1. What is JavaScript?
JavaScript is a high-level, interpreted programming language primarily used for adding interactivity to web pages.

2. What are the data types in JavaScript?
JavaScript has six primitive data types: string, number, boolean, null, undefined, and symbol, along with a complex data type called object.

3. What is the difference between null and undefined?
Null represents the intentional absence of any object value, while undefined indicates the absence of a value or an uninitialized variable.

4.What is the difference between == and === in JavaScript?
The == operator in Javascript is used for comparing two variables, but it ignores the datatype of variable, while the === operator is used for comparing two variables, but this operator also checks datatype and compares two values.

5. What is the purpose of the let keyword in JavaScript?
The let keyword is used to declare block-scoped variables in JavaScript. Variables declared with let are only accessible within the block where they are defined.

6. What is the purpose of the const keyword in JavaScript?
The const keyword is used to declare block-scoped variables in JavaScript that cannot be re-assigned. However, it does not make objects or arrays immutable.

7. What is the difference between let and var in JavaScript?
Var is function-scoped, let and const are block-scoped. Var has hoisting, let and const do not.

8.charAt() — Returns a character at a specified position inside a string.

9.charCodeAt() — Gives you the unicode of character at that position.

10.concat() — Concatenates (joins) two or more strings into one.

11.indexOf() — Provides the position of the first occurrence of a specified text within a string.

12.lastIndexOf() — Same as indexOf() but with the last occurrence, searching backwards.

13.**replace()** — Find and replace specified text in a string.

14.**search()** — Executes a search for a matching text and returns its position.

15.**slice()** — Extracts a section of a string and returns it as a new string.

16.**split()** — Splits a string object into an array of strings at a specified position.

17.**substring()** — Also similar to slice() but can't accept negative indices

Array Methods

18.join() — Combine elements of an array into a single string and return the string

19.pop() — Removes the last element of an array

20.push() — Add a new element at the end

21.shift() — Remove the first element of an array

22.slice() — Pulls a copy of a portion of an array into a new array

23.sort() — Sorts elements alphabetically

24.splice() — Adds elements in a specified way and position

25.toString() — Converts an array to strings

26.unshift() — Adds a new element to the beginning

27. flat() — Flattening is useful when you want to convert a multi-dimensional array into a one-dimensional array.

28.entries() — The entries() method returns an Array Iterator object with key/value pairs.

29.keys() — The Array.keys() method returns an Array Iterator object with the keys of an array.

30.What is the purpose of the forEach() function in JavaScript?
The forEach() function is used to execute a provided function once for each element in an array. It provides an easy way to iterate over array elements and perform operations on them.

31.What is the purpose of the map() function in JavaScript?
The map() function is used to create a new array by applying a given function to each element of an existing array. It allows transforming and manipulating array elements easily.

32. What is the purpose of the filter() function in JavaScript?
The filter() function is used to create a new array containing elements that pass a certain condition defined by a provided function. It allows filtering elements from an array based on specific criteria.

33. What is the purpose of the reduce() function in JavaScript?
The reduce() function is used to reduce an array to a single value by applying a function to each element and accumulating the result. It is often used to perform calculations or transformations on arrays.

34. What is a callback function in JavaScript?
A callback function is a function that is passed as an argument to another function and gets executed at a later time or in response to an event.

35. What is an anonymous function in JavaScript?
An anonymous function is a function without a name. It can be assigned to a variable or passed as an argument to another function. They are often used for onetime or callback functions.

36. What are closures in JavaScript?
A closure is the combination of a function bundled together(enclosed) with references to its surrounding state(the lexical environment). In other words, a closure gives you access to an outer function's scope from an inner function.

37. What is a high-order function in JavaScript?
In JavaScript, a higher-order function is a function that can accept other functions as arguments, return functions, or both.

38. What is a first-class function in JavaScript?
A first-class function in JavaScript is a function that can be treated like any other variable. This means that you can assign a function to a variable, pass it as an argument to another function, and return it from a function.

33. What are arrow functions?
Arrow functions are a concise syntax for defining functions in JavaScript.

34.Write the difference between map() method and foreach() method?
The map() method returns an entirely new array. While The for-Each() method does not returns a new array based on the given array.

The map() method returns the newly created array according to the provided callback function, while The for-Each() method returns "undefined".

35. What is object in javascript?
An object is a collection of properties, and a property is an association between a name (or key) and a value. A property's value can be a function, in which case the property is known as a method.

