**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

Deploy your static website using Github Pages : Host your local Git repository’s static website directly using Github pages

Name: Vijaya Nandana M Department : CSE



**Introduction**

GitHub Pages is a static site hosting service designed to publish your projects directly from a GitHub repository. It allows developers to showcase their work, create personal websites, or host documentation in an efficient, free, and straightforward way.

**Objective**

1. Learn the fundamentals of GitHub Pages and its deployment process.

2. Understand the importance of static website hosting and its use cases.

3. Gain hands-on experience in using Git and GitHub for project versioning and hosting.

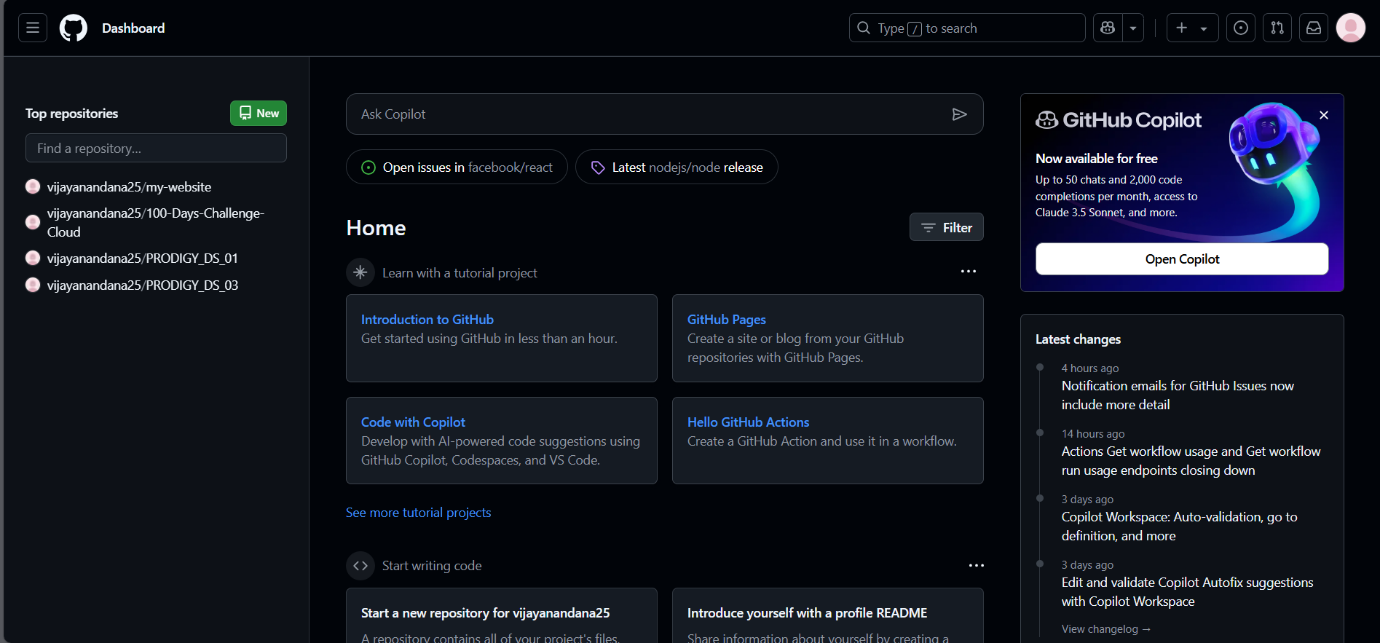
4. Successfully publish a static website and make it publicly accessible.

**Step-by-Step Overview**

Step1:

Once you're logged in, click the green "New" button on the top- right of your GitHub homepage to create a new repository.

