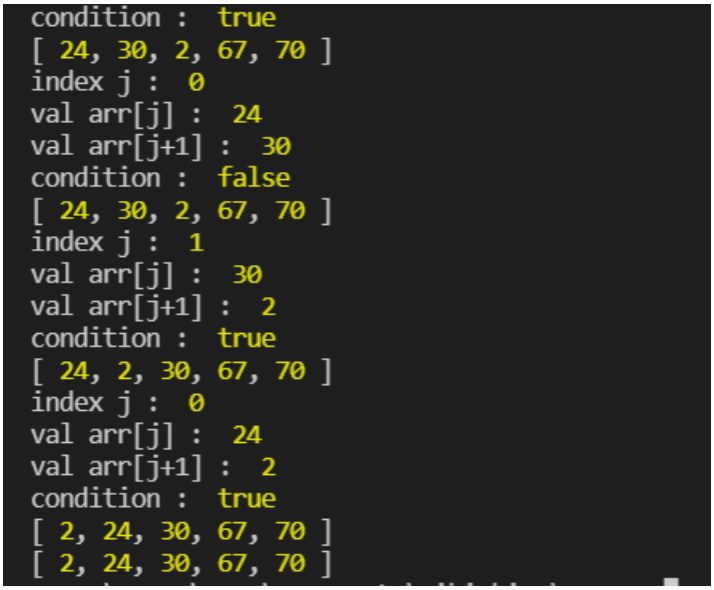
```
// Creating the bblSort function
function bblSort(arr) {

  for (let i = 0; i < arr.length; i++) { // 5 kali perulangan

    // Last i elements are already in place
    for (let j = 0; j < (arr.length - i - 1); j++) {// sisa dari jumlah array

      // Checking if the item at present iteration
      // is greater than the next iteration
      console.log('index j : ', j);
      console.log('val arr[j] : ', arr[j]);
      console.log('val arr[j+1] : ', arr[j + 1]);
      console.log('condition : ', arr[j] > arr[j + 1]);
      if (arr[j] > arr[j + 1]) {

        // If the condition is true then swap them
        let temp = arr[j]
        arr[j] = arr[j + 1]
        arr[j + 1] = temp;
      }
      console.log(arr);
    }
  }
  // Print the sorted array
  console.log(arr);
}


// This is our unsorted array
let arr = [67, 24, 30, 70, 2];


// Now pass this array to the bblSort() function
bblSort(arr);
```


condition : true
[ 24, 30, 2, 67, 70 ]
index j : 0
val arr[j] : 24
val arr[j+1] : 30
condition : false
[ 24, 30, 2, 67, 70 ]
index j : 1
val arr[j] : 30
val arr[j+1] : 2
condition : true
[ 24, 2, 30, 67, 70 ]
index j : 0
val arr[j] : 24
val arr[j+1] : 2
condition : true
[ 2, 24, 30, 67, 70 ]
[ 2, 24, 30, 67, 70 ]

```
function insertionSort(inputArr) {
    let n = inputArr.length;
    for (let i = 1; i < n; i++) {
        // Choosing the first element in our unsorted subarray
        let current = inputArr[i];

        console.log('inputArr[i] : ', inputArr[i]);
        // The last element of our sorted subarray
        let j = i - 1;

        console.log('j : ', j);
        console.log('current', current);
        console.log('inputArr[j] : ', inputArr[j]);

        console.log('val ', (current < inputArr[j]));

        while ((j > -1) && (current < inputArr[j])) {

            console.log("J ", j);

            inputArr[j + 1] = inputArr[j];

            console.log('inputArr[j+1] : ', inputArr[j + 1]);
            j--;
            console.log(inputArr);
        }
        inputArr[j + 1] = current;

        console.log('inputArr[j+1] 2 : ', inputArr[j + 1]);
        console.log(inputArr);
    }
    return inputArr;
}

let inputArr = [5, 2, 4];
insertionSort(inputArr);
console.log(inputArr);
```

```
inputArr[i] :  2
j :  0
current 2
inputArr[j] :  5
val  true
J  0
inputArr[j+1] :  5
[ 5, 5, 4 ]
inputArr[j+1] 2 :  2
[ 2, 5, 4 ]
inputArr[i] :  4
j :  1
current 4
inputArr[j] :  5
val  true
J  1
inputArr[j+1] :  5
[ 2, 5, 5 ]
inputArr[j+1] 2 :  4
[ 2, 4, 5 ]
[ 2, 4, 5 ]
```

```
// // binary search
let iterativeFunction = function (arr, x) {

    let start = 0, end = arr.length-1;

    while (start <= end){

        console.log("start", start);
        console.log("end", end);

        // Find the mid index
        let mid = Math.floor((start + end)/2);
        console.log('Median : ', mid);
        console.log('val mid : ', arr[mid]);

        console.log("condition", arr[mid] < x);

        // If element is present at mid, return True
        if (arr[mid] === x) return true;

        // Else look in left or right half accordingly
        else if (arr[mid] < x)
             start = mid + 1;
        else
             end = mid - 1;
    }

    return false;
}

let arr = [1, 3, 5, 7, 8, 9, 10];
let x = 10;

iterativeFunction(arr, x) ? console.log("Element found!")
