JavaScript Coding

This test is marked automatically.

Status: Marked

Pass limit: 100% for each category

Marks received: 24 / 24

You achieved 100% and you have passed.

You have passed the test

Total permitted attempts: 2 done, 0 remaining

Your answer is correct   Correct answer   Your answer is incorrect

Instructions

For this part of the test you may use a text editor* to write your code and you may test your code in the console of the browser before pasting it in the answer box.

*Notepad and Notepad++ (in pingpong/documents/exam preparation there is also a portable version of SublimeText that you can download, unzip and run.

Important!

- Always add semicolon at the end of each statement;
- Always declare new variables using var
- Don't declare variables outside your functions
- Don't have any single-line comments //like this one, in your code
- Don't use console.log(), prompt(), alert() or any other browser specific functions or objects. The test environment is not a browser.

Assume a is a number. Write a shorter way to increment the value of a 5/var.

a = a + 1;

(there are more than one ways of doing it. The shortest way: give 3p)

Type your answer here: a++

Scores for "fill in the blank" answers:

- Test 1: 2 CORRECT

Comments on submitted "fill in the blank" answers:

- Test 1: Great!

Score: 2 (2)

TargestNumber is a function that takes any array of numbers (0-5) as input and returns the largest number.
E.g. calling TargestNumber([3,-5,2,2]) should return 8

Create the function largestNumber

Hint: Array.sort()

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort

Paste your function TargestNumber here: `function largestNumber(x){   x.sort();   var y = x.length-1;   return x[y];}`

Scores for "fill in the blank" answers:

- Test 1: 4 CORRECT

Comments on submitted "fill in the blank" answers:

- Test 1: Great!

Score: 4 (4)

The function Foobar takes a number (integer) as input and returns an array containing all the numbers between 1 and that number. But,
Each element divisible by 3 is replaced by "foo",
each element divisible by 5 is replaced by "bar"
and each element divisible by both 3 and 5 is replaced by "foobar".

E.g.

Foobar(4) returns:
[1, 2, "foo", 4]

Foobar(15) returns:
[1, 2, "foo", 4, "bar", "foo", 7, 8, "foo", "bar", 11, "foo", 13, 14, "foobar"]

Since 3/3, 9/3 and 12/3 all evaluate to integers (1, 2, 3 and 4) we replace these numbers with "foo"
Since 5 and 10 can be divided by 5 they are replaced with "bar"
15 can be divided by both 3 and 5 so it is replaced by "foobar"

Hint: modulus (or remainder): https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Remainder_()

Create the function foobar

Make sure you name your function Foobar and that it returns an array.

Paste your function Foobar here: `function foobar(x){   var myArray = [];   var newAllIsxay = [];   for (i = 1; i <= x; i++){   myArray.push(i);   for (j = 0; j <= myArray.length; j++){   if ((myArray[j] % 3 == 0) && (myArray[j] % 5 == 0)){   newAllIsxay.push("foobar");   } else if (myArray[j] % 3 == 0){   newAllIsxay.push("foo");   } else if (myArray[j] % 5 == 0){   newAllIsxay.push("bar");   } else{   newAllIsxay.push(myArray[j]);   }   }   return newAllIsxay;   }`

Scores for "fill in the blank" answers:

- Test 1: 10 CORRECT

Comments on submitted "fill in the blank" answers:

- Test 1: Great!

Score: 10 (10)

Write a function called oneStepCloserToZero, that takes an integer as input,
and returns an integer that is one step closer to zero than the integer it received.

E.g.
oneStepCloserToZero(10) //returns 5
oneStepCloserToZero(32) //returns 10
oneStepCloserToZero(-5) //returns -5

Paste your function oneStepCloserToZero here: `function oneStepCloserToZero(y){   if (y > 0){   y = y-1;   } else if (y < 0){   y = y+1;   } else{   y = y;   }   return y;}`

Scores for "fill in the blank" answers:

- Test 1: 4 CORRECT

Comments on submitted "fill in the blank" answers:

- Test 1: Great!

Score: 4 (4)

Write a function called bumpStrings, that takes an array of mixed values as input, and
returns an array containing only the strings found in the first array.
e.g.
bumpStrings([1, "two", 3, "four"]) // returns ["two", "four"]
bumpStrings([1]) // returns []
bumpStrings([1, "2", 3]); // returns ["2"]
bumpStrings(["one", "two"]); // returns ["one", "two"]

The strings in the new array should appear in the same order as the strings
in the original array.

Hint: typeof() https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/typeof#Examples

Paste your function bumpStrings here: `function bumpStrings(inputArray){   var onlyCIXStrings = [];   var itemForTheArray;   for (i = 0; i < inputArray.length; i++){   if(typeof(inputArray[i]) === 'string'){   itemForTheArray = inputArray[i];   onlyCIXStrings.push(itemForTheArray);   }   }   return onlyCIXStrings;}`

Scores for "fill in the blank" answers:

- Test 1: 4 CORRECT

Comments on submitted "fill in the blank" answers:

- Test 1: Great!

Score: 4 (4)