

Interview Q & A Series -14

Interview Q & A In

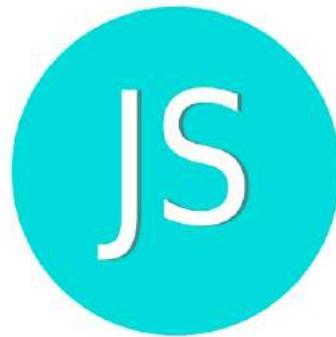


**90% Frequently asked
Interview Q & A**

By Bandana Ojha

Interview Q & A Series-14

Interview Questions & Answers in



100+ Frequently asked Interview Q & A

90% Frequently asked Interview Q & A in
JavaScript

By Bandana Ojha

All Rights Reserved © 2018

Introduction

The authors of this book “Interview Questions & Answers in JavaScript”

conducted so many interviews at various companies and meticulously collected the most effective questions with simple, straightforward explanations. Rather than going through comprehensive, textbook-sized reference guides, this book includes only the information required to start his/her career as a web developer. Answers of all the questions are short and to the point.

We assure that you will get here the 90% frequently asked interview questions and answers.

Good luck to ALL !!!



1. What is JavaScript(JS)?

JavaScript is a lightweight, object-oriented, cross-platform, scripting language. It is widely used for client-side validation. The JavaScript Translator (embedded in the browser) is responsible for translating the JavaScript code for the Web browser.

2. What are the differences between Java and JavaScript.

Java is a complete programming language while JavaScript is used to develop client - server applications. Java is an object - oriented programming (OOPS) or structured programming language like C++ or C whereas JavaScript is a client-side scripting language and focuses how the page should react after triggering the JavaScript code.

3. What is the difference between JavaScript and JScript?

JavaScript is a scripting language developed by Netscape Communications designed for developing client and server Internet applications. Netscape Navigator is designed to interpret JavaScript embedded into Web pages.

Microsoft JScript is an open implementation of Netscape's JavaScript. JScript is a high-performance scripting language designed to create active online content for the World Wide Web. JScript allows developers to link and automate a wide variety of objects in Web pages, including ActiveX controls and Java programs. Microsoft Internet Explorer is designed to interpret JScript embedded into Web pages.

4. What is Scope in JavaScript?

The scope determines the accessibility of variables, objects, and functions in particular part of your code. In JavaScript, each function gets its own scope. Scope is basically a collection of variables as well as the rules for how those variables are accessed by name. A variable name has to be unique within the same scope. A scope can be nested inside another scope. If one scope is nested inside another, code inside the innermost scope can access variables from either scope.

5. List some features of JavaScript.

Following are the features of JavaScript –

- Lightweight, interpreted programming language.

- Designed for creating network-centric applications.
- Complementary to and integrated with Java.
- Complementary to and integrated with HTML.
- Open and cross-platform.

6. List some of the advantages of JavaScript.

Some of the advantages of JavaScript are:

- Server interaction is less
- Feedback to the visitors is immediate
- Interactivity is high
- Interfaces are richer

7. List some of the disadvantages of JavaScript.

Some of the disadvantages of JavaScript are:

- No support for multithreading
- No support for multiprocessing
- Reading and writing of files is not allowed
- No support for networking applications.

8. What is DOM?

DOM stands for Document Object Model. DOM is an object-oriented representation of the HTML elements. All the elements (or nodes) are part of window.document.

9. What is BOM?

BOM stands for Browser Object Model. It provides interaction with the browser. The default object of a browser is a window. So, you can call all the functions of the window by specifying the window or directly. The window object provides various properties like document, history, screen, navigator, location, innerHeight, innerWidth, etc.

10. What is the use of history object?

The history object of a browser can be used to switch to history pages such as back and forward from the current page or another page. There are three methods of history object.

history.back() - It loads the previous page.

history.forward() - It loads the next page.

history.go(number) - The number may be positive for forward, negative for backward. It loads the given page number.

