@Aakash Kanojiya

Top 30 Interview Questions for JavaScript !!!

>>>

@Aakash Kanojiya

What is JavaScript and how is it used in web development?

JavaScript is a programming language used to create interactive effects within web browsers. It can be used for front-end development to create interactive user interfaces, and also for back-end development to create server-side applications. JavaScript is one of the three core technologies used in web development, alongside HTML and CSS.

What is the difference between JavaScript and other programming languages?

JavaScript is a dynamically-typed, high-level programming language that is used mainly for front-end and back-end web development. It is unique in that it is executed on the client-side, meaning that it runs within a web browser. Other programming languages may be compiled or interpreted and can be used for a wide variety of applications, such as desktop or mobile applications.

>>>

What is the Document Object Model (DOM) in JavaScript?

The Document Object Model (DOM) is a programming interface for HTML and XML documents. It represents the page so that programs can change the document structure, style, and content. The DOM provides a way for programmers to access and manipulate HTML and XML documents, which makes it essential for building dynamic web pages and applications.

What are the different data types in JavaScript?

JavaScript has seven different data types: undefined, null, boolean, number, string, object, and symbol.

How do you declare variables in JavaScript?

Variables can be declared using the var, let, or const keywords in JavaScript. The var keyword is used for declaring variables with global or functional scope, while let and const are used for block-scoped variables.

What is hoisting in JavaScript?

Hoisting is a behavior in JavaScript where variable and function declarations are moved to the top of their respective scopes. This means that a variable or function can be used before it has been declared, although it will still be undefined until it is assigned a value.

What is scope in JavaScript?

Scope refers to the accessibility of variables, functions, and objects in different parts of a JavaScript program. There are two types of scope in JavaScript: global scope and local scope.

What are closures in JavaScript?

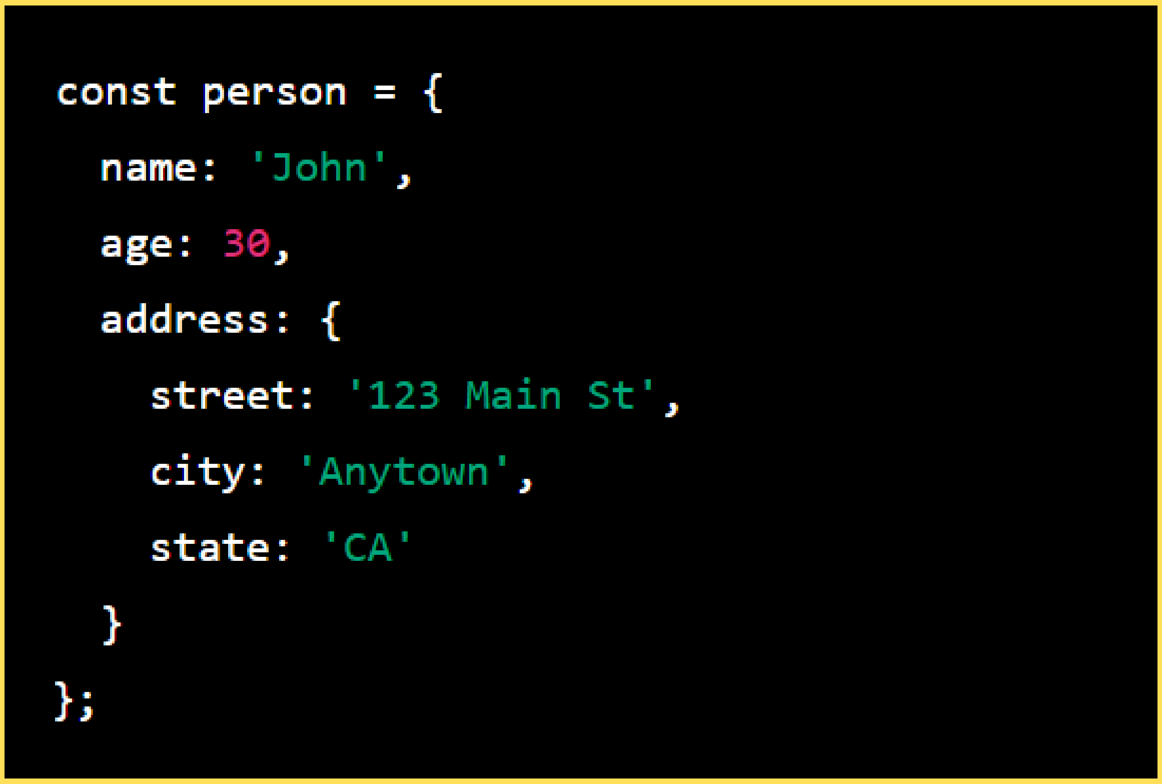
Closures are functions that have access to variables in their outer (enclosing) functions, even after the outer function has returned. This is possible because the inner function retains a reference to the variables in the outer function, which are kept alive in memory.

