Q) find triplet from an array to get sum of target number
let arr =[9,1,2,3,8,4,5,6]; let target = 11;

Q) let students = [{ name: "ABC", gender: "M" }, { name: "1232", gender: "F" }, { name: "ASDF", gender: "M" }, { name: "QWER", gender: "F" }, { name: "ZXCV", gender: "F" }, { name: "ASXCV", gender: "M" }]

output -

{ M: [array of students whose gender is M], F: [whose gender is F] }

```
convertToObject(students,"gender")
function convertToObject( arr, keyname ){
}
```

Q) Remove duplicates

```
let chars = ['A', 'B', 'A', 'C', 'B']; // const arr = [1,1,1,2,1,3,3,4,4,4,4,5]
let result=[];
for(let i=0;i<chars.length;i++){
  if(!result.includes(chars[i])){
    result.push(chars[i])
  }
}
console.log(result)
```

Or

```
[...new Set(chars)]
```

Or

```
const arr = [1,1,1,2,1,3,3,4,4,4,4,5]
let result = {};
for(let i=0; i<arr.length; i++){
  if(arr.indexOf(arr[i]) == i ){
    result[arr[i]] = 1
  }
  else{
    result[arr[i]] +=1
  }
}
console.log(result)
```

Q) Concat 2 objects

```
let obj1 = {name:'abc'};
let obj2 = {age:20};

let res= {...obj1, ...obj2}
console.log(res)
```

Q) Remove that object from arrOfObj which have volume that is present in arr array .

```
let arrOfOjb = [
  { name: 'a', volume: '3', subject: 'abc' },
  { name: 'b', volume: '8', subject: 'cde' },
  { name: 'c', volume: '6', subject: 'abc' },
  { name: 'd', volume: '2', subject: 'xyz' },
  { name: 'e', volume: '1', subject: 'xyz' },
  { name: 'f', volume: '5', subject: 'abc' },
  { name: 'g', volume: '9', subject: 'opq' }
];
let arr = [2, 3, 1, 8];

arrOfOjb= arrOfOjb.filter(obj=> !arr.includes(parseInt(obj.volume)))
console.log(arrOfOjb);
```

Q) Print Fibonacci series upto n = 10

```
let n=10;
let result=[];
if(n>=0){
  result.push(0)
}
if(n>=1){
  result.push(1)
}
for(let i=2;i<n;i++){
  result.push(result[i-1]+result[i-2])
}
```

```
console.log(result)
```
Q) console.log([] == [])

Q) const user = {
```
        name: 'John',
        getName: () => {
                return this.name;
                }
       }
console.log(user.getName())
```

Q) Count how many pairs of each element are there
Outpt {red:2, ….}

```
let arr = ['red', 'green', 'yellow', 'green', 'red', 'blue', 'red', 'white', 'red', 'green', 'blue'];
let obj={};

for(let i=0;i<arr.length;i++){
  if(arr.indexOf(arr[i]) == i){
    obj[arr[i]] = 1
  }else{
    obj[arr[i]] += 1
  }
 }


for(let elem in obj){
  obj[elem] = Math.floor(obj[elem]/2)
}
```