11. What is the difference between Client-side JavaScript and Server-side JavaScript?

Client-side means that the JavaScript code is run on the client machine, which is the browser. Server-side JavaScript means that the code is run on the server which is serving web pages. One runs in the browser (client side), the other runs on the server

Client-side components usually consist of HTML, CSS, JavaScript, Ajax, JS libraries, images, and whatever other files that are to be downloaded to the browser. On the server, you need a listener to process requests, fetch resources or information, and manipulate them so that they can be sent back to the client, like Node.js.

12. What is negative infinity?

Negative Infinity is a number in JavaScript, which is derived by 'dividing negative number by zero'. It can be better understood as a number that is lower than any other number. Its properties are as follows:

- A number of objects need not to be created to access this static property.
- The value of negative infinity is the same as the negative value of the infinity property of the global object.

13. What are JavaScript Data Types?

Following are the JavaScript Data types:

Number	represents numeric values, e.g., 10
String	represents a sequence of characters, e.g., "hello world"
Boolean	represents Boolean value either false or true
Undefined	represents an undefined value
Object	represents an instance through which we can access members
Array	represents a group of similar values
Null	represents null, i.e., no value



14. What is the difference between undefined value and null value?

Undefined value: A value that is not defined and has no keyword is known as undefined value.

For example:
str string;
//Here, string has an undefined value.

Null value: A value that is explicitly specified by the keyword "null" is known as a null value.

For example:
String str=null;
//Here, str has a null value.

15. What are the valid scopes of a variable in JavaScript?

The scope of a variable is the region of your program in which it is defined. JavaScript variable will have only two scopes.

Global Variables - A global variable has global scope which means it is visible everywhere in your JavaScript code.

Local Variables - A local variable will be visible only within a function where it is defined. Function parameters are always local to that function.

16. Which type of variable among global and local, takes precedence over other if names are same?

A local variable takes precedence over a global variable with the same name.

17. What are the meanings of undefined variables and undeclared variables?

Undefined variables are the types of variables which gets declared in the program. However, the value of undefined variables is not provided in JavaScript programming language. The value gets returned if one tries to read the undefined value.

Undeclared variables are the types of variables which do not even exist in a program and are also not declared properly. The value also can't be read properly when tried to be read and hence a runtime error is shown.

18. What are anonymous functions in JavaScript?

An anonymous function allows a developer to create a function that has no name. In other words, anonymous functions can be used to store a bit of functionality in a variable and pass that piece of functionality around.

19. What is the difference between an Anonymous Function and a named function?

Anonymous functions exist only after they are called; whereas, named functions to exist even if not called.

20. Explain the terms synchronous and asynchronous code.

Synchronous code is something that should be finished before anything else can happen, or in other words, the synchronous code is a blocking code.

Asynchronous code is something in which actions can happen and is not dependent on other actions- in other words, it is a non-blocking code.

21. What is event bubbling?

Bubbling just works like the bubbles, the event gets handled by the innermost element and then propagated to the outer element. JavaScript allows DOM elements to be nested inside each other. In such a case, if the handler of the child is clicked, the handler of parent will also work as if it were clicked too.

22. What is the definition of global variable? In what way, these variables are declared?

Global variable is a special kind of variable in JavaScript. This kind of variable is something that is easy to use and also available across the entire length of the JavaScript code. Mainly, the var keyword is used whether to declare a global or local variable.

23. Name the problems that are associated with the use of global variables?

Even though global variables are easy to use, these have some shortfalls. While using this type of variables, the problem of clashing of the variable names of different global and local scope occurs. The code that is often relied on the global variable also gets difficult to be tested and debugged.

24. What are the various functional components in JavaScript?

The different functional components in JavaScript are-

First-class functions: Functions in JavaScript are utilized as first-class objects. This usually means that these functions can be passed as arguments to other functions, returned as values from other functions, assigned to variables or can also be stored in data structures.

Nested functions: The functions, which are defined inside other functions, are called Nested functions. They are called 'every time' the main function is invoked.

25. What is callback function?

Callback is a typical function of the JavaScript which can be passed as an option or argument of JavaScript. Sometimes, callbacks can also be termed as simple events. Users are given calls to react to different kind of triggered situations.

