Assignment

Date: 9 July 2023

Topics: Asynchronous Programming, Promises, Callbacks, fetch

Tasks:

1: Write a function that uses setTimeout to log the message "Hello, World!" after a delay of 2 seconds.

2: Implement a function that uses setInterval to display the current time in the format "HH:MM:SS" every second.

3: Create a function called multiply that takes two numbers as arguments and returns their product. Implement a callback function called handleResult that logs the result to the console after a delay of 1 second.

4: Write a function called countdown that takes a number as an argument and logs the numbers from the given number to 1 with a delay of 1 second between each number. Use promises to handle the asynchronous behavior.

5: Implement a function called fetchData that fetches data from the "https://jsonplaceholder.typicode.com/posts" API using fetch. Handle the response using promises and log the data to the console.

6: Create a function called sum that takes an array of numbers as an argument and returns a promise. The promise should resolve with the sum of all the numbers in the array after a delay of 2 seconds.

7: Implement a function called repeat that takes a function and a number as arguments. The function should execute the given function repeatedly, with a delay of 500 milliseconds between each execution, for the specified number of times.

8: Write a function called fetchRandomUser that uses fetch to retrieve data from the "https://randomuser.me/api/" API. The function should return a promise that resolves with the first name and last name of a random user after a delay of 1 second.

9: Create a function called waitForCondition that takes a condition function and an interval as arguments. The function should repeatedly check the condition every specified interval, and once the condition evaluates to true, it should resolve a promise. Implement a callback function that logs a success message after the condition is met.

10: Write a function called fetchMultipleUrls that takes an array of URLs as an argument. The function should fetch data from all the URLs concurrently using promises and return an array of the resolved data. Each request should have a timeout of 3 seconds, and if any request takes longer than the timeout, it should be skipped and the corresponding element in the result array should be set to null.