Bai3:Hoisting la cơ chế mac dinh cua TS de dinh nghia cac bien truoc hoac sau khi duoc môtả

Bài 4 từ khóa Let trong typescript

+Block scope: trong phạm vi 1 block vi du if else,for,switch

+can notre-assign:không ho tro khai bao bien lai lần 2

+not support hoisting:ko hỗtrợ hoisting có nghĩa là bat buoc khai bao biến let truoc khi su dụng

Bai 5 Let trong for

Giải quyết van de ko cần dung iif(immediately invoked function express) de tham chiếu bien đó vao trong 1 ham khác

for(let i=1;i<=5;i++){

//console.log(i);

setTimeout(function(){

console.log(i);

},1000);

}

//cach 2:dung iif (immediately invoked funtion)

for(vari=1; i<=5;i++){

(function(x){

setTimeout(function() {

console.log(x);

},2000);

})(i)

}

Bai6 const la hằng chi duoc phép read only va co the thay đổi gia trị nhưng ko duoc reassign lại

let number;

const number1 = 11;

number = 10;

const obj ={

name:"TEDU"

}

//obj.name="TEDU1";

/\*obj={

name:"Kenny",

email:"kennytran1989@gamil.com"

}\*///ko duoc re assign obj vi no da ducodinhnghia o tren la 1 const

console.log(obj.name);

Các buoc cài lite-server cua nodejs de chạy,

+npm install:de cai node module

+npm install lite-server -g :cai dat lite server global cho toàn cục

Sau khi cài xong restart lai visual studio code

+npm init de generate file package.json co noi dung

{

"name": "typescript",

"version": "1.0.0",

"description": "",

"main": "index.js",

"dependencies": {

"lite-server": "^2.3.0"//mapping gói lite server

},

"devDependencies": {},

"scripts": {

"start":"lite-server"//start de run lite server chay typescript

},

"author": "",

"license": "ISC"

}

+npm install: de install cac module trong node module

Bai 7 kieu du lieu trong type script

Kiểu number (decimal,hex,binary…)

let decimal:number = 10;

console.log(decimal);

Kieu boolean (true false)

let bool:boolean=true;

console.log(bool);

Kieu string

let color:string ="blue";

console.log(color);

Trong typescript để nội suy 1 bien khac trong mot chuoi ta dung dau ` ` and ${bienduocnoisuy}

let fullname:string ="kenny";

let sentense:string=`

hello,My name is ${fullname}

`;

Vd trên ta nội suy bien fullname trong chuoi cua biến sentence

Kieu Mảng

//cach 1 khai bao mang

let mang:number[]=[1,2,3];

console.log(mang);

//cach 2

let mang2:Array<number> = [4,5,6];

console.log(mang2);

//mảng số va chữ

let mang3:[string,number]= ["hello",10];

console.log(mang3);

//lỗi vì khai báo type la string number nhung value la number string

//let mang4:[string,number]=[10,"hello"];

//console.log(mang4);

Bai 8: Arrow function cách viết function tra ve kết quả 1 dòng thay cho return

let calculate= function(){

return10000000;

}

console.log(calculate());

let calculateArrow = () =>20000000;

console.log(calculateArrow());

them paramater vao function

let calculateSum = (bonus:number) =>200 + bonus;

console.log(calculateSum(20));

Bai 9 tu khoa this

var employ={

id:1,

info:function(){

console.log(this.id);

}

}

employ.info();

var employ2={

id:2,

info:function(){

setTimeout(function(){

console.log(this.id);

},1000)

}

}

employ2.info(); //undefined do tu this luc nay dang hiêu trong function cua hàm settimeout nen no ko hieu duoc thuoc tinh id cua object employ

var employ3={

id:3,

info:function(){

var self = this;

setTimeout(function(){

console.log(self.id);

},1000)

}

}

employ3.info();

var employ4={

id:4,

info:function(){

var self = this;

setTimeout(()=>console.log(this.id),1000) //dung arrow function de xuât thuoc tinh id cua object employ4

}

}

employ4.info();

bai 10 :gia tri mac dinh cho tham so cua function

let getBonus = function(value =10,tax=20){

console.log(value + tax);

}

getBonus(40);//truyền paramater cho value

getBonus(undefined,30);//truyen paramater cho value la undefined de function hieu lấy value =10 la mac dinh

var getBonus2 = function(value =10,tax=value \* 0.1){

console.log(value + tax);

}

getBonus2(20,undefined);//tham so paramater thu 2 se mac dinh theo tham so do duoc truyen dinh nghia tronghàm

let getPersent=()=>2;//arrow function

var getBonus3 = function(value =10,tax=value \* getPersent()){//gán gia tri tham so thu 2 bang tham so thu nhat nhan voi 1 gia tri cua arrow function

console.log(value + tax);

