

<activity04 />

During Class

Activity04 during class

1.- Create a web page with personal information and portfolio of projects.

2.- Create a personal Git hub page: <https://<user>.github.io/personal/>

<https://github.com/sarfarazmohammed880/personal/>

2.- Create a personal Git hub page:

<https://<user>.github.io/personal/>



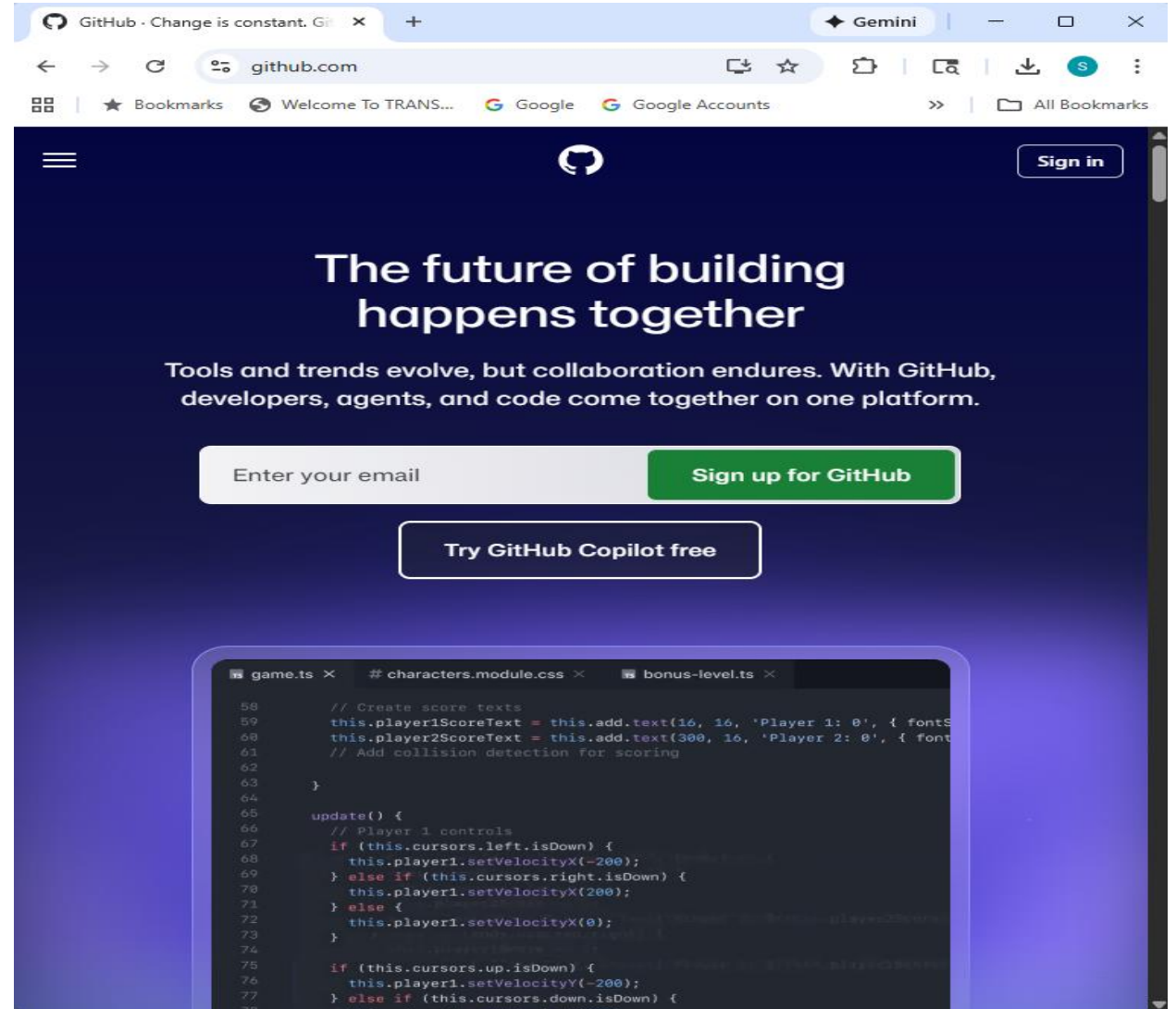
A step-by-step guide to Git web pages



Step 1: Create a GitHub account

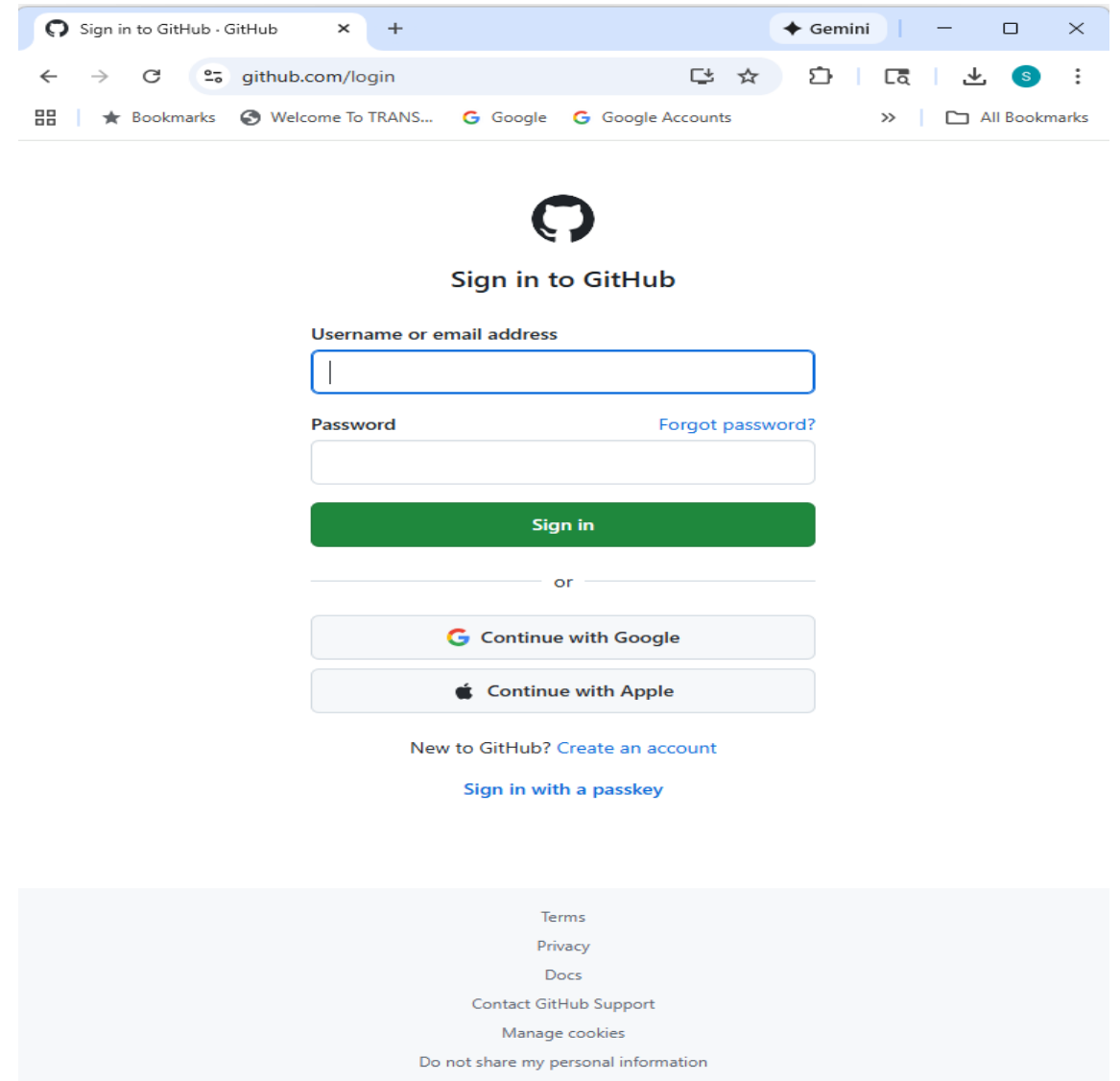
<https://github.com/>

We recommend
using your personal
email ...