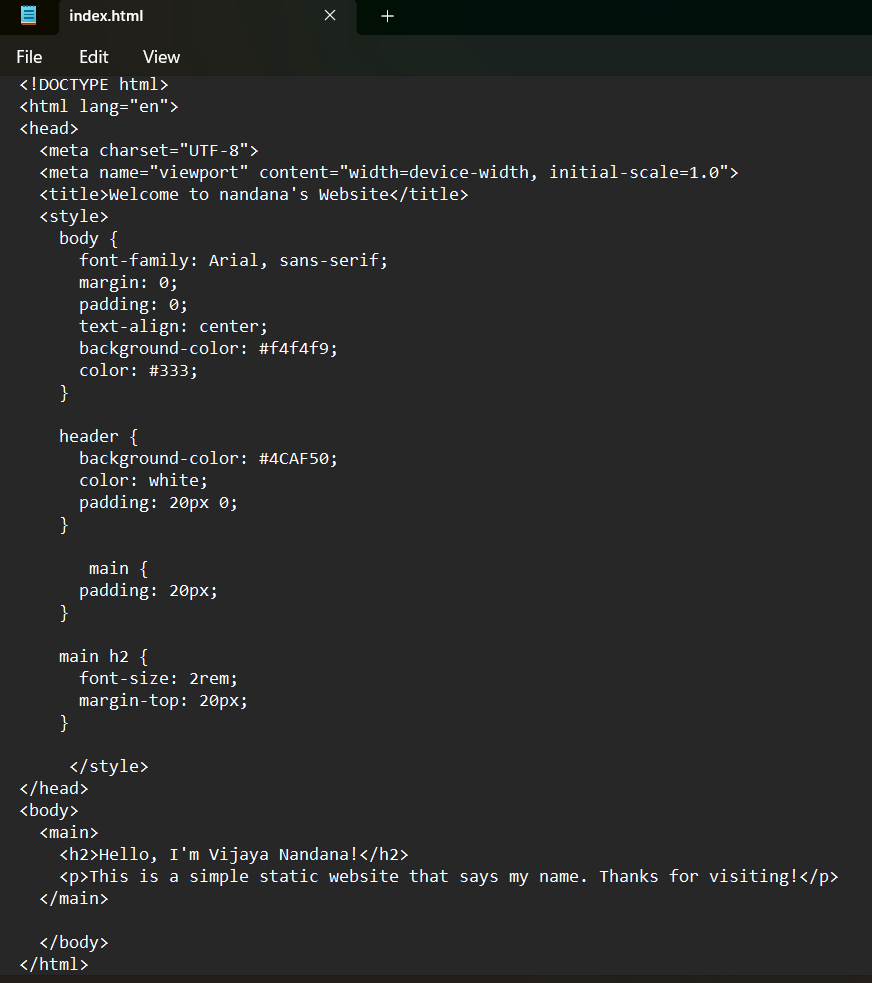
Give your repository a name, for example, my-static-website. Leave the other settings as default, and click "Create repository".



Step 2 :

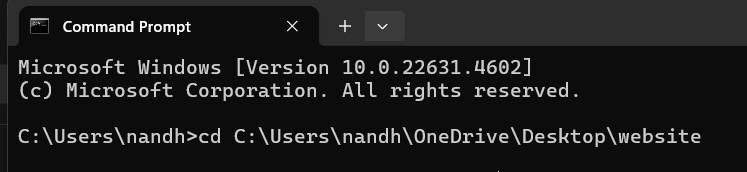
Create a folder (e.g., website) where you’ll keep all your website files. Inside that folder, create the main file for your website, called index.html.

Here’s a simple example of what to put in your index.html:

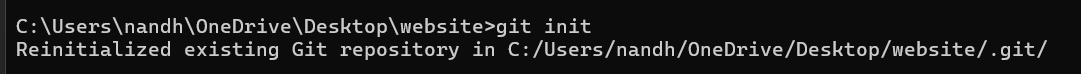


Step 3 :

* Open Command Prompt and navigate to the folder where your index.html file is saved.



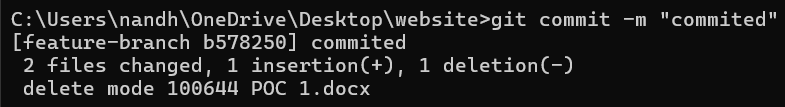
* Initialize the repository using “git init”.



* Add your website files to the repository.



* Save the changes in Git with a commit message.



* link your local repository to the GitHub repository using

**git remote add origin <repo url>**

* push your files to the repository.

Step 4 :

