

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: Jenith Melkeena R 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.

Step-by-Step Overview

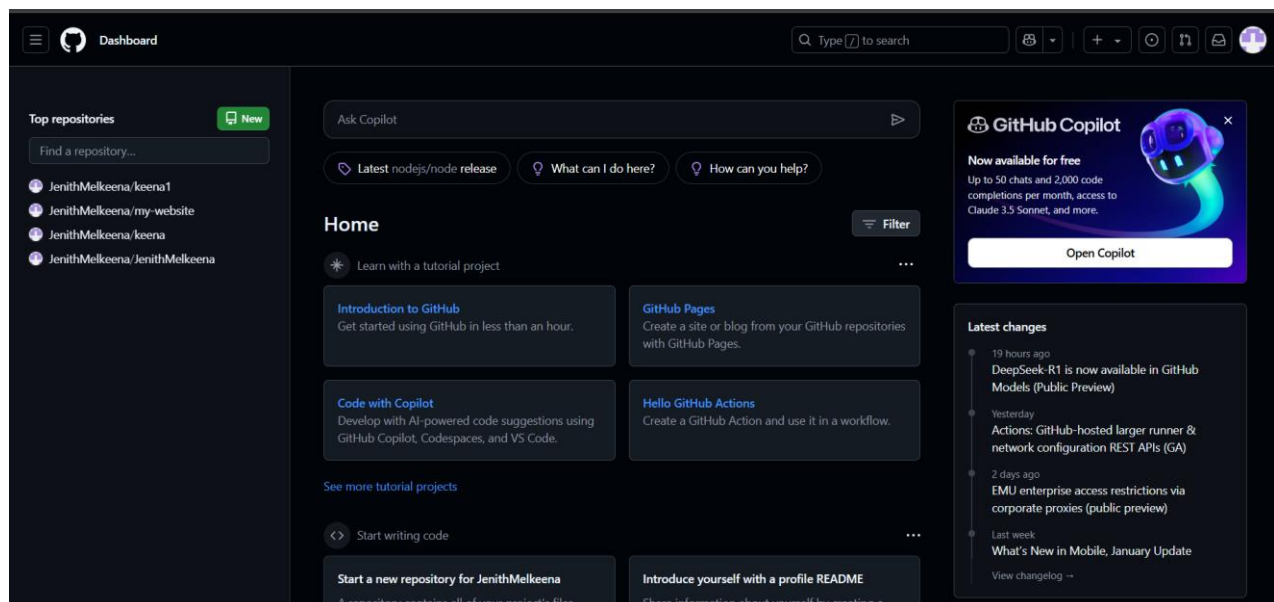
Step 1:

Create a New Repository:

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"**.