If you already have a GitHub account: [login](https://github.com/login)

<https://github.com/login>



The screenshot shows the GitHub login page in a web browser. The browser's address bar displays 'github.com/login'. The page features the GitHub logo at the top, followed by the heading 'Sign in to GitHub'. Below this, there are two input fields: 'Username or email address' and 'Password'. A 'Forgot password?' link is positioned to the right of the password field. A prominent green 'Sign in' button is located below the password field. Underneath the button, the word 'or' is centered, followed by two buttons: 'Continue with Google' and 'Continue with Apple'. At the bottom of the main content area, there are links for 'New to GitHub? Create an account' and 'Sign in with a passkey'. The footer contains a list of links: 'Terms', 'Privacy', 'Docs', 'Contact GitHub Support', 'Manage cookies', and 'Do not share my personal information'.

Sign in to GitHub · GitHub

github.com/login

Sign in to GitHub

Username or email address

Password [Forgot password?](#)

Sign in

or

[Continue with Google](#)

[Continue with Apple](#)

New to GitHub? [Create an account](#)

[Sign in with a passkey](#)

[Terms](#)

[Privacy](#)

[Docs](#)

[Contact GitHub Support](#)

[Manage cookies](#)

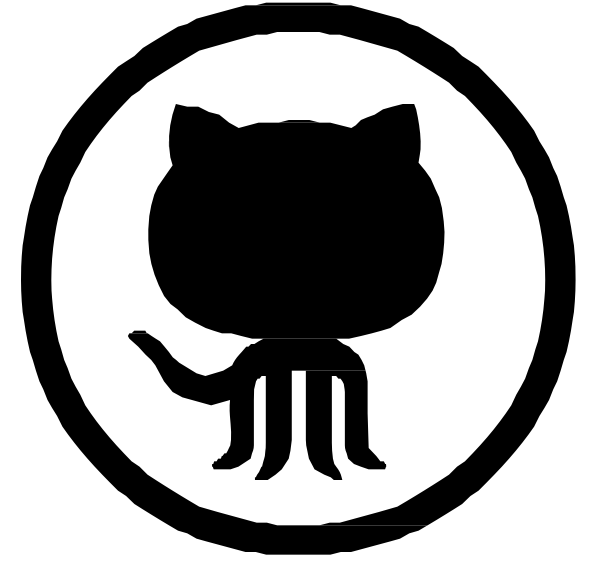
[Do not share my personal information](#)

Publish your Web site

<https://pages.github.com/>

Reference:

<https://stackoverflow.com/questions/8446218/how-to-see-an-html-page-on-github-as-a-normal-rendered-html-page-to-see-preview>



