JS Quiz

Total points15/65

Email\*

capson.f@sankeysolutions.com

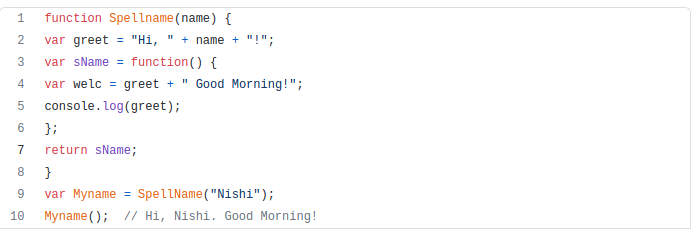
Induction Batch\*

Enter your Induction program batch number, e.g. B10, B11

B4

Which function represents closure?\*

0/5



Myname

None of these

Spellname

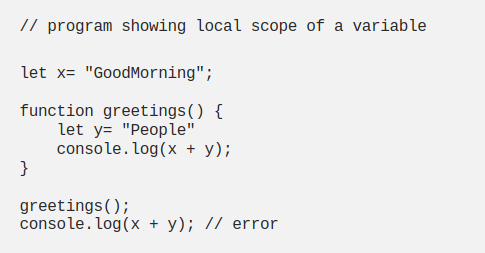
sName

Correct answer

sName

Which of the flg is true?\*

0/5



y is a local variable & x is a global variable

x & y are both local variables

x is a local variable & y is a global variable

x & y are both global variables

Correct answer

x is a local variable & y is a global variable

What is wrong in the code shown in above question\*

···/5

The scoping of the variable is creating a conflict in the above code, where the y variable is a local variable which means the variable is initialized or created inside the function so it cannot be accessed outside the function and whereas x is the global variable which can be accessed outside the function.

Give below an example of Hoisting in JS\*

···/5

Variable hoisting:- console.log(num); //undefined var num = 6; //initialization and declaration console.log(num); //6 Function hoisting:- catName("Billa"); function catName(name) { console.log(`My cat's name is ${name}`); //my cat's name is Billa } Variable hosting using let and const:- console.log(num); //Reference error let num = 6; console.log(num); console.log(num); //Reference error const num = 6; console.log(num);

Give below an example of destructuring\*

···/5

Destructuring basically makes it easier to assign values from an array-like object to individual variables. const arr=[1,2,3,4,5]; const[a,b,c,d,e]=arr console.log(a,b,c,d,e); OUTPUT: 1 2 3 4 5

let x = [1, 2, 3];                                                                                                                            print(...x)\*

What will be the output

0/5

1, 2, 3

2

3

Invalid Syntax

Correct answer

1, 2, 3

Array.map() updates the array provided in the argument\*

0/5

Statement is Invalid

Statement is False

Statement is True

Correct answer

Statement is False

console.log(100 === '100');\*

5/5

False

True

let day1 = {                                                                                    squirrel: false,                                                                                 events: ["work", "touched tree", "pizza","running"]                       };                                                                        console.log(day1.wolf);\*

5/5

false

work

undefined

""

let obj = {x: 0, y: 0, z: 2}                                                                       Write below the command which will give the following output ['x', 'y', 'z']\*

···/5

Object.keys(obj)

let kim = "Kim"                                                                                  kim.age = 88;                                                     console.log(kim.age);\*

5/5

88

undefined

"Kim"

Error

Print the following output without loops and using a single js function                                                                                  [1, 2, 3, 4, 5]\*

···/5

function execute(initial,final){ if(initial<=final){ console.log(initial); execute(initial+1,final); } } execute(1,100)

Explain your understanding of JS Prototypes in 10-15 words with an example\*

···/5

According to my understanding, Each and every object be it of type array or strings or object itself has properties of its own known as a prototype, from where it inherits its unique list of properties.