Placement Empowerment Program

Cloud Computing and DevOps Centre

Automate Static Website Deployment Locally

Create a script that updates your server whenever changes are pushed.

Name: Madhumitha H

Department: ADS

Introduction:

In this task, we are going to deploy a static website using GitHub Pages. GitHub Pages allows you to host simple static websites directly from a GitHub repository, making it an excellent choice for personal projects, portfolios, or documentation. We will start by creating a project directory on our local machine, initializing it as a Git repository, and linking it to a remote GitHub repository. After that, we'll create a basic index.html file containing the structure for our website and push it to GitHub. Once the files are uploaded, we'll configure GitHub Pages to serve the website. Finally, we'll access our live website via a unique GitHub Pages URL. This process gives you hands-on experience with Git, GitHub, and static website deployment.

Overview:

Here's an overview of the task :

- **Create a local project directory** and initialize it as a Git repository.
- **Link the local repository** to a remote GitHub repository for version control.
- **Create an index.html file** with the basic structure for a static website.
- **Add and commit the file** to the Git repository.
- **Push the changes to GitHub**, uploading the website files.
- **Configure GitHub Pages** to serve the website from the main branch.
- **Access the live website** through the provided GitHub Pages URL.

Objectives:

The objective of this task is to automate the deployment of a static website on a local server using Node.js. It aims to streamline development by detecting file changes in real time and restarting the server automatically. The task involves using **PM2** for process management and **Chokidar** for file watching. The goal is to eliminate manual restarts and ensure the latest website updates are always served.

Step-by-Step Overview

Step 1: **Set Up the Project Directory**

Create a Project Folder (Directory)

- Create a new folder (directory) for your website. For example, name it **website**.

Step 2

Initialize a Git Repository

- Inside your **website** folder, initialize a **Git repository in terminal**.
- using command:git init

Step 3

Add a Remote GitHub Repository

- Now, link your local Git repository to the remote GitHub repository (where your project will be hosted).

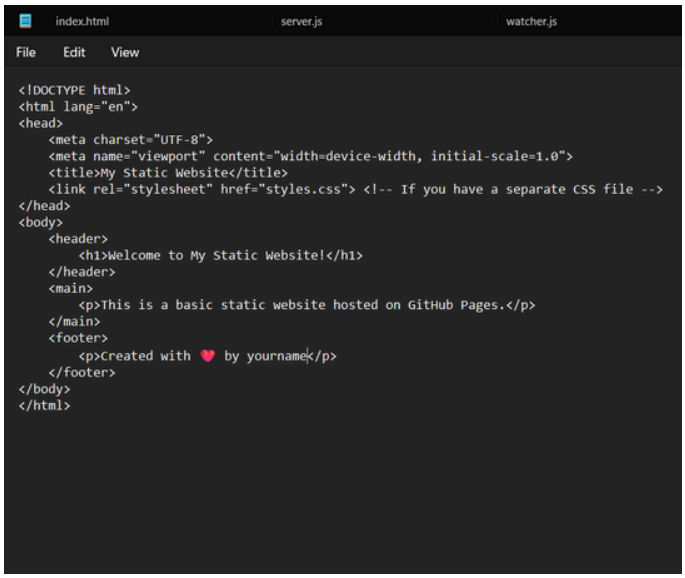
git remote add origin https://github.com/madhuh1210/my-static-website.git

Step 4:

Create an index.html File

- Inside the **website** folder, create a new file named **index.html**.
- Open the file and add the basic HTML structure for your static website:

give whatever you want your website to show . hwere i have just generated random html file



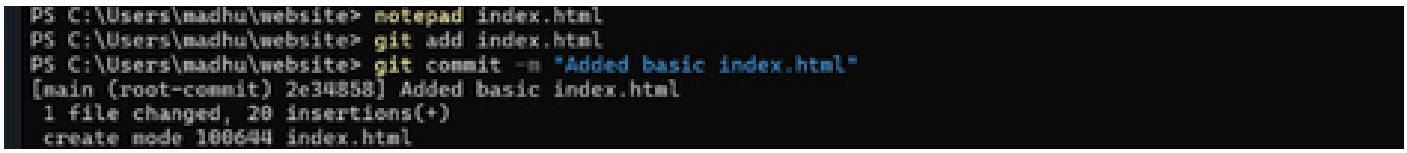
Step 5

Add and Commit the File

- Once you've created and saved the index.html file, you need to **add** it to Git and **commit** it.

