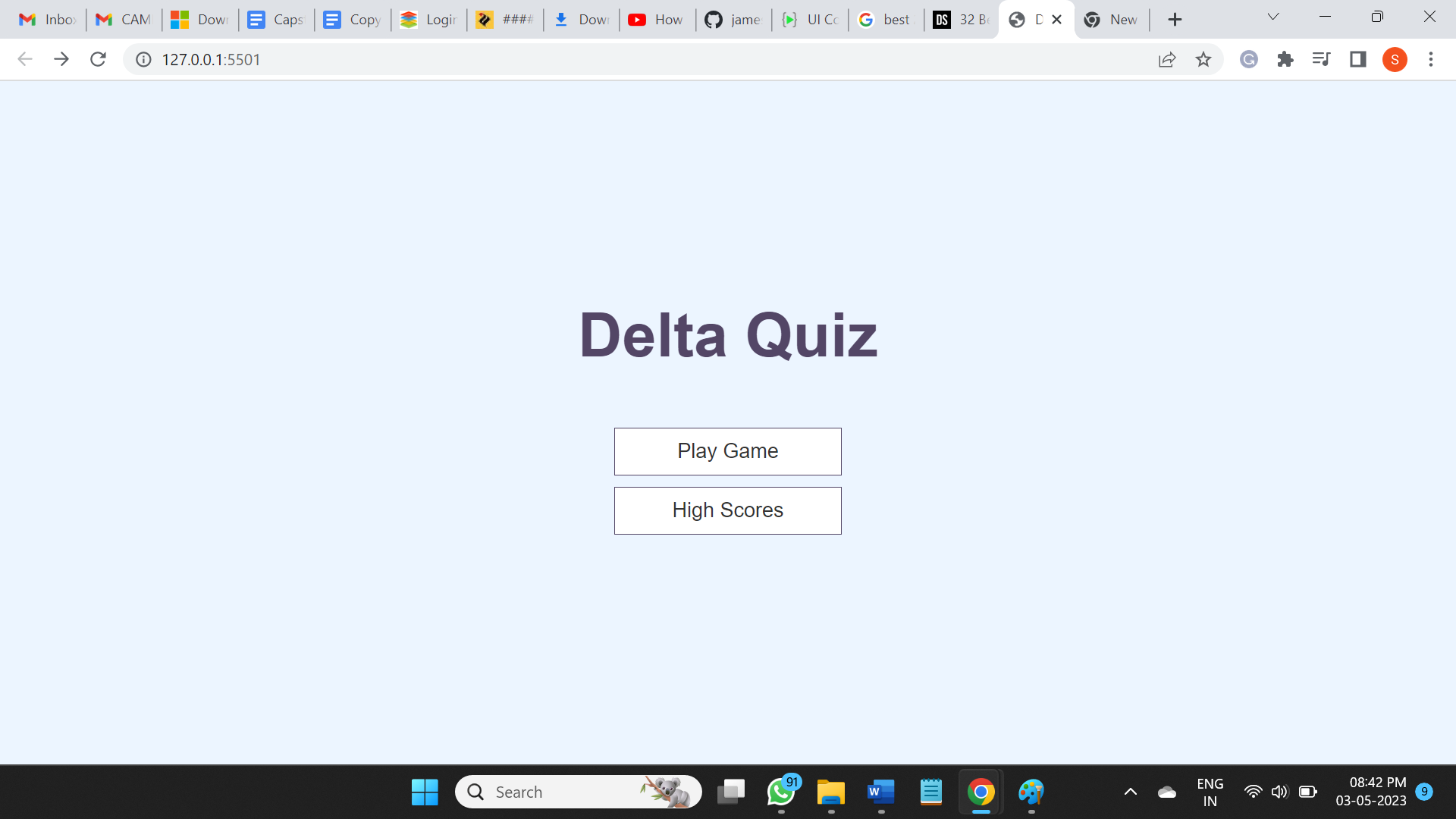
DELTA QUIZ

**Build a Delta Quiz App with HTML, CSS, and JavaScript**

Home Screen



* Save high scores in Local Storage
* Create a progress bar
* Create a spinning loader icon
* Dynamically generate HTML in JavaScript
* Fetch trivia questions from Open Trivia DB API
* JavaScript - Array Functions (splice, map, sort), Local Storage, Fetch API
* ES6 JavaScript Features - Arrow Functions, the Spread Operator, Const and Let, Template Literals
* CSS - Flexbox, Animtations, and REM units

**1. Create and Style the Home Page**

I am create the home page along with a good chunk of the necessary CSS. The home page will consist of a few links for the Game and High Scores pages.