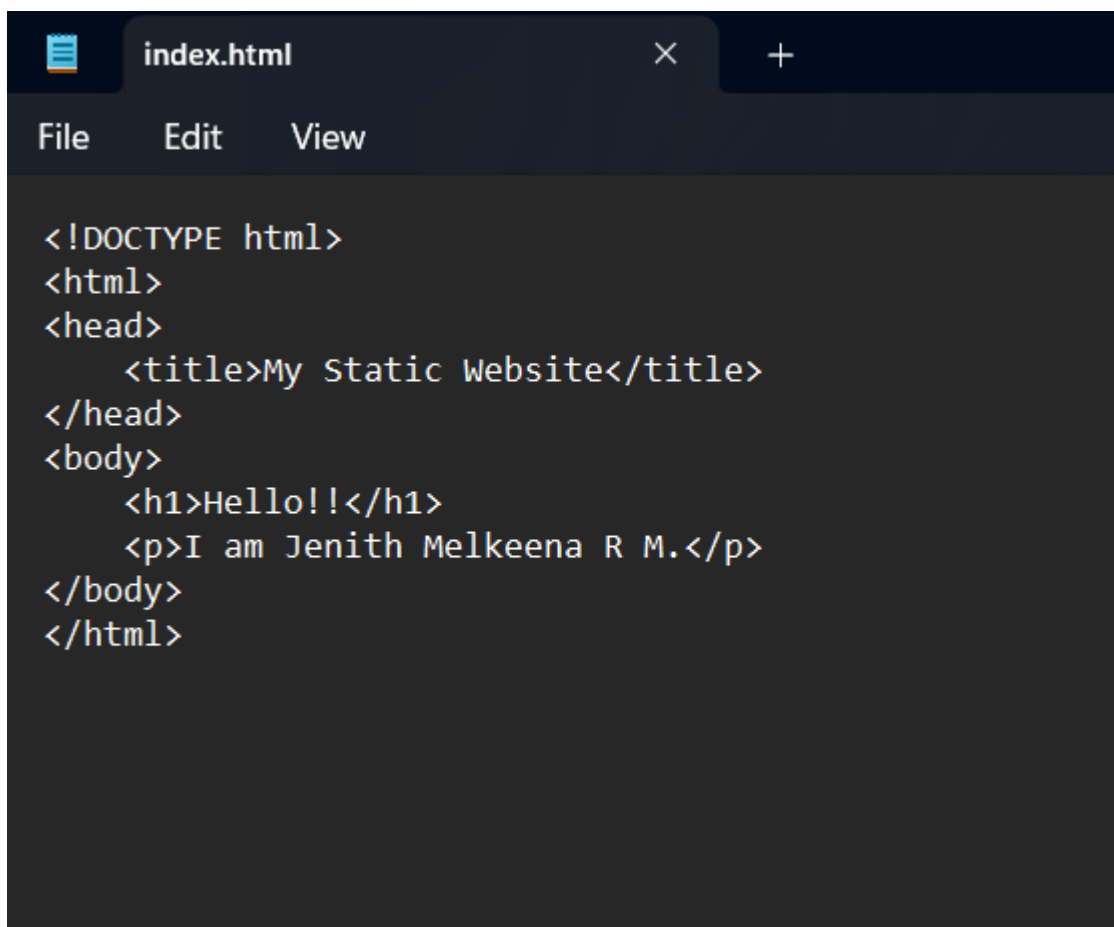


Step 2:

Create a folder (e.g., my-static-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:

A screenshot of a code editor window with a dark theme. The title bar shows a tab labeled 'index.html' with a close button (X) and a plus sign (+). The menu bar includes 'File', 'Edit', and 'View'. The code is written in a light-colored font on a dark background. It is a basic HTML document with a DOCTYPE declaration, a head section with a title, and a body section with an h1 heading and a paragraph.

```
<!DOCTYPE html>
<html>
<head>
  <title>My Static Website</title>
</head>
<body>
  <h1>Hello!!</h1>
  <p>I am Jenith Melkeena R M.</p>
</body>
</html>
```

Step 3:

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

Use the cd command to navigate.

```
C:\Users\jenit>cd C:\Users\jenit\OneDrive\Desktop\my-static-website  
C:\Users\jenit\OneDrive\Desktop\my-static-website>|
```

Step 4:

Initialize a Git repository by running:

```
C:\Users\jenit\OneDrive\Desktop\my-static-website>git init  
Initialized empty Git repository in C:/Users/jenit/OneDrive/Desktop/my-static-website/.git/
```

Step 5:

Add your website files to the repository:

```
C:\Users\jenit\OneDrive\Desktop\my-static-website>git add .  
  
C:\Users\jenit\OneDrive\Desktop\my-static-website>
```

Step 6:

Save the changes in Git with a commit message:

```
C:\Users\jenit\OneDrive\Desktop\my-static-website>git commit -m "Initial commit"  
[master (root-commit) 3c58d83] Initial commit  
1 file changed, 10 insertions(+)  
create mode 100644 index.html
```

Step 7:

Go to your GitHub repository (the one you created earlier).