                          :  console.log("Element not found!");
```

```
PS C:\Users\NIDA\Documents\Dibimbing\Day 9> node .\search.js
start 0
end 6
Median :  3
val mid :  7
condition true
start 4
end 6
Median :  5
val mid :  9
condition true
start 6
end 6
Median :  6
val mid :  10
condition false
Element found!
PS C:\Users\NIDA\Documents\Dibimbing\Day 9>
```

```
function selectionSort(inputArr) {
    let n = inputArr.length;

    for(let i = 0; i < n; i++) {
        // Finding the smallest number in the subarray
        let min = i;// awal ditambung itu index ke 0
        for(let j = i+1; j < n; j++){

            console.log("arr j ", inputArr[j]);
            console.log("min " ,inputArr[min]);
            console.log('val ', inputArr[j] < inputArr[min]);

            if(inputArr[j] < inputArr[min]) { // berubah menjadi index j
                min=j;
            }
         }
         console.log('min2 : ', min , i);
         if (min != i) {
             // Swapping the elements
             let tmp = inputArr[i];  // 67
             inputArr[i] = inputArr[min]; // 24
             console.log(tmp);
             console.log(inputArr[min]);
             inputArr[min] = tmp;

        }
        console.log(arr);
    }
    return inputArr;
}

let arr = [20,67,24];
console.log(selectionSort(arr));
```

```
PS C:\Users\NIDA\Documents\Dibimbing\Day 9> node .\selection-sort.js
arr j  67
min  20
val  false
arr j  24
min  20
val  false
min2 :  0 0
[ 20, 67, 24 ]
arr j  24
min  67
val  true
min2 :  2 1
67
24
[ 20, 24, 67 ]
min2 :  2 2
[ 20, 24, 67 ]
[ 20, 24, 67 ]
```

```
class Car {
    constructor(name, year, price) {
        this.name = name;
        this.year = year;
        this.price = price;
    }

    //setter
    setName(name) {
        this.name = name;
    }


    //getter
    getName() {
        return this.name;
    }

}


let mazda = new Car("Mazda", 2000, 100000);
console.log(mazda);
mazda.setName("Lambo");
console.log(mazda.getName());
console.log(mazda);
```

```
Car { name: 'Lambo', year: 2000, price: 100000 }
PS C:\Users\NIDA\Documents\Dibimbing\Day 7>
```

```
class Animal {
    constructor(name, type, food) {
        this.name = name;
        this.type = type;
        this.food = food;
    }

    showDesc = () => console.log("My Name is " + this.name + " Im a " + this.type
+ " and my food is " + this.food);

}


// inheritance
class Cat extends Animal {
    constructor(name, food) {
        super(name, 'Cat', food)
        this.name = name;
        this.food = food;
    }

    // polymorphism
    // showDesc = () => console.log("My Name is " + this.name + " and my food is
" + this.food);
}

let kitty = new Cat('kitty', 'fish');
kitty.showDesc();

module.exports = { Animal };
```

```
My Name is kitty Im a Cat and my food is fish
PS C:\Users\NIDA\Documents\Dibimbing\Day 7>
```

```
// deklarasi sebuah function
function showName(name) {
    console.log("My Name Is ", name);
}

showName("Fiqri");

function withoutParam() {
    console.log("This Is function Without param");
}

withoutParam();

// // function yang tidak mengembalikan nilai
console.log("Chat App");

function showMessage(from, text) {
    console.log(from.toUpperCase() + ': ' + text);
}

showMessage('Ann', "Hello");
showMessage('Jeanni', "Hello Ann");
showMessage('Ann', "How Are You?");
showMessage('Jeanni', "Im Fine");

function yang mengembalikan nilai
function sum(a, b) {
    return a + b;
}
```

```
function substract(c, d) {
```

```