Or

```
for(let [key,value] of Object.entries(obj)){
  obj[key] = Math.floor(value/2)
}


 console.log(obj)
```

```
fun();
function fun() {
 console.log(1);
 setTimeout(() => {
 console.log(2);
 })
 const task = new Promise((resolve, reject) => {
 console.log(3);
 resolve(() => console.log(4))
 reject(() => console.log(5))
 })
 task
 .then(() => console.log(6))
 .catch(() => console.log(7))
 console.log(8);
}
// 1 3 8 6 2


console.log("Hello, World!");
//reverse each word in the sentence
let str = "hii all i am aditi priya"
let arr = str.split(" ")
let reversedWordArr = arr.map(elem=> elem.split("").reverse().join(""))
let result = reversedWordArr.join(" ")
console.log(result)



// let n= 5;

// function fact(n){
```

```
//   if(n==0){
//     return 1;
//   }
//   return n*fact(n-1);
// }


// let result = fact(n)
// console.log(result)

// aabc -> c b aa
// abca -> ac ab

// debit card -> bad credit


// let str = "Tom Marvolo Riddle"
// let comp_str = "I am Lord Voldemort"

// let arr = str.replace(/\s/g,'').toLowerCase().split('')
// let sortedstr = arr.sort()

// let comparr = comp_str.replace(/\s/g,'').toLowerCase().split('')
// let sorted_compstr = comparr.sort()

// console.log(sortedstr,sorted_compstr)
// if(sortedstr.join('') == sorted_compstr.join('')){
//   console.log("Anagrams ")
// }
// else{
//   console.log("Not anagrams")
// }

// Create a function called multiply that takes an integer n as input and returns a Promise with the output as n*2 after waiting for n seconds

let n= 5 ;
// function multiply(n){
//   return new Promise((resolve,reject) => {
//     setTimeout(()=>{
```

```
//       resolve(2*n)

//     },1000*n)
//   })
//   .then((res) => {
//     console.log(res)
//   })
// }
// multiply(n)

// async function multiply(n){
//   return setTimeout((n)=>{
//     return n*2
//   }, n*1000)
// }
// let res = multiply(n)
// console.log(res)

// let arr = [2398,34,7876,3413,5435,456,5462,9284,2998]
// let result = arr.sort((a,b) => a-b)
// console.log(result)


Promise.resolve(1)
.then((r) => {
  console.log(r);
  return r*10;
})
.then((r) => {
  console.log(r);
  return r*10;
})
.then((r) => {
  return new Promise((resolve, reject)=>{
    setTimeout(()=> {resolve(r*10)
      console.log(r)},1000)
  })
})
.then((r) => {
  console.log(r);
```

```
  return r*10;
})
.catch((err) => {
  console.log(err);
});


// let n= 5;

// function fact(n){
//   if(n==0){
//     return 1;
//   }
//   return n*fact(n-1);
// }


// let result = fact(n)
// console.log(result)

// aabc -> c b aa
// abca -> ac ab

// debit card -> bad credit


// let str = "Tom Marvolo Riddle"
// let comp_str = "I am Lord Voldemort"

// let arr = str.replace(/\s/g,'').toLowerCase().split("")
// let sortedstr = arr.sort()

// let comparr = comp_str.replace(/\s/g,'').toLowerCase().split("")
// let sorted_compstr = comparr.sort()

// console.log(sortedstr,sorted_compstr)
// if(sortedstr.join("") == sorted_compstr.join("")){
//   console.log("Anagrams ")
// }
// else{
```

```
//   console.log("Not anagrams")
// }

// Create a function called multiply that takes an integer n as input and returns a Promise with the output as n*2 after waiting for n seconds

let n= 5 ;
// function multiply(n){
//   return new Promise((resolve,reject) => {
//     setTimeout(()=>{
//       resolve(2*n)

//     },1000*n)
//   })
//   .then((res) => {
//     console.log(res)
//   })
// }
// multiply(n)

// async function multiply(n){
//   return setTimeout((n)=>{
//     return n*2
//   }, n*1000)
// }
// let res = multiply(n)
// console.log(res)

// let arr = [2398,34,7876,3413,5435,456,5462,9284,2998]
// let result = arr.sort((a,b) => a-b)
// console.log(result)


Promise.resolve(1)
.then((r) => {
  console.log(r);
  return r*10;
})
.then((r) => {
  console.log(r);
```

```
  return r*10;
})
.then((r) => {
  return new Promise((resolve, reject)=>{
    setTimeout(()=> {resolve(r*10)
      console.log(r)},1000)
  })
})
.then((r) => {
  console.log(r);
  return r*10;
})
.catch((err) => {
  console.log(err);
});


let bear = {
  sound: 'roar',
  roar() {
    console.log(this.sound);
  },
};

bear.sound = 'grunt';
let bearSound = bear.roar;
bearSound();
```

When you assign a method to a variable, it loses its original context (`this` binding). So, when `bearSound()` is called, the `this` inside the `roar` method is no longer the `bear` object, but it becomes the global object (or `undefined` in strict mode).

```
// str1 = "anagram"
// str2 = "nagaram"

// str1 = "cat"
// str2 = "rat"
```

// obj1: {a:3 ,n:1 , g:1 ,r:1 ,m:1}

// obj2: {n:1 ,a: 3,g:1, r: 1, m:1}

// {c:1,a:1,t:1}
// {r:1,a:1,t:1}

// Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.

