1. What is JavaScript?

Ans: JavaScript is a lightweight, cross-platform, single-threaded , high-level, object oriented, multiparadigm programming language. It is also known as scripting language. JavaScript is used to make websites interactive and to provide dynamic content.

1. Types of datatypes in JavaScript:

Ans: There are 3 major datatypes in JavaScript i.e. primitive, special, non-primitive or composite.

Primitive datatypes are the datatypes which hold single values and are immutable. It includes number, string, boolean and symbol.

Special datatypes are undefined and null. Undefined is used when we declare a variable and not initialize it whereas null is used to explicitly represents absence of a value.

Non-primitive or composite datatypes are the datatypes which holds multiple values and are mutable. It includes objects, arrays, functions.

1. What is variable?

Ans: A variable in JavaScript is a named storage location which can hold a value. The value can be of any type like number,string, object, array or even functions. We can create variables in JavaScript using var, let or const keyword.

1. What is typeOf operator in JavaScript?

Ans: typeOf operator in JavaScript is a unary operator which accepts only one operand. Operand can be of any data type. The typeOf operator returns a string indicating the data type of the passed operand.

1. What is a function?

Ans: A function is a reusable piece of code designed to perform a particular task.

There are 2 types of functions: pre-defined and user-defined.

Pre-defined functions are the functions which are already defined in the JavaScript and we don’t have to define them. For example: console.log(), prompt(), alert(),etc.

User-defined functions are the functions which are defined by us as a programmer and we can use them in our code.

There are 3 types user can create functions – named , anonymous and arrow function.

1. What is named, anonymous and arrow function?

Ans: Named functions are declared using the function keyword followed by the name of the function. We can invoke the named function using the function name.

Whereas anonymous functions do not have a name so invoke the anonymous function we store it in a variable.

Arrow functions is the concise way to write a function. They were introduced in ES6 version and they are also known as lambda functions.

1. What is a string?

Ans: String is a sequence of characters. We write strings in double or single quotes.

1. What is an index?

Ans: Index is a position of an element and it always starts from 0.

1. What is the property of strings?

Ans: The property we can apply to strings is length property. By using length property, we can find the length of the string.

1. List the methods we have for strings?

Ans.

1. charAt(index) – returns character value present at the specified index.
2. toUpperCase() – capitalize the string.
3. toLowerCase() – small case the string.
4. trim() – it trims the whitespace from the left and the right side of the string.
5. replace(searchValue, replaceValue) – accepts 2 arguments. First is the word which we want to replace and second is the word with which we want to replace. If you want to replace all the characters you can use replaceAll().
6. Split(“separator”) – it splits a string into substring array, then returns that newly created string.
7. indexOf(“searchString”) – returns the index of the searchString.
8. lastIndexOf(“searchString”) - returns the last index of the searchString.
9. startsWith(“searchString”) – returns Boolean values.
10. endsWith(“searchString”) - returns Boolean values.
11. Slice(start,end) – returns the section of the string.
12. substring(start,end) – returns the section of the string.
13. subStr(start,end) - returns the section of the string.
14. toString() – returns string representation.
15. parseInt(“strNumber”) – convert string to number.
16. new Number(“strNumber”) – convert string to number.
17. What is the difference between var, let, const?

Ans:

|  |  |  |
| --- | --- | --- |
| var | let | const |
| Global or function scope | Block scope always | Block scope always |
| Can be redeclare and reinitialized | Can be reinitialized but cannot redeclare | Can be reinitialized but cannot redeclare |
| Not necessary to initialize while declaring. | Not necessary to initialize while declaring. | Necessary to initialize while declaring. |

1. What is break and continue jump statement?

Ans: break and continue are the two most important statements of javascript.

Break statement is used to exit a loop early. It is used to terminate the loop prematurely and skip over the rest of the iterations of loop.

Continue statement is used to skip the current iteration of the loop and continue with the next iteration of the loop.

1. What is an infinite loop?

Ans: Infinite loop is a piece of code that lacks a functional exit so it repeats itself indefinitely.

1. When to use switch case statement?

Ans: you can use switch statement when you have a variable to check against various possible values and want to execute the different code blocks for different conditions.

1. What is global, function and block scope?

Ans: Scope refers to accessibility of the variables and functions.

There are 3 scopes: global, function and block.

Global scope: variables declared outside of all the functions are known as global variables and in the global scope. They can be accessed from anywhere in the program.

Local scope or function scope: variables declared inside a function are known as local variables and have local scope. They can only be accessed from within the function in which they are declared.

