1. List all ES6 features of js.

Ans:

* Let and const keyword.
* Arrow functions
* Spread operator.
* Map and set.
* Promises.
* Classes.
* Rest parameter.
* JavaScript modules.

1. What is destructuring?

Ans: Destructuring is a JavaScript expression that makes it possible to unpack values from arrays, or properties from objects, into distinct variables. That is, we can extract data from arrays and objects and assign them to variables.

1. What is promises?

Ans: A promise is an object that encapsulates the result of an asynchronous operation.

A promise object has a state that can be one of the following:

* Pending
* Fulfilled with a value.
* Rejected for a reason.

In the beginning, the state of a promise is pending, indicating that the asynchronous operation is in progress. Depending on the result of the asynchronous operation, the state changes to either fulfilled or rejected.

Promise constructor accepts 2 callback functions resolve() and reject().

If the asynchronous operation completes successfully, the executor will call the resolve() function to change the state of the promise from pending to fulfilled with a value.

In case of an error, the executor will call the reject() function to change the state of the promise from pending to rejected with the error reason.

1. Explain map, reduce and filter method.

Ans: The map() method takes an array and a callback function as arguments. The callback function is applied to each element of the array, and the results are returned in a new array.

The filter() method takes an array and a callback function as arguments. The callback function is applied to each element of the array, and the elements that return true are returned in a new array.

The reduce() method takes an array and a callback function as arguments. The callback function is applied to each element of the array, and the results are accumulated into a single value.

1. What is higher order functions and callback functions? Difference between them.

Ans: Higher Order Function: A function that accepts a function as an argument and/or returns a function as its value.

Callback Function: A function that's passed as a argument to another function.

1. Difference between map and set.

Ans: A map is a collection of key-value pairs. Each key is unique, and each value can be any data type. Maps are often used to store data that needs to be accessed quickly, such as a lookup table.

A set is a collection of unique elements. Each element can be any data type, and the order of the elements is not important. Sets are often used to store data that needs to be unique, such as a list of unique identifiers.

1. What is hoisting in JavaScript?

Ans: In JavaScript, hoisting refers to the built-in behavior of the language through which declarations of functions, variables, and classes are moved to the top of their scope – all before code execution. In turn, this allows us to use functions, variables, and classes before they are declared.

1. What are the differences between == and ===?

Ans: In JavaScript, the === operator is a strict equality comparison operator that matches by both value and data type. The == operator is an equality comparison operator that matches by value only.

1. What are the different ways to create an object in JavaScript?

Ans:

* Object literal
* Constructor function
* Classes
* Object.create()

1. What is the event loop in JavaScript?

Ans: The event loop is a mechanism in JavaScript that allows the execution of code to be scheduled in a non-blocking way. It is a single-threaded loop that continuously checks for events and then executes the appropriate callback functions. The event loop allows JavaScript to be responsive to user input, network events, and other asynchronous operations.

1. What is asynchronous programming?

Ans: Asynchronous programming is a technique that enables your program to start a potentially long-running task and still be able to be responsive to other events while that task runs, rather than having to wait until that task has finished. Once that task has finished, your program is presented with the result. This is achieved by using callbacks, promises, or async/await functions.

1. What is the prototype chain in JavaScript?

Ans: Every object in JavaScript has a built-in property, which is called its prototype. The prototype is itself an object, so the prototype will have its own prototype, making what's called a prototype chain. The chain ends when we reach a prototype that has null for its own prototype.

1. What is the purpose of the this keyword in JavaScript?

Ans: The this keyword in JavaScript is a powerful tool that allows you to reference the current object. This can be useful in a variety of situations, such as when you need to access the properties or methods of the current object, or when you need to pass the current object to a function.

The value of the this keyword depends on the context in which it is used. In an object method, this refers to the object that owns the method. In a function, this refers to the global object. In strict mode, this is undefined.

1. What is the use of the isNaN function?

Ans: The isNaN() function is used to check if a value is NaN (Not a Number). isNaN() returns true if a number is Not-a-Number.

1. What is call, apply and bind method?

Ans: The call method sets the this inside the function and immediately executes that function.

The apply() method is similar to call(). The difference is that the apply() method accepts an array of arguments instead of comma separated values.

The bind method creates a new function and sets the this keyword to the specified object.

1. What is DOM?
2. What is callback hell?
3. Difference between promise, callback and async and await.
4. What is IIFE?
5. What is strict mode?
6. What is web storage?
7. Difference between local storage and session storage?
8. What is cloning? Different ways of deep cloning and shallow cloning?
9. Different ways to perform deep cloning?
10. How can we perform deep clone for nested object?
11. How to merge object?
12. What is self invoking functions?
13. Difference between normal function and function expression
14. What is closure?
15. What is function constructor?
16. What is pass by value or pass by reference?
17. What is instanceof operator?
18. What is spread and rest operator?
19. Difference between var, let and const.