Resources

* Emmet in Visual Studio Code
* Understanding REM Units
* A Complete Guide to Flexbox

**2. Create and Style the Game Page**

I will create the Game Page and display static question and answer information. Eventually, I will load questions from an API.

**3. Display Feedback for Correct/Incorrect Answers**

In this user will check the user's answer for correctness and display feedback to the user before loading the next question.

Resources

* Bootstrap 4 Colors
* Triple vs Double Equals

**4. Create Head's Up Display (HUD)**

In this I am create a Heads Up Display (HUD) for our quiz app. This will display the user's score and current question number.

**5. Create a Progress Bar**

In this I will take HUD one step further by creating a visual progress bar to track the user's progress through the questions.

**6. Create and Style the End Page**

I am create End page where I will display the user's achieved score. This screen will provide a form for saving the score and links for playing again or going home.

Resources

* Local Storage

**7. Save High Scores in Local Storage**

I will save and maintain a high scores array in Local Storage. To do this, I will need to JSON.stringify() and JSON.parse() to convert our high score array to a string and visa versa.

Resources

* Local Storage

**8. Load and Display High Scores from Local Storage**

In this video, we will create our High Scores page. We will have to load the high scores from Local Storage, iterate through them, and display them on the screen.

Resources

* JSON Parse and Stringify
* Array Sort
* Array Map
* Array Join

**9. Fetch API to Load Questions From Local JSON File**

In this, I will Fetch API to load Question from local json file. This will help clean up our Game.js file

**10. Create a Spinning Loader**

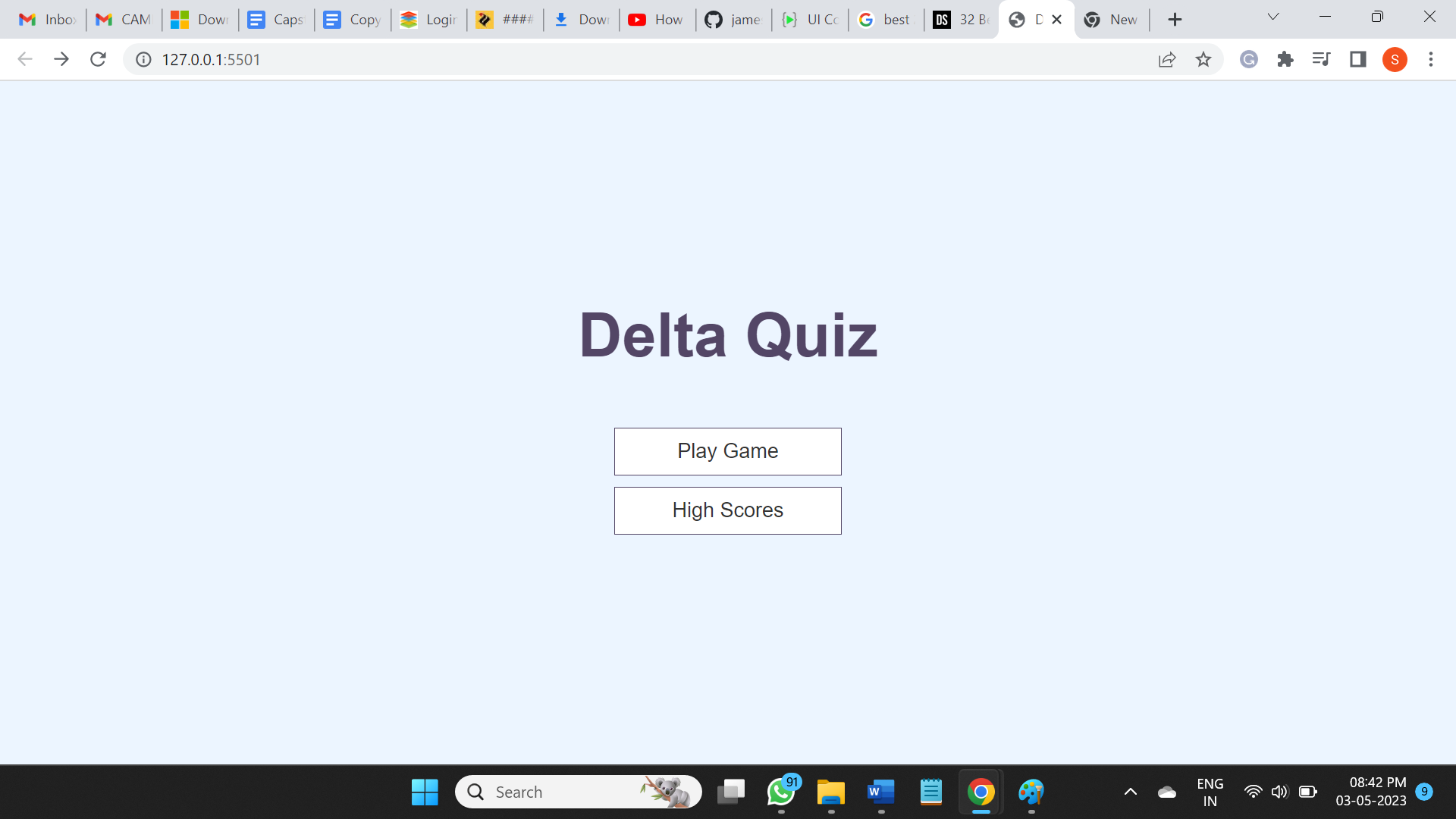
In this, I we will create a simple spinning loader in CSS that will be displayed until we are finished requesting/loading questions from the API.