26. What are the primitive data types in JavaScript?

A primitive is a basic data type that's not built out of other data types. It can only represent one single value. All primitives are built-in data types by necessity, (the compiler has to know about them,) but not all built-in data types are primitives.

In JavaScript there are 5 primitive data types are available they are undefined, null, Boolean, string and number are available. Everything else in JavaScript is an object.

27. What are the different types of errors in JavaScript?

There are three types of errors:

Load time errors: Errors which come up when loading a web page like improper syntax errors are known as Load time errors and it generates the errors dynamically.

Run time errors: Errors that come due to misuse of the command inside the HTML language.

Logical Errors: These are the errors that occur due to the bad logic performed on a function which is having different operation.

28. What is Currying in JavaScript?

A partial invocation of a JavaScript function is called Currying. Few arguments of a function are processed, and a function is returned. Few more arguments are added by the returning function.

29. What is Hoisting in JavaScript?

Hoisting is the JavaScript interpreter's action of moving all variable and function declarations to the top of the current scope.

There are two types of hoisting:

variable hoisting - rare

function hoisting - more common

Wherever a var (or function declaration) appears inside a scope, that declaration is taken to belong to the entire scope and accessible everywhere throughout.

30. Is JavaScript case sensitive?

Yes, JavaScript is a case-sensitive language. This means that language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters.

31. How many types of objects are there in JavaScript?

There are two types of objects in JavaScript -

Date object - This type of object is built within the JavaScript programming language. The date objects are created with the help of new Date. It can be operated with the help of a bunch of methods once it is created. The methods allow the inclusion of the year, the month, day, hour and even minutes, seconds and millisecond of the date object. These are set with the help of local standard time or universal time.

Number object - This type of number object also includes the dates as it solely numerical dates are represented by it like integers and fractions. The literals of the numbers get converted to the number class automatically.

32. Explain “use strict” in JavaScript ?

“use strict” is a JavaScript directive that is introduced in Es5. The purpose of using “use strict” directive is to enforce the code is executed in strict mode. In strict mode we can’t use a variable without declaring it. “use strict” is ignored by earlier versions of JavaScript.

33. Explain typecasting in JavaScript?

In Programming whenever we need to convert a variable from one data type to another typecasting is used. In JavaScript, we can do this via library functions. There are basically 3 typecasts are available in JavaScript Programming, they are:

Boolean(value): Casts the inputted value to a Boolean

Number(value): Casts the inputted value to an Integer or Floating-point Number.

String(value) : Casts the inputted value a string

34. How to create objects in JavaScript?

There are 3 ways to create an object in JavaScript.

By object literal

By creating an instance of Object

By Object Constructor

35. What is the difference between View state and Session state?

-View State is a state of a page within a browser wherein the values of controls persist when post back operation is done. When another page is loaded, the previous page data is no longer available.

- Session State is the data of a user session and is maintained on the server side. This data available until user closes the browser or session time-outs.

36. Write the point of difference between web-garden and a web-farm?

Both web-garden and web-farm are web hosting systems. The only difference is that web-garden is a setup that includes many processors in a single server while web-farm is a larger setup that uses more than one server.

37. What is NaN in JavaScript?

The NaN property represents "Not-a-Number" value. This property indicates that a value is not a legal number. The NaN property is the same as the Number. Nan property.

38. Explain the working of timers in JavaScript?

Timers are used to execute a piece of code at a set time or also to repeat the code in a given interval of time. This is done by using the functions setTimeout, setInterval and clearInterval.

The setTimeout(function, delay) function is used to start a timer that calls a particular function after the mentioned delay.

The setInterval(function, delay) function is used to repeatedly execute the given function in the mentioned delay and only halts when cancelled.

The clearInterval(id) function instructs the timer to stop.

39. Name the different types of pop up boxes in JavaScript.

There are three types of pop up boxes in JavaScript

(i) alert() provides some information to the user with just an OK button

(ii) confirm() asks a question to the user with two options Ok and cancel

(iii) prompt() takes an input from the user

40. What are the types of functional components available in JavaScript

There two types of functional components those are available in JavaScript. These are -

1. Nested functions - As the name suggests, nested functions are the ones which are included in other functions. They are basically called each time when the invoked functions are provided.

