

Javascript Interview Questions & Answers 2015: JS Basic & Advanced

Javascript interview questions: JavaScript is a very widely used scripting language in web development. If you are a fresher who wants to build a career in JavaScript or an experienced developer, these frequently asked JavaScript interview Questions will definitely help you in preparing yourself for a job interview. These interview questions of Java Script have been prepared by the experts who are experts in web development and are involved in the recruitment of JavaScript developers. You will rarely find such a gold mine of interview questions based on JS and its frameworks.

The compatibility of JavaScript with HTML and languages based on OOPS makes it the most preferred scripting language that adds to the interactivity of the website. You can pursue a very lucrative career with JavaScript as it is a technology that is here to stay. In this post, you can find the answers to both basic and advanced JavaScript Interview questions that will be beneficial for experienced as well as the freshers.

If you' re about to apply for a new exciting position and you feel like giving your JavaScript skills a test, this is the best moment to revisit your knowledge and make sure that you come for your interview well-prepared. Here are top 20 most common questions about JavaScript that professionals usually get asked during job interviews.

Some Frequently Asked JavaScript Interview Questions



Question: Differentiate between Java and JavaScript?

Answer: While Java is a complete programming language, JavaScript is basically a coded program which developers can introduce to HTML pages. Java and JavaScript are two languages that aren't inter-dependent – they were devised to serve completely different purposes.

It's worth to mention that Java has been programmed keeping the object-oriented programming (OOPS) (or structured programming language) factor in mind – just like C++ or C. When it comes to JavaScript, it is a client-side scripting language and it's generally considered to be an example of unstructured programming.

Question: When working with JavaScript, how do you organize your code?

Answer: When asking this kind of question, the interviewer usually wants to know how the candidates design and maintain their code. Employers will be interested to know whether you design code which is application-specific and will have no other use, or whether you use the module pattern or class inheritance to build a reusable code.

There exist approaches which allow a number of developers to work together on a project without interfering with each other's work. It's obvious that testing modular code or classes is a more approachable situation than a complex mess of code thrown together by one developer.

Question: How to differentiate between null and undefined?

Answer: Let's enter the tricky land of technical JavaScript questions. JavaScript basically has two distinct values to stand for nothing – null and undefined. While undefined means that value of the variable is not defined, null stands for empty or non-existent value used by programmers to indicate “no value”.

A few words about undefined. JavaScript has a global variable undefined bearing the value of “undefined”, as well as a type of undefined which is also “undefined”. Don't get confused here – undefined is not a keyword or constant – it's a type bearing a single value: undefined. Even if you assign a new value to it, you won't change the value of the type undefined.

Null, on the other hand, isn't a type or an object – it's a primitive value. One of its characteristics as a value is the fact that you can't add properties to it. Many people wrongly assume null to be an object, mostly because type of null returns “object”.

Question: What is event bubbling?

Answer: This is a common question that pops up during interviews. Event bubbling basically defines the action of events in the Document Object Model (DOM) – in child and parent nodes, where all child node events are conveyed to their parent nodes automatically.

The main advantage of this method is the increased speed of coding, because the code

will need to pass over the DOM tree once only. Event bubbling is a good option if you' re placing more than one event listener on a DOM element – you can use all of the elements to put just one listener, if you want to improve code simplicity and reduction. With event bubbling, you can create one event listener on the page' s body element that will react to any click event that will occur within that element.

Question: Highlight the main differences between GET and POST methods in HTML form?

Answer: Using a GET method, you can get your key and values appended at the end of the URL in the form of a query string. Most developers avoid using GET method when they' re passing some sensitive data over the internet. When you' re sending data, the GET method basically will extend the length of the URL – and we all know that it' s limited (latest browsers can support not more than 2048 characters).

POST method is used by developers to pass sensitive data in such a way as to not append it and display it in the URL. In POST, there are no limitations when it comes to the amount of data that can be passed.

Question: How do you handle errors in JavaScript?

Answer: If we' re talking about exceptions which occur at runtime, those are best handled by means of try / catch / finally blocks – this way you' ll avoid those awful error notifications. The finally block is only optional, but try/catch is a must.