Resources

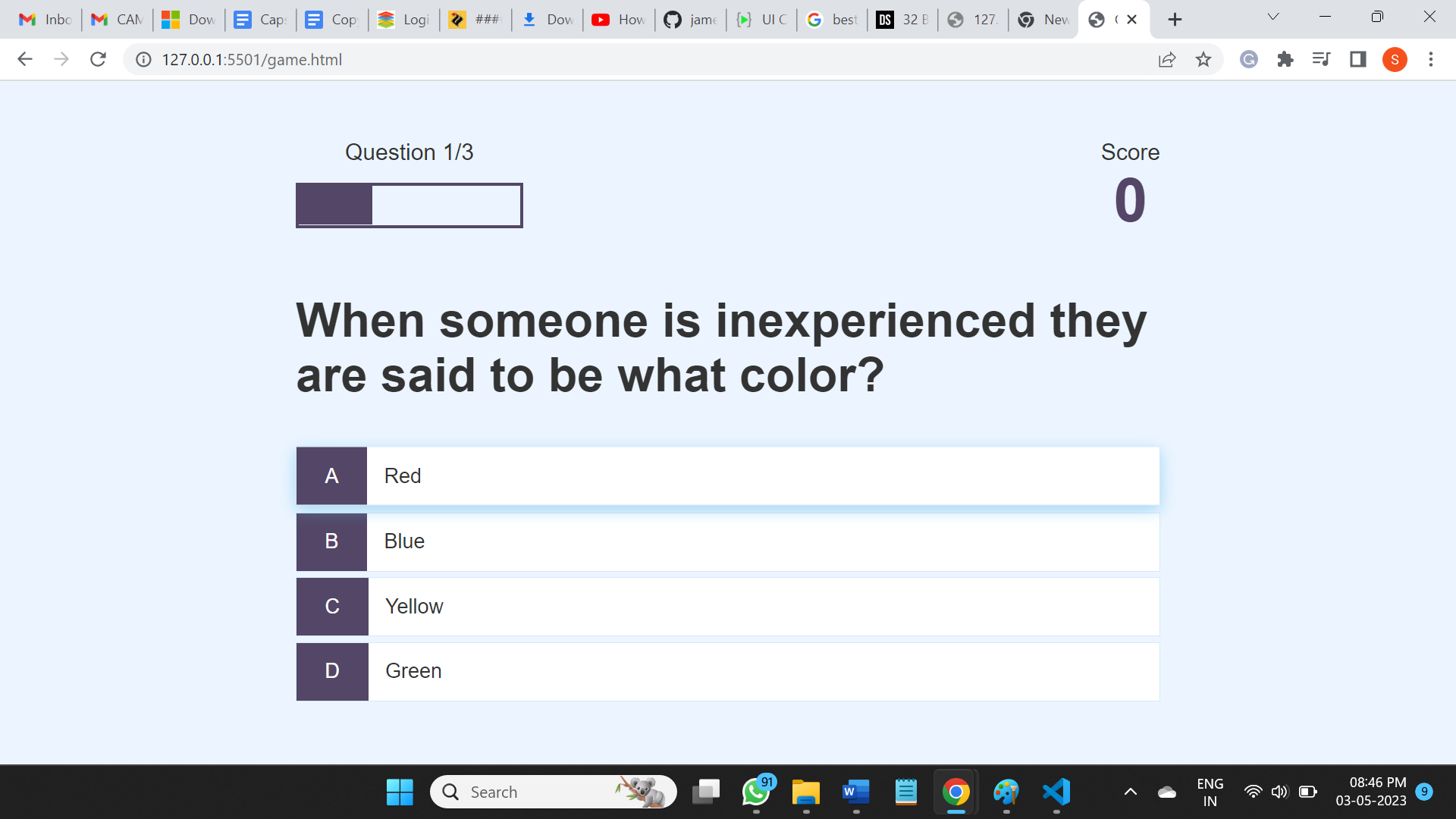
* Create a CSS Loader