git add index.html

git commit -m "Add index.html for static website"



Step 6:

Push the File to GitHub Repository

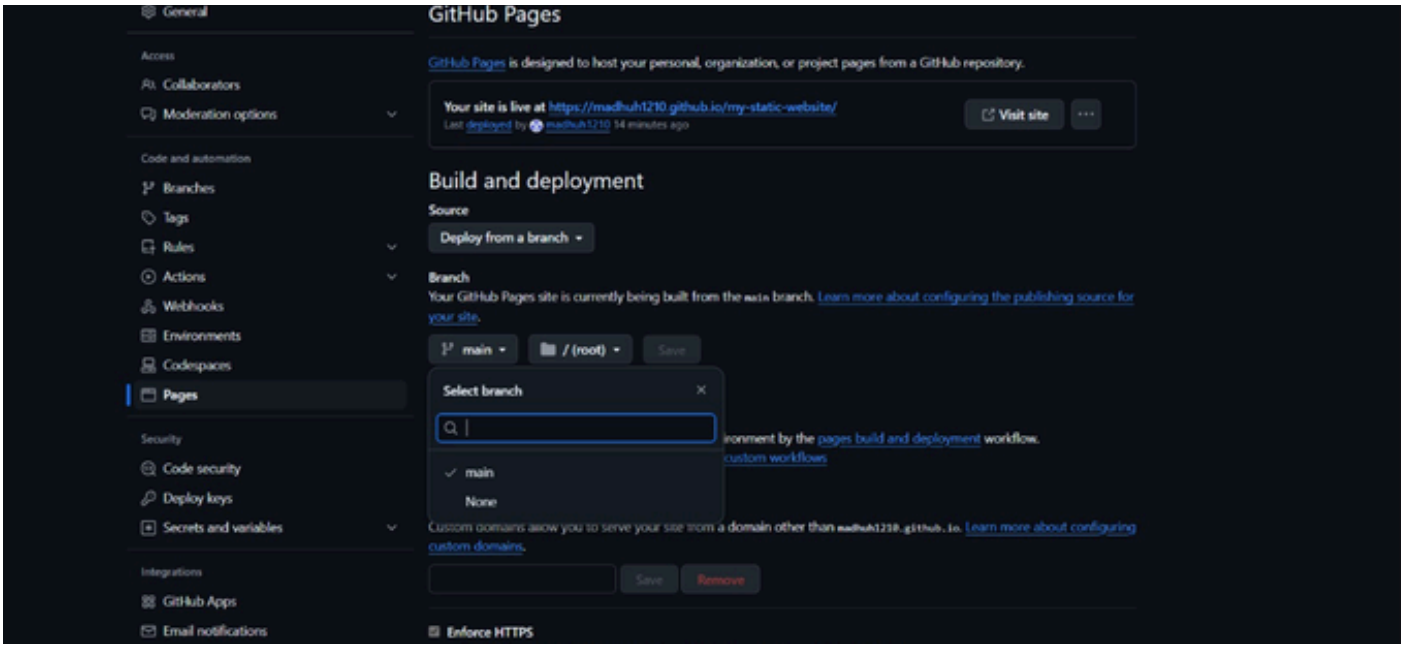
- Push the changes to your **GitHub repository** (this will upload the file to GitHub).

git push -u origin main ->put this command in terminal

Step 7:

7. Configure GitHub Pages

- Go to your **GitHub repository** in your browser.
- Navigate to **Settings** → **Pages**.
- In the **Source** section, select the main branch as the source for your GitHub Pages site.
- Click **Save**.