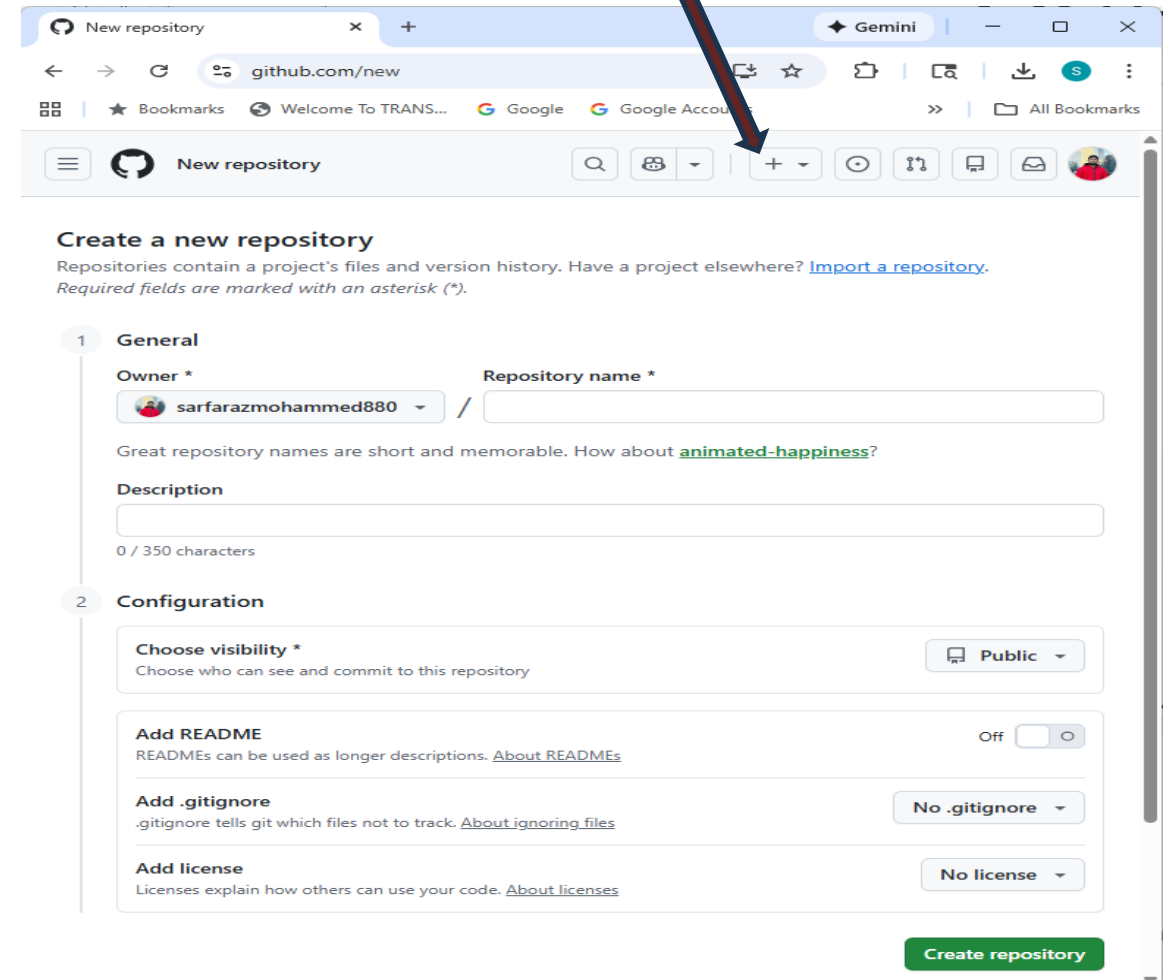
Create repository
– personal

Create a new repository – personal

A repository is like a place or a container where something is stored.

In this case we're creating a Git repository to store **code**.

New repository



The screenshot shows the GitHub 'New repository' page. The browser address bar is 'github.com/new'. The page title is 'New repository'. The main heading is 'Create a new repository'. Below this, it says 'Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository.](#) Required fields are marked with an asterisk (*).

1 General

Owner * sarfarazmohammed880 / Repository name *

Great repository names are short and memorable. How about [animated-happiness?](#)

Description

0 / 350 characters

2 Configuration

Choose visibility * Public

Choose who can see and commit to this repository

Add README Off

READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore No .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

Add license No license

Licenses explain how others can use your code. [About licenses](#)

[Create repository](#)

The new repository will be under the name you just created:

<https://github.com/sarfarazmohammed/personal>

Public →

README →

github.com/new

New repository

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).
Required fields are marked with an asterisk (*).

- General**

Owner * sarfarazmohammed880 / **Repository name *** personal
personal is available.

Great repository names are short and memorable. How about [crispy-octo-invention](#)?

Description

0 / 350 characters
- Configuration**

Choose visibility * Public
Choose who can see and commit to this repository

Add README On
READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore No .gitignore
.gitignore tells git which files not to track. [About ignoring files](#)

