MUHAMMAD SHAHRIYAR

***ASSIGNMENT ARRAY METHODS***

***QUESTION NO 01:***

Let’s try 5 array operations.

1. Create an array styles with items “Jazz” and “Blues”.

2 .Append “Rock-n-Roll” to the end.

3. Replace the value in the middle with “Classics”. Your code for finding the 4.

middle value should work for any arrays with odd length.

4. Strip off the first value of the array and show it.

5. Prepend Rap and Reggae to the array.

*ANSWER:*

 // 1. make array

let styles=['jazz','blues'];

console.log(styles)

// 2. Append “Rock-n-Roll” to the end.

styles.push('ROck-n-Roll');

console.log(styles);

// 3. Replace the value in the middle with “Classics”.

styles.splice((styles.length/2),1,'classics')

    console.log(styles);

// 4. Strip off the first value of the array and show it.

   const first\_value=styles.shift();

  console.log(first\_value);

// 5.Prepend Rap and Reggae to the array.

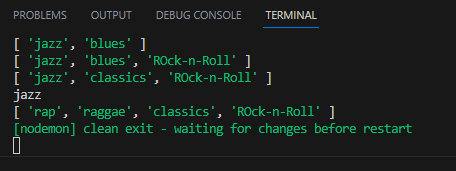
  //  styles.splice(0,0,'rap','reggae')

 //    console.log(styles)

    styles.unshift('rap','raggae');

    console.log(styles);

*OUTPUT:*

**

*QUESTION NO 02:*

Write the function sumInput() that:

Calculates and returns the sum of array items.

*ANSWER:*

    const arr=[2,3,4,5];

function sumInput(a){

    let sum=0;

    for(let i=0;i<a.length;i++)

    {

        sum+=a[i];

    }

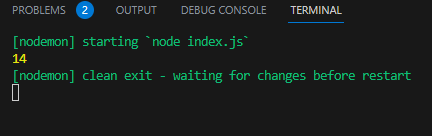
   return sum;

}

let result = sumInput(arr);

console.log(result);

*OUTPUT:*

**

*QUESTION NO 03:*

Write a function filterRange(arr, a, b) that gets an array arr, looks for elements with values higher or equal to a and lower or equal to b and return a result as an array.

The function should not modify the array. It should return the new array.

For instance:

let arr = [5, 3, 8, 1];

let filtered = filterRange(arr, 1, 4);

console.log( filtered ); // 3,1 (matching values)

console.log( arr ); // 5,3,8,1 (not modified)

*ANSWER:*

let arr1=[5,3,8,1]

function filterRange(arr1,a,b) {

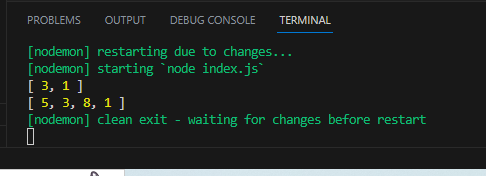
  return arr1.filter((x)=>x>=a&&x<=b)

}

console.log(filterRange(arr1,1,4));

console.log(arr1);

*OUTPUT:*



*QUESTION NO 04:*

Write a function filterRangeInPlace(arr, a, b) that gets an array arr and removes from it all values except those

 that are between a and b. The test is: a ≤ arr[i] ≤ b.

The function should only modify the array. It should not return anything.

For instance:

let arr = [5, 3, 8, 1];

filterRangeInPlace(arr, 1, 4); // removed the numbers except from 1 to 4

console.log( arr ); // [3, 1]

*ANSWER:*

  let arr1=[5,3,8,1]

function filterRangeInPlace(arr1, a, b) {

    let l=arr1.length;

    for(let i=0;i<l;i++)

      if(!(arr1[i]>=a && arr1[i]<=b)){

        arr1.splice(i,1);

        i--;

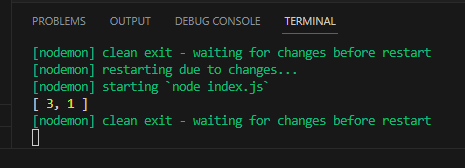
        l--;

      }

  }

  filterRangeInPlace(arr1,1,4)

  console.log(arr1);

*OUTPUT: *

*QUESTION NO 05:*

You have an array of user objects, each one has user.name. Write the code that converts it into an array of names.