Enable GitHub Pages

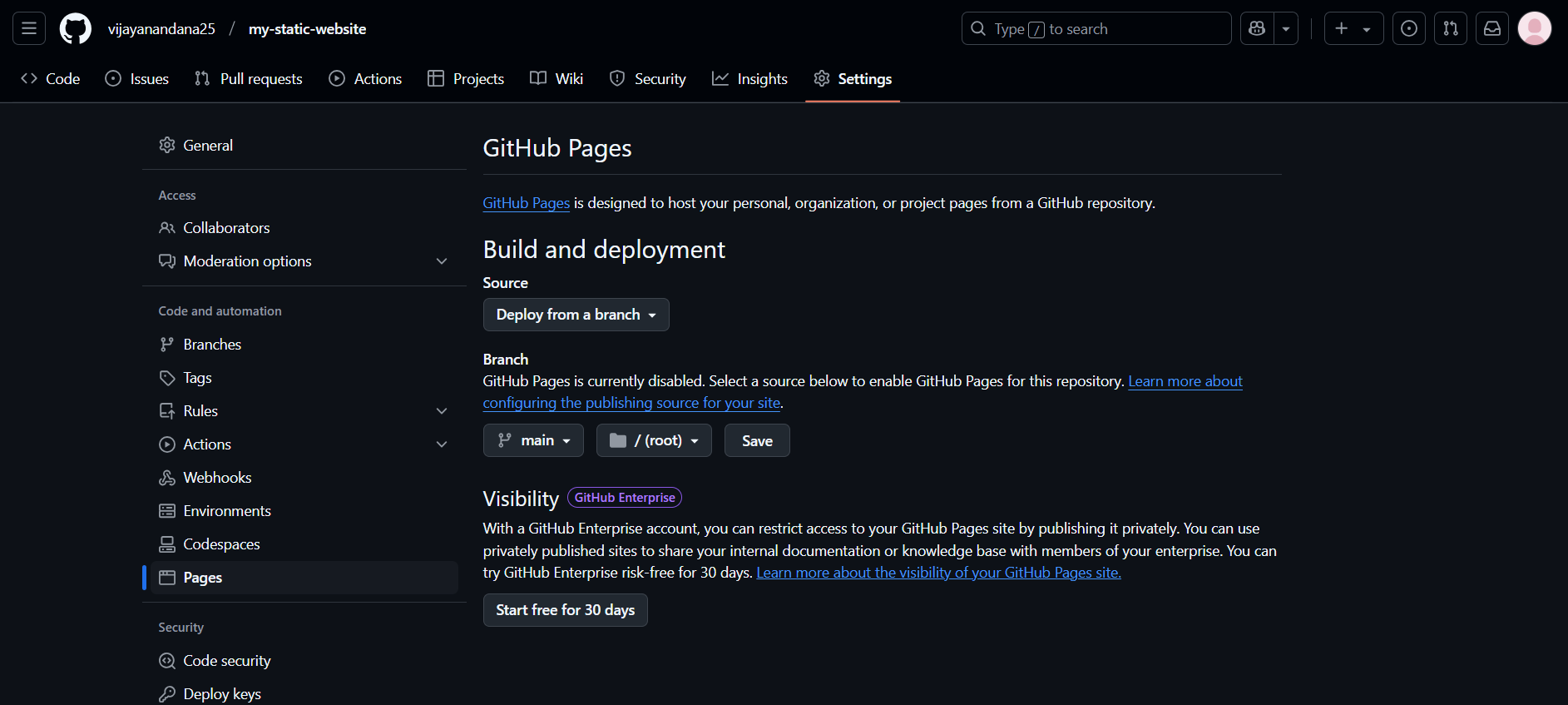
1. Go to your repository on GitHub.

2. Click on the Settings tab (it’s near the top, next to Code, Issues, etc.).

3. Scroll down to the Pages section (on the left menu, under "Code and automation").

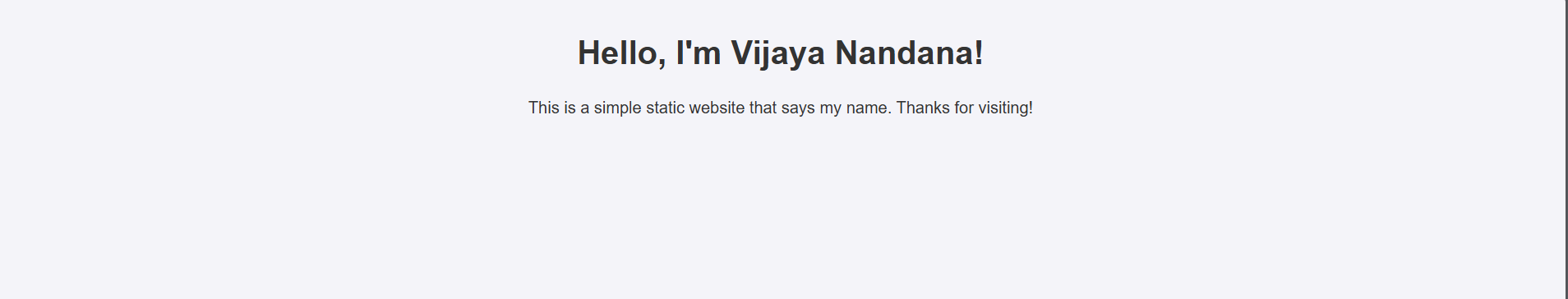
4. Under Source, select: o Branch: main o Folder: / (root)

5. Click Save.



Step 7 :

Access Your Website at https://<yourusername>.github.io/<yourrepo>



**Outcome**

By completing this PoC, we have successfully deploying a static website using GitHub Pages and gained hands-on experience with Git commands like git init, git add, git commit, git remote add, and git push.