

// Problems on array: January 28 2020

// Q1. Write a JavaScript function to remove 'null', '0', '', 'false', 'undefined' and 'NaN' values from an array.

// Sample array: [NaN, 0, 15, false, -22, "", undefined, 47, null]

// solutions ///

```
let array = [NaN, 0, 15, false, -22, "", undefined, 47, null];
a = array.filter(Boolean);
console.log(a);
```

// Q2. Write a JavaScript program to flatten a nested (any depth) array into single level.

// Sample array:

// a) [1, [2], [3, [[4]]], [5, 6]]

// Expected output: [1, 2, 3, 4, 5, 6]

// solution 1 :

```
let a = [1, [2], [3, [[4]]], [5, 6]];
a = a
  .toString()
  .replace(/\]/, "")
  .split(",");
a = a.map(Number);
```

// solution 2 :

```
a = a.flat(Infinity);
```

// Q3. There are two arrays with individual values, write a JavaScript program to compute the sum of respective index value from the given arrays.

// Sample array:

// a ) array1 = [1, 0, 2, 3, 4];

// array2 = [3, 5, 6, 7, 8, 13];

// Expected Output:

// [4, 5, 8, 10, 12, 13]

// basic common code solution :

```
let array1 = [1, 0, 2, 3, 4];
```

```
let array2 = [3, 5, 6, 7, 8, 13];
```

```
let inner = [],
```

```
    outer = [];
```

```
if (array1.length > array2.length) {
```

```

    inner = array2;
    outer = array1;
} else {
    inner = array1;
    outer = array2;
}
/////      option 1:  /////

```

```

let sum = [],
    tryif = 0;
for (let i = 0; i < outer.length; i++) {
    if (typeof inner[i] === "undefined") tryit = 0;
    else tryit = inner[i];

    sum.push(outer[i] + tryit);
}
console.log(sum);

```

```

////      option 2:  /////

```

```

let temp = outer.map((el, i) => {
    let tryit = 0;
    if (typeof inner[i] === "undefined") tryit = 0;
    else tryit = inner[i];

    return el + tryit;
});

console.log(temp);

```

```

// b )
// array1 = [4, [31], [21, 32, 45], [3], 4, 78, 89];
// array2 = [6, 19, 50, 12, 34, 56, 78, 90];

```

```

// to convert array at level 1
var array1 = [4, [31], [21, 32, 45], [3], 4, 78, 89];
var array2 = [6, 19, 50, 12, 34, 56, 78, 90];
array1 = array1.flat();
console.log(array1);
let inner = [],
    outer = [];
// to decide which array is bigger
if (array1.length > array2.length) {

```

```

    inner = array2;
    outer = array1;
} else {
    inner = array1;
    outer = array2;
}

let temp = outer.map((el, i) => {
    let tryit = 0;
    if (typeof inner[i] === "undefined") tryit = 0;
    else tryit = inner[i];

    return el + tryit;
});

console.log(temp);

```

// Q4. There are two arrays with individual values, write a JavaScript program to get the difference values between these two arrays.

// Sample array:

```

// a) var array1 = [1, 2, 3], var array2 = [100, 2, 1, 10]
// Expected output: ["3", "10", "100"]

```

// solution 1:

```

var array1 = [1, 2, 3];
array1 = array1.toString().split(",");
var array2 = [100, 2, 1, 10];
array2 = array2.toString().split(",");
var temp = [];

array1.forEach(el => {
    if (!array2.includes(el)) temp.push(el);
    else temp.push(null);
});

array2.forEach(el => {
    if (!array1.includes(el)) temp.push(el);
    else temp.push(null);
});

```

```
temp = temp.filter(Boolean).sort((a, b) => a - b);
```

```
/// solution 2: /////
```

```
var array1 = [1, 2, 3];
```

```
var array2 = [100, 2, 1, 10];
```

```
var inner = [],
```

```
    outer = [];
```

```
var outArr1 = array1.filter(x => !array2.includes(x));
```

```
var outArr2 = array2.filter(x => !array1.includes(x));
```

```
var finalArray = outArr1.concat(outArr2).sort((a, b) => a - b);
```

```
console.log(finalArray);
```

```
/// better code ///
```

```
var array1 = [1, 2, 3];
```

```
var array2 = [100, 2, 1, 10];
```

```
var inner = [],
```

```
    outer = [];
```

```
var finalArray = array1
```

```
    .filter(x => !array2.includes(x))
```

```
    .concat(array2.filter(x => !array1.includes(x)))
```

```
    .sort((a, b) => a - b);
```

```
console.log(finalArray);
```