    console.log(c - d);
    return c - d;
}

let num1 = sum(10, 100); // 110
let num2 = sum(20, 10); // 30

substract(num1, num2); // 80

console.log("Result = ", result);


// callback
// memasukan function lain ke dalam parameter function tertentu
function ask(question, yes, no) {
    confirm(question) ? yes() : no();
}

let showOk = () => alert('You Agreeed.');


function showCancel() {
    alert('You Canceled the Excecution.');
}


ask('Do you agree?', showOk, showCancel);

function sum(a, b) {
    return a + b;
}

// arrow function
// penyederhanaan penulisan sebuah function
let sum = (num1, num2) => num1 + num2;

console.log(sum(1, 100));
console.log(sum(2, 100));
console.log(sum(3, 100));
console.log(sum(4, 100));

let showStr = str => console.log(str);
showStr("tampil string");
```

```
// deklarasi sebuah function
function showName(name) {
    console.log("My Name Is ", name);
}

showName("Fiqri");

function withoutParam() {
    console.log("This Is function Without param");
}

withoutParam();

// // function yang tidak mengembalikan nilai
console.log("Chat App");

function showMessage(from, text) {
    console.log(from.toUpperCase() + ': ' + text);
}

showMessage('Ann', "Hello");
showMessage('Jeanni', "Hello Ann");
showMessage('Ann', "How Are You?");
showMessage('Jeanni', "Im Fine");

// function yang mengembalikan nilai
function m(a, b) {
    return a + b;
}

function substract(c, d) {
    console.log(c - d);
    return c - d;
}
```

```
let num1 = m(10, 100); // 110
let num2 = m(20, 10); // 30

substract(num1, num2); // 80

console.log("Result = ", substract);


// callback
// memasukan function lain ke dalam parameter function tertentu
function ask(question, yes, no) {
    confirm(question) ? yes() : no();
}

let showOk = () => alert('You Agreeed.');


function showCancel() {
    alert('You Canceled the Excecution.');
}


ask('Do you agree?', showOk, showCancel);

function dash(a, b) {
    return a + b;
}

// arrow function
// penyederhanaan penulisan sebuah function
let must = (num1, num2) => num1 + num2;

console.log(must(1, 100));
console.log(must(2, 100));
console.log(must(3, 100));
console.log(must(4, 100));

let showStr = str => console.log(str);
showStr("tampil string");
```

```
const _ = require('lodash');

// // // map array lodash
let num = [1, 2, 3, 4];

// num6 / array method js
// untuk memecah sebuah array
num.map(n => console.log(n));

console.log("-----------");

// // //lodash map
_.map(num, n => console.log(n));

// // // map obj lodash
let data = {
    nama: 'fiqri',
    umur: 15
}

data.map(n => console.log(n));

_.map(data, (value, prop) => console.log(prop + " : " + value));


let arr = [
    { n: 1 },
    { n: 2 }
];

// ES6
console.log('ES6', arr.map(x => x.n)); //dia harus memakai function lagi untuk
mengakses key dalam object array arr
```

```
// lodash
console.log('lodash', _.map(arr, 'n')); // bisa langsung akses key dalam object
tanpa menggunakan . lagi

let arr2 = [
    { a: [{ n: 1 }] },
    { a: [{ n: 2 }] }
];

//ES6
console.log(arr2.map((obj) => obj.a[0].n));
//lodash
console.log(_.map(arr2, 'a[0].n'));

// // lodash filter
let users = [
    { 'user': 'barney', 'age': 36, 'active': true },
    { 'user': 'fred', 'age': 40, 'active': false }
];

// // ES6
console.log(users.filter(user => user.active));

// //lodash
console.log(_.filter(users, 'active'));
console.log(_.filter(users, ['active', false]));
console.log(_.filter(users, { 'age': 36, 'active': true }));
```