Block scope: variables declared inside a block of code, such as curly braces are known as block scoped variables and have block scope. They can only be accessed from within the block in which they are declared.

Var is the function scoped whereas, let and const is block scoped.

1. What is an array?

Ans: Array is a non-primitive data type which can store multiple values. Array allows duplicate elements. Each value in an array has a numeric position, known as index and its start from 0.

1. What is a map?

Ans: Map is used to store collection of elements in key-value- pairs.

1. What is set?

Ans: set is a collection of unique elements or values.

1. What is object?

Ans: an object is a non-primitive data type in javascript which is used to store data in key and value pair.

1. What is function declaration and function expression?

Ans: Function declaration is a that can be used before initialization whereas function expression is a function value that is essentially stored in a variable.

1. What is implicit and explicit type conversion in javascript?

Ans: implicit type conversion is when javascript automatically converts a value to a different data type based on the context. It is also known as type coercion.

For instance, when we are trying to add 42 with a Boolean value true javascript will return output as 43. It will automatically converts Boolean true as 1 and then add with 42.

Explicit type conversion is when we explicitly try to convert a value to a different data type using some functions such as toString(), number(), parseInt(), parseFloat(), Boolean().

1. Is javascript is dynamically typed or statically typed?

Ans: javascript is dynamically typed language which means data types of the variables is determined by the values they hold at runtime and they can change throughout the program as we assign different values to them.

1. What is isNaN function?

Ans: NaN stands for Not-a-Number. isNaN is a function we use to check whether the passed operand is a number or not. It will returns true if the value is not a number and if the value is number it will returns false.

1. What is passed by value and passed by reference?

Ans: in javascript primitive data types are passed by values and non-primitive data types are passed by reference.

Passed by values means that it will create a copy of the original value. So,if we try to change the clone value it will not affect the original value.

In passed by reference when we try to clone a value it does not create any copy of the original value, it will just point or refer to the original value. So, if we try to make any changes in the cloned value it will change the original value.

1. What is DOM?

Ans: DOM stands for document object model and it is an API to manipulate the html document. The DOM provides function to add, remove and modify html document effectively.

1. What is object?

Ans: An object is a non-primitive data type in javascript which is used to store data in the form of key and value pair.

There are 3 different ways to create an object :

1. Object literal
2. Class
3. Function constructor.
4. What is oop in js?

Ans: oop stands for object oriented programming it is an programming paradigm which is based in the concepts of objects and classes. Oop is based on several concepts such as abstraction, inheriteance, polymorphism, encapsulation. These are also known as 4 pillars of oop.

1. What is class?

Ans: javascript class is a blueprint for creating objects. They encapsulates data and functions to manipulate that data. Classes are just special function in js just as you can define expressions and function declaration classes can be define in two ways class expressions and class declarations.

So we can say that classes are just syntactic sugar on our function constructor.

1. What is class constructor?

Ans: Constructor method is a special method used for creating and initializing an object instance of that class.

1. What is instance of an object?

Ans: An instance of a class is an object which is created from that class.

We use new keyword to create an object instance of a class.

1. What is function constructor?

Ans: A function constructor is a special type of function which is used to create objects. It is called when we create objects using new keyword.

1. How to add methods in function constructor and in class?

Ans: In classes we can create methods after the closing tag of constructor method. It means we can have all our data members and methods in one block of code.  
but in function constructor we have to define method using prototype keyword.

Syntax : functionName.Prototype.methodName = function(){function body}

1. What is an instanceOf operator?

Ans: instanceOf operator is used to check if an object is the instance of the particular class. It will return true if the object is the instance of that class else it will return false.

1. What is shallow copying and deep copying in js?

Ans: In js when can copy objects in 2 different ways shallow copy and deep copy.  
shallow copy creates a new object with reference to the same memory location as the original object. This means that when we made any changes to the original object it will be reflected to the new object.

Deep copy creates new object with new memory location for all of its methods and nested objects or arrays. This means that changes mades in the original object will not be reflected in the new object.

1. What is global objects in js?

Ans: In javascript global objects are the objects that are always exists in global scope. We can access them from anywhere. For example : window, console, document, JSON.

1. What is hoisting?

Ans: Hositing in js is a behaviour of variable or functions that they can be used before declarations. There are 2 types of hoisting in js variable hoisting and function hoisting.  
variable hoisting is when we can use a variable before its declaration. All the variables created with the var keyword are hoisted in js. Let and const variables are not hoisted in js.

Function hoisting is when we can call or invoke function before declaration. Only regular functions are hoisted in js not function expressions.

1. What is web storage?

Ans: Web storage allows us to store the data in the browser. There are 2 types of web storage session storage and local storage.

