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: Sandhana Jefrina J

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

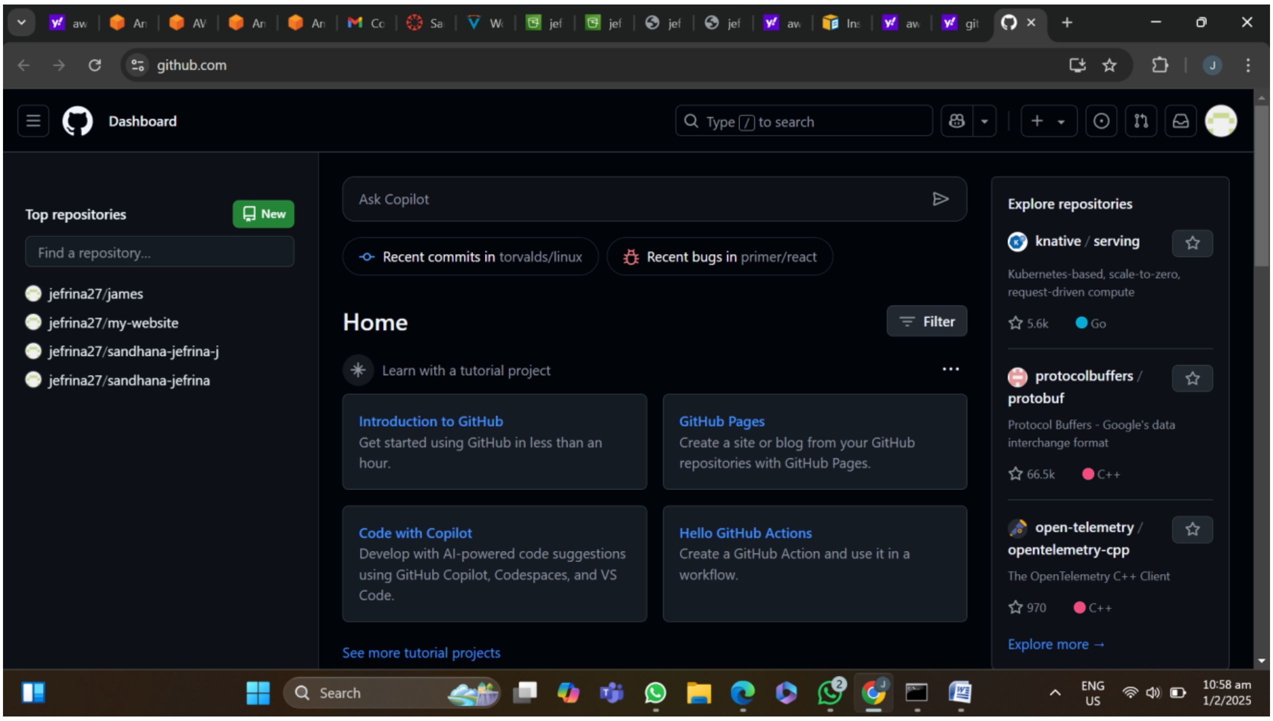
Step 1:

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**".


github.com
Dashboard
Type / to search
Top repositories
New
Find a repository...
jefrina27/james
jefrina27/my-website
jefrina27/sandhana-jefrina-j
jefrina27/sandhana-jefrina
Ask Copilot
Recent commits in torvalds/linux
Recent bugs in primer/react
Home
Filter
Learn with a tutorial project
Explore repositories
knative / serving
5.6k
Go
protocolbuffers / protobuf
66.5k
C++
open-telemetry / opentelemetry-cpp
970
C++
Explore more →
See more tutorial projects
Search
ENG
US
10:58 am
1/2/2025

Introduction to GitHub
Get started using GitHub in less than an hour.

GitHub Pages
Create a site or blog from your GitHub repositories with GitHub Pages.

Code with Copilot
Develop with AI-powered code suggestions using GitHub Copilot, Codespaces, and VS Code.