2. First class functions - These are the types of functions which are used as first-class objects. Such functions can be used as arguments and opinions against other types of functions. The data structures can store the values of such functional components and the variables can also be stored properly in the data structures.

41. What is closure?

Closures are created whenever a variable that is defined outside the current scope is accessed from within some inner scope.

42. What is Date object in JavaScript?

Date object is a datatype built into the JavaScript language. Date objects are created with the new Date().

Once a Date object is created, a number of methods allow you to operate on it. Most methods simply allow you to get and set the year, month, day, hour, minute, second, and millisecond fields of the object, using either local time or UTC (universal, or GMT) time.

43. What is Number object in JavaScript?

Number object represents numerical date, either integers or floating-point numbers. In general, you do not need to worry about Number objects because the browser automatically converts number literals to instances of the number class.

Creating a number object -

```
var val = new Number(number);
```

If the argument cannot be converted into a number, it returns NaN (Not-a-Number).

44. What is a prompt box?

A prompt box is a box which allows the user to enter input by providing a text box. Label and box will be provided to enter the text or number.

45. What are JavaScript Cookies?

Cookies are the small text files stored in a computer and it gets created when the user visits the websites to store information that they need. Example could be User Name and password from the previous visits.

46. Can you access Cookie using JavaScript?

JavaScript can also manipulate cookies using the cookie property of the Document object. JavaScript can read, create, modify, and delete the cookie or cookies that apply to the current web page.

47. How does one read a cookie in JavaScript?

JavaScript is easily used to read a cookie. The cookie is basically just the value of another object which is the document.cookie. This string can be used properly to access various cookies. The same string keeps accounts of the name list and writes different cookies in different ways like by using semicolons. Generally, the value of the cookie is just the string value.

48. How to delete Cookies using JavaScript?

Deleting a cookie refers to not being able to read a cookie. This happens if the expiration date of the cookie is set to a specific time in the past. In this way, the deleted cookie is not being able to be viewed by any user.

49. How to create an object in JavaScript?

An object in JavaScript can be created using two was:

New Key word:

When a JavaScript variable is declared with the keyword "new", the variable is created as an object:

```
var x = new String(); // Declares x as a String object
```

Anonymous Object:

Anonymous objects can be created using pair of curly braces containing property name and value pairs.

```
var car = {type: "BMW", model: "X6", color: "Black"};
```

50. What is the role of break and continue statements?

Break statement is used to come out of the current loop while the continue statement continues the current loop with a new recurrence.

51. What is JavaScript screen object?

The JavaScript screen object holds information of browser screen. It can be used to display screen width, height, colorDepth, pixelDepth etc.

52. What are the properties of screen object?

There are many properties of screen object that returns information of the browser.

width	returns the width of the screen
height	returns the height of the screen
availWidth	returns the available width
availHeight	returns the available height
colorDepth	returns the color depth
pixelDepth	returns the pixel depth.

53. Why it is not advised to use innerHTML in JavaScript?

innerHTML content is refreshed every time and thus is slower. There is no scope for validation in innerHTML and, therefore, it is easier to insert rogue code in the document and, thus, make the web page unstable.

54. What is the difference between the operators '==' and '==='?

The operator '==' compares the value; whereas, the operator '===' compares both value and type.

55. What is the difference between typeof and instanceof operators in Javascript?

The typeof operator returns a string of what type the operand is. Whereas, the instanceof operator does not work with primitive data types; but works with objects and checks on what type the object is.

56. What is the difference between firstChild and firstElementChild?

A firstChild when used returns the first node. It could be an HTML element or even a space, or a new line. Whereas, firstElementChild, when used returns the first HTML element only.

57. What are the disadvantages of using inner HTML in JavaScript?

The use of inner HTML in JavaScript has the following disadvantages -

1. The process of using inner HTML is much slower than the rest of the variables as its content is slowly built into different elects and takes time to get re-parsed.
2. While using the inner HTML, the content gets replaced in JavaScript.
3. Appending to inner HTML can't be used properly.
4. Using inner HTML can also break the document of the JavaScript. Since no validation is required by it, any type of valid inner HTML can be used. Even broken HTML can also be used, and this can cause problems.
5. The old content also gets easily replaced

58. Name the different type of groups of data types that are used in JavaScript?

There are two basic groups of data types -

Reference type - These are complex types of data which can mainly include dates and strings.

