

## Javascript

### Interview question and answers

① Difference between `=` and `==`.

→ The difference between the `=` and `==` operator.

is the `==` operator does the type conversion of the operands before comparison, whereas the `==` operator compares the values as well as the data types of the operands.

② What are the primitive and non-primitive data types

→ Primitive data types are

Number, String, Boolean, Null, Undefined, Symbol

Non-primitive data types are

Array, Object, Function, Date, RegEx

③ What is the use of 'this' keyword

→ This keyword refers to an object that is executing the current piece of code

- It references the object that is executing the current function.

- If the function being referenced is a regular function that time "this" references the global object

④ Difference between Regular javascript functions and arrow function

→ Regular functions created using function declarations or expressions are 'constructible' and 'callable'. Since regular functions are constructible, they can be called using the 'new' keyword.

- The arrow functions are only 'callable' and not constructible

Eg:- Example of arrow functions  
Var square = (x) => {  
    return (x\*x);  
};

console.log(square(3));

Output:- 9

Example of regular functions

let square = function(x)

{  
    return (x\*x);  
};

console.log(square(3));

Output:- 9

⑤ Difference between map and forEach  
→ The map method receives a function as a parameter.  
Then it applies it on each element and returns an  
entirely new array populated with the results of  
calling the provided function

This means that it returns a  
new array that contains an image of each element  
of the array. It will always return the same number  
of items

Eg:- const A = [5, 4, 3, 2, 1]

A.map(n => n \* n)

Output:- [25, 16, 9, 4, 1]

→ The forEach() method receives a function as an argument  
and executes it once for each array element. However,  
instead of returning a new array like map, it returns  
undefined

```
const Array1 = [  
  { id:1, name: "Raju" },  
  { id:2, name: "Ramesh" },  
  { id:3, name: "Kumar" }  
]
```

```
Array1.forEach(element => console.log(element.name))
```

Output:  
Raju  
Ramesh  
Kumar

### ⑥ What is Closure

- A closure is the combination of a function bundled together (enclosed) with references to its surrounding state (the lexical environment)
- or
- A closure gives you access to an outer function scope from an inner function.

What is hoisting

Hoisting in javascript is a behavior in which a function or a variable can be used before declaration

Ex:-

```
function x() {  
  console.log('x is ' + x);  
}  
x();
```

Output:-

dp:- 2

Q) What is Promise

→ Promises are used to handle asynchronous operations in javascript

They are easy to manage when dealing with

multiple asynchronous operations where callbacks can create callback hell leading to unmanageable code

Eg:-

```
var promise = new Promise(function(resolve, reject){  
  const n = "gadag";  
  const y = "gadag";  
  if (n === y) {  
    resolve();  
  } else {  
    reject();  
  };});
```

Promise.then(function(){

console.log("Success, given n is equal to y");

}).

catch(function(){

console.log("Some unsuccess, given n is not equal to y");

});

Output:

Success, given  $n$  is equal to  $\lambda$

⑨ Difference between null and undefined

→ Null! It is the intentional absence of the value

It is one of the Primitive Values of JavaScript

Undefined! It means the value does not exist in the Compiler

⑩ What are the ES6 Features

- let keyword
- const keyword
- Arrow functions
- The ... Operator
- For/of
- Map object
- Set object
- Classes
- Promises
- Symbols
- Default Parameter
- String.includes()
- String.startsWith()
- String.endsWith()
- Array.from()
- Array.keys()
- Array.find()
- New Global methods

Q1) What is Prototype Chain  
→ Every function and object has a property named  
prototype by default

Eg:- function Person() {  
    this.name = 'Akshay';  
    this.age = 26;  
}

const Person = new Person();

console.log(Person.prototype);

Output:- Person{}

- In the above Example we are trying to access the Prototype Property of a Person Construction function

Q2) What are truthy and Falsy values in JavaScript  
- A value that is considered true when encountered

in a boolean context are called truthy values

Eg:- a = 10, b = 10

a == b

True

- A value that evaluate to false in a boolean context are called Falsy values than are  
0, -0, undefined, null, NaN, ""

Output:-

console.log(adding())

Output:- undefined

⑬ Write the differences between let, Var and const

↑

Var	let	const
globally scoped or function scoped. attached to window object	globally scoped or block scoped	globally scoped or block scoped
is attached to window object. Not attached to window object	attached to window object	attached to window object
Hoisting Var is hoisted to top of its execution (either global or function) and initialized as undefined	let is hoisted to top of its execution (either global or block) and left uninitialized	const is hoisted to top of its execution (either global or block) and left uninitialized
Re-declaration within scope.	no	no
named within scope	yes	no

Q1. What is Destructuring in javascript?

