**Movie Search Application**

Scenario: You have been assigned the task of developing a Movie Search Application that allows users to search for movies using a third-party open API. The application should utilize asynchronous JavaScript concepts and make use of the Fetch API to fetch movie data from the API. [OMDb API - The Open Movie Database](https://www.omdbapi.com/)

**Requirements:**

* Design a user interface (UI) for the Movie Search Application. Include elements such as an input field for movie search, a button to initiate the search, and a section to display the movie results.
* Use HTML and CSS to create the UI for the Movie Search Application.
* Implement JavaScript code that handles the user input, fetches movie data from a third-party open API using the Fetch API, and displays the movie results.
* Handle error cases, such as empty search input or failed API requests, gracefully by displaying appropriate error messages to the user.
* Display relevant movie information for each search result, such as the movie title, release year, poster image, and a brief overview.
* Add styling to enhance the visual appeal of the Movie Search Application.
* Test the application by searching for different movies and verifying that the movie results are displayed correctly.
* Provide attribution to the third-party API used in your application, following their usage guidelines and requirements.

Detailed Steps:

* Set up the project structure:
  + Create a new directory for the project.
  + Create an index.html file for the HTML structure.
  + Create a styles.css file for the CSS styling.
  + Create a main.js file for the JavaScript logic.
* Design the user interface:
  + Build the HTML structure for the Movie Search Application, including input fields and display areas.
  + Style the elements using CSS to create an appealing and user-friendly interface.
* Implement JavaScript code:
  + Write JavaScript code in the main.js file to handle user interactions and fetch movie data.
  + Attach an event listener to the button element to handle the click event.
  + Retrieve the user input from the input field and validate it if necessary.
  + Use the Fetch API to send a GET request to a movie API, passing the search term as a parameter.
  + Handle the response using promises or async/await and extract the relevant movie information.
  + Display the retrieved movie data on the webpage.
* Implement error handling:
  + Validate user input to ensure it is not empty or invalid.
  + Handle errors that may occur during the API request or data retrieval, displaying appropriate error messages to the user.
* Style the Movie Search Application:
  + Modify the styles.css file to apply custom styles to the UI elements, making the application visually appealing and responsive.
* Test the application:
  + Open the index.html file in a web browser.
  + Enter different movie search terms in the input field and click the search button.
  + Verify that the movie results are displayed correctly and error messages are shown when necessary.
* Provide attribution:
  + Make sure to follow the usage guidelines and requirements of the third-party API used in your application.
  + Provide proper attribution to the API in your application, such as displaying their logo or mentioning their name.

By following these steps, you will create a Movie Search Application using asynchronous JavaScript and the Fetch API to fetch movie data from a third-party open API. The exercise focuses on implementing the application logic, error handling, and styling. Feel free to customize the design and add additional features as per your requirements.