Primitive type - These are types of data includes number data.

59.What is JavaScript function?

A JavaScript function is a block of code designed to perform a particular task.

A JavaScript function is executed when "something" invokes it (calls it).

Example

```
function myFunction(f1, f2) {  
    return f1 + f2;      // The function returns the sum  
    of f1 and f2  
}
```

60. How would you create a private variable in JavaScript?

To create a private variable in JavaScript that cannot be changed you need to create it as a local variable within a function. Even if the function is executed the variable cannot be accessed outside of the function.

61. Explain Revealing Module Pattern design pattern.

A variation of the module pattern is called the Revealing Module Pattern. The purpose is to maintain encapsula-

tion and reveal certain variables and methods returned in an object literal.

62. Explain Prototype Design Pattern.

The Prototype Pattern creates new objects, but rather than creating non-initialized objects it returns objects that are initialized with values copied from a prototype - or sample - object. The Prototype pattern is also referred to as the Properties pattern.

Classical languages rarely use the Prototype pattern, but JavaScript being a prototypal language uses this pattern in the construction of new objects and their prototypes.

63. Explain JavaScript Event Delegation Model.

When capturing and bubbling, allow functions to implement one single handler to many elements at one particular time then that is called event delegation. Event delegation allows you to add event listeners to one parent instead of specified nodes and that particular listener analyzes bubbled events to find a match on the child elements.

64. Explain arrow functions.

Arrow functions allow a short syntax for writing function expressions.

Arrow functions do not have their own this. They are not well suited for defining object methods.

Arrow functions are not hoisted. They must be defined before they are used.

Using const is safer than using var, because a function expression is always constant value.

You can only omit the return keyword and the curly brackets if the function is a single statement. Because of this, it might be a good habit to always keep them:

Example

Example

// ES5

```
var x = function(x, y) {
```

```
    return x * y;
```

```
}
```

// ES6

```
const x = (x, y) => x * y;
```

65. What is the difference between the substr() and substring() functions in JavaScript?

substr() allows you to specify the maximum length to return

substring() allows you to specify the indices and the second argument is NOT inclusive

The second argument of substring method takes end index, which is excluded from final substring, while second argument of substr method takes maximum length of expected substring.

```
var str = "JavaScript"  
alert(str.substr(4, 6)); // this will return Script  
alert(str.substring(4, 6)); // this will return "Sc"
```

66. How to write a comment in JavaScript?

There are two types of comments in JavaScript.

Single Line Comment: It is represented by // (double forward slash)

Multi-Line Comment: Slash represents it with asterisk symbol as /* write comment here */

67. How to convert Javascript date to ISO standard?

toISOString() method is used to convert JavaScript date to ISO standard. It converts JavaScript Date object into a string, using the ISO standard.

68. What is anonymous function?

It is a function that has no name. These functions are declared dynamically at runtime using the function operator instead of the function declaration. The function operator is more flexible than a function declaration.

For example:

```
var x = function (a, b) {return a * b};  
var z = x(4, 3);
```

The function above is an anonymous function (a function without a name).

Functions stored in variables do not need function names. They are always invoked (called) using the variable name.

69. What is purpose of onError event handler in JavaScript?

The onerror event handler is a feature to facilitate error handling for JavaScript. The error event is fired on the window object whenever an exception occurs on the page.

The onerror event handler provides three pieces of information to identify the exact nature of the error -

Error message - The same message that the browser would display for the given error.

URL - The file in which the error occurred.

Line number - The line number in the given URL that caused the error.

70. What is the use of the 'this' keyword?

The JavaScript "this" keyword refers to the object it belongs to. The "this" keyword has different values depending on where it is used. In a method, "this" refers to the owner object. In a function, "this" refers to the global object.

71. Define various types of errors which occur in JavaScript ?

JavaScript has three types of errors as follows-

Run time errors - This type of error is the outcome of the misuse of the use of command within the HTML language.