The Destructuring assignment syntax is a Javascript expression that makes it possible to unpack values from arrays, or properties from objects, into distinct variables.

Example:-

```
let fullstack = [ ['HTML', 'CSS', 'JavaScript'],
                  ['Django', 'Express.js', '.Net'] ]
```

```
let [ frontend, Framework ] = fullstack
console.log(Framework)
```

Output:-

```
['Django', 'Express.js', '.Net']
```

Q2. What is Callback?

A callback function is a function passed into another function as an argument, which is then invoked inside the other function to complete some kind of routine or action.

```
Eg:- function greeting(name){
      alert('Hello, ${name}');
    }
```

```
function processUserInput(callback){
  const name = prompt("Please enter your name.");
  callback(name);
}
processUserInput(greeting);
```

The above example is a synchronous

Q16 What is an event in javascript  
→ Events are actions or occurrences that happen in the system you are programming, which the system tells you about so your code can react to them.  
Types of events are:

- ① Keyboard events
- ② Mouse events
- ③ Form events
- ④ CSS events
- ⑤ Form Events
- ⑥ Focus Events
- ⑦ Document events

Q17 What is Block scope/Function scope  
→ The Block scoped/Function scoped Variable means that the variable defined within a function/Block will not be accessible from outside the function.

Q18 What is this keyword  
→ The this keyword refers to the current object in a method or constructor  
- The most common use of this keyword is to eliminate the confusion between class attributes and Parameters with the same name.

Q19 What is Pure function  
→ A Pure function is a function that always returns the same result if the same arguments are passed.  
- It does not depend on any state or data change during a program execution rather it only depends on its argument

Eg:-

```
function calculateGST(ProductPrice)  
{  
    return ProductPrice * 0.05;  
}
```

Q) What is the difference between Classical Inheritance and Prototypal Inheritance

Classical Inheritance	Prototypal Inheritance
- classes are immutable. You can't modify or add new methods to them at runtime.	- Prototypes are flexible. They may be either mutable or immutable.
- classes may or may not support multiple inheritance.	- Objects can inherit from multiple prototypes.
- It's verbose and complicated. You have abstract classes, final classes, interfaces, etc.	- It's simple. You only have objects and extending objects. is the only operation required.

Q) Is Javascript is statically typed or Dynamically typed language

Javascript is Dynamically typed language, because we can declare the variable, functions, objects and anything without declaring the type.

Q) What is the difference between Local storage and Session storage.

→

## Local Storage

- The storage capacity of local storage is 5MB

- As it is not session based, it must be deleted via JS or manually

- The client can only read local storage

- There is no transfer of data to the server

- ② Is JavaScript is single threaded or multi-threaded  
→ JavaScript is a single threaded language because while running code on a single thread, it can be really easy to implement as we don't have to deal with the complicated scenarios that arise in the multi-threaded environment like deadlock

- ③ What does super keyword do when it is used?  
→ The super keyword is used to call the constructor of its parent class to access the parent's properties and methods

## Session Storage

- The storage capacity of session storage is 5MB

- It is session-based and works per window or tab, based on setting. This means that data is stored only for the duration of a session, i.e. until the browser is closed.

- The client can only read local storage

- There is no transfer of data to the server

- Both clients and servers can read and write the cookies

- Data transfer to the server is in JSON

## Cookies

- The storage capacity of cookies is 4KB

- Cookie engine

- Per tab and cookie

- Per tab and cookie

### ⑥ What is Functional Programming

Functional Programming is a paradigm of building computer programs using expressions and functions without mutating state and data. By respecting these restrictions, Functional Programming aims to write code that is clearer to understand and more bug-resistant.

### ⑦ Is Javascript Asynchronous Programming language?

Yes, Javascript is an asynchronous programming language, because in Javascript multiple related operations can run concurrently without waiting for other tasks to complete.

### ⑧ What is Pass by value and Pass by reference in Javascript

→ Pass by value: In Pass by value function is called

by directly passing the value of the variable as an argument to any change made inside the function does not affect the original value.

Pass by reference: In Pass by reference function is called

by directly passing the reference/address of the variable as an argument to changing the value inside the function also change the original value.