Here are practice problems on JavaScript functions, callback hell, promises, and async/await:

Functions Practice

1. Write a function addNumbers that takes two numbers as arguments and returns their sum.
2. Create a function greet that takes a name as an argument and logs a personalized greeting message.
3. Write a function isEven that takes a number as an argument and returns true if it's even, false otherwise.
4. Create a function filterEvens that takes an array of numbers as an argument and returns a new array with only the even numbers.
5. Write a function calculateArea that takes the radius of a circle as an argument and returns its area.

Callback Hell Practice

1. Write a function getUserData that takes a callback function as an argument. The callback function should be called with an object containing user data.
2. Create a function getProductDetails that takes a product ID and a callback function as arguments. The callback function should be called with an object containing product details.
3. Write a function authenticateUser that takes a username, password, and a callback function as arguments. The callback function should be called with a boolean indicating authentication success.
4. Create a function fetchData that takes a URL and a callback function as arguments. The callback function should be called with the fetched data.
5. Write a function sendNotification that takes a message and a callback function as arguments. The callback function should be called with a boolean indicating notification success.

Promises Practice

1. Write a function getUserData that returns a promise resolving with an object containing user data.

2. Create a function getProductDetails that returns a promise resolving with an object containing product details.
3. Write a function authenticateUser that returns a promise resolving with a boolean indicating authentication success.
4. Create a function fetchData that returns a promise resolving with the fetched data.
5. Write a function sendNotification that returns a promise resolving with a boolean indicating notification success.

Async/Await Practice

1. Write an async function getUserData that returns an object containing user data.
2. Create an async function getProductDetails that returns an object containing product details.
3. Write an async function authenticateUser that returns a boolean indicating authentication success.
4. Create an async function fetchData that returns the fetched data.
5. Write an async function sendNotification that returns a boolean indicating notification success.

Mixed Practice

1. Write a function processData that takes an array of numbers, filters out odd numbers, and returns the sum of the remaining numbers using callbacks.
2. Create a function fetchAndProcessData that fetches data from an API, processes the data using promises, and logs the result.
3. Write an async function authenticateAndFetchData that authenticates a user, fetches data from an API, and logs the result.