

37 Essential JavaScript Interview Questions *

Toptal sourced essential questions that the best JavaScript developers and engineers can answer. Driven from our community, we encourage experts to submit questions and offer feedback.

[Hire a Top JavaScript Developer Now](#)

INTERVIEW QUESTIONS

What is a potential pitfall with using `typeof bar === "object"` to determine if `bar` is an object? How can this pitfall be avoided?

[View answer](#)



What will the code below output to the console and why?

```
(function(){  
  var a = b = 3;  
})();  
  
console.log("a defined? " + (typeof a !== 'undefined'));  
console.log("b defined? " + (typeof b !== 'undefined'));
```

[View answer](#)



What will the code below output to the console and why?

```
var myObject = {  
  foo: "bar",  
  func: function() {  
    var self = this;  
    console.log("outer func:  this.foo = " + this.foo);  
    console.log("outer func:  self.foo = " + self.foo);  
    (function() {  
      console.log("inner func:  this.foo = " + this.foo);  
      console.log("inner func:  self.foo = " + self.foo);  
    })();  
  }  
};  
myObject.func();
```

[View answer](#)

Apply to Join Toptal's Development Network

and enjoy reliable, steady, remote [Freelance JavaScript Developer Jobs](#)

[Apply as a Freelancer](#)

What is the significance of, and reason for, wrapping the entire content of a JavaScript source file in a function block?

[View answer](#)

What is the significance, and what are the benefits, of including `'use strict'` at the beginning of a JavaScript source file?

[View answer](#)

Consider the two functions below. Will they both return the same thing? Why or why not?

```
function fool()
{
  return {
    bar: "hello"
  };
}

function foo2()
{
  return
  {
    bar: "hello"
  };
}
```

[View answer](#)

What will the code below output? Explain your answer.

```
console.log(0.1 + 0.2);
console.log(0.1 + 0.2 == 0.3);
```

[View answer](#)

In what order will the numbers 1-4 be logged to the console when the code below is executed? Why?

```
(function() {
  console.log(1);
  setTimeout(function(){console.log(2)}, 1000);
  setTimeout(function(){console.log(3)}, 0);
  console.log(4);
})();
```

[View answer](#)

Write a simple function (less than 160 characters) that returns a boolean indicating whether or not a string is a **palindrome**.

[View answer](#)

Write a `sum` method which will work properly when invoked using either syntax below.

```
console.log(sum(2,3)); // Outputs 5
console.log(sum(2)(3)); // Outputs 5
```

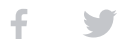
[View answer](#)

Consider the following code snippet:

```
for (var i = 0; i < 5; i++) {
  var btn = document.createElement('button');
  btn.appendChild(document.createTextNode('Button ' + i));
  btn.addEventListener('click', function(){ console.log(i); });
  document.body.appendChild(btn);
}
```

(a) What gets logged to the console when the user clicks on “Button 4” and why?

(b) Provide one or more alternate implementations that will work as expected.

[View answer](#)

Assuming `d` is an “empty” object in scope, say:

```
var d = {};
```

...what is accomplished using the following code?

```
[ 'zebra', 'horse' ].forEach(function(k) {  
    d[k] = undefined;  
});
```

[View answer](#)

What will the code below output to the console and why?

```
var arr1 = "john".split('');  
var arr2 = arr1.reverse();  
var arr3 = "jones".split('');  
arr2.push(arr3);  
console.log("array 1: length=" + arr1.length + " last=" + arr1.slice(-1));  
console.log("array 2: length=" + arr2.length + " last=" + arr2.slice(-1));
```

[View answer](#)

What will the code below output to the console and why ?

```
console.log(1 + "2" + "2");  
console.log(1 + +"2" + "2");  
console.log(1 + -"1" + "2");  
console.log(+ "1" + "1" + "2");  
console.log( "A" - "B" + "2");  
console.log( "A" - "B" + 2);
```

[View answer](#)

The following recursive code will cause a stack overflow if the array list is too large. How can you fix this and still retain the recursive pattern?

```
var list = readHugeList();  
  
var nextListItem = function() {  
    var item = list.pop();  
  
    if (item) {  
        // process the list item...    }  
}
```

```
        nextListItem();  
    }  
};
```

[View answer](#)

What is a “closure” in JavaScript? Provide an example.