Copy the **repository URL**:

In your Command Prompt, link your local repository to the GitHub repository:

```
C:\Users\jenit\OneDrive\Desktop\my-static-website>git remote add origin https://github.com/JenithMelkeena/my-static-website
```

Step 8:

Push your files to GitHub:

```
Enumerating objects: 3, done.  
Counting objects: 100% (3/3), done.  
Delta compression using up to 12 threads  
Compressing objects: 100% (2/2), done.  
Writing objects: 100% (3/3), 363 bytes | 363.00 KiB/s, done.  
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
```

Step 9:

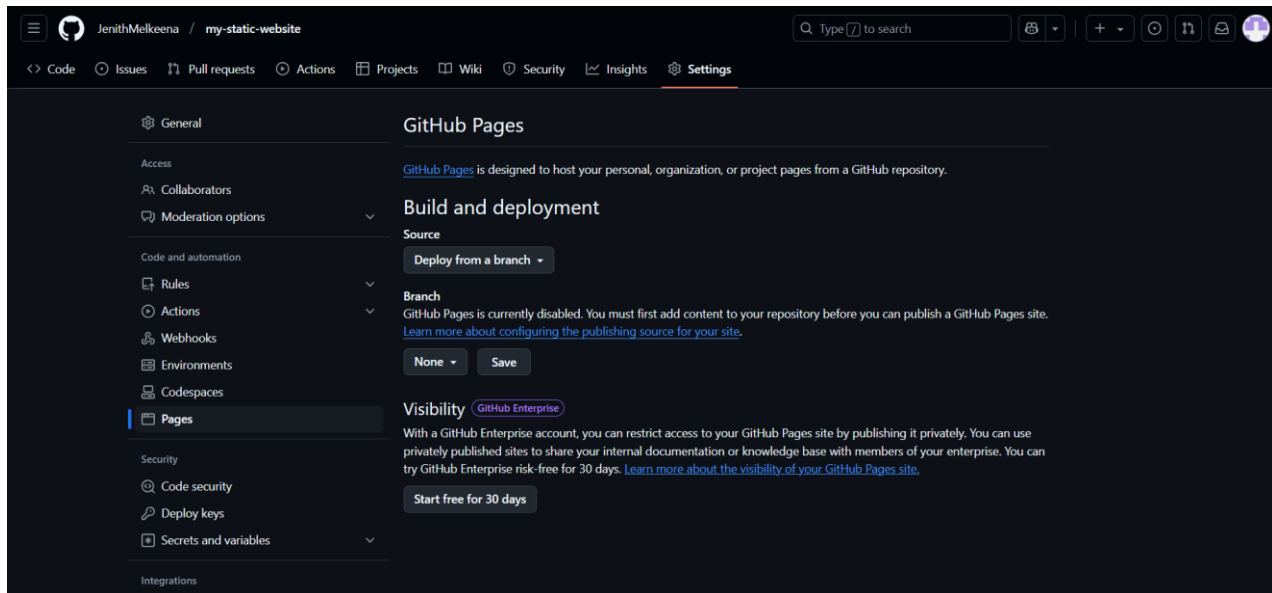
Enable GitHub Pages

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:

- **Branch:** main
- **Folder:** / (root)

5. Click **Save**.



Step 10:

Access Your Website

Wait a few minutes for GitHub Pages to deploy your site.

Visit your website at:

<https://<your-username>.github.io/<your-repository>>

Hello!!

I am Jenith Melkeena R M.

Outcome

By completing this PoC of deploying a static website using GitHub Pages, you will:

1. Successfully create and configure a GitHub repository for your project.
2. Initialize a Git repository in your local project folder and link it to GitHub.
3. Upload your static website files (HTML, CSS, JavaScript) to GitHub.
4. Enable GitHub Pages in the repository settings to host your static website.
5. Access your static website live on the web via a GitHub Pages URL.
6. Gain hands-on experience with Git commands like `git init`, `git add`, `git commit`, `git remote add`, and `git push`.

7. Understand the process of hosting a static site for free using GitHub Pages.