Coding Question

// Coding Question With Code and Output

How do you find the missing number in a given integer array of 1 to 100?

```
var a = [1,2,4,6,8,9],
  count = 10;
var missing = new Array();

for (var i = 1; i <= count; i++) {
  if (a.indexOf(i) == -1) {
    missing.push(i);
  }
}
console.log(missing);
```

```
output = [3, 5, 7, 10]
```

How do you find the duplicate number on a given integer array?

```
const yourArray = [1, 1, 2, 3, 4, 5, 5]

let duplicates = []

const tempArray = [...yourArray].sort()

for (let i = 0; i < tempArray.length; i++) {
  if (tempArray[i + 1] === tempArray[i]) {
    duplicates.push(tempArray[i])
  }
}

console.log(duplicates) //[ 1, 5 ]
```

```
output =[ 1, 5 ]
```

How do you find the largest and smallest number in an unsorted integer array?

```
const arr = [112, 24, 31, 44, 101, 203, 33, 56];
const findMaxMin = (arr) => {
  let max = arr[0];
  let min = arr[0];
  for(let i = 0; i < arr.length; i++) {
    if(arr[i] > max) {
      max = arr[i];
    }
    else if (arr[i] < min) {
      min = arr[i];
    }
  };
  return {
    min, max
  };
};
console.log(findMaxMin(arr));

output = { min: 24, max: 203 }
```

How do you find all pairs of an integer array whose sum is equal to a given number?

```
function twoSum(nums, target_num) {
    var map = [];
    var indexnum = [];

    for (var x = 0; x < nums.length; x++)
    {
        if (map[nums[x]] != null)
        // what they meant by map[nums[x]]
        {
            index = map[nums[x]];
            indexnum[0] = index+1;
            indexnum[1] = x+1;
            break;
        }
        else
        {
            map[target_num - nums[x]] = x;
        }
    }
    return indexnum;
    }
console.log(twoSum([10,20,10,40,50,60,70],50));

Output = [ 3, 4 ]
```

How do you find duplicate numbers in an array if it contains multiple duplicates?

```
const numbers = [1, 2, 3, 2, 4, 5, 5, 6];

const unique = Array.from(new Set(numbers));

if(numbers.length === unique.length) {
    console.log(`Array doesn't contain duplicates.`);
} else {
    console.log(`Array contains duplicates.`);
}
Output =  Array contains duplicates.
```

How do you reverse an array in place in JavaScript? In place means you cannot create a new array. You have to update the original array.

```
let yourArray = [1,2,3,4]
let reverseArray = yourArray.slice().reverse()

console.log(reverseArray)

output = [ 4, 3, 2, 1 ]
```

How do you print duplicate characters from a string?

```
function removeDuplicateChar(string) {
  return string
    .split(")
    .filter(function(item, pos, self) {
      return self.indexOf(item) == pos;
    })
    .join(");
}
console.log(removeDuplicateChar("banana"));

output = ban
```

How do you check if two strings are anagrams of each other?

```
var str1,str2
str1='LISTEN'
str2='SILENT'
console.log( (str1.split("").sort().join(""))===(str2.split("").sort().join("")) )

output = true
```

How do you print the first non-repeated character from a string?

```
function firstNonRepeatedCharacter(string) {
    return string.split(").filter(function (character, index, obj) {
        return obj.indexOf(character) === obj.lastIndexOf(character);
    }).shift();
}

console.log(firstNonRepeatedCharacter('aabcbd'));
```

```
output = c
```

How can a given string be reversed using recursion?

```
function reverseString(str) {
  if (str === "") // This is the terminal case that will end the recursion
    return "";

  else
    return reverseString(str.substr(1)) + str.charAt(0);
}
console.log(reverseString("hello"));
```

```
Output = olleh
```