What you do is basically attempt to run code in the try block located between the braces. Execution is then shifted to the catch block of code in case runtime errors take place. Once the try/catch block is completed, your code execution will transfer to the finally code block. This logic is similar to C# or Java.

Question: How do you distinguish undefined variable and undeclared variable?

Answer: The variable that are declared in the program but no value is assigned to them, are called undefined variable. Undeclared variables are the ones that are not declared in the program.

```
1 Undeclared Variable:
2 <script>
3 var p;
4 alert(a); // This will give error because a is undeclared variable.
5 </script>
6
7 undefined variable
8 <script>
9 alert(a); //This will give error because a is undefined.
10 </script>
```

Question: Why do developers include ‘use strict’ at the beginning of a JavaScript source file? What are the benefits of this strategy?

Answer: Using ‘use strict’ , developers enforce stricter parsing and error handling on their JavaScript code at runtime. Code errors that could have been ignored or failed with no sign whatsoever will now generate errors or throw exceptions.

All in all, using ‘use strict’ is an excellent measure against losing time later on improving the code. Its main benefits are as follows: it makes debugging easier, it eliminates a ‘this’ value coercion, it prevents the creation of accidental global variations, it throws error on invalid use of ‘delete’ , it makes eval() safer and it doesn’t allow duplicate property names or parameter values.

Question: How to request data from the server without the page being reloaded in the browser?

Answer: JavaScript that is being executed on the clients' system can request the data from the server by using XMLHttpRequest object. This method opens the connection and XMLHttpRequest.send() function sends the request to the server in real time. The following code makes it simpler to understand.

```
1 var oRequest = new XMLHttpRequest();
2 var sURL = "http://" + self.location.hostname + "/hello/requested_file.htm";
3 oRequest.open("GET", sURL, false);
4 oRequest.setRequestHeader("User-Agent", navigator.userAgent);
5 oRequest.send(null)
6 if (oRequest.status == 200) alert(oRequest.responseText);
7 else alert("Error executing XMLHttpRequest call!");
```

Question: What are JavaScript timers and what are their benefits/drawbacks?

Answer: Another popular interview question. Timers basically allow developers to execute code at a defined time or do it repeatedly when using an interval. This is done with the following functions: setTimeout, setInterval and clearInterval.

The setTimeout(function, delay) function will initiate a timer which calls a specific function after a specified delay – it returns an id value which can later be used to access it. The setInterval(function, delay) function is similar to the former, but it executes repeatedly on the specified delay and stops only when canceled. The clearInterval(id) function stops the timer.

Timers can sometimes be tricky to use because they operate within a single thread, and events might queue up when waiting to be executed.

Question: What are the main differences between ‘==’ and ‘===’ ?

Answer: The answer is simple: == is tolerant and won't check types and === will

definitely check whether both sides are of the same type. Still, `==` converts to its convenient type in order to have both sides in same type and then does the comparison.

`===`, on the other hand, compares the types and values. If both sides aren't the same type, the answer is always false. For instance, if you're comparing two strings of code, they must have character sets that are identical. Primitives like number or boolean must share the same value.

Question: What is the function of `unshift()` method in Javascript?

Answer: The `unshift()` method resembles the `push()` method with the difference that it works only at the start of the array. You can prepend one or more elements to the array by using this method. It is, hence, used to alter the array on which the method is called.

```
1 var vegetable= [ "potato" ];  
2 vegetable.unshift( "brinjal" );  
3 vegetable.unshift( "cabbage", "jackfruit" );  
4 console.log(vegetable);  
5  
6 we get the following console output:  
7 ["cabbage", " jackfruit ", " brinjal ", " potato "]
```

Question: What JavaScript types can you list?

Answer: Some interviewers will be interested in theory basics too, so make sure to refresh your knowledge. JavaScript has been referred as a loosely-typed language, which means that no type declarations are required when you're creating variables.

In JavaScript, strings and numbers can be mixed together with no problems. JavaScript will also easily identify them and determine what the type should be. The types supported in JavaScript are as follows: Number, String, Boolean, Function, Object, Null, and Undefined.

Question: How to write script-generated content to another window?

