\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ES6 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. Block-Scoped Const, Let

‘use strict’ : Che do nghiem ngat. Phat hien loi va cac hanh dong k an toan.

=> var: Khai báo kiểu global

=> let: Khai báo kiểu block scoped phạm vi hoạt động trong {} được khai báo và scoped con {}

=> const: Khai báo kiểu hằng số( ngăn chặn việc gán lại biến, (arr , function) thì vẫn có thể thay đổi ) phạm vi hoạt động trong {} được khai báo và scoped con {}

1. Arrow function

Function an danh

Const multiploer = (item,multi) => item \* multi;

Neu chi co 1 tham so co the bo qua ()

Const x = item => item \* 2

1. Rest parameter( tham so con lai)

Function howMany(...args){

Return “Length Args :” + args.length

}

howMany(0,1,2,[3,4,5]) // log 4

--Rest parameter loai bo su can thiet kiem tra mang args va cho phep ap dung map(), filter(), reduce() tren mang tham so.

-Math.max() : chi so dung cho (1,2,3,4) khong su dung duoc cho arr=[1,2,3,4] , Khi do ta dung Rest parameter.

Ví dụ:

Math.max(arr) // NaN

Math.max(...arr) // 4

- Sao chép 1 mảng vs Rest parameter

Let arr2 = (‘aaa’,’bbb’,...arr1)

1. Destructuring Assignment

- ex1 Arr

let date = [10,03,2020]

let [d,m,y] = date d=10,m=03,y=2020

- ex2 Obj

var person = {name:'name',email:'email.com'}

var {name: myname,email: email} = person

- ex3 gan truc tiep

var person = {name:'name'}

var {name: myname,,email:'email.com'} = person

=> Tiện cho việc gán nhiều giá trị không cần phải truy xuất nhiều

1. Class trong ES6

Class People{

Contructor(param1,param2...){  
 this.property1 = param1;

this.property2 = param2;

This.property... = param...;

}

get name(){

Return this.name;

}

Set name(nameUpdate){  
 this.name = nameUpdate;

}

}

Let A = new People(‘Nguyen A’,22,’sv’);

1. Export chia se khối mã

export default chỉ sử dụng 1 lần trong 1 file. Ngoài ra k thể sử dụng var let const.

Export.js

Const sum = (x,y) => {

Return x + y;

}

Const product = (a,b) =>{  
 return a \* b;

}

export {sum, product};

-------------------------------------------------------------

Import.js

Import {sum,product} from ‘./Export.js’

Or

Import \* as myExport from ‘./Export’

1. Promise trong javascript

Const myPromise = new Promise((resolve,reject)=>{

If(...){  
 resolve(data); Or resolve(func\_data);

}else{

reject(‘Message error !’);

}

})

Promise co 3 trang thai : pending,fulfiled, reject.

Resolve : trang thai tra ve thanh cong

Reject : trang thai tra ve that bai

.then()=> “Resolve”

.catch(err) => “Error”.