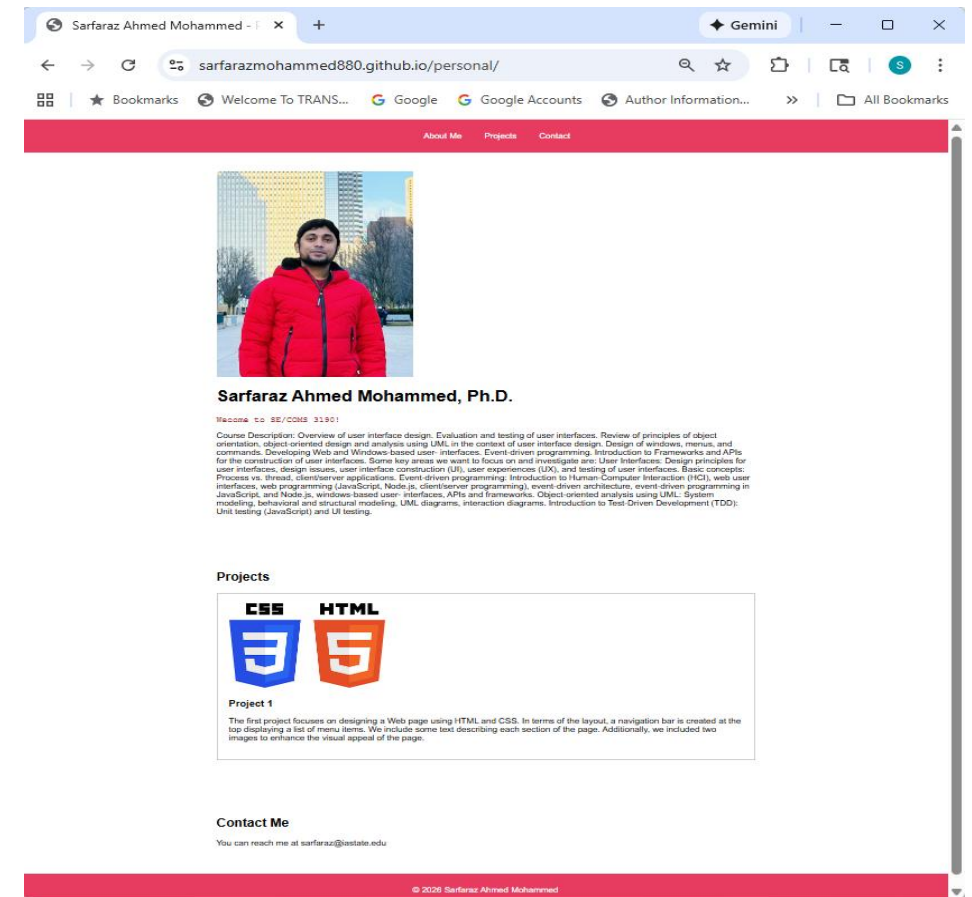
Add license No license
Licenses explain how others can use your code. [About licenses](#)

