



EAST WEST UNIVERSITY

Lab-4

Submitted By:

Name: Yougshree Saha Urmy

Id: 2021-1-60-127

Section: 01

Course Code: CSE479

Course Name: Web Programming

Submitted To:

Nishat Tasnim Niloy

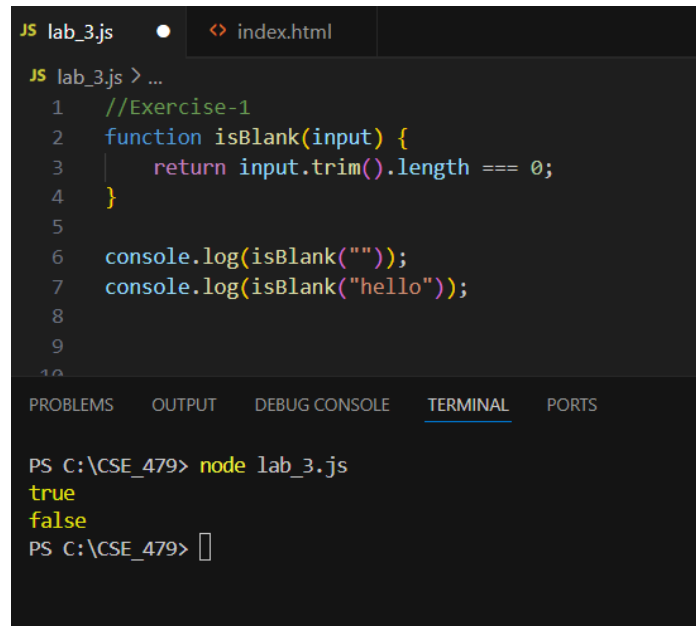
Lecturer

Department of Computer Science and Engineering

Date of submission: 5-12-2024

Exercise 1: Check whether or not a string is blank

Write a JavaScript function ***isBlank(input)*** that checks whether or not the ***input*** String is blank.

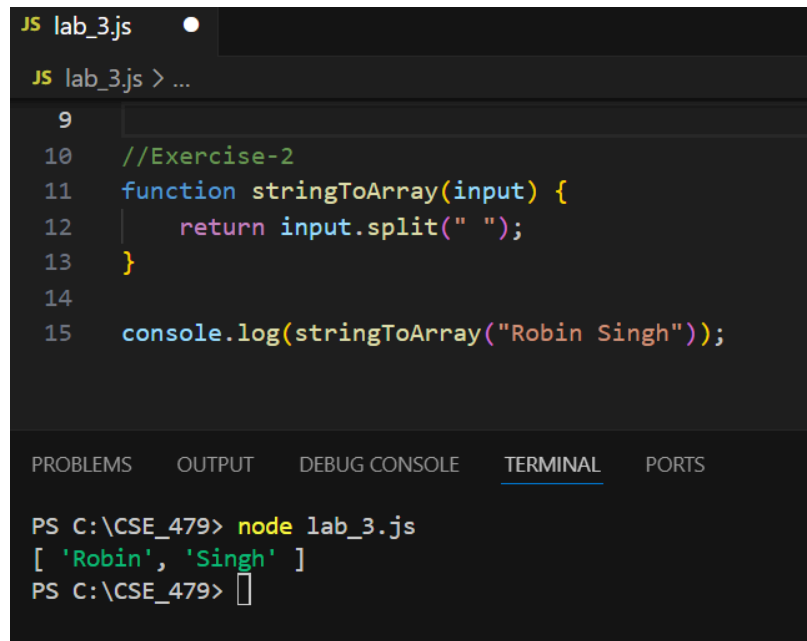


```
JS lab_3.js  ●  index.html
JS lab_3.js > ...
1  //Exercise-1
2  function isBlank(input) {
3      return input.trim().length === 0;
4  }
5
6  console.log(isBlank(""));
7  console.log(isBlank("hello"));
8
9
10
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\CSE_479> node lab_3.js
true
false
PS C:\CSE_479> 
```

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.



```
JS lab_3.js  ●
JS lab_3.js > ...
9
10 //Exercise-2
11 function stringToArray(input) {
12     return input.split(" ");
13 }
14
15 console.log(stringToArray("Robin Singh"));
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\CSE_479> node lab_3.js
[ 'Robin', 'Singh' ]
PS C:\CSE_479> 
```