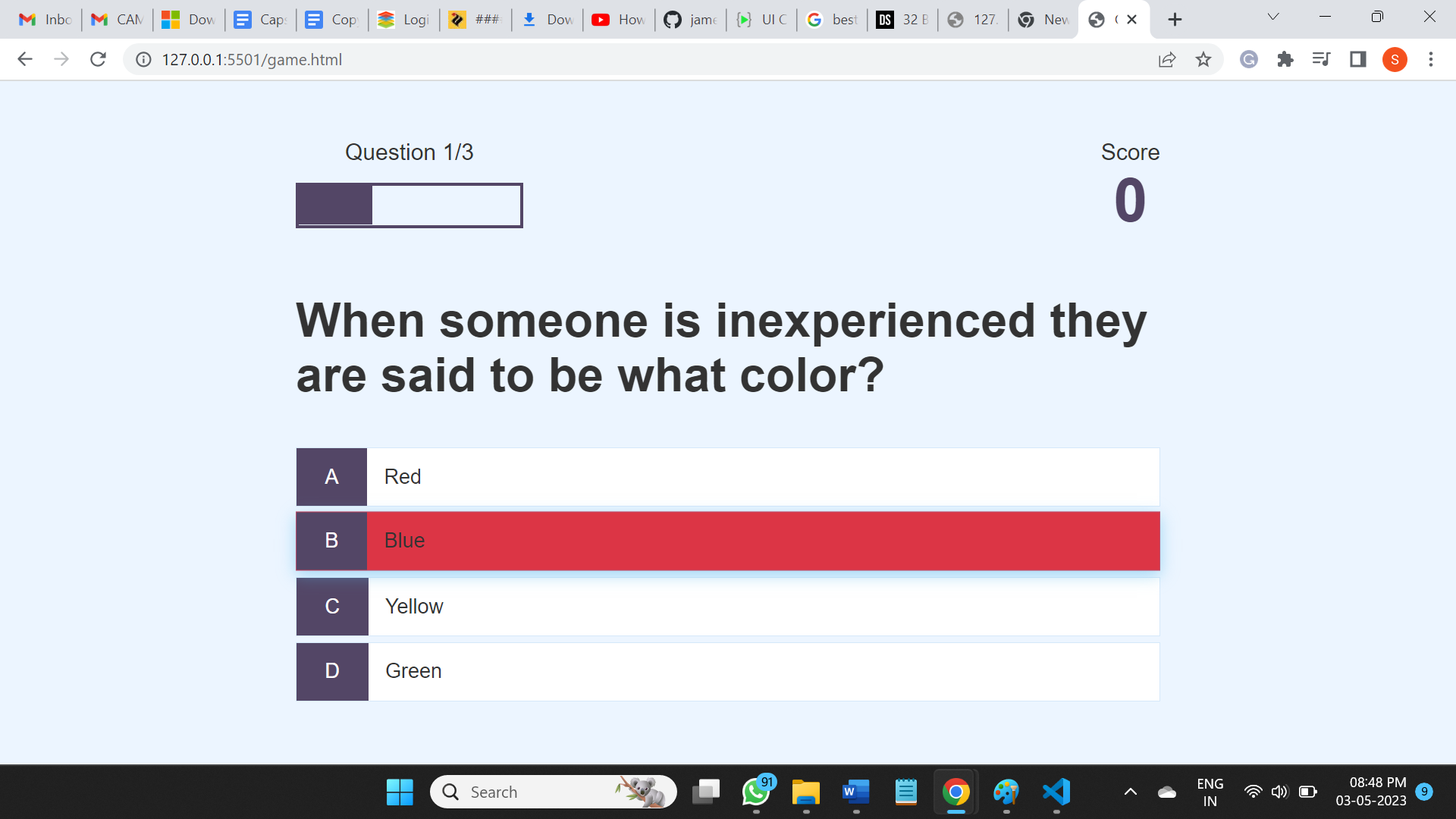
**11.Screenshots of the Project**