Create repository

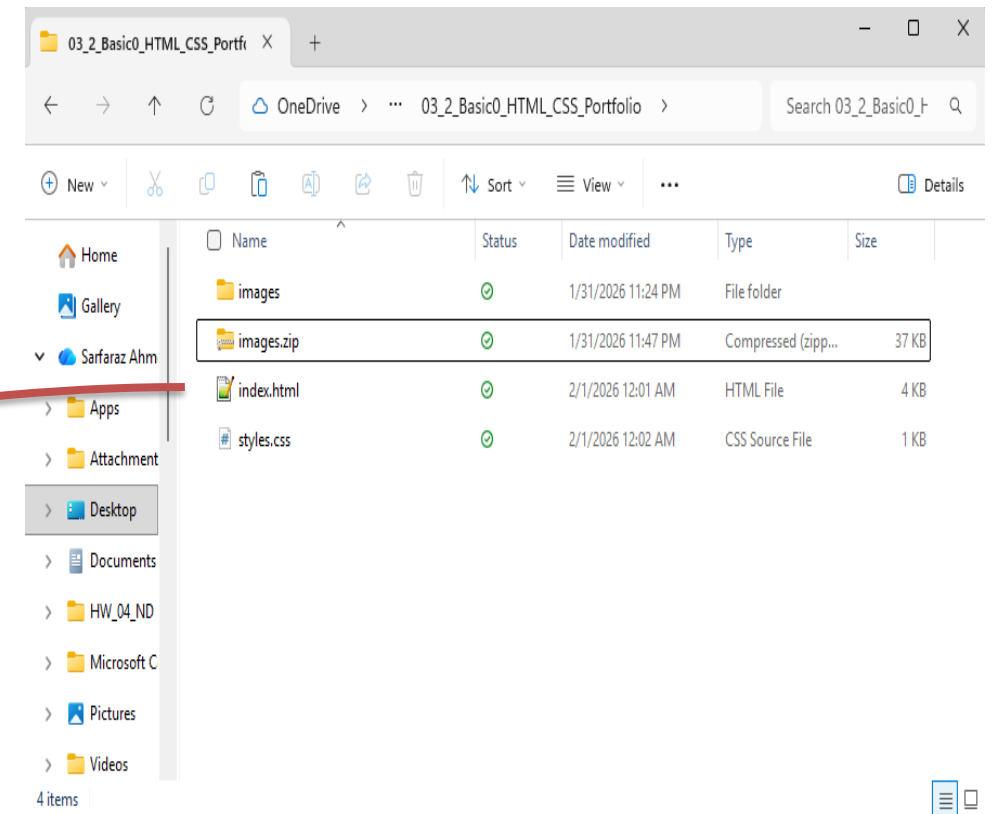
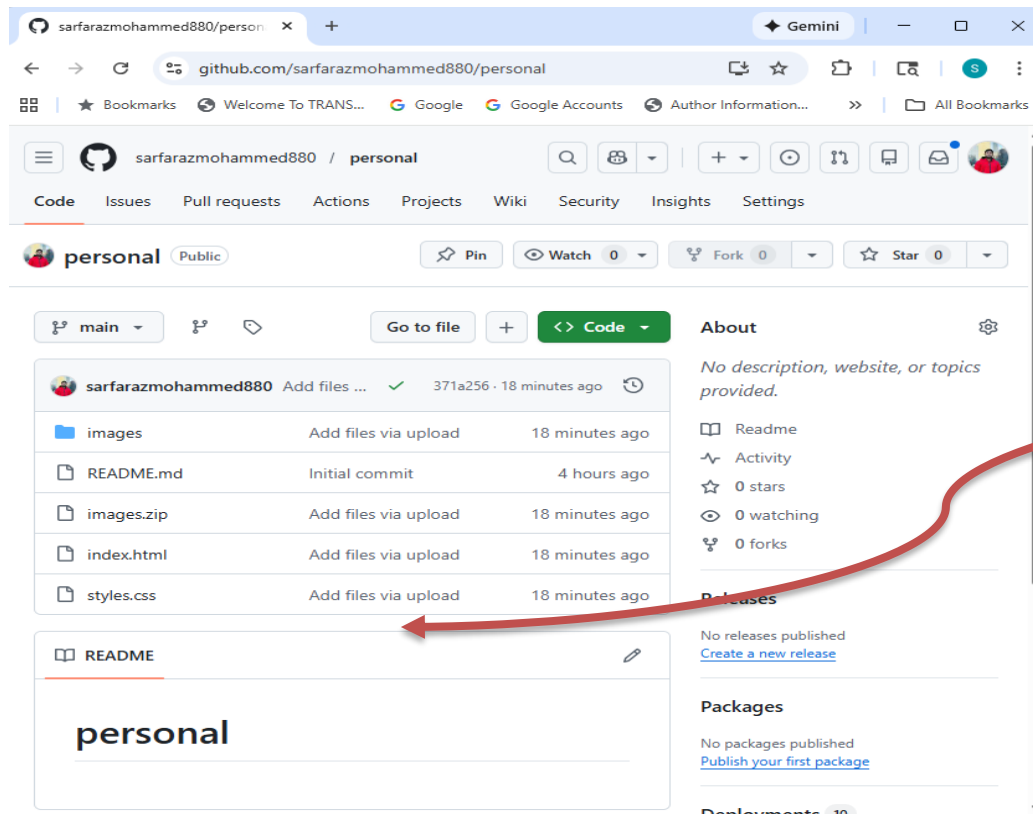
Add your personal web page to the repository `personal` in Github.

`index.html`

`style.css`



Drag files from to repository personal:



Set up your web page:

1 Settings

2 Pages

3 Save

The screenshot shows the GitHub Pages settings page for the user sarfarazmohammed880. The page is titled 'GitHub Pages' and includes a left sidebar with navigation options: General, Access, Collaborators, Moderation options, Code and automation, Branches, Tags, Rules, Actions, Models, Webhooks, Copilot, Environments, Codespaces, Pages (highlighted), and Security. The main content area shows the 'Build and deployment' section, which includes a 'Source' dropdown set to 'Deploy from a branch', a 'Branch' dropdown set to 'main', and a 'Save' button. A red arrow labeled '1' points to the 'Settings' tab in the top navigation bar. A red arrow labeled '2' points to the 'Pages' option in the left sidebar. A red arrow labeled '3' points to the 'Save' button in the 'Build and deployment' section.

