 

**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

**Deploy your static website using GitHub page**

Name: MIRDULA C Department:ADS



GitHub Pages is a free service that allows you to host static websites directly from your GitHub repositories. It's an excellent choice for personal projects, portfolios, documentation, and more. By leveraging GitHub Pages, you can deploy your static website without the need for additional hosting services.

**Overview:**

1. **Create a Repository:** Start by creating a new repository on GitHub. For user or organization sites, name the repository as username.github.io or organization.github.io. For project sites, any repository name will suffice.

[docs.github.com](https://docs.github.com/articles/creating-project-pages-manually?utm_source=chatgpt.com)

1. **Add Your Website Files:** Upload your HTML, CSS, JavaScript, and other assets to the repository. Ensure that your main HTML file is named index.html, as this will serve as the entry point to your site.
2. **Configure GitHub Pages:** Navigate to the repository's settings, find the "Pages" section, and select the branch and folder (if any) that contains your website files. GitHub will then build and deploy your site.

[docs.github.com](https://docs.github.com/articles/creating-project-pages-manually?utm_source=chatgpt.com)

1. **Access Your Site:** After deployment, your site will be accessible at https://username.github.io for user or organization sites, or https://username.github.io/repository-name for project sites.

1. Create a GitHub Repository

* Go to [GitHub](https://github.com).
* Click on the + icon in the top right corner and select **New** repository.
* Give your repository a name (e.g., my-static-website).
* Choose Public.
* Select Initialize this repository with a README.
* Click Create repository.

2. Prepare Your Static Website

* Create a folder on your local machine and add your HTML, CSS, and JavaScript files.
* Make sure the main HTML file is named index.html, as this will be the homepage of your website.

3. Push Your Website Files to GitHub

* Open Git Bash or your terminal.
* Navigate to the folder where your website files are located.
* Initialize a Git repository if you haven't already:

**git init**

* Add your remote GitHub repository:

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

* Add all the files to Git:

**git add** .

* Commit the changes:

**git commit -m "Initial commit"**

* Push the files to GitHub:

**git push -u origin master**

4. Enable GitHub Pages

* Go to your GitHub repository.
* Click on the Settings tab.
* Scroll down to the GitHub Pages section.
* Under Source, select main or master branch.
* Your site will be deployed at **https://yourusername.github.io/my-static-website/.**

5. Access Your Static Website

* After a few minutes, visit the link **https://yourusername.github.io/my-static-website/ to see your deployed static website.**