Hello GitHub Actions
Create a GitHub Action and use it in a workflow.

Kubernetes-based, scale-to-zero, request-driven compute

Protocol Buffers - Google's data interchange format

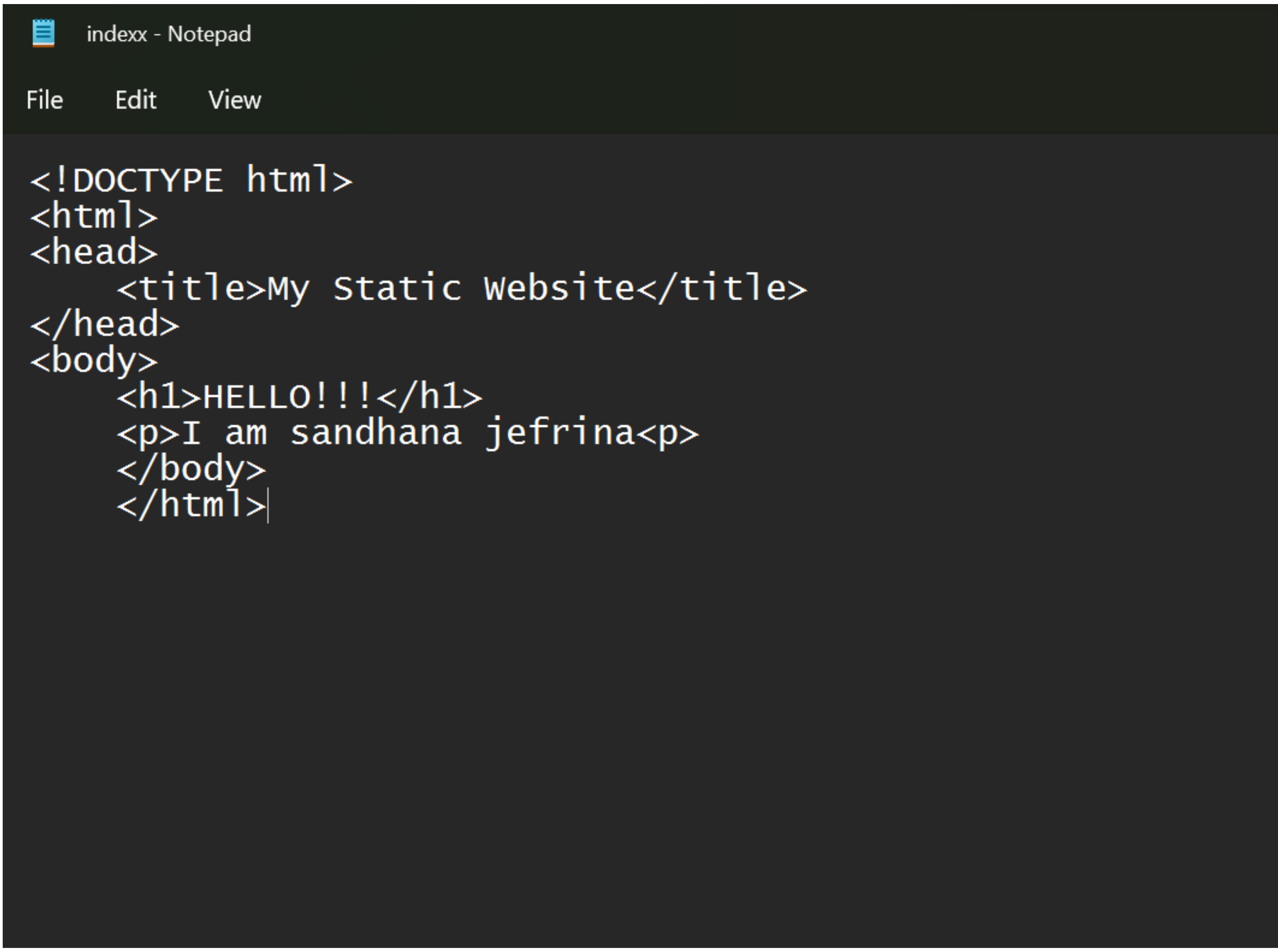
The OpenTelemetry C++ Client

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:

