

# Callback and Promises

## Assignment Questions



## Assignment 1

Write a function `sumAsync` that takes two numbers as arguments and uses a callback to return their sum after a delay of 1 second.

## Assignment 2

Create a function `getData` that returns a Promise. The Promise should resolve after 2 seconds with a message "Data fetched successfully."

## Assignment 3

Write an asynchronous function `fetchData` that uses the Fetch API to retrieve data from a given URL and returns the parsed JSON response.

### API to be used

<https://jsonplaceholder.typicode.com/todos/1>

## Assignment 4

Write an asynchronous function `fetchData` that uses the Fetch API to retrieve data from a given URL (<https://jsonplaceholder.typicode.com/todos/1>) and returns the parsed JSON response.

## Assignment 5

Implement a function `multiplyWithCallback` that takes an array of numbers and a callback function. The function should multiply each element of the array by 2 and pass the result to the callback.

## Assignment 6

Create a function `fetchUserDataAndPosts` that takes a user ID and fetches the user details and their posts using separate API calls. Use promise chaining to ensure the posts are retrieved only after the user details are fetched. Return an object with user details and posts.

### API to be used

For user: [https://jsonplaceholder.typicode.com/users/\\${userId}](https://jsonplaceholder.typicode.com/users/${userId})

For post: [https://jsonplaceholder.typicode.com/posts?userId=\\${userId}](https://jsonplaceholder.typicode.com/posts?userId=${userId})

## Assignment 7

Write a function `fetchMultipleData` that takes an array of URLs and uses `Promise.all()` to fetch data from all the URLs concurrently. Return an array of responses.

### API to be used

Change todo id for each API call

<https://jsonplaceholder.typicode.com/todos/1>

## Assignment 8

Create a function `racePromises` that takes an array of promises and returns the result of the first promise that resolves or rejects. Use `Promise.race()` to implement this.