For instance:

let john = { name: "John", age: 25 };

let pete = { name: "Pete", age: 30 };

let mary = { name: "Mary", age: 28 };

let users = [ john, pete, mary ];

let names =  ... your code

console.log( names ); // John, Pete, Mary

*ANSWER:*

let john = { name: "John", age: 25 };

let pete = { name: "Pete", age: 30 };

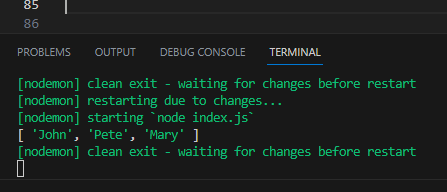
let mary = { name: "Mary", age: 29 };

let users = [ john, pete, mary ];

let name=users.map((x)=>x.name);

console.log(name);

*OUTPUT:*

**

*QUESTION NO 06:*

You have an array of user objects, each one has name, surname and id.

Write the code to create another array from it, of objects with id and fullName, where fullName is generated from name and surname.

For instance:

let john = { name: "John", surname: "Smith", id: 1 };

let pete = { name: "Pete", surname: "Hunt", id: 2 };

let mary = { name: "Mary", surname: "Key", id: 3 };

let users = [ john, pete, mary ];

let usersMapped =  ... your code ...

usersMapped = [

  { fullName: "John Smith", id: 1 },

  { fullName: "Pete Hunt", id: 2 },

  { fullName: "Mary Key", id: 3 }

]

console.log( usersMapped[0].id ) // 1

console.log( usersMapped[0].fullName ) // John Smith

So, actually you need to map one array of objects to another. Try using => here. There’s a small catch.

\*/

*ANSWER:*

let johny = { name: "John", surname: "Smith", id: 1 };

let petee = { name: "Pete", surname: "Hunt", id: 2 };

let maryy = { name: "Mary", surname: "Key", id: 3 };

let users2 = [ johny, petee, maryy ];

let usersMapped =users2.map((x)=>{

  let a={};

  a.fullname=x.name+x.surname;

  a.id=x.id;

  return a;

})

console.log(usersMapped);

console.log( usersMapped[0].id);

console.log( usersMapped[0].fullname );

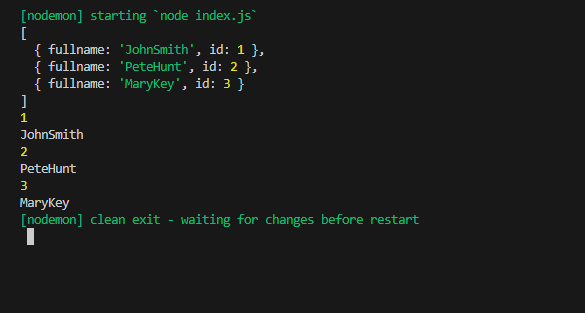
console.log( usersMapped[1].id);

console.log( usersMapped[1].fullname );

console.log( usersMapped[2].id);

console.log( usersMapped[2].fullname );

*OUTPUT:*

**

*Question no 07:*

// Write the function getAverageAge(users) that gets an array of objects with property age and returns the average age.

// The formula for the average is (age1 + age2 + ... + ageN) / N.

// For instance:

// let john = { name: "John", age: 25 };

// let pete = { name: "Pete", age: 30 };

// let mary = { name: "Mary", age: 29 };

// let arr = [ john, pete, mary ];

// console.log( getAverageAge(arr) ); // (25 + 30 + 29) / 3 = 28

*ANSWER:*

let john = { name: "John", age: 25 };

let pete = { name: "Pete", age: 30 };

let mary = { name: "Mary", age: 29 };

let users = [ john, pete, mary ];

function getAverageAge(users) {

    let age=users.map((user)=>user.age);

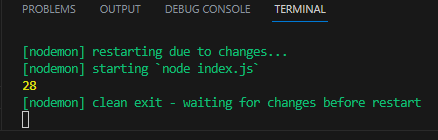
    let avgAge=age.reduce((a,b)=>a+b);

    return avgAge/age.length;

  }

  console.log(getAverageAge(users));

*OUTPUT:*

**

*Question no 08:*

Let arr be an array.

Create a function unique(arr) that should return an array with unique items of arr.

For instance:

function unique(arr) {

  your code

}

let strings = ["Hare", "Krishna", "Hare", "Krishna",

  "Krishna", "Krishna", "Hare", "Hare", ":-O"

];

alert( unique(strings) );

*ANSWER:*

function unique(a){

    let l=a.length;

    for (let i = 0; i < l; i++)

      for(let j = i+1; j < l; j++)

        if(a[i]===a[j]){

          a.splice(j,1);

          j--;

          l--;

        }

  }

  let strings = ["Hare", "Krishna", "Hare", "Krishna",

    "Krishna", "Krishna", "Hare", "Hare", ":-O"

  ];

  unique(strings)

  console.log(strings);

*OUTPUT:*