Load time errors - Load time errors are syntax errors which are improper and arise when a web page is tried to be loaded. This type of error is generated dynamically.

Logical errors - A function often has a different operation and this type of error arises when the logic of the function is badly performed.

72. How to compare two objects in JavaScript?

JavaScript object deep comparison. Comparing `x === y`, where x and y are values, return true or false. Comparing `x === y`, where x and y are objects, returns true if x and y refer to the same object. Otherwise, returns false even if the objects appear identical.

73. What are the difference between var, let, and const keywords in JavaScript?

Var: The JavaScript variables statement is used to declare a variable and, optionally, we can initialize the value of that variable.

Let: The let statement declares a local variable in a block scope. It is similar to var, in that we can optionally initialize the variable.

Const: const statement values can be assigned once and they cannot be reassigned. The scope of const statement works similar to let statements.

74. What are Transpilers?

Transpilers, or source-to-source compilers, are tools that read source code written in one programming language and produce the equivalent code in another language.

75. What does the instanceof operator do?

The instanceof operator tests whether the prototype property of a constructor appears anywhere in the prototype chain of an object. It checks whether the object is an instance of a class or not.

76. Which type of variable among global and local, takes precedence over other with same name?

If two variables exist with the same name, one global and one local, the local has priority.

Example:-

```
int i=3;  
void setup() {  
    int i=7;  
    println(i);  
    println(this.i);  
}
```

output is 7,3 because local variable only gives first priority when the variable inside the method with same name.

77. What is the use of Math object in JavaScript?

The JavaScript Math object allows you to perform mathematical tasks on numbers.

Examples:

```
Math.round(4.7); // returns 5
```

```
Math.sqrt(64); // returns 8
```

```
Math.sin(90 * Math.PI / 180); // returns 1
```

78. What is the use of a Boolean object in JavaScript?

The JavaScript Boolean is an object that represents value in two states: true or false. You can create the JavaScript Boolean object by Boolean() constructor.

```
Boolean(10 > 9) // returns true
```

79. What are the WeakSet objects in JavaScript?

WeakSet objects are collections of objects. An object in the WeakSet may occur only once; it is unique in the WeakSet's collection.

The main differences to the Set object are:

In contrast to Sets, WeakSets are collections of objects only and not of arbitrary values of any type.

The WeakSet is weak: References to objects in the collection are held weakly. If there is no other reference to an object stored in the WeakSet, they can be garbage collected. That also means that there is no list of current objects stored in the collection. WeakSets are not enumerable.

80. What is a map object?

A Map is an object that maps keys to values. A map cannot contain duplicate keys: Each key can map to at most one value. It models the mathematical function abstraction.

81. What is the use of a Map object in JavaScript?

The JavaScript Map object is used to map keys to values. It stores each element as key-value pair. It operates the elements such as search, update and delete on the basis of specified key.

82. What is the use of a WeakMap object in JavaScript?

The JavaScript WeakMap object is a type of collection which is almost similar to Map. It stores each element as a key-value pair where keys are weakly referenced. Here, the keys are objects and the values are arbitrary values.

83. List out some comparison operators supported by Javascript?

Javascript supports below comparison operators

- > Greater than
- < Less than
- <= Less than or equal to
- >= Greater than or equal to
- == Equal to

`!=` Not Equal to
`==` Equal to with datatype check
`!==` Not equal to with datatype check

84. Can you assign an anonymous function to a variable? Can you pass an anonymous function as an argument to another function?

Yes, an anonymous function can be assigned to a variable.

Yes, an anonymous function can be passed as an argument to another function.

85. How to print a web page using JavaScript?

JavaScript helps you to implement this functionality using print function of window object. The JavaScript print function `window.print()` will print the current web page when executed.

86. How to handle exceptions in JavaScript?

The latest versions of JavaScript added exception handling capabilities. JavaScript implements the try, catch & finally construct as well as the throw operator to handle exceptions.

```
Try{  
    Code  
}  
Catch(exp){  
    Code to throw an exception  
}  
Finally{  
    Code runs either it finishes successfully or after catch  
}
```

87. What is the difference between `window.onload` and `onDocumentReady`?

The `onload` function is not run until all the information on the page is loaded. This leads to a substantial delay before any code is executed.

