Async

# Data Fetching :

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Async Function Example 1</title>

</head>

<body>

    <h2>Fetching Data (Async/Await)</h2>

    <p id="output">Waiting for data...</p>

    <script>

        // Example 1: Fetch data using async/await

        async function fetchData() {

            let data = await new Promise((resolve) => {

                setTimeout(() => {

                    resolve("Data fetched successfully!");

                }, 2000); // Wait for 2 seconds

            });

            return data;

        }

        // Call the async function

        fetchData().then(result => {

            document.getElementById("output").innerText = result; // Update the result

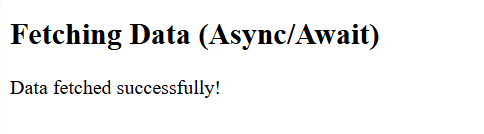
        });

    </script>

</body>

</html>

# OutPut :



# User Authentication :

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Async Function Example 3</title>

</head>

<body>

    <h2>Simulating User Authentication</h2>

    <p id="output">Authenticating user...</p>

    <script>

        // Example 3: Simulate user authentication with async/await

        async function authenticateUser(username, password) {

            // Simulate checking credentials

            let isAuthenticated = await new Promise((resolve) => {

                setTimeout(() => {

                    // Simulate successful login if username is 'user' and password is 'pass'

                    if (username === "user" && password === "pass") {

                        resolve("User authenticated successfully!");

                    } else {

                        resolve("Invalid username or password!");

                    }

                }, 2000); // Wait for 2 seconds

            });

            return isAuthenticated;

        }

        // Simulating authentication attempt

        authenticateUser("user", "pass").then(result => {

            document.getElementById("output").innerText = result; // Display authentication result

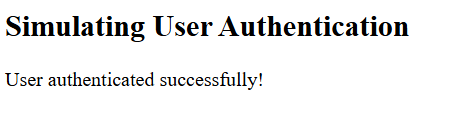
        });

    </script>

</body>

</html>

# OutPut :



# Error handling :

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Async Function Example 2</title>

</head>

<body>

    <h2>Async Function with Error Handling</h2>

    <p id="output">Fetching data...</p>

    <script>

        // Example 2: Async function with try-catch for error handling

        async function fetchData() {

            try {

                let data = await new Promise((resolve, reject) => {

                    // Simulating a failure (error)

                    setTimeout(() => {

                        reject("Failed to fetch data!");

                    }, 2000); // Simulate a failure after 2 seconds

                });

                return data;

            } catch (error) {

                return error; // Catching error and returning the error message

            }

        }

        // Call the async function

        fetchData().then(result => {

            document.getElementById("output").innerText = result; // Display the error message

        });

    </script>

</body>

</html>

# OutPut :

