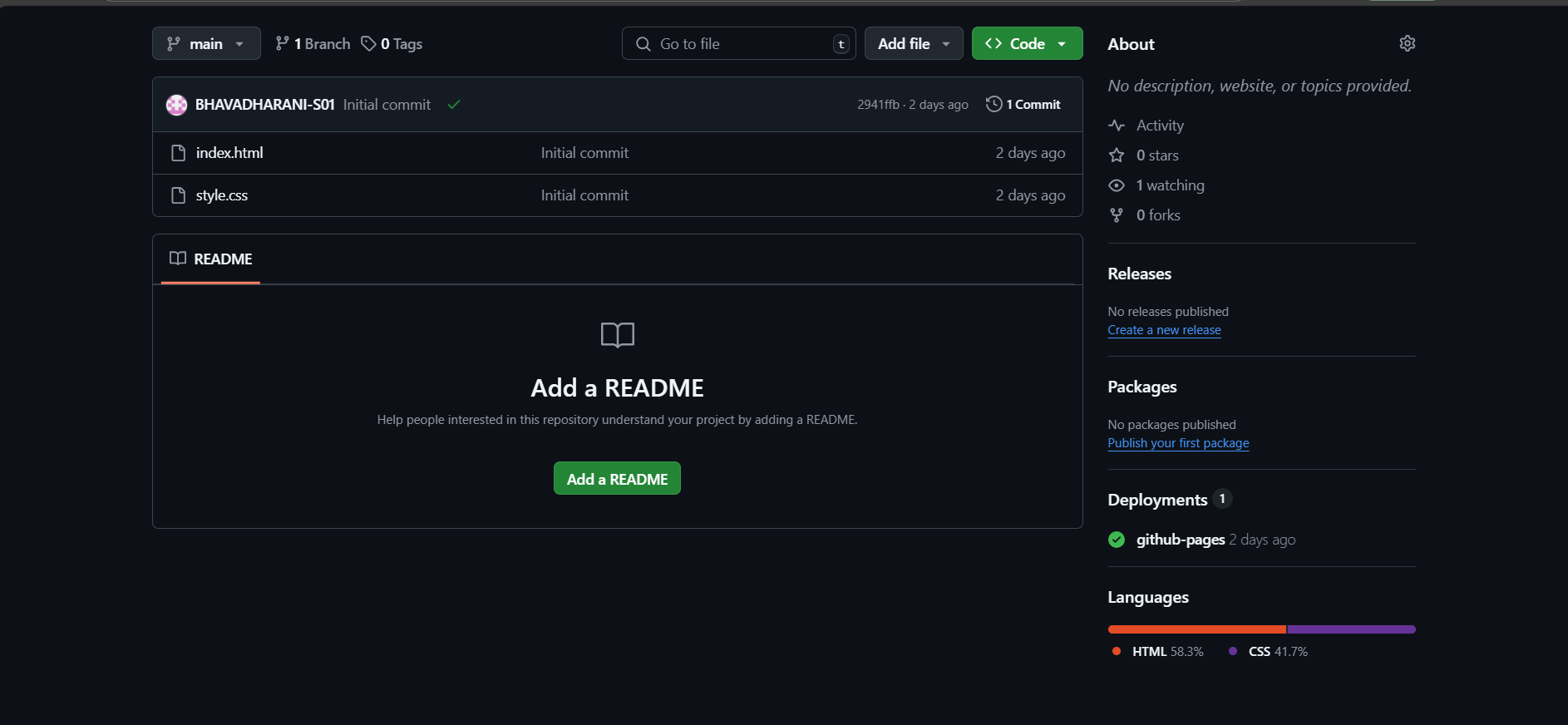
Step 9:

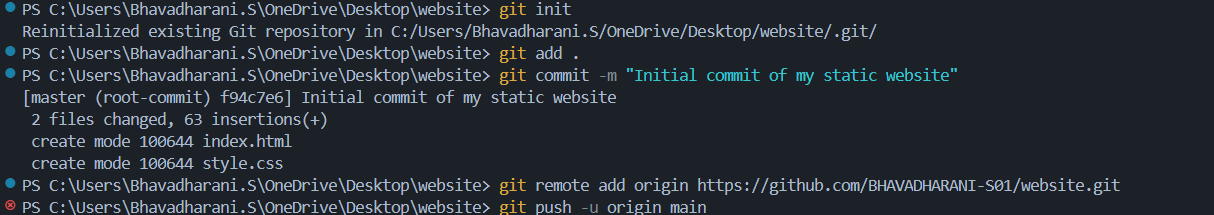
1. Back in the terminal, link your local repository to the remote GitHub repository:

git remote add origin https://github.com/yourusername/my-website.git

Replace yourusername with your GitHub username and my-website with your repository’s name.







**Expected Outcome :**

By completing this PoC of setting up a local Git repository, you will:

* Successfully initialize a Git repository in your local static website folder.
* Track changes made to your website files (HTML, CSS, etc.) using Git version control.
* Understand the basic Git commands (git init, git add, git commit) for version control.
* Commit your changes locally with a descriptive commit message.
* Gain hands-on experience with Git and how it helps manage and track website file changes.