Session storage stores data for one session it means when you close the tab the data will be lost.  
it will not clear the data after refreshing it will just delete the data after the tab is closed.

Local storage stores data that will not expire. It means data is stored until you will manually delete it.

1. Explain the 4 pillars of oop.

Ans: Abstraction – abstraction means to hide the internal data which is not require on the user side. Suppose we are using array.push() method we know that it is used to add the element in the end of the array but we don’t know its implementation and it is not required also.

Encapsulation – encapsulation means binding data members and member function into a single unit. Encapsulation is also helpful because it allows us to control how data is accessed or modified.

Inheritance – inheritance means child class or derivaed classes that extends their parents class can inherit parents properties or methods and can have their own.

Polymorphism – polymorphism means many shapes. Suppose you have more user class and it has 2 subclass admin and student. Inside the user class we have called login. Both the subclasses will inherit it from the parents class but we can overwrite the login process for the admin so this is known as polymorphism. For the purpose of login we have 2 different forms of method.

1. What is method overloading and method overriding?

Ans: method overloading is the ability to have multiple methods with the same name but different parameters. Suppose you want to have a method called ad and you are using it for different data types.

Method overriding is the ability to have a method in subclass that have same name and signature as in super class. This allows you to provide a different implementation for the subclass.

1. What is super class?
2. What is JSON?

Ans: JSON stands for javascript object notation , is a way to store information in an organized way and easy to access manner.

1. Explain JSON.parse() and JSON.stringify().

Ans: JSON.parse() – converts json string into javascript object.

JSON.stringify() – converts javascript object into json strings.

1. What is closure?

Ans: A closure is a function which has access to its own variable , its outer functions variable and the global variables.

1. What are callbacks?

Ans: in javascript we can also pass function as an argument to another function. The function that is passed as an argument is known as callback function.

1. Explain setTimeOut().

Ans: setTimeOut() is a bulit-in function which calls the function or evaluates an expression after a given period of time in millisecond.

List of array methods :

|  |  |
| --- | --- |
| name | description |
| push() (end) | Adds one or more elements to the end of an array. |
| unshift() (start) | Adds one or more elements to the beginning of an array. |
| pop() (end) | Removes the last element from an array. |
| shift() (start) | Removes the first element from an array. |
| splice(start,deleteCount, item) | Changes the contents of an array by adding,removing or replacing elements. |
| reverse() | reverse the array |
| sort() |  |
| fill() |  |
| map() |  |
| filter() |  |
| slice(index,itemCount) | Returns a shallow copy of a portion of an array. |
| concat() | Joins two or more arrays and returns a new array. |
| flat() |  |
| flatMap() |  |
| indexOf() | Returns the first index at which a specified element is found in an array. |
| findIndex() |  |
| find() |  |
| includes() | check whether the given character is included in the array or not |
| some() |  |
| every() |  |
| join() | joined the 2 array |
| reduce() |  |
| forEach() (loop) | Calls a function for each element in the array. |

List of string methods :

|  |  |
| --- | --- |
| name | description |
| charAt() | Returns the character at the specified index in a string |
| charCodeAt() |  |
| concat() | Combines two or more strings. |
| fromCharCode() |  |
| indexOf() | Returns the index within the calling string of the first occurrence of the specified value. |
| lastIndexOf() | Returns the index within the calling string of the last occurrence of the specified value. |
| match() |  |
| replace(from, to) | Searches a string for a specified value or a regular expression and returns a new string where the specified values are replaced. |
| search() |  |
| slice(start,end) | Extracts a section of a string and returns a new string. |
| split() | Splits a string into an array of substrings. |
| subString(from,to) | Returns the part of the string between the start and end indexes. |
| toLowerCase() | converts the string in lowercase |
| toUpperCase() | converts the string in uppercase |
| includes() | check whether a string contains the specified string or characters |
| endsWith() | check whether a string ends with the given string or characters. |
| repeat() |  |
| valueOf() |  |
| trim() | removes the whitespace from the start and end of the string. |
| startsWith() | check whether a string starts with the given string or characters. |
| toString() |  |

List of array and strings shared methods

|  |  |
| --- | --- |
| name | method |
| concat() | Combines two or more arrays/strings and returns a new array/string. |
| indexOf() | Returns the index of the first occurrence of a specified value. |
| lastIndexOf() | Returns the index of the last occurrence of a specified value. |
| slice() | Extracts a portion of an array/string and returns a new array/string. |
| length | Property that returns the number of elements in an array/string. |
| includes() | check whether a string/array contains the specified string or characters or array element |