How do you check if a string contains only digits

```
function checkIfStringHasOnlyDigits(_string)
{
    if(_string.match(/^[0-9]+$/) != null)
    {
        console.log("String contains only numbers")
    }
    else
    {
        console.log("String does not contain only numbers")
    }
}
checkIfStringHasOnlyDigits("123ThisPointer.com")
checkIfStringHasOnlyDigits("8965")
checkIfStringHasOnlyDigits("89.65")
checkIfStringHasOnlyDigits("")
```

```
output =
String does not contain only numbers
String contains only numbers
String does not contain only numbers
String does not contain only numbers
```

How are duplicate characters found in a string?

```
const getRepeatedChars = (str) => {
    let result = [];
    str.map(each => {
      let repeatedChars = 0;
      for (let i = 0; i < each.length - 1; i++) {
          if (each[i] === each[i + 1] && each[i] !== each[i - 1]) {
            repeatedChars += 1;
          }
      }

        result.push(repeatedChars);
    });

    return result;
};

console.log(getRepeatedChars(["aaabbbkdnndicccoekdczufnrz", "awsfgds"]));

Output =  [4, 0]
```

How do you count a number of vowels and consonants in a given string?

```
// program to count the number of vowels in a string


let vowels = ["a", "e", "i", "o", "u"]

function countVowel(str) {

    let count = 0;


    for (let letter of str.toLowerCase()) {
        if (vowels.includes(letter)) {
            count++;
        }
    }

    return count
}


let string = prompt('Enter a string: ');

const result = countVowel(string);

console.log(result);

output = prompt[ a,e,I,o,u] = 5
prompt[svgfh] = 0
```

How do you count the occurrence of a given character in a string?

```
var str = "A,B,C,D,E,d,";
var ch = ',';

var count = str.split(ch).length - 1;
console.log(count);
```

```
Output = 6
```

How do you find all permutations of a string ?

```
let stringPermutations = function (str) {
  if (str.length <= 2) return str.length === 2 ? [str, str[1] + str[0]] : [str];
  return str
    .split('')
    .reduce(
      (acc, letter, i) =>
        acc.concat(stringPermutations(str.slice(0, i) + str.slice(i + 1)).map(val => letter +
val)),
      []
    );
};

console.log(stringPermutations("abc"));
console.log(stringPermutations("dp"));
```

```
output =
[ 'abc', 'acb', 'bac', 'bca', 'cab', 'cba' ]
[ 'dp', 'pd' ]
```

How do you reverse words in a given sentence without using any in-built method?

```
function reverse(str,start,end)
    {
        let temp;
        while (start <= end)
        {
            temp = str[start];
            str[start]=str[end];
            str[end]=temp;
            start++;
            end--;
        }
    }
 function reverseWords(s)
    {
        s=s.split("");
        let start = 0;
        for (let end = 0; end < s.length; end++)
        {
            if (s[end] == ' ')
            {
                reverse(s, start, end);
                start = end + 1;
            }
        }
        reverse(s, start, s.length - 1);
        reverse(s, 0, s.length - 1);
        return s.join("");
    }
var s = "i love  gaming";
document.write(reverseWords(s));
```

output = gaming love i

How do you check if two strings are a rotation of each other?

```
function RotEq (str1, str2) {
    if (str1 === str2) return true;
    if (str1.length !== str2.length) return false;

    var start2 = str2.indexOf(str1[0]);
    if (start2 === -1) return false;

    return str1 === str2.slice(start2) + str2.slice(0, start2)
}
console.log(RotEq("abcd", "abcd"))
```

```
let j = RotEq("abcd", "acdb")
console.log(j)
```

```
output =
true
false
```

How do you check if a given string is a palindrome ?

```
function checkPalindrome(str) {
    const arrayValues = string.split('');
    const reverseArrayValues = arrayValues.reverse();
    const reverseString = reverseArrayValues.join('');
    if(string == reverseString) {
        console.log('It is a palindrome');
    }
    else {
        console.log('It is not a palindrome');
    }
}
const string = prompt('Enter a string : ');
checkPalindrome(string);
```

```
Output  =  Enter a string : abc
It is not a palindrome

Enter a string : aba
It is a palindrome
```