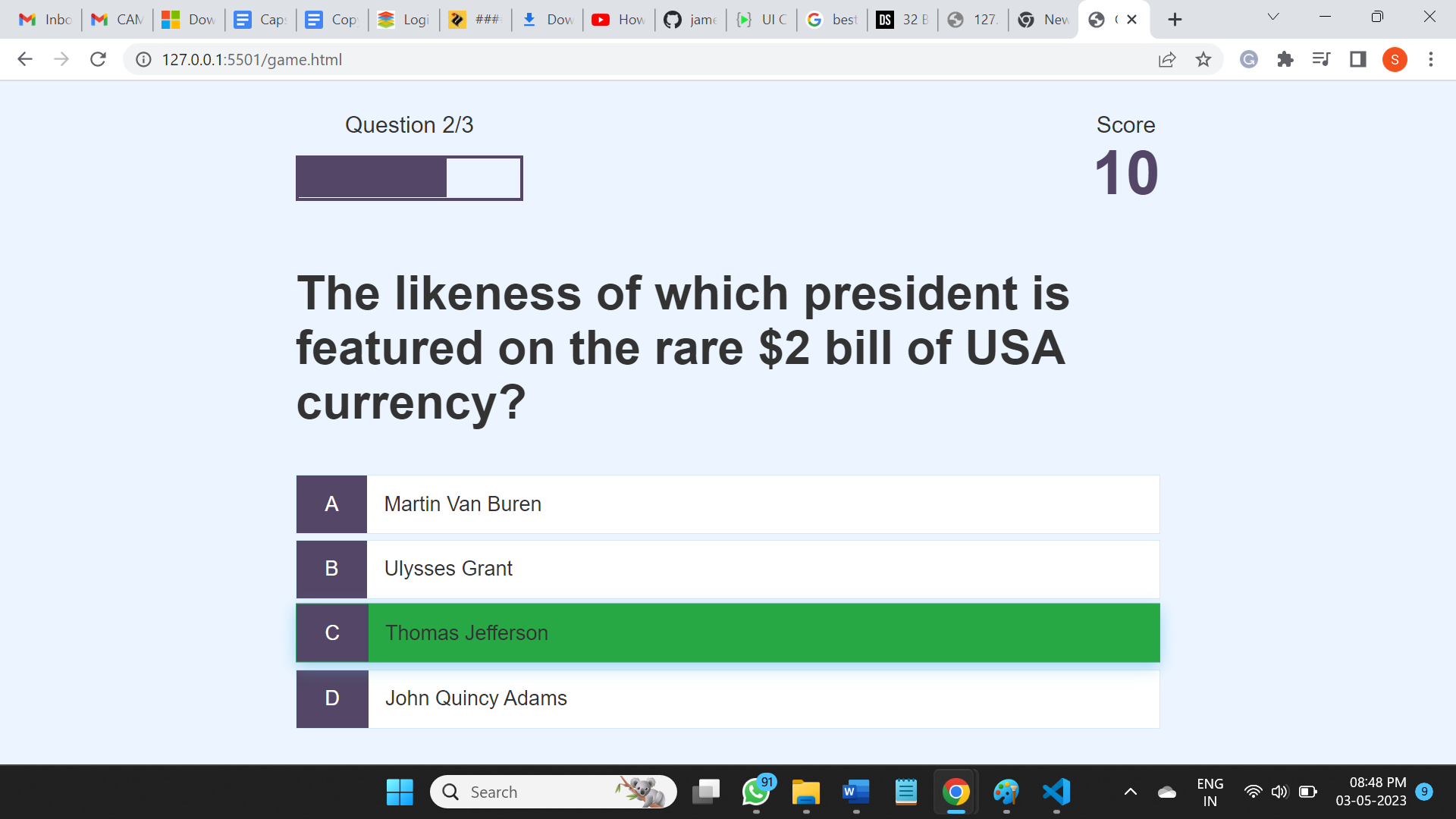
**1.Home page**



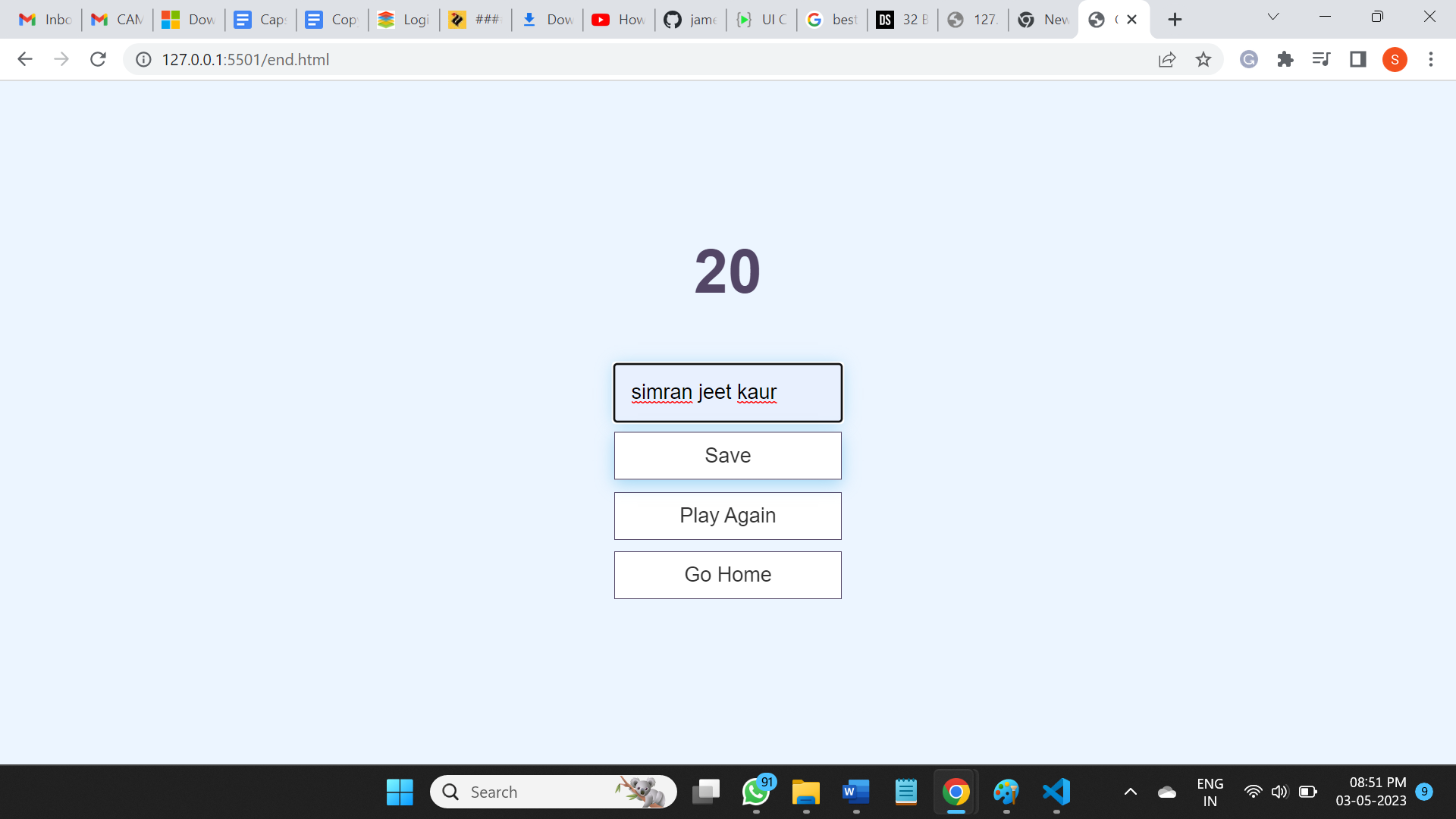
