

Programs to practice in the Lab

Topics Covered: Callbacks and Callback Hell

Question 1: Write a function greet that takes a name and a callback function. The function should return a greeting message by calling the callback with the message.

Question 2: Create three functions that take a number, multiply it by 2, subtract 3, and then add 10. Use callbacks to chain these operations together.

Question 3: Write a function delayedMessage(message, delay) that prints the message after the specified delay using setTimeout. Use a callback for when the message has been printed.

Question 4: Write a function fetchDataWithCallback that simulates fetching data from an API with a delay. Implement error handling in the callback, allowing it to receive either a successful response or an error message. Test the function by simulating both success and failure scenarios.

Question 5: In a payment processing system, you need to:

Authenticate the user, verify the validity of the payment method, process the payment, update the user's account balance, and notify the user of the payment result via email.

Design this payment processing system using callbacks to ensure that each step occurs in the correct order.

Question 6: Build a course enrolment system for a university. The process involves:

- Checking if the student meets the prerequisites for the course.
- If prerequisites are met, check if the course has available seats.
- Reserve the student's seat in the course.
- Add the student to the course roster.
- Send a confirmation email to the student.

Implement the course enrolment process using nested callbacks.

Question 7: Develop a school management system where the following tasks need to be performed:

- Register a new student.
- Assign the student to a class and a tutor.
- Add the student to the class schedule.
- Track the student's grades and attendance.
- Generate and send a report card at the end of the term.

Structure this workflow using callbacks. Ensure each step happens in order.

Question 8: Develop a JavaScript program to simulate student examination result processing using callbacks, similar to a university results portal.
Functional Modules (Each MUST use callbacks):

Fetch Student Details

- Input: studentId
- Return name, department
- Delay: 1 second

Fetch Marks

- Return marks for 5 subjects
- Delay: 2 seconds

Calculate Result

- Calculate:
 - Total
 - Average
 - Pass/Fail (pass if all subjects ≥ 40)

Assign Grade

- Based on average:
 - $\geq 75 \rightarrow$ Distinction
 - 60–74 \rightarrow First Class
 - 50–59 \rightarrow Second Class
 - Else \rightarrow Fail

Generate Result Sheet

- Display full result details after 1 second

Handle invalid student ID

Handle missing marks

Use callback error-first pattern: `callback(error, result)`

#####@##@#