BCVD1007 – Full Stack Development II – Lab 3

- JavaScript Arrays

Developer Note:

- Please create a separate JavaScript file for each exercise
- You may use the HTML page to trigger your scripts or you may use the JS Playground Editors (REPL) to program and just submit the JS file.
    - https://repl.it/languages/javascript
    - https://jsfiddle.net/

Exercise 1:

Create a function named **buildArray** that has the following requirements:

- Takes a parameter ***num***
- Create a local variable named ***myArray***
- Write a loop that will continue to iterate until the counter is greater than num.
- For each iteration, it add to your counter value to your variable ***myArray***
- Before the function exists it will return the value of ***myArray***

Expected output is as follows:

```
buildArray(2);
```

```
[ 0, 1 ]
```

```
buildArray(25);
```

```
[
   0,  1,  2,  3,  4,  5,  6,  7,
   8,  9, 10, 11, 12, 13, 14, 15,
  16, 17, 18, 19, 20, 21, 22, 23,
  24
]
```

Exercise 2:

Create a function named **emptyArray** that has the following requirements:

- Takes a parameter ***myArray*** which will always be an array
- Write a loop that will continue to iterate until the array is empty
- For each iteration, remove an element from the array
- Before the function exists it will return the value of ***myArray***

Expected output is as follows:

```
emptyArray([1,2,3])

emptyArray(["egg","bacon","toast","coffee","random","juice"]);
```

```
[]
```

Exercise 3:

Create a function named **reverseIt** that has the following requirements:

- Takes a parameter ***myArr*** which will always be an array
- Will **return** an array in reverse order

Expected output is as follows:

```
reverseIt([1, 2, 3, 4]) // [4, 3, 2, 1]

reverseIt([9, 9, 2, 3, 4]) // [4, 3, 2, 9, 9]

reverseIt([]) // []
```

Exercise 4:

Create a function named **findArrIndex** that has the following requirements:

- Takes a parameter ***myArr*** which will always be an array
- Takes a parameter ***str*** which always be a string
- Will the position of the string in the array
    - Otherwise -1 for not found

Expected output is as follows:

```
findArrIndex(["abc", "edabit", "yyy", "xxx"], "yyy") // 2

findArrIndex(["a", "g", "y", "z"], "z") // 3

findArrIndex(["egg","bacon","toast","coffee","random","juice"], "coffee") // 3

findArrIndex(["egg","bacon","toast","coffee","random","juice"], "mcmuffin") //-1
```

Exercise 5:

Create a function named **hasNumber** that has the following requirements:

- Takes a parameter ***myArr*** which will always be an array
- Takes a parameter ***num*** which always be a number
- Returns true if the number exists in the array
- Returns false if the number does not exist in array

Expected output is as follows:

```
hasNumber([1, 2, 3, 4, 5], 3) // true

hasNumber([1, 1, 2, 1, 1], 3) // false

hasNumber([5, 5, 5, 6], 5) // true

hasNumber([], 5) // false
```

Challenge:

Create a function named **speakHacker** that has the following requirements:

- Takes a parameter ***str*** which always be a string
- Replace all "a"s with 4, "e"s with 3, "i"s with 1, "o"s with 0, and "s"s with 5.

Expected output is as follows:

```
speakHacker("Long Weekend!") // L0ng W33k3nd!

speakHacker("matrix 4 is coming out") //m4tr1x 4 15 c0m1ng 0ut

speakHacker("marvel movies are delayed") //m4rv3l m0v135 4r3 d3l4y3d
```