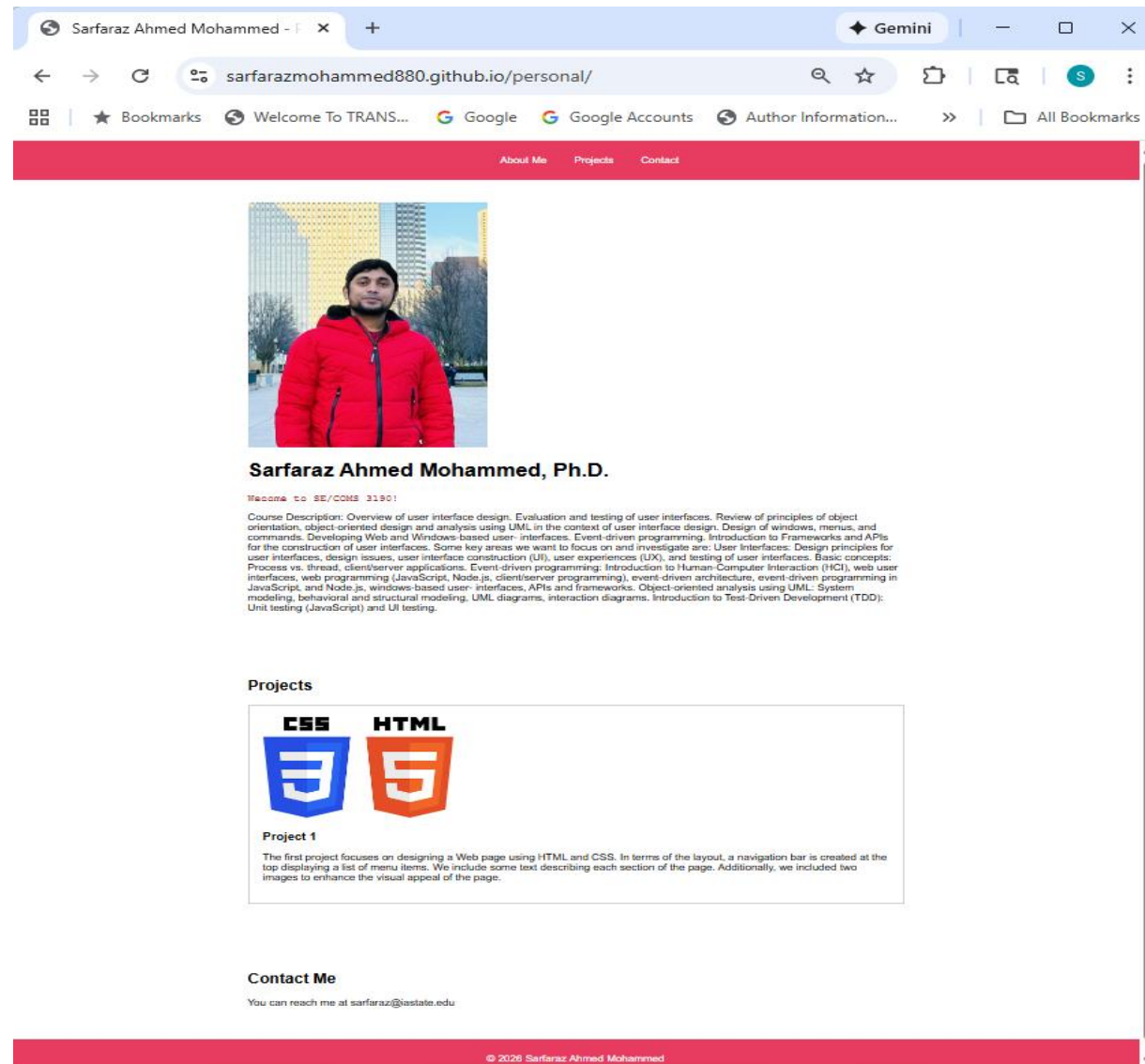
If you have configured GitHubPages you can get your public URL like this:

`https://<username>.github.io/personal/index.html`

e.g.,

`https://sarfarazmohammed880.github.io/personal/index.html`

Show your Web page: <https://sarfarazmohammed880.github.io/personal/index.html>



Sarfaraz Ahmed Mohammed, Ph.D.
Ali Jannesari, Ph.D.



For this Activity upload to Canvas:

- Portfolio URL e.g., <https://sarfarazmohammed880.github.io/personal/index.html>
- and files: index.html, style.css and folder with images