What is the difference between an object and an array in JavaScript?

An object is a collection of properties, where each property is a key-value pair. An array is an ordered list of values, where each value is accessed by its index number.

How do you create an object in JavaScript?

Objects can be created using object literals or using the Object constructor function. For example, an object can be created using an object literal like this:



```
const person = {
  name: 'John',
  age: 30,
  address: {
    street: '123 Main St',
    city: 'Anytown',
    state: 'CA'
  }
};
```

What is prototypal inheritance in JavaScript?

Prototypal inheritance is a way of creating new objects based on existing objects, by inheriting their properties and methods. In JavaScript, every object has a prototype, which is a reference to another object that it inherits properties and methods from.

@Aakash Kanojiya

What is the difference between synchronous and asynchronous programming in JavaScript?

Synchronous programming is when tasks are executed sequentially, one after the other, in a predictable order. Asynchronous programming is when tasks are executed concurrently, and the order of execution is not guaranteed.

What is a promise in JavaScript?

A promise is a JavaScript object that represents a value that may not be available yet but will be resolved in the future.

What is async/await in JavaScript?

Async/await is a way to write asynchronous code that looks like synchronous code. It allows developers to write code that waits for a promise to resolve before executing the next line of code.

What are the different types of operators in JavaScript?

JavaScript has several types of operators, including arithmetic, assignment, comparison, logical, bitwise, and string operators.

What is the difference between == and === in JavaScript?

The == operator compares values for equality, whereas the === operator compares values for equality and type.

What is event bubbling in JavaScript?

Event bubbling is the process by which an event propagates from the innermost element that triggered the event up to the document root, triggering all event handlers along the way.

What is event delegation in JavaScript?

Event delegation is a technique in which a single event handler is added to a parent element, rather than multiple event handlers being added to multiple child elements.

What is a decorator in JavaScript?

A decorator in JavaScript is a function that is used to modify the behavior of another function or object. Decorators are typically used to add functionality to an existing function or object without modifying its original source code.

What is the difference between onclick and addEventListener in JavaScript?

The main difference between onclick and addEventListener in JavaScript is how they handle multiple event listeners for the same element.