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.

```
JS lab_3.js > abbreviateName

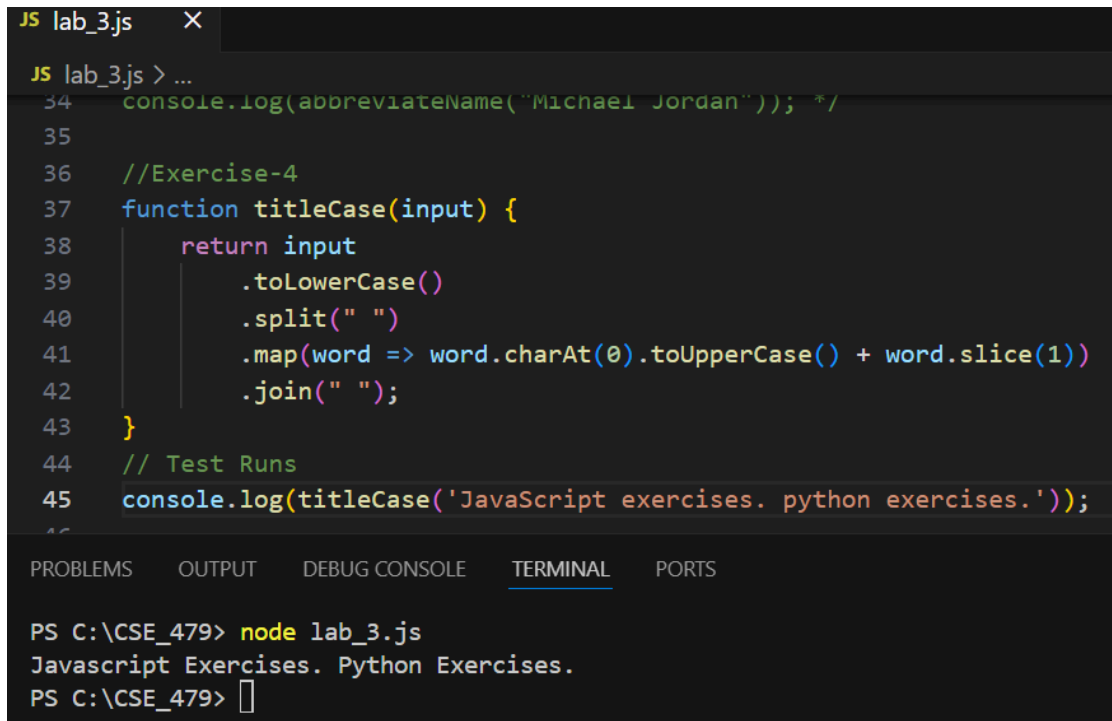
17 //Exercise-3
18 function abbreviateName(name) {
19 // Split the name by spaces
20     const nameParts = name.trim().split(" ");
21 // If there's only one part, return it
22     if (nameParts.length === 1)
23     {
24         return nameParts[0];
25     }
26 // Otherwise, abbreviate the last name
27     const firstName = nameParts[0];
28     const lastNameInitial = nameParts[1][0].toUpperCase();
29
30     return `${firstName} ${lastNameInitial}`;
31 }
32 // Test Runs
33 console.log(abbreviateName("Robin Singh"));
34 console.log(abbreviateName("Michael Jordan"));
35
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Robin S.
Michael J.
PS C:\CSE_479>

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.



```
JS lab_3.js X
JS lab_3.js > ...
34 console.log(abbreviateName("Michael Jordan")); */
35
36 //Exercise-4
37 function titleCase(input) {
38     return input
39         .toLowerCase()
40         .split(" ")
41         .map(word => word.charAt(0).toUpperCase() + word.slice(1))
42         .join(" ");
43 }
44 // Test Runs
45 console.log(titleCase('JavaScript exercises. python exercises.'));
46
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\CSE_479> node lab_3.js
Javascript Exercises. Python Exercises.
PS C:\CSE_479> 
```

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

```
JS lab_3.js
JS lab_3.js > ...
47 //Exercise-5
48 function firstN(array, n= 1)
49 {
50     if (n< 0)
51     {
52         return [];
53     }
54
55     if (n> array.length)
56     {
57         return array;
58     }
59
60     return array.slice(0, n);
61 }
62
63 // Test Runs
64 console.log(firstN([7, 9, 0, -2]));
65 console.log(firstN([], 3));
66 console.log(firstN([7, 9, 0, -2], 3));
67 console.log(firstN([7, 9, 0, -2], 6));
68 console.log(firstN([7, 9, 0, -2], -3));
69
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\CSE_479> node lab_3.js
[ 7 ]
[]
[ 7, 9, 0 ]
[ 7, 9, 0, -2 ]
[]
PS C:\CSE_479> 
```

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.