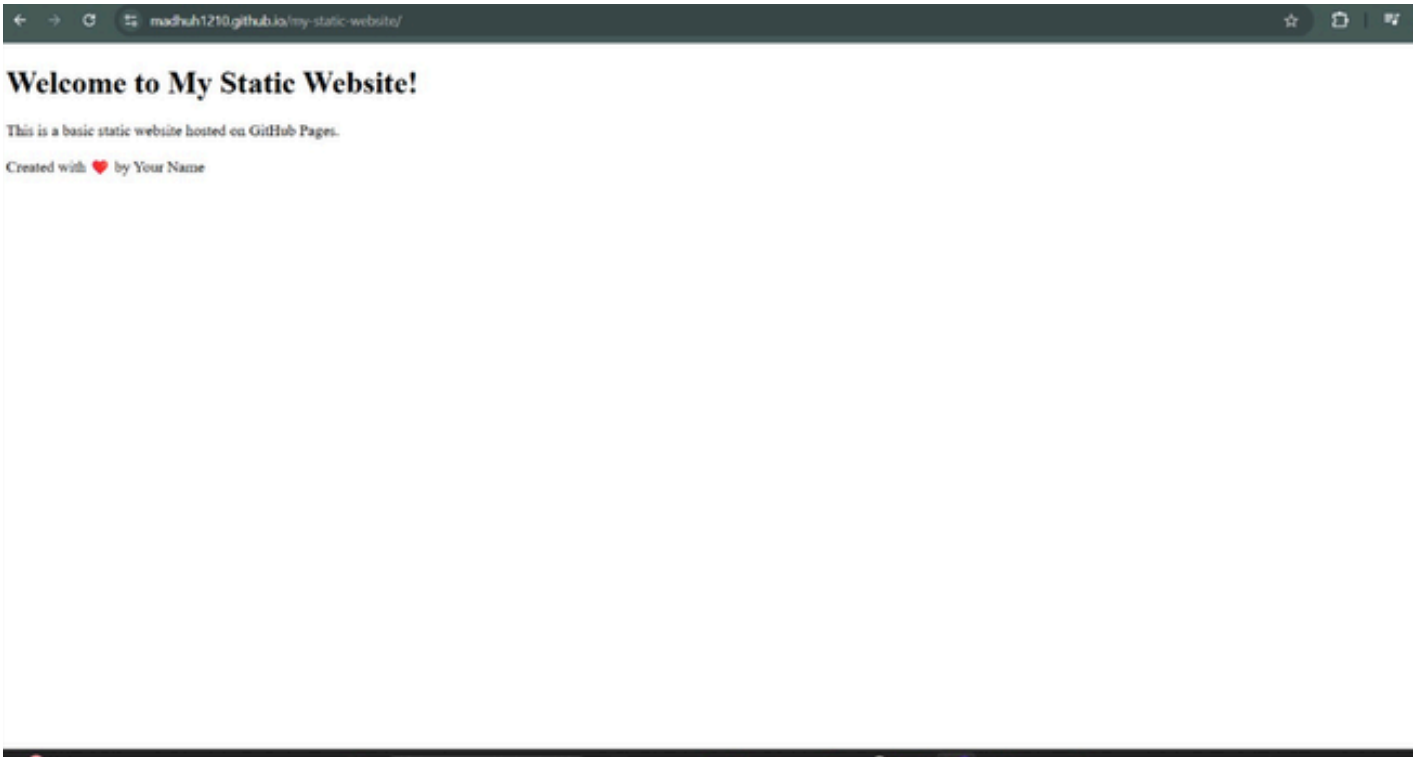
Step 8:

Access Your Website

- After a few minutes, your website will be live at the following UR

https://madhuh1210.github.io/my-static-website/

it be consisting of your github link followed by your my static website repo



Outcomes

Git Repository Initialized: Your folder is now a Git repository with version control.

Remote Repository Added: You can push/pull changes to and from GitHub.

index.html Created: Your basic website content is ready.

Changes Pushed to GitHub: The code is now uploaded to GitHub.

GitHub Pages Set Up: Your static website is now live and accessible at the GitHub Pages URL.