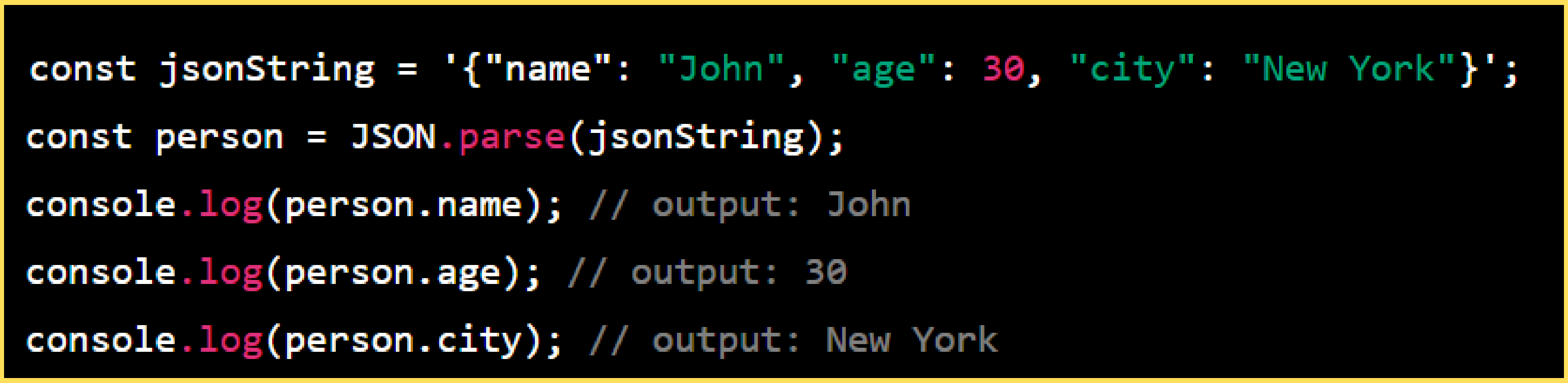
onclick is an inline event handler attribute that can be added to an HTML element to execute a JavaScript function when the element is clicked. It can only handle a single function at a time, which means that if multiple onclick attributes are added to the same element, only the last one will be executed. Also, it cannot be removed once it is set.

On the other hand, addEventListener is a method that allows you to attach multiple event listeners to an element. It can handle multiple functions at the same time and the order in which they are added is preserved. Also, it allows you to remove a specific listener from the element using the removeEventListener method.

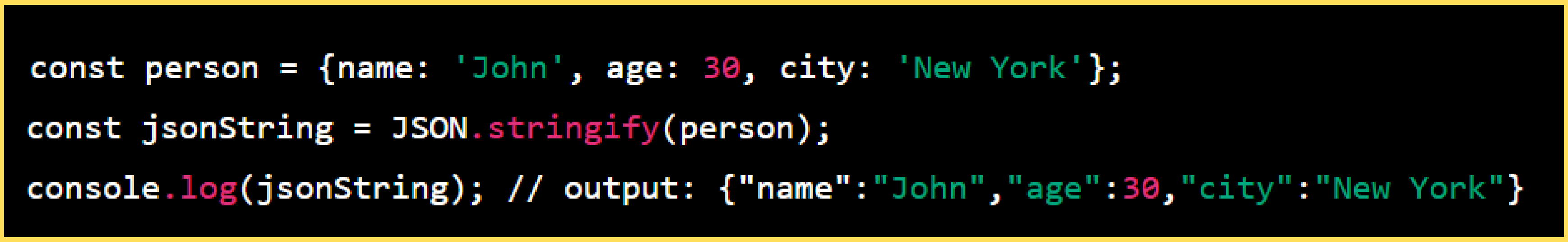
What is JSON and how is it used in JavaScript?

JSON (JavaScript Object Notation) is a lightweight data interchange format that is easy for humans to read and write and easy for machines to parse and generate. It is a text format that is used to represent data in a structured way and is widely used for data exchange between web servers and clients.

In JavaScript, JSON is used to exchange data between a web server and a web application. The JSON format is based on a subset of the JavaScript programming language and is natively supported by most modern web browsers.



```
const jsonString = '{"name": "John", "age": 30, "city": "New York"}';
const person = JSON.parse(jsonString);
console.log(person.name); // output: John
console.log(person.age); // output: 30
console.log(person.city); // output: New York
```