Answer: This is one of the advanced JavaScript interview questions that is asked frequently. Two methods namely `winRef.document.writeln()` or `winRef.document.write()` can be used for this purpose. WinRef stands for Window reference and is returned by `Window.open()` method in the following code.

```
1writeConsole('Hello Guys!');
2function writeConsole(content) {
3top.consoleRef=window.open('', 'myconsole', 'width=350,height=250'
4+',menubar=0')
5top.consoleRef.document.writeln( '<html>
6<head>
7<title>My Console</title>
8</head>' + ' top.consoleRef.document.close() }
```

Question: How can `Math.max` be used to find the max value in a given array?

Answer: You should basically use ‘`apply`’ on `Math.max` and, as ‘`apply`’ covers an array of arguments. Given the fact that we’re not analyzing anything from this array or using it in some way, we can basically pass ‘`null`’ as first parameter and get it over with.

Question: How would you write a simple function that will indicate whether 2 is as parameter passed or not?

Answer: First things first, the basics – ‘`arguments`’ is a local variable, which is available inside all the functions. It shows all the arguments that are passed to the function. ‘`arguments`’ is an array-like object – it has a certain length but it doesn’t include the methods like `forEach`, `indexOf` and others.

To tell whether 2 is passed as parameter, you should first convert ‘`arguments`’ to an

array. You can do that by calling slice method on an array and pass arguments. After this, you should go for indexOf, and you're done.

One thing to remember: Avoid naming your arguments as 'arguments' or creating local variables that are called 'arguments' – you don't want to risk overriding build in arguments object.

Question: Which is faster between the two- JavaScript or ASP script?

Answer: This is one of the most frequently asked JavaScript interview questions. JavaScript is the faster between the two. This is because JavaScript is a client-side scripting language and does not require the web server to execute. ASP, on the contrary, is a server-side script and requires longer time duration to execute.

The following code will make the point clear:

```
01 var band = {
02   "name": "The Red Hot Chili Peppers",
03   "members": [
04     {
05       "name": "Anthony Kiedis",
06       "role": "lead vocals"
07     },
08     {
09       "name": "Michael 'Flea' Balzary",
10       "role": "bass guitar, trumpet, backing vocals"
11     },
12     {
13       "name": "Chad Smith",
14       "role": "drums, percussion"
15     },
16     {
17       "name": "John Frusciante",
18       "role": "Lead Guitar"
19     }
20   ],
21   "year": "2009"
22 }
```

Question: Explain 'this' in JavaScript?

Answer: When executing any kind of function, JavaScript engine will set a property to a function called 'this' which refers to the execution context. 'this' always refers to an object and will impact on how a function is called. 'this' can have different values, here's an overview:

- Inside a function or in the global context, ‘this’ refers to the window object
- Inside immediate invoking function using “use strict” , the value of ‘this’ will be undefined
- When executing a function in terms of an object, the object will become the value of ‘this’
- Inside a setTimeout function, the value of ‘this’ will be the window object
- If you use a constructor (new keyword, for instance) to form an object, the value of ‘this’ will refer to the newly created object
- The value of ‘this’ can be set to any arbitrary object as you pass the object as the first parameter of apply, call, or bind

Question: How would you add new elements dynamically with the help of JavaScript?

Answer: This question is one of the most commonly asked JavaScript interview questions and answers for experienced developers. The following code can be written to perform the addition of new elements dynamically.

```
1
2 <html>
3 <head> <title>Title</title>
4 <script type="text/javascript">
5 function addNode() { var newP = document.createElement("p");
6 var textNode = document.createTextNode(" This is a new text node");
7 newF.appendChild(textNode); document.getElementById("firstP").appendChild(newP); }
8 </script> </head>
9 <body>
10 <p id="firstP">First Paragraph<p>
11 </body>
12 </html>
13
```

Question: How would you perform the execution of cache for any function?

Answer: You can use the following method: you pass a function and the function will internally maintain a cache object where the calculated value will be cached. When calling the function with the same argument, the value which has already been cached will be invoked here.

Question: Global variables – what are they, how are they declared and what problems can they generate?