36.What are template literals in JavaScript?
Template literals, denoted by backticks (`), are a way to create strings in JavaScript that support interpolation of variables and multi-line strings.

37.What is the this keyword in JavaScript?
The this keyword refers to the object that is currently executing the code. Its value is determined by how a function is called, and it provides a way to access object properties and methods within a function.

38.What is the purpose of the call() method in JavaScript?
The call() method is used to invoke a function with a specified this value and arguments provided individually. It allows borrowing methods from other objects and explicit invocation of functions.

39. What is the purpose of the apply() method in JavaScript?
The apply() method is used to invoke a function with a specified this value and arguments provided as an array or an array-like object. It allows borrowing methods from other objects and explicit invocation of functions.

47. What is the purpose of the bind() method in JavaScript?
The bind() method is used to create a new function with a specified this value and initial arguments. It allows explicit binding of the this value within a function.

48.What is array of object in javascript?
An array of object is a collection of homogenous data, that stores a sequence of numbered objects at a single place.

49.What is strings in javascript?

A strings is a sequence of characters that represents text. Javascript strings are primitive and immutable.

50.What is an array in javascript?

An array in javascript is a type of global object that is used to store data.

51.What is JSON in javascript?

Javascript Object Notation(JSON) is a standard text-based format for representing structured data based on javascript object syntax.

52.Why JSON() is used?

JSON is perfect for storing temporary data. JSON can also be used as a serialization data format for any programming language to provide a high level interoperability.

53.What is JSON parse and JSON Stringify in JavaScript?

stringify() : This method takes a JavaScript object and then transforms it into a JSON string.
JSON. parse() : This method takes a JSON string and then transforms it into a JavaScript object.

54.What is rest parameter in javascript?

The rest parameter syntax allows us to represent an indefinite number of arguments as an array.

55.What is spread operator in javascript?

It allows us the privilege to obtain a list of parameters from an array.

56. What is the purpose of the startsWith() method in JavaScript?

The startsWith() method is used to check if a string starts with a specified substring. It returns true if the string starts with the substring, and false otherwise.

57. What is the purpose of the endsWith() method in JavaScript?

The endsWith() method is used to check if a string ends with a specified substring. It returns true if the string ends with the substring, and false otherwise.

58. What is the purpose of the includes() method in JavaScript?

The includes() method is used to check if a string contains a specified substring.+ It returns true if the substring is found, and false otherwise.

59. Can you explain the difference between an async function and a regular function in JavaScript?

Async functions are functions that allow you to use the await keyword to wait for a promise to resolve before continuing execution of the function. Regular functions do not have this ability, and will instead execute the code inside of them immediately.

60.What do you understand about await in JavaScript?

The await keyword in JavaScript is used to pause the execution of a function until a Promise is resolved. This allows you to write asynchronous code that looks and feels like synchronous code.

61.What is object destructuring in JavaScript?

Object destructuring is a feature that allows extracting properties from objects and assigning them to variables. It provides a concise way to extract values and work with object properties.

62.What is a promise?

A promise is an object that represents the result of an asynchronous operation. A promise can be in one of three states: pending, fulfilled, or rejected. A pending promise means that the asynchronous operation has not yet completed. A fulfilled promise means that the asynchronous operation has completed successfully. A rejected promise means that the asynchronous operation has failed.

63.Can you explain what the .then() method does?

The .then() method is used to specify what should happen when a promise is resolved. This is usually used to chain together multiple promises, so that each promise is executed in order. The .then() method can also take two arguments: a success callback and a failure callback. The success callback will be executed if the promise is resolved, and the failure callback will be executed if the promise is rejected.

64.Can you explain what the .catch() method does?

The .catch() method is used to handle errors that may occur in a Promise chain. The .catch() method takes a single argument, which is a function that will be executed if an error occurs. The .catch() method is typically used after a .then() method to handle any errors that may have occurred in the .then() method.

65.What does the .resolve() method do when working with promises?

The .resolve() method is used to resolve a promise and return the corresponding value. This is typically used when a promise has been fulfilled and the corresponding value is known.

66.What does the .reject() method do when working with promises?

The .reject() method is used when you want to indicate that a promise has failed. This is typically used when you want to catch an error that has occurred inside of a promise.

67. What are the two types of module exports in JavaScript?

There are two types of exports: Named Exports and Default Exports.

68.What is the purpose of export keyword in JavaScript modules?

The export keyword in JavaScript is used to export functions, objects, or primitive values from a module so they can be used by other modules.