}

getBonus3(20);

Bai 11 Res parameter la cach khai bao truyen tung phan tu vào function de dua vao thành 1 mảng

let displayColors= function(...color:string[]){

console.log(color);

}

displayColors('Red');

displayColors('Red','Green');

displayColors('Red','Green','Blue');

vd duoi day la sai cach khai bao 1 rest parameter vi 1 rest parameter la 1 mảng phải la duoc khai báo cuoi cung trong 1 list tham so duoc truyen vao 1 function cho nên khai báo tham số old duoi day la sai quy cách

let displayColors= function(message,...color:string[],old){

//console.log(message);

console.log(color);

console.log(old);

}

let message="Hello";

let old=29;

displayColors(message,'Red',old);

displayColors(message,'Red','Green',old);

displayColors(message,'Red','Green','Blue',old);

Bai 12 Spread operator khai bao truyen vào 1 mảng vào function và boc tach tưng phần tử trong mảng đó (nguoc lại với rest paramater)

let displayColorSpread= function(message,...color:string[]){

console.log(message);

console.log(color);

for(let i in color){

console.log(color[i]);

}

}

let messageSpread="Hello";

let colorSpread=["red","green","blue"];

displayColorSpread(messageSpread,...colorSpread);

Bai 13 Destructing array (Bóc tách array ra các bien doc lap)

let employees= ['A','B','C'];

let [emp1,emp2,emp3] = employees;

//let [,,emp3] = employees; lay value cua phan tu thu 3 của mảng employees

//let [emp1,...emp2] = employees; lấy value của phan tu thu 1 cua mảng employees và dua gia tri cua phan tu thu 2 va thu 3 vao mang emp2 KQ:A ['B','C']

console.log(emp1);

console.log(emp2);

console.log(emp3);

Bai 14 Destructing Object (Boc tach doi tuong)

let emp = {

fname:'Technology',

lname:'Education',

level:2

}

let {fname,lname,level} = emp;

console.log(fname);

console.log(lname);

console.log(level);

co the dat dinh danh (alias) moi cho tung thuoc tinh

let emp = {

fname:'Technology',

lname:'Education',

level:2

}

let {fname :f,lname:l,level:lv} = emp;

console.log(f);

console.log(l);

console.log(lv);

Bai 15 String template

letname3="TEDU";

letmess=`

Welcome ${name} to ES6 and TypeScript

This is tutorial series for ES6 and Typescript

`;

console.log(mess);

Addsourcecode typescript den git repository

+Tao repo trengit hub

<https://github.com/kennytran1989/typescript-es6.git>

+ add origin den repo vua tạo

git remote add origin <https://github.com/kennytran1989/typescript-es6.git>

+hiển thị remote origin vua add

git remote show origin

+Pull repo tu origin vua tao ve

git pull origin

+push len origin master cua repo do

git push origin master

Nếu xay ra loi do trong thu muc chua co file .gitinore va file README.md cho nen ta tiep tuc run lenh

git push origin master –allow -unrelated -histories (pull nhung file con thieu local tu tren origin master cua repo đo ve va merge chúng lại)

Bai 16 for off loop

let colorName= ["Red","Green","Blue","Yellow","Purple"];

//liet ke value cua mang thong qua index

for(let index in colorName){

console.log(colorName[index]);

}

//liêt ke value truc tiep cua mảng

for(let item of colorName){

console.log(item);

}

let name4 ="TEDU";

for(let c of name4){

console.log(c);

}

Bai 17 class typescript

class person{

run(){

console.log("Person is running...");

}

}

let p1=new person();

console.log(p1.run());

class trong type typescript ko ho tro hoisting phai khai bao truoc rui moi tao doi tuong cua class do

Bai 18 khai bao phuong thuc trong class

class person2{

constructor(nameinput){

this.name = nameinput;

console.log(this.name+ " is Constuctor.");

}

static talk(){

console.log("This is method static");

}

run2(){

console.log("This is method normal");

}

}

let p2=new person2("TEDU");

p2.run2();

person2.talk();

Bai 19 ke thua trong typescript

class person3{

constructor(name){

console.log(name+ " Person3 constructor...");

}

getid(){

return 10;

}

}

class employ5 extends person3{

constructor(name){

super(name);//goi constructor cua lop cha giong parent php

console.log(name + " Employ constructor...");

}

getId(){

return super.getid();//goi ham getID() cua lop cha person3

}

}

let empl=new employ5("TEDU");

console.log(empl.getId());

Bai 20 Module

Module duoc hieu la tung phan cua chuong trinh duoc chia nho ra de de control va develop

trong typescript biến va function duoc dinh nghia trong 1 module,muốn sử dung trong nhung module khac nhung bien va hàm đó phải duoc export value ra sau đó nhung module khác mới sử dung duoc