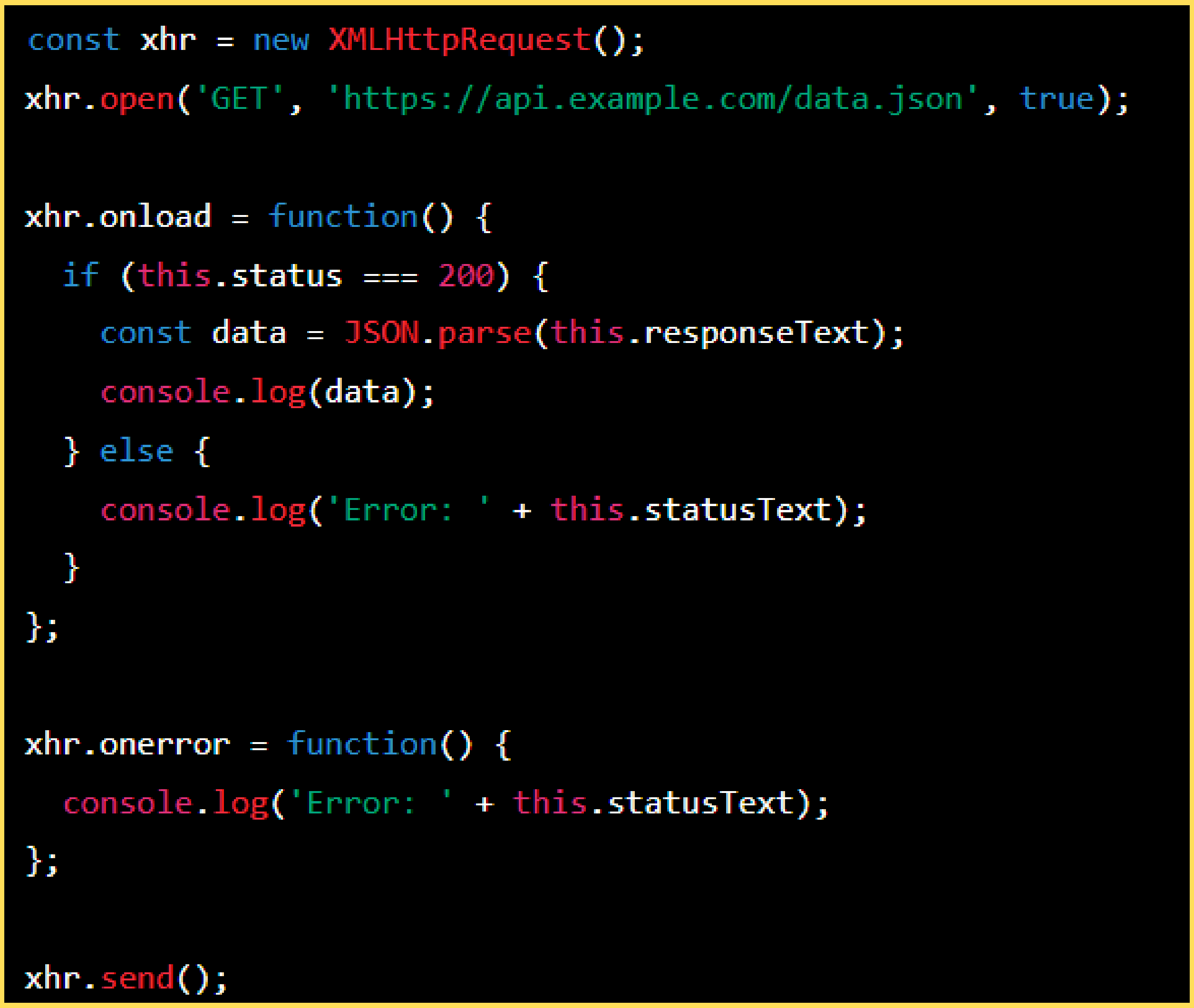
```
const person = {name: 'John', age: 30, city: 'New York'};
const jsonString = JSON.stringify(person);
console.log(jsonString); // output: {"name":"John","age":30,"city":"New York"}
```

What is AJAX in JavaScript?

AJAX (Asynchronous JavaScript and XML) is a technique used in web development to exchange data with a web server asynchronously, without the need to reload the entire page. It allows web applications to update parts of a web page without refreshing the entire page, providing a smoother and more responsive user experience.

In JavaScript, AJAX is typically implemented using the XMLHttpRequest (XHR) object, which allows you to send HTTP requests to a server and receive responses in different formats, such as HTML, XML, JSON, or plain text.