```
JS lab_3.js > ...
68  */
69  //Excercise-6
70
71  function lastN(array, n) {
72      if (n === undefined) {
73          return array[array.length - 1];
74      }
75
76      if (n < 0) {
77          return [];
78      }
79      if (n > array.length) {
80          return array;
81      }
82
83      return array.slice(-n);
84  }
85
86  // Test Runs
87  console.log(lastN([7, 9, 0, -2]));
88  console.log(lastN([7, 9, 0, -2], 3));
89  console.log(lastN([7, 9, 0, -2], 6));
90
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\CSE_479> node lab_3.js
-2
[ 9, 0, -2 ]
[ 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.

```
70 //Excercise-7
71
72 function sumPair(numbers, target) {
73     for (let i = 0; i < numbers.length - 1; i++) {
74         if (numbers[i] + numbers[i + 1] === target) {
75             return [i, i + 1];
76         }
77     }
78     return [-1, -1];
79 }
80
81 // Test Run
82 const numbers = [10, 20, 10, 40, 50, 60, 70];
83 const target = 50;
84 console.log(sumPair(numbers, target));
85
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\CSE_479> node lab_3.js

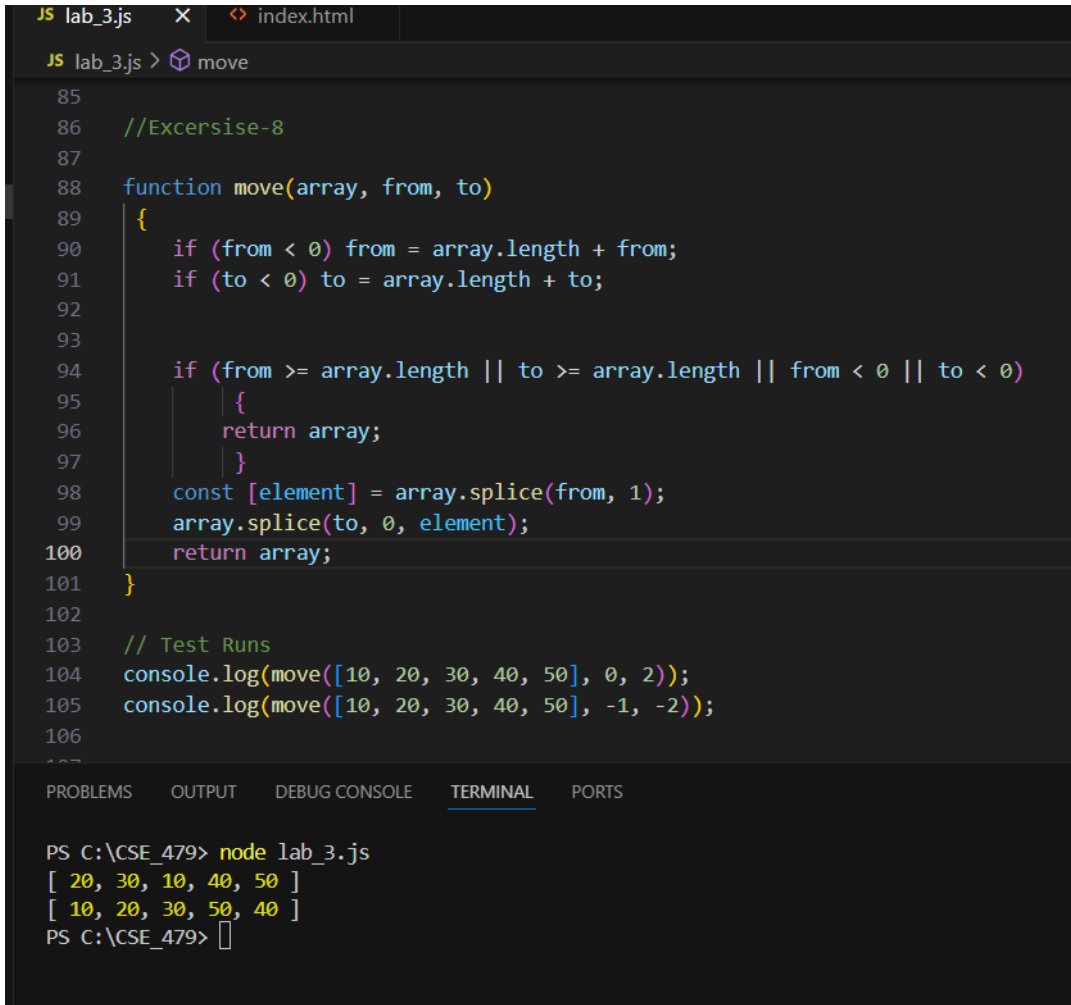
[2, 3]

PS C:\CSE_479>

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.



```
JS lab_3.js X index.html
JS lab_3.js > move

85
86 //Excercise-8
87
88 function move(array, from, to)
89 {
90     if (from < 0) from = array.length + from;
91     if (to < 0) to = array.length + to;
92
93
94     if (from >= array.length || to >= array.length || from < 0 || to < 0)
95     {
96         return array;
97     }
98     const [element] = array.splice(from, 1);
99     array.splice(to, 0, element);
100     return array;
101 }
102
103 // Test Runs
104 console.log(move([10, 20, 30, 40, 50], 0, 2));
105 console.log(move([10, 20, 30, 40, 50], -1, -2));
106
107
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\CSE_479> node lab_3.js
[ 20, 30, 10, 40, 50 ]
[ 10, 20, 30, 50, 40 ]
PS C:\CSE_479> 
```