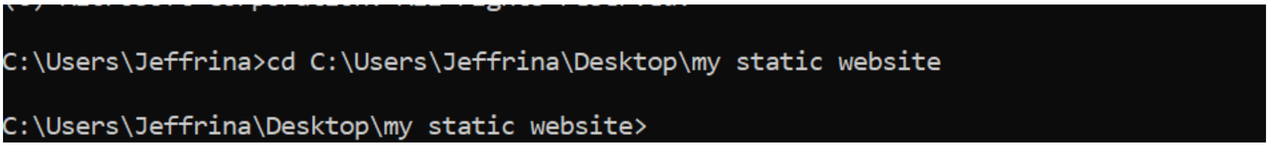


```
<!DOCTYPE html>
<html>
<head>
    <title>My Static Website</title>
</head>
<body>
    <h1>HELLO!!!</h1>
    <p>I am sandhana jefrina<p>
    </body>
    </html>
```

Step 3:

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

Use the cd command to navigate.



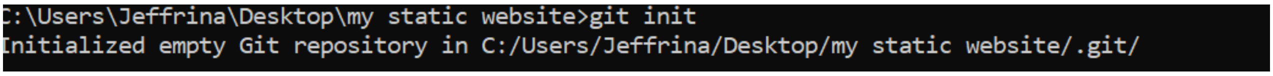
```
(c) Microsoft Corporation. All rights reserved.

C:\Users\Jeffrina>cd C:\Users\Jeffrina\Desktop\my static website

C:\Users\Jeffrina\Desktop\my static website>
```

Step 4:

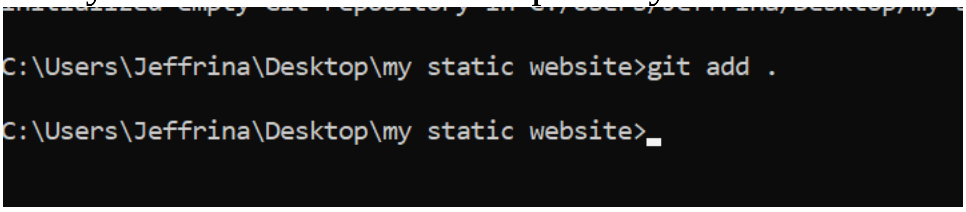
Initialize a Git repository by running:



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

Step 5:

Add your website files to the repository:

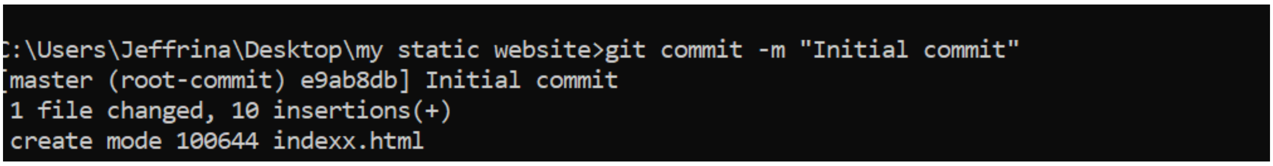


```
C:\Users\Jeffrina\Desktop\my static website>git add .

C:\Users\Jeffrina\Desktop\my static website>_
```

Step 6:

Save the changes in Git with a commit message:



