East West University
Department of Computer Science and Engineering
Course: CSE479- Web Programming (Section 1)
Fall 2024

Lab Work -03

Objectives:

You will write and execute

- Code that uses the JavaScript String data type
- Code that uses the JavaScript Array data type
- Basic JavaScript instructions

Exercises:

Exercise 1: Check whether a string is blank

Write a JavaScript function ***isBlank(input)*** that checks whether the ***input*** String is blank. The function returns ***true*** if the ***input*** String is blank, ***false*** otherwise.

Test case:

1. **console.log(isBlank(""));**
   true
2. **console.log(isBlank("hello"));**
   false

Exercise 2: Split a string and convert it to an array

Write a JavaScript function ***stringToArray(input)*** to split a string and convert it into an array of words. The function should return the array of words.

Test case:

1. **console.log(*stringToArray*("Robin Singh"));**
   ["Robin", "Singh"]

Exercise 3: Convert a first and last name to abbreviated form

Write a JavaScript function ***abbreviateName(name)*** to convert a string (a person's first and last names) to abbreviated form. If there is only a first name listed, then return just that name.

Test case:

1. ***console.log(abbreviateName("Robin Singh"));***
   "Robin S."

Exercise 4: Convert a string to title case

Write a JavaScript function ***titleCase(input)*** to convert an input string to title case. Note that titleCase has the first letter of each word capitalized and every other letter lowercase. Note in the example below that "JavaScript" becomes "Javascript" for the final result.

Test case:

1. ***console.log(titleCase('JavaScript exercises. python exercises.'));***
   "Javascript Exercises. Python Exercises."

Exercise 5: Get the first n elements of an array

Write a JavaScript function ***firstN(array, n)*** to get the first ***n*** elements of an array.

- If n is not given, get the first element.
- If n < 0, return the empty array.
- If n > number of elements in the array, return the entire array

Test case:

1. ***console.log(firstN([7, 9, 0, -2]));***
   7
2. ***console.log(firstN([],3));***
   []
3. ***console.log(firstN([7, 9, 0, -2],3));***
   [7, 9, 0]
4. ***console.log(firstN([7, 9, 0, -2],6));***
   [7, 9, 0, -2]
5. ***console.log(firstN([7, 9, 0, -2],-3));***
   []

Exercise 6: Get the last n elements of an array

Write a JavaScript function ***lastN(array, n)*** to get the last **n** element of an array.

- If n is not given, return the last element.
- If n < 0, return the empty array.
- If n > number of elements in the array, return the entire array.

Test case:

1. ***console.log(lastN([7, 9, 0, -2]));***
   -2
2. ***console.log(lastN([7,9,0,-2],3));***
   [9, 0, -2]
3. ***console.log(lastN([7, 9, 0, -2],6));***
   [7, 9, 0, -2]

Exercise 7: Pair of consecutive elements that sum to target

Write a JavaScript function ***sumPair(numbers, target)*** to find the indices of a pair of consecutive elements from a given array whose sum equals a specific target number. The function should return an array of the indices of the pair of consecutive elements or the array [-1, -1] if a pair is not found.

Test case:

1. Input:
   const numbers= [10,20,10,40,50,60,70], target=50;
   console.log(***sumPair(numbers, target));***
   Output:
   [2, 3]

Exercise 8: Challenge, move an item from one position to another

Write a JavaScript function ***move(array, from, to)*** to move an array element from one position to another. The function should return the changed array.

Test case:

1. ***console.log(move([10, 20, 30, 40, 50], 0, 2));***
   [20, 30, 10, 40, 50]

2. ***console.log(move([10, 20, 30, 40, 50], -1, -2));***   ***// use the splice method for this to work or check if an index is < 0 and add array length to it.***
   [10, 20, 30, 50, 40]
