MSIMBO

JavaScript Practice

# Description

This is going to be a separated assignment helping you to learn and practice JasvaScript. Finish the following questions in assignment2.js.

# Questions

1. Write the function printInt(n) taking one parameter n and print all natural numbers from 1 to n in console.
2. Write the function printIntRev(n) taking one parameter n and print all natural numbers in reverse (from n to 1) in console.
3. Write the function checkInput(x) taking one parameter x and return the string ‘number’ if x is a number; return the string ‘string’ if x is a string; and return boolean if x is a boolean. Otherwise returns -1.

E.g. checkInput(5) => ‘number’

checkInput(‘hello’)=> ‘string’

checkInput(‘5’) => ‘string’

checkInput(true)=> boolean

checkInput(none) => -1

1. Write the function simpleEvenAdding(num) taking a number and add up all the even numbers from 1 to num.

Examples: simpleEvenAdding(5) => 6 (because 2+4 = 6)

simpleEvenAdding(10) => 30 (because 2+4+6+8+10 = 30)

simpleEvenAdding(11) => 6 (because 2+4+6+8+10 = 30)

simpleEvenAdding(1) => 0

simpleEvenAdding(0) => 0

1. Write the function letterCapitalize(str) taking a string and capitalizaed the first letter of each word. The given words will be separated by only one space.

Examples:

letterCapitalize(“hello world”) = “Hello Word”

letterCapitalize(“you cannot find the answer online”) = “You Cannot Find The Answer Online”

Hint: There is a built-in function in javaScript that change string into uppercase. There is also function to separate string into an array of characters.

1. Write the function simpleReverse(str) taking a string and return the string in reversed order.

Examples:

simpleReverse(‘hello’) => ‘olleh’

simpleReverse(‘I Love Code’) => ‘edoC evoL I’

Hint: Think of how you can loop through a string or array of characters backwards to produce a new string

1. Write the function findDiff(arr) taking a array of numbers as parameter and return the difference between the maximum number and the minimum number (max - min).

Examples:

findDiff([1,2,4,6,20, 3]) => 19 (Because 20 - 1 = 19)

findDiff([24, 22, 23, 22, 24]) => 2 (Because 24 - 22 = 2)

findDiff([1, 1, 1, 1, 1]) => 0

findDiff([1]) => 0;

findDiff([]) => 0;

1. Write the function timeConvert(num) taking a number as parameter and return the number of hours and minutes the parameter converts to. Seperate the number of hours and minutes with a colon.

Example:

timeConvert(61) => 1:1

timeConvert(63) => 1:3

timeConvert(120)=> 2:0

timeConvert(59)=> 0:59

Hints: Dividing and modulo the number 60.

1. Write the function findStr(str, long) taking two strings as parameters and return how many str you can find in long. Assume Str is not empty string.

Example:

findStr(“hi”, “dasdhidasdahidashi”) => 3

findStr(“o”, “daodo”) => 2

findStr(“ha”,”abcde”)=> 0

1. Write the function selfDividingNumbers(left, right) taking two number bound as parameters and returns an array of every possible self dividing number between

them, including the bounds.

Hint: A self-dividing number is a number that is divisible by every digit it contains. For example, 128 is a self-dividing number because 128 % 1 ==0, 128 % 2 == 0 and 128 % 8 == 0.

Examples:

selfDividingNumbers(1, 22) => [1,2,3,4,5,6,7,8,9,11,12,15,22]

selfDividingNumbers(1, 10) => [1,2,3,4,5,6,7,8,9]

selfDividingNumbers(12, 21) => [12, 15]

Extra Credit:

Write the function moveZeros(nums) taking an arry of numbers and move all 0’s to the end of it while maintaining the relative order of the non-zero elements. For example:

moveZeros([0,1,0,3,12]) => [1,3,12,0,0]

moveZeros([1,2,0,0,2,312,33,0,0]=>[1,2,2,312,33]

moveZeros([0,0,0])=>[0,0,0]

moveZeros([1,2,312,11,2]) => [1,2,312,11,2]

# Deliverables

Send your assignment2.js while email to [shiyu3169@gmail.com](mailto:shiyu3169@gmail.com)