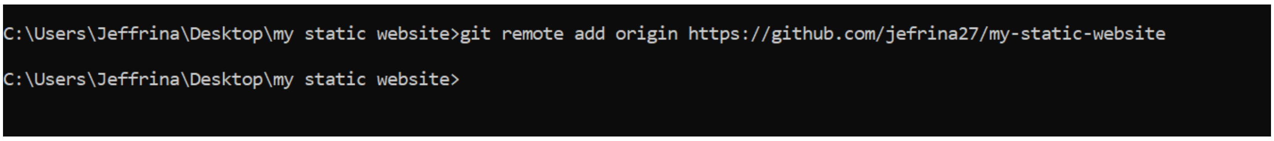
```
C:\Users\Jeffrina\Desktop\my static website>git commit -m "Initial commit"
[master (root-commit) e9ab8db] Initial commit
 1 file changed, 10 insertions(+)
 create mode 100644 indexx.html
```

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:

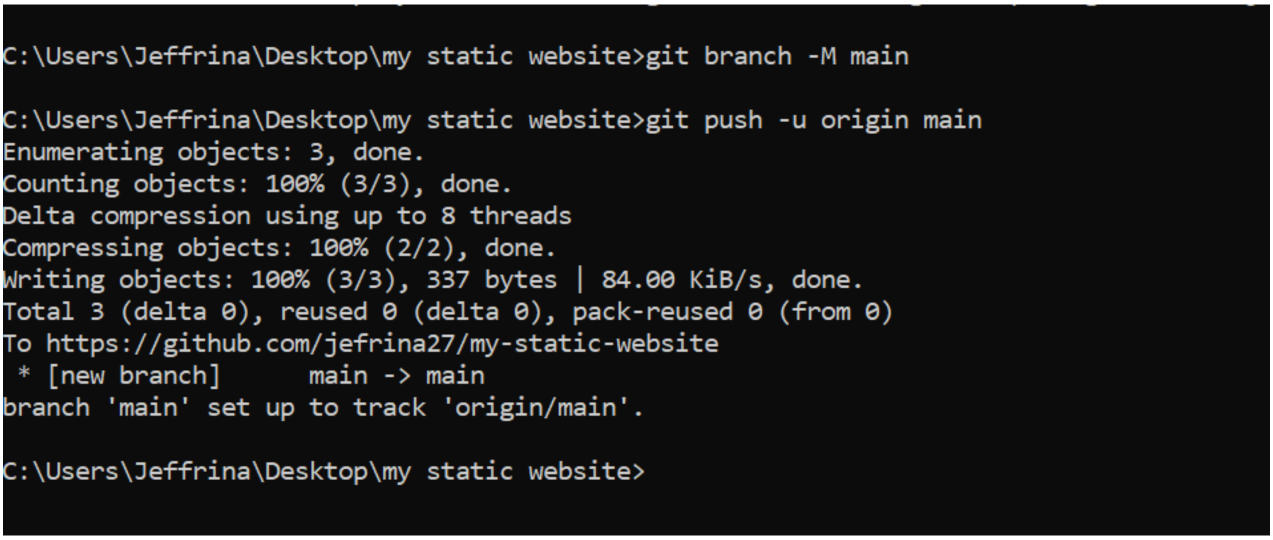


```
C:\Users\Jeffrina\Desktop\my static website>git remote add origin https://github.com/jefrina27/my-static-website

C:\Users\Jeffrina\Desktop\my static website>
```

Step 8:

Push your files to GitHub:



```
C:\Users\Jeffrina\Desktop\my static website>git branch -M main

C:\Users\Jeffrina\Desktop\my static website>git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 337 bytes | 84.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/jefrina27/my-static-website
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

C:\Users\Jeffrina\Desktop\my static website>
```

Step 9:

