Short Answer:

Answer the following questions with complete sentences in **your own words**. You are encouraged to conduct your own research online or through other methods before answering the questions. If you research online, please consult multiple sources before you write down your answers.

1. **What are the disadvantages of synchronous code?**

- Slow load time
- No built-in callback methods.
- When a thread is locked, others get blocked as well.
- Cannot execute multiple operations at the same time which can affect user experience.
- Once a request fails, the entire program becomes unresponsive
- huge resources is needed to handle more threads if requests become overwhelming.

2. **What is asynchronous code in JavaScript?**

asynchronous programming provides opportunities for a program to continue running other code while waiting for a long-running task to complete.

3. **How does JavaScript achieve asynchronous code?**

Callbacks: provide functions to call once the asynchronous method has finished running
promises: chain together promise methods together using then()
async/await keywords: provides a cleaner method of writing promises and gives the user more control over execution order.

4. **What does the event loop do? What data structures does it use?**

The Event Loop monitor the Call Stack and the Callback Queue. If the Call Stack is empty, the Event Loop will take the first event from the queue and will push it to the Call Stack, which effectively runs it.

5. **What is the callback queue?**

A callback queue is a queue of tasks that are executed after the current task. The callback queue is handled by the JavaScript engine after it has executed all tasks in the microtask queue

6. **What is an HTTP request and HTTP response?**

An HTTP request is made by a client, to a server. The aim of the request is to access a resource on the server in order to receive or submit data.
-what a browser send to a server to retrieve or submit data.

An HTTP response is to provide the client with the resource it requested with the HTTP request or to inform the client that the action it requested has been carried out; or to inform the client that an error occurred with the request.
-what browers receive from a serve to from that the request was received

7. **How many HTTP methods are there? Explain each one.**
    1. **What is the difference between GET and POST? What about POST and PUT?**

There are 5 HTTP methods
-Get: to receive the data
-Post: to create the data;
-Put: to update or edit the data;
-Delete: to delete the data;

8. **Could you explain the different classes of HTTP status codes? What are some common status codes?**

Code 100-199: information**;**
Code 200-299: Success; ex. 200: okay; 201: Created
Code 300-399: Redirect;
Code 400-499: Client error; ex. 401: Bad Request; 401: Unauthorized; 403: Forbidden; 404: Not Found
Code 500-599: Server Error; 500: Server Error; 503: Server not available

9. **What is AJAX?**
10.

AJAX (Asynchronous JavaScript and XML) help create better web applications dealing with asynchronous functions with XML, HTML, CSS, XMLHTTP Request. and Java Script

11. **What is XHR?**

XHR is a JavaScript API to create AJAX requests to get data.

12. **What is a Promise?**

A Promise is an object that is used to deal with asynchronous operations in JavaScript

13. **How many states does a Promise have? What are they?**

Promises have 4 states.

fulfilled: promise succeeded
rejected: promise failed
pending: Promise is pending
settled: Promise has been settled by either being fulfilled or rejected

14. **What is callback hell?**

Callback hell is multiple callback functions that depends on each other, are nested together. This issue makes the code hard to read and it is easy to get lost.

15. **What is the advantage of Promises over callbacks?**

Promises can handle multiple asynchronous operations easily and provide better error handling than callbacks and events

16. **Explain Promise.all() vs Promise.allSettled().**

The Promise.all() method takes a list of promises as an input, and returns a single Promise that is reduced to an array of the results of the input promises.

Promise.allSettled() returns a list of promises that have been **resolved** after all of the given promises have either been fulfilled or rejected, with an array of objects that describes the outcome of each promise.

The big difference between them is that Promise.all() will reject once a promise is rejects while Promise.allSettled() is will never reject.

17. **What is the Microtask Queue?**

A microtask Queue manages data structure where microtasks areadded and executed. They are processed *after callbacks/scripts/timeouts* as long as no other JavaScript is running, and at the end of each task.

18. **What is the difference between making server requests via fetch and XHR?**

Fetch allows for cleaner and simpler API request and calls by avoiding callback hell and having to remember the complex API of XMLHttpRequest.

19. **What is async & await? How do we use them?**

Async means that the function will always return a promise. Await is what makes the JavaScript code wait until the promise is finished.

They are used only in async function with a try and catch.

20. **Explain localStorage vs sessionStorage.**

SessionStorage: stores memory when the tab is opened and is removed when the tab is closed.

LocalStorage: store memory even when the browser is closed.

21. **What do the async and defer attributes do when loading scripts?**

Async loads script when the HTML hasn’t been fully parsed.
Defer loads script when the HTLM is fully parsed.

22. **What are ES6 modules? Why are they useful? How do we make these modules available in other JS files?**

ES6 modules are files containing JavaScript. They are useful because it helps organizes JavaScript code. These modules are available in other JS files through export and import.

Coding Questions:

Use HTML/CSS/JS to solve the following problems. You are highly encouraged to present more than one way to answer the questions. Please follow best practices when you write the code so that it would be easily readable, maintainable, and efficient. Clearly state your assumptions if you have any. You may discuss with others on the questions, but please write your own code.

1. Given a url **"https://jsonplaceholder.typicode.com/users"**, send a GET request to display the data on the page in a table. Errors should be handled properly.
   - Do this with fetch and XHR.

2. Create a webpage with text input and a search button. When you input a user ID and click search, it should display that user's information, posts, and todos all in the same page in a table. **Hint: Promise.all() or Promise.allSettled().**

- For example, when the user types 2, display the data from the following urls:
  https://jsonplaceholder.typicode.com/users/2
  https://jsonplaceholder.typicode.com/posts?userId=2
  https://jsonplaceholder.typicode.com/todos?userId=2

- If the user ID is invalid (no data in the response), there should be an error message says “User was not found. Please try another user ID”.

3. Implement a function **delayedRequest(url)** that retrieves data from the specified url and outputs it to the console after 2 seconds.
- Test it with any of the “https://jsonplaceholder.typicode.com/users" urls.

Paired Programming:

Use JS to solve the following problems with your partner. Please remember to label who was acting as the driver and navigator for each problem and write down your feedback on their performance. **Feedback will be confidential**.

Leetcode #66: Plus One
Leetcode #70: Climbing Stairs