**2.Play Game Pages**





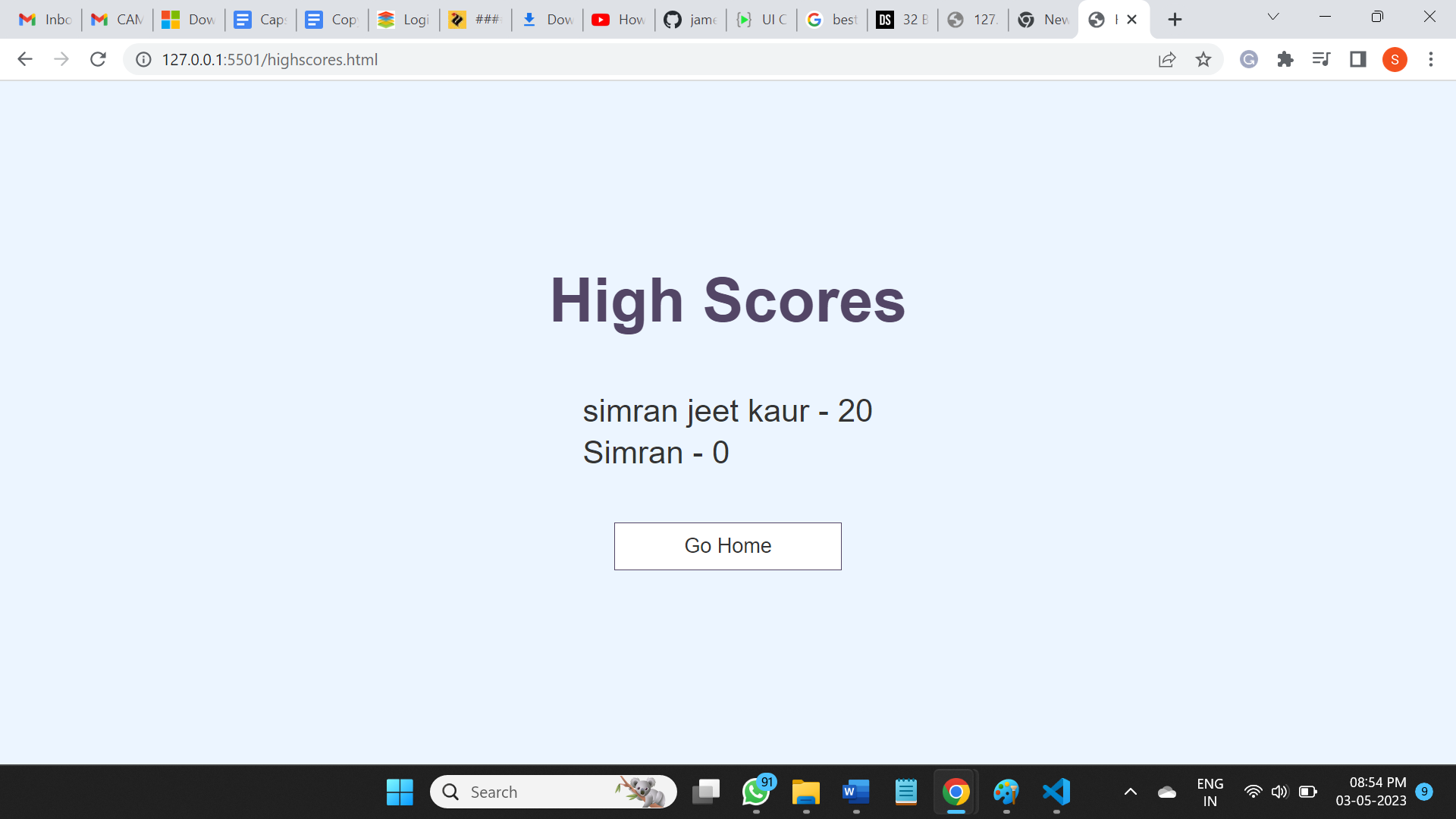


**4.Score Screen After Quiz**



Total Score after quiz

**5. High Score Screen**



**GitHub link :-**