Enable GitHub Pages

1. Go to your repository on GitHub.

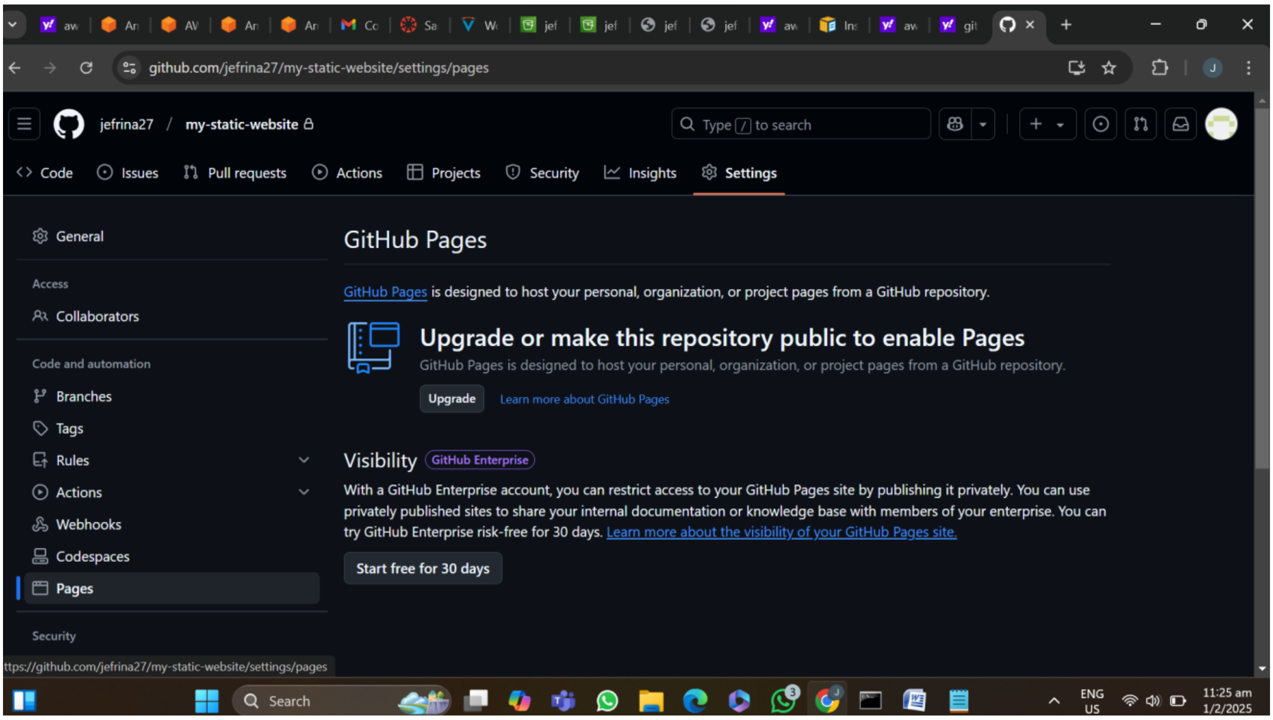
2. Click on the **Settings** tab (it’s near the top, next to Code, Issues, etc.).

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**.



github.com/jefrina27/my-static-website/settings/pages

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Upgrade or make this repository public to enable Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Visibility GitHub Enterprise

With a GitHub Enterprise account, you can restrict access to your GitHub Pages site by publishing it privately. You can use privately published sites to share your internal documentation or knowledge base with members of your enterprise. You can try GitHub Enterprise risk-free for 30 days. Learn more about the visibility of your GitHub Pages site.