69.What is the purpose of import keyword in JavaScript modules?

The `import` keyword in JavaScript is used to bring in functions, objects, or values from another module into the current module, facilitating code reusability and organization.

70.What is the purpose of the alert function in JavaScript?

The alert function shows a modal window with a text message and waits for the user to press "OK".

71.What is the purpose of the prompt function in JavaScript?

The prompt function shows a modal window with a text message asking the user to input text.

72.What is the purpose of the confirm function in JavaScript?

The confirm function shows a modal window with a message and waits for the user to press OK or Cancel.

73. What is the difference between a Do-While and a While Loop?

The main difference is that a do-while loop executes its body at least once before checking the condition. A While loop, on the other hand, only executes its loop body if the loop condition is initially true.

74.What is the difference between the break statement and the continue directive?

In JavaScript, The break statement terminates the entire loop, while the continue directive skips the current iteration and proceeds to the next iteration in the loop.

75.What is the difference between the While and For loops in JavaScript?

In JavaScript, while-loops check a condition before execution, while for-loops have an initialization, condition, and update expression within the loop statement.

76.What are JavaScript classes?

JavaScript classes are a way to define objects with shared properties and behaviors. They provide a template for creating multiple instances of objects with similar characteristics.

77.What is inheritance in JavaScript?

Inheritance is a mechanism in JavaScript where an object can inherit properties and methods from another object. It allows for code reuse and creating hierarchical relationships between objects.

78. What are JavaScript modules?

JavaScript modules are reusable pieces of code that encapsulate related functionality. They allow for better organization, encapsulation, and code reuse in larger JavaScript applications.

79. What is the DOM in JavaScript?

The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document.

80.What is constructor in javascript?

A constructor is a special function that creates and initializes an object instance of a class. In JavaScript, a constructor gets called when an object is created using the new keyword.

81.What is the purpose of the typeof operator in JavaScript?

The **typeof** operator in JavaScript is used to determine the data type of a given operand. It returns a string representing the type of the operand. For example, **typeof 5** returns **'number'**, **typeof 'hello'** returns **'string'**, and so on.

82.Discuss the differences between JavaScript's apply() and call() and bind() methods:

apply() accepts an array of arguments, while
call() accepts an argument list.
bind() can preset arguments when creating the new function, but additional arguments can be passed when the new function is invoked.

83.Differences between for...in and for...of loops:

for...in loop iterates over the enumerable properties of an object, including inherited ones. It's commonly used for iterating over object properties.

for...of loop iterates over iterable objects like arrays, strings, maps, sets, etc., extracting their values. It doesn't iterate over object properties, just the values.

84.Differences between synchronous and asynchronous code execution:

Synchronous code executes line by line, blocking further execution until the current operation is completed.

Asynchronous code allows other operations to continue while waiting for an asynchronous operation to complete. Callbacks, promises, and async/await are common patterns for handling asynchronous code.

85.Benefits of using the spread operator:

The spread operator (...) can be used to clone arrays or objects, concatenate arrays, or spread elements of an iterable (like arrays) into another array or object.

86.Differences between filter(), map(), and reduce() array methods:

filter() creates a new array with elements that pass a certain condition.
map() creates a new array by applying a function to each element of the original array.
reduce() applies a function to each element of the array (from left to right) to reduce it to a single value.

87.Handling of the this keyword in different contexts:

In JavaScript, the value of this depends on how a function is called:
In the global context or in a function called without an explicit context, this refers to the global object (in non-strict mode) or undefined (in strict mode).
In a method, this refers to the object that the method is called on.
In an event handler, this refers to the element that fired the event.

88.What is the purpose of the use strict directive in JavaScript?
The **'use strict'** directive is used to enable strict mode in JavaScript, which imposes stricter parsing and error handling rules on the code. Strict mode helps catch common coding mistakes and promotes writing more secure and optimized code by disallowing certain unsafe or deprecated features.

89.Differences between indexOf() and search():
The **indexOf()** method returns the index of the first occurrence of the specified substring. The **search()** method returns the index of the first character of the match.

The **indexOf()** method does not support regular expressions. The **search()** method supports regular expressions.

90.What is super() method in javascript?
The super keyword is used to call the constructor of its parent class to access the parent's properties and methods.

91.What is hoisting in JavaScript?
Hoisting is a JavaScript behavior where variable and function declarations are moved to the top of their containing scope during the compilation phase, allowing them to be used before they are declared.