

JavaScript Interview Questions – Functions

Home › JavaScript › JavaScript Interview Questions – Functions



Before you head into that interview, make sure you understand how functions work, and the various components of their behavior.

Functions play a major role in the JavaScript language. In addition to providing private scope, they allow you to organize blocks of code for later execution. While most front-end developers have used JavaScript functions many times, there are some core areas of knowledge that are important to know.

I consider the following topics to be the baseline for a decent understanding of JavaScript functions.

```
1 function foo(a,b,c){  
2   return 10;  
3 }
```

Q: What is the length of the above function named: “foo”?

A: 3

Hint: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Function/length

Q: In a function, what does the word “arguments” refer to?

A:

It is a property of the function, and is an array-like list of the arguments that were actually passed into the function

Hint: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions_and_function_scope/arguments

```
1 function hello(){  
2   alert('hello');  
3 };
```

Q: What does the above function return?

A: “undefined” (without quotes)

Hint: If you do not specify a return value in a function, it returns **undefined** .

[return – JavaScript | MDN](#)

[undefined – JavaScript | MDN](#)



```

1 function foo(a,b,c){
2   console.log(a + b + c);
3 };
4
5 foo('hello', ' there ');

```

Q: Given the above code, what will be the console output when foo() is executed?

A: “hello there undefined”

Hint: When a function executes, any arguments that are not passed-in have a value of: **undefined**.

```

1 function hello(){
2   alert('hello');
3 };

```

Q: Given the code above, how would you programatically obtain a string representing the source code of the function?

A: foo.toString()

Hint:

[Function.prototype.toString\(\) – JavaScript | MDN](#)

```

1 function executeFunction(func){
2   func();
3 };
4
5 function sayHello(){
6   alert('hello');
7 };
8
9 executeFunction(/* what would you put here? */);

```

Q: Given the above code, how would you change the call to: “executeFunction” function, so that it will execute the “sayHello” function?

A:

```

1 function executeFunction(func){
2   func();
3 };
4
5 function sayHello(){
6   alert('hello');
7 };
8
9 executeFunction(sayHello);

```

Hint:

[javascript pass function as parameter – Stack Overflow](#)

```

1 function alertGreeting(msg){
2   alert(msg);
3 };
4
5 function getGreetingText(){
6   return 'hello';
7 };
8
9 alertGreeting( /* what would you put here? */ );

```

Q: Given the above code, how would you change the call to the: “alertGreeting” function, so that it will alert the return-value of the “getGreetingText” function ?

A:

```

1 function alertGreeting(msg){
2   alert(msg);
3 };
4
5 function getGreetingText(){
6   return 'hello';
7 };
8
9 alertGreeting( getGreetingText() );

```

Hint:

[Functional Programming — Eloquent JavaScript](#)

```

1 function isFunction(func){
2   /* what would you put here? */
3 };

```



```

4
5 // For Example:
6
7 console.log( isFunction() )//should return: false
8 console.log( isFunction("hello") )//should return: false
9 console.log( isFunction(123) )//should return: false
10 console.log( isFunction(['a','b','c']) )//should return: false
11
12 console.log( isFunction(isFunction) )//should return: true
13 console.log( isFunction(function(){})) //should return: true
14 console.log( isFunction(document.getElementById) )//should return: true
15 console.log( isFunction( new Function() ) )//should return: true

```

Q: Given the code above, how would you change the “isFunction” function so that when it is passed a function, it returns true, and if not, it returns false?

A:

```

1 function isFunction(func){
2     if (func && func instanceof Function){
3         return true
4     } else {
5         return false
6     }
7 };

```

Hint:

Use the JavaScript “instanceof” operator: [instanceof – JavaScript | MDN](#)

```

1 function sayHello(){
2     alert('hello');
3 };

```

Q: Given the code above, how would you change the “sayHello” function so that it executes immediately after it is defined? (i.e. there is no need to call it, it executes right-away)

A:

```

1 (function sayHello(){
2     alert('hello');
3 })();

```

Hint: You can turn that function into a self-executing or “immediate” function by wrapping it in parentheses, and adding a pair of parentheses after that: **(yourFunctionHere)()**.

[Immediate Functions in JavaScript – The Basics | Kevin Chisholm – Blog](#)

[Immediately-invoked function expression – Wikipedia, the free encyclopedia](#)