[View answer](#)

What would the following lines of code output to the console?

```
console.log("0 || 1 = "+(0 || 1));  
console.log("1 || 2 = "+(1 || 2));  
console.log("0 && 1 = "+(0 && 1));  
console.log("1 && 2 = "+(1 && 2));
```

Explain your answer.

[View answer](#)

What will be the output when the following code is executed? Explain.

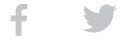
```
console.log(false == '0')  
console.log(false === '0')
```

[View answer](#)

What is the output out of the following code? Explain your answer.

```
var a={},  
    b={key: 'b'},  
    c={key: 'c'};  
  
a[b]=123;  
a[c]=456;
```

```
console.log(a[b]);
```

[View answer](#)

What will the following code output to the console:

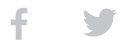
```
console.log((function f(n){return ((n > 1) ? n * f(n-1) : n)})(10));
```

Explain your answer.

[View answer](#)

Consider the code snippet below. What will the console output be and why?

```
(function(x) {  
  return (function(y) {  
    console.log(x);  
  })(2)  
})(1);
```

[View answer](#)

What will the following code output to the console and why:

```
var hero = {  
  _name: 'John Doe',  
  getSecretIdentity: function () {  
    return this._name;  
  }  
};  
  
var stoleSecretIdentity = hero.getSecretIdentity;  
  
console.log(stoleSecretIdentity());  
console.log(hero.getSecretIdentity());
```

What is the issue with this code and how can it be fixed.

[View answer](#)

Create a function that, given a DOM Element on the page, will visit the element itself and *all* of its descendents (*not just its immediate children*). For each element visited, the function should pass that element to a provided callback function.

The arguments to the function should be:

- a DOM element
- a callback function (that takes a DOM element as its argument)

[View answer](#)

Testing your `this` knowledge in JavaScript: What is the output of the following code?

```
var length = 10;
function fn() {
    console.log(this.length);
}

var obj = {
    length: 5,
    method: function(fn) {
        fn();
        arguments[0]();
    }
};

obj.method(fn, 1);
```

[View answer](#)

Consider the following code. What will the output be, and why?


```
(function () {  
  try {  
    throw new Error();  
  } catch (x) {  
    var x = 1, y = 2;  
    console.log(x);  
  }  
  console.log(x);  
  console.log(y);  
})();
```

[View answer](#)

What will be the output of this code?

```
var x = 21;  
var girl = function () {  
  console.log(x);  
  var x = 20;  
};  
girl ();
```

[View answer](#)

```
for (let i = 0; i < 5; i++) {  
  setTimeout(function() { console.log(i); }, i * 1000 );  
}
```

What will this code print?

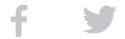
[View answer](#)

What do the following lines output, and why?

```
console.log(1 < 2 < 3);  
console.log(3 > 2 > 1);
```

[View answer](#)

How do you add an element at the beginning of an array? How do you add one at the end?

[View answer](#)

Imagine you have this code:

```
var a = [1, 2, 3];
```

a) Will this result in a crash?

```
a[10] = 99;
```

b) What will this output?

```
console.log(a[6]);
```

[View answer](#)

What is the value of `typeof undefined == typeof NULL`?

[View answer](#)

What would following code return?

```
console.log(typeof typeof 1);
```

[View answer](#)

What will be the output of the following code:

```
for (var i = 0; i < 5; i++) {  
    setTimeout(function() { console.log(i); }, i * 1000 );  
}
```

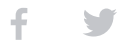
Explain your answer. How could the use of closures help here?

[View answer](#)



What is `NaN`? What is its type? How can you reliably test if a value is equal to `NaN`?

[View answer](#)



What will the following code output and why?

```
var b = 1;
function outer(){
    var b = 2
    function inner(){
        b++;
        var b = 3;
        console.log(b)
    }
    inner();
}
outer();
```

[View answer](#)



Discuss possible ways to write a function `isInteger(x)` that determines if `x` is an integer.

[View answer](#)



How do you clone an object?

[View answer](#)

