typescript 开发环境

在线compiler 官网 playground

本地

npm node package manager

tsc hello.ts

hello.ts🡪hello.js

* 多行字符串

var a=`hello

world`;

对应的javascript代码

var a="hello\nworld";

* 字符串模板

`${value1}`

`${getName()}`

* 拆分字符串

function test(val1, val2, val3){  
 console.log(val1, val2, val3);

}

var a="helloworld";

var b=19;

test`你好世界${a}, I am ${b}`

第一个参数["你好世界", ", I am ", ""]

第二个参数"helloworld"

第三个参数19

用一个字符串模板传递参数

typescript类型推断机制

var value1="fawfew";

此时value1=18;

报错

想变类型的话这样写

var value1 : any="fafwa";

var value2 : number=120;

var isMan : boolean = true;

参数默认值

function test(a: string = "hello);

类

class Person(){

name; //不用写var

test(){  
 console.log("555");

}

}

var p1=new Person();

p1.name="nihao"

访问控制符

public 默认

private

protect

constructor(){}

构造函数

constructor(name: string){

this.name=name;

}

var p1=new Person("wuyifan");

的效果等同于

constructor(public name:string){

}

继承

class Son extends Father {

age: number;

constructor(name: string, age: number){

super(age);//子类构造方法必须先super();

}

work(){

super.eat();//super还可以调用父类方法

}

}