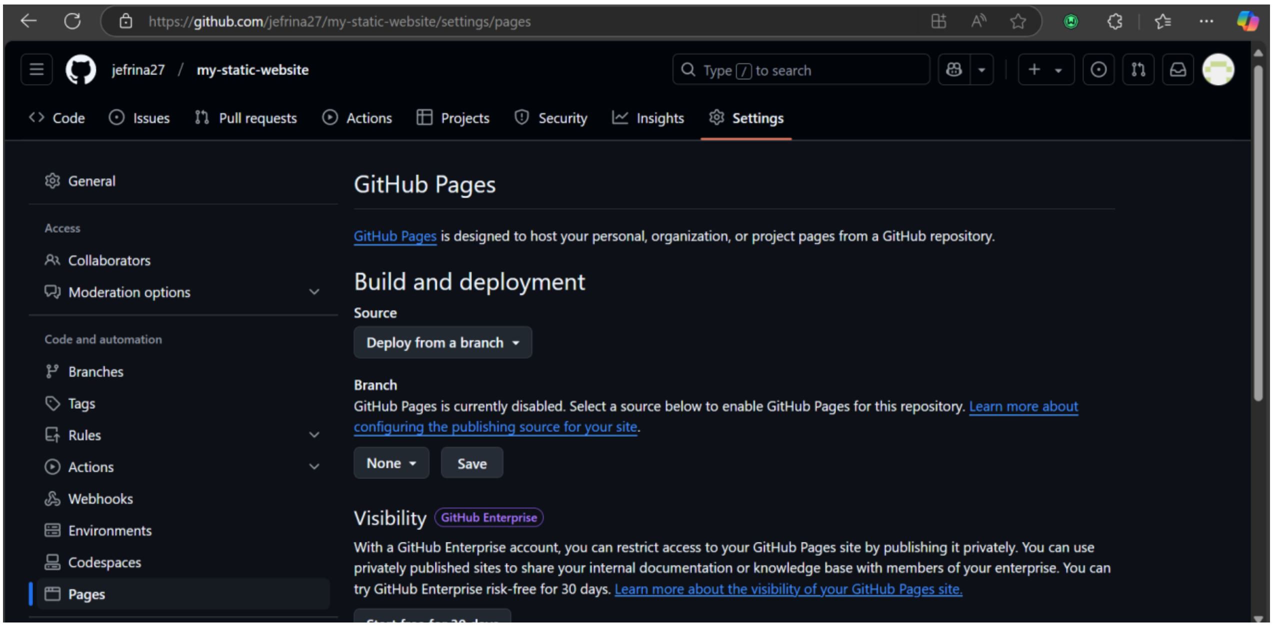
Step 10:

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>



https://github.com/jefrina27/my-static-website/settings/pages

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Build and deployment

Source

Deploy from a branch

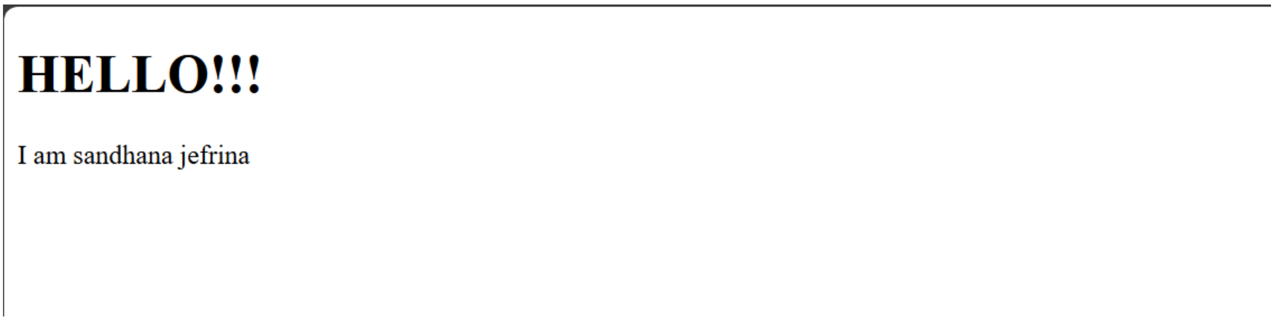
Branch

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. Learn more about configuring the publishing source for your site.

None | Save

Visibility GitHub Enterprise

With a GitHub Enterprise account, you can restrict access to your GitHub Pages site by publishing it privately. You can use privately published sites to share your internal documentation or knowledge base with members of your enterprise. You can try GitHub Enterprise risk-free for 30 days. Learn more about the visibility of your GitHub Pages site.



HELLO!!!

I am sandhana jefrina

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.