Here's an example of how to use the XHR object to send an HTTP GET request to a web server and receive a JSON response:



```
const xhr = new XMLHttpRequest();
xhr.open('GET', 'https://api.example.com/data.json', true);

xhr.onload = function() {
  if (this.status === 200) {
    const data = JSON.parse(this.responseText);
    console.log(data);
  } else {
    console.log('Error: ' + this.statusText);
  }
};

xhr.onerror = function() {
  console.log('Error: ' + this.statusText);
};

xhr.send();
```

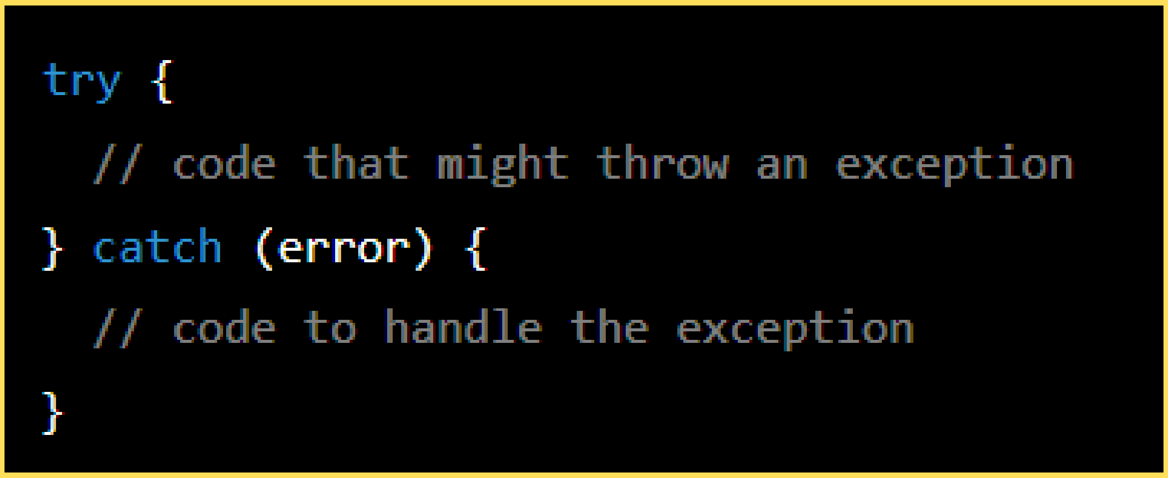
@Aakash Kanojiya

What are the different types of errors in JavaScript?

1. Syntax errors: These occur when the syntax of a program is incorrect and prevent the program from running. Syntax errors are usually caused by typographical errors, missing brackets, or invalid statements. For example, forgetting to close a bracket or using an incorrect operator can cause a syntax error.
2. Runtime errors: These occur during the execution of a program and can cause it to crash or behave unexpectedly. Runtime errors are usually caused by invalid data or unexpected conditions, such as dividing a number by zero or trying to access a non-existent object property.
3. Logical errors: These occur when a program runs without crashing but produces incorrect or unexpected results. Logical errors are usually caused by errors in the program's algorithm or logic, such as using the wrong formula for a calculation or not handling a specific condition.

What is try-catch in JavaScript?

try-catch is a JavaScript statement that allows you to handle exceptions that occur during the execution of a program. The try block contains the code that might throw an exception, while the catch block handles the exception by executing a set of statements.
Here's an example:



```
try {
  // code that might throw an exception
} catch (error) {
  // code to handle the exception
}
```

What is the difference between null and undefined in JavaScript?

null is a value that represents the intentional absence of any object value. It is often used to indicate that a variable has no value assigned to it. undefined, on the other hand, is a value that represents an uninitialized variable or a missing property in an object.

@Aakash Kanojiya

What is the event loop in JavaScript?

The event loop is a mechanism in JavaScript that allows for asynchronous programming. It continuously monitors the call stack and the message queue, and when the call stack is empty, it takes the first message from the queue and adds it to the call stack.

Thanks for reading !

Follow my profile for more coding related contents

Like Comment Share

@Aakash Kanojiya