Common JS interview questions

What are the different data types in JavaScript?

JavaScript offers primitive data types like strings, numbers, booleans, null, undefined, and Symbol (introduced in ES6). It also has objects, which are non-primitive and can hold complex data structures.

Explain the difference between == and === operators.

Both == and === perform comparisons, but === is for strict equality, considering both value and type. In contrast, == loosely compares values, allowing for type coercion (implicit conversion) that can lead to unexpected results.

What are closures in JavaScript?

A closure is a function that has access to variables from its outer (enclosing) function, even after the outer function has returned. This allows the inner function to remember and manipulate the outer function's variables, creating a private scope.

How does the this keyword work in JavaScript?

The this keyword refers to the current object context where a function is being executed. Its value depends on how the function is called. Understanding this is essential for working with object methods and event handlers.

Explain prototypal inheritance in JavaScript.

JavaScript uses a prototype-based inheritance model. Objects inherit properties and methods from their prototype objects. If a property or method is not found in the current object, JavaScript searches up the prototype chain to locate it.

What is asynchronous programming in JavaScript?

Asynchronous programming allows code to run operations (like network requests or user input) without blocking the main program thread. This keeps your web application responsive while waiting for these tasks to complete.

What are the advantages of using let and const keywords over var?

let and const (introduced in ES6) offer block-level scoping, preventing variable hoisting issues that can occur with var. They also improve code clarity by indicating whether a variable can be reassigned (let) or not (const).

Explain the concept of hoisting in JavaScript.

Hoisting refers to the behavior where variable declarations using var are lifted to the top of their scope (function or global scope) during code parsing. This can lead to unexpected results if not handled carefully. let and const are not hoisted.

What's the difference between call, apply, and bind methods?

These methods are used to define the this keyword when calling a function. call and apply allow you to pass arguments individually (for call) or as an array (for apply). bind creates a new function with a pre-defined this value.

Follow For More