Answer: Global variables can be added throughout your code – they basically have no scope. Local variables have a scope and are restricted to where they’re declared, for instance within a given function. This is where the var keyword will be used in order to declare a local variable or object – omitting the var keyword, you’ll create a global variable.

It’s a fact that the majority of JavaScript developers try to avoid global variables. Such variables tend to generate naming conflicts between local and global, not to mention that code which depends on global is often difficult to test and maintain.

Question: What is the value returned by prompt() return if the user clicks the Cancel button?

Answer: It depends on the user’s browsers. Most of the browsers return a null value and a few of them return an empty string. You can use the following code to test it.

```
1 userInput = prompt('Prompt text','your Suggested input');  
2 if (userInput) {  
3 // You can do something with the input
```

Question: How can you explain the deferred scripts using event handlers in JavaScript ?

Answer: This is among the most frequently asked JavaScript interview questions. When you use an Event handler, a trigger is allowed to be generated whenever a user clicks a button, an image, etc. The OnLoad event handler handles all the components of the page including Java applets, images and embedded multimedia, and loads them in the

browser. The code below explains how onLoad event handler triggers the done() function.

```
1
2
3
4 <HTML>
5 <HEAD>
6 <TITLE>Event_handler Code</TITLE>
7 <SCRIPT LANGUAGE=" Text/JavaScript" >
8 <!--
9 function done() {
10 alert( "Loading Finished" )
11 }
12 // -->
13 </SCRIPT>
14 </HEAD>
15 <BODY Event_handler=" done()" >
16 </BODY>
17 </HTML>
18
19
20
21
22
23
24
```

Question: Describe undeclared and undefined variables?

Answer: Undeclared variables are variables that don't exist in a program and therefore, aren't declared. Once the program tries to read the value of an undeclared variable, it'll encounter a runtime error.

Undefined variables, on the other hand, are the variables which are declared in the program but they don't bear any value. Once the program runs and tries to read the value of an undefined variable, you'll just have an undefined value returned to you.

Question: Explain what is a prompt box?

Answer: A prompt box is a box that allows users to enter input – it provides a box and a label for the users to enter text or numbers.

Question: Does JavaScript pass parameters by value or by reference?

Answer: While primitives such as string or number are passed by value, reference is used for objects for the same. Now, if you change the property of the passed object, the change will be reflected in the code. If you decide to allocate a new object to the already passed object, this kind of alteration will have no reflection in the code.

Question: List the functional components in JavaScript?

Answer: The functional components in JavaScript are as follows:

First-class functions - Utilized in JavaScript as first class objects, you can pass the functions to other functions as arguments, assigned to variables, returned as values from other functions or stored in data structures.

Nested functions - The functions explicated within other functions are called Nested functions. They will be automatically called ‘everytime’ the main function is invoked.

Question: Why is unescape() method used?

Answer: unescape() function helps in decoding the encoded string

Syntax: unescape(string1)

The string1, here is the string that is to be decoded.

```
1 <script>
2 document.write(unescape( "Latest%20Java%20Interview%20Questions%20and%20Answers" ));
3 </script>
```

These javascript interview questions and answers have been prepared by the experts of the subject ,who have been working on JavaScript for years. The list contains both basic JavaScript interview question and answer and advanced JavaScript interview questions and answers. These questions and answers on JavaScript will surely help you crack the interviews in any company and will provide you with a confidence which is a must if you

want to impress the interviewer and secure your position. While the list contains some JavaScript interview questions and answers for experienced, there are javascript interview questions and answers for freshers too.

Preparing to answer such questions will help you to showcase your expertise and land a great job as a JavaScript developer to work on a bunch of exciting projects.

We have put in a lot of efforts to compile this list of JavaScript Interview Questions. However, if you are not able to find the answer to the question that you are looking for, you can post your questions in the comment section below. Our experts will definitely post the answers to the questions raised by you.

Related Posts

- [Must to Know AJAX and JavaScript Cheat Sheets for Developers](#)
- [WordPress Interview Questions & Answers: WordPress CMS Interview Questions](#)
- [SEO Interview **Question:** SEO Interview Questions Answers 2015](#)