`onDocumentReady` loads the code just after the DOM is loaded. This allows early manipulation of the code.

88. What are exports and imports?

Imports and exports help us to write modular JavaScript code. Using Imports and exports we can split our code

into multiple files. Imports allow taking only some specific variables or methods of a file. We can import methods or variables that are exported by a module.

89. What is the use of blur function?

Blur function is used to remove the focus from the specified object.

90. What does `2 + 4 + "6"` evaluate to?

Since 2 and 4 are integers, this is number arithmetic, since 6 is a string, it is concatenation, so the result is 66.

91. What does "`2`" + `4` + `6` evaluate to?

Since 2 is a string, everything is a string, so the result is 246.

91. What will happen if an infinite while loop is run in Javascript?

The browser will crash.

92. What are the variable naming conventions in JavaScript?

While naming the variables in JavaScript following rules should be kept in mind:

Any JavaScript reserved keyword should not be used as variable name.

JavaScript variable names should not start with a numeral (0-9). They must begin with a letter or the underscore character.

JavaScript variable names are case sensitive. For example, Student and student are two different variables.

93. Which built-in method removes the last element from an array and returns that element?

`pop()` method removes the last element from an array and returns that element.

94. Which built-in method adds one or more elements to the end of an array and returns the new length of the array?

`push()` method adds one or more elements to the end of an array and returns the new length of the array.

95. What is the use of `Void(0)`?

`Void(0)` is used to prevent the page from refreshing and parameter "zero" is passed while calling.

`Void(0)` is used to call another method without refreshing the page.

96. List out some Mouse Events.

Below are the 9 mouse events:

click: A single click

dblclick: A double click

mousedown: When the mouse button is clicked down.

mouseup: When the mouse button is released up.

mouseenter: When the mouse cursor enters an external element.

mouseleave: When the mouse cursor leaves an external element.

mouseover: When the mouse cursor enters an internal and external element.

mouseout: When the mouse cursor leaves an internal and external element.

mousemove: When the mouse cursor is moved.

97. What is a token in JavaScript?

A token is a single element of a programming language. There are five categories of tokens: 1) constants, 2) identifiers, 3) operators, 4) separators, and 5) reserved words. For example, the reserved words "var" and "let" are tokens of the JavaScript language.

98. What is IIFEs ? When it is used?

It is the short form of Immediately Invoked Function Expressions.

It executes immediately after it's created. It is often used when trying to avoid polluting the global namespace, because all the variables used inside the IIFE (like in any other normal function) are not visible outside its scope.

99. What is prototypal Inheritance?

Every object has a property called a prototype, where we can add methods to it and when you create another object from these the newly created object will automatically inherit its parent's property.

100. What is Lexical Scope in JS?

Whenever you see a function within another function, the inner function has access to the scope in the outer function, this is called Lexical Scope or Closure - also referred to as Static Scope.

101. Explain the unshift() method ?

This method is functional at the starting of the array, unlike the push(). It adds the desired number of elements to the top of an array.

For example -

```
var name = [ "banana" ];  
name.unshift( "orange" );  
name.unshift( "apple", "grapes" );  
console.log(name);
```

The output is shown below:

```
["apple ", "grapes ", "orange ", "banana "]
```

102. How to read and write a file using JavaScript?

There are two ways to read and write a file using JavaScript

- Using JavaScript extensions
- Using a web page and Active X objects

103. How can we detect OS of the client machine using JavaScript?

The navigator.appVersion string can be used to detect the operating system on the client machine.

104. How to create an array in JavaScript?

There are 3 ways to create an array in JavaScript.

By array literal

By creating an instance of Array

By using an Array constructor

105. Explain Scope Chain.

Scope chains establish the scope for a given function. Each function defined has its own nested scope as we know, and any function defined within another function has a local scope which is linked to the outer function - this link is called the chain. It's always the position in the code that defines the scope. When resolving a variable, JavaScript starts at the innermost scope and searches outwards until it finds the variable/object/function it was looking for.