Bai 21 cách cài đặt module trong ES6

+tao thư mục es6module và thu mục src chứa 2 file moduleA.js và moduleB.js

+Cài đặt gói es6-module-loader

npm install es6-module-loader --save

+Cài đặt traceur

npm install traceur --save

+Cài đặt systemjs

npm install systemjs

Sau khi cài đặt tạo file index.html với nôi dung gọi những gói module vừa install

<html>

<head>

<title>ES6 Moduel Loader</title>

<script src="node\_modules/traceur/bin/traceur.js"></script>

<script src="node\_modules/systemjs/dist/system.src.js"></script>

<script src="node\_modules/es6-module-loader/dist/es6-module-loader-dev.js"></script>

</head>

<body>

<script>

System.import('src/moduleA.js');

</script>

</body>

</html>

Bai 23 export và import module

Cú pháp:

export let name\_variable="Value ";

//cach 1

//export let fname="TEDU";

//export let lname="Online chanel";

//cach 2

let fname="TEDU";

let lname="Online chanel";

export{fname,lname};

import {name\_variable} from 'file\_name';

import {fname,lname} from './moduleB.js';

console.log(`${fname} ${lname}`);

Các vấn đề khác

let obj={

name:"TEDU"

};

export{fname,lname,obj};

import {fname,lname,obj} from './moduleB.js';

//fname="kenny" sai ko the thay doi gia tri cua thuoc tinh duoc import

obj.name="Changed";// doi value cua thuoc tính cua object duoc import

obj.old="19";// co the thêm thuoc tính cho object duoc import

console.log(obj);

console.log(obj.name);

console.log(obj.old);

console.log(`${fname} ${lname}`);

//co the dat alias cho bien duoc import vào

//import {fname as f,lname as l} from './moduleB.js';

//console.log(`${f} ${l}`);

Bai 23 Export default

let fname2= "Kenny";

export default fname2;// export default 1 biến bất kỳ

ben file import

import {default as name2} from './moduleB.js'; // co the dat dinh danh alias cho bien duoc export la default

console.log(name2);

\*Note: Export default la mặc định chi duoc export 1 bien duy nhất

Bai 24 Export class va function

export function greet(message){

console.log(message);

}

export class Employee{

constructor(){

console.log("Employee Constructor");

}

greeting(message){

return message;

}

}

import {greet,Employee} from "./moduleB.js";

console.log("Example cho import function va class");

greet("TEDU");

let em=new Employee();

console.log(em.greeting("Wellcome to TEDU"));

Bai 25 kieu array va tuple

kieu array

let arr1:string[]= ["Hello","World"];

let arr2:number[]= [1,2,3,4,5];

let arrAny:any[]=[10,"Kenny",true];

let arr3:Array<string>=["TEDU","Online"];

console.log(arr1[0]);

kieu tuple là kiểu mảng chỉ định ra các kiểu dữ liệu khác nhau được cho phép bên trong 1 mảng

let student: [string,number,boolean];

student=["Kenny",29,true];

//student=["kenny","Developer","true"]; //sai vì mang studen chỉ định theo thu tự la string,number,boolean

student[2]=false;

//student[2]="test"; // sai vi phan tu thu 3 duoc xac dinh la kieu boolean

student[3]="developer";

console.log(student);

Bai 26 Su dung constructor

//cach1

class employee{

public fname:string;

public lname:string;

constructor(fname:string, lname:string){

this.fname = fname;

this.lname = lname;

}

}

//cách 2

class employee1{

constructor(public fname:string, public lname:string){

}

}

let e=new employee1("TEDU","Online");

console.log(e.fname + " " +e.lname);

Bai 27 Interface

interface person1{

fname:string,

lname:string,

old?:number//dau ? de xac dinh thuoc tinh do ko bat buoc khi mot bien su dung interface

}

let pe1 : person1 = {

fname:"Tedu",

lname:"Online",

old:2

}

console.log(pe1.fname+" "+pe1.lname+" "+pe1.old);

let pe2 : person1 = {

fname:"Tedu2",

lname:"Online2",

}

Các vấn đề khác

interface LabelledValue {

readonly label: string;

}

function printLabel(labelledObj: LabelledValue) {

//labelledObj.label=11; // sai vi label la thuoc tinh chỉ đọc của interface LabelledValue

console.log(labelledObj.label);

}

let myObj = {size: 10, label: "Size 10 Object"};

printLabel(myObj);

//printLabel(myObj.size); // sai vi interface LabelledValue ko co thuoc tính size

Bai 28

Decorator trong type script giong nhu 1 decorator design pattern no cho them va chỉnh sua cac function tính năng gốc để cho ra các function với tính năng mới