// You may assume that each input would have exactly one solution, and you may not use the same element twice.

// You can return the answer in any order.

// Example 1:
// Input: nums = [2,11,15,7], target = 9
// Output: [0,3]
// Explanation: Because nums[0] + nums[1] == 9, we return [0, 1].
// Example 2:

// Input: nums = [3,2,4], target = 6
// Output: [1,2]
// Example 3:

// Input: nums = [3,3], target = 6
// Output: [0,1]

// obj = {2:0,11:0,15:0,7:0} -> O(1)
// for x in arr:
// y = target - x
// if y is in obj:

// [x,y]

```
let str = "ppprqq";
let result = "";
let temp =[]; no adjacent char same
no 2 same char together  pqrpqp



console.log("Hello, World!");

// let arr = [58,3,32,45,67]
// let res= arr[0]
// let temp;
// for(let i = 0; i< arr.length ; i++){

//   if(arr[i+1]<arr[i]){
//     temp= arr[i]
//     arr[i]=arr[i+1]
//     arr[i+1]=temp
//   }
// }
// res=arr[arr.length-2]
// console.log(res,arr)



//anagrams or not

let str1= "bat";
let str2="tab";
let str3 = "abc";
let str4= "hjse";

// function anag(a,b){
//   if(a.length != b.length){
//     console.log("Not anagrams")
//   }else{
//     let str1Obj={};
//     for(let i=0;i<str1.length;i++){
//       for(let j=0;j<str2.length;j++){

//         if(str1[i] != str2[j]){
//           if( j==str2.length){
//               console.log("not anagrams")
```

```
//           }
//         }
//         if(str[i] == str[j]){
//           break
//         }
//       }
//     }
//     if(i== str1.length){
//       console.log("anagrams")
//     }
//   }

// }
// anag(str1,str2)


function outer(){
  let a= 10;
  function inner(){
    console.log(a)
  }
  inner()
}
outer()









// let a={};
// let b = new Object();
console.log("Hello, World!");

          let       Input= "ABDEFGGEABEF";
          let inputArr = Input.split("")
          let maxLength = 0;
//        console.log(inputArr,"arr")
          for(let i = 0; i<inputArr.length; i++){
            let temp = [];
            temp.push(inputArr[i])
```

```
    for(let j=i+1;j<inputArr.length; j++){
      if(!temp.includes(inputArr[j])){
        temp.push(inputArr[j])
      }else{
        break;
      }
    }
    maxLength = temp.length> maxLength ? temp.length : maxLength
  }

  console.log(maxLength)
```

N ropes ,join so that minimum sum comes
[1,2,3,4] => 19

```
async function foo() {
console.log(2);
await null;
console.log(4);
}

console.log(1);
foo();
console.log(3);
```

cache

https://bigfrontend.dev/problem/validate-an-ip-address

```
isValidIP("1.267.34.abc")
isValidIP("abc")
```

```
console.log("Hello, World!");
// let str= "2.23.34.55"

function isValidIP(str) {
  // your code here
  let arr = str.split('.')
  console.log(arr)
  for(let i=0 ;i<arr.length;i++){
    console.log(parseInt(arr[i]))
```

```
    if((arr[i].length == 1 && arr[i]< 1 ) || (arr[i].length == 2 &&
arr[i]< 10 ) || (arr[i].length == 3 && arr[i]< 100 )|| arr[i]>= 255){
      console.log("not valid ipv4")
      break;
    }else if(i == arr.length -1){
      console.log("valid ipv4")
    }
  }
}
isValidIP("12.34")
```

there is two collection of users and books , we need to find the number of sandrala books of user afreen and user adity

quest :- Write a function that generates a random alphanumeric string of a given length.
ques :- Write a function that takes an array of integers and returns the largest difference between any two numbers in the array.