### ES567测试题

### ###1.简述一下js有几种数据类型？

//1.基础数据类型：（简记：nubnss）

null，undefined，boolean，number，string，symbol

//2.对象(引用，复合)类型

Object，Array，Function

### ###2.typeof的返回值有几种？typeof有什么特点？

>Typeof：一个不正确，两个不精确，其余都准确。

>一个不正确：typeof(null)的结果是object。

>两个不精确：typeof []和{} 的结果都是object

typeof 1 // 'number'

typeof '1' // 'string'

typeof undefined // 'undefined'

typeof true // 'boolean'

typeof console.log // 'function'

typeof Symbol() // 'symbol'

typeof b // b 没有声明，但是还会显示 undefined

typeof [] // 'object'

typeof {} // 'object'

typeof null // 'object'

### ###3.说一下null与undefined？

>undefined：没有赋值（一般不会将一个变量手动赋值为undefined，因为没有意义）

var a;

console.log(a)//undefined

console.log(b)//undefined

var b = 1;

>null：赋值了, 但是值为null

//一个约定俗称的小规矩，一般赋值为null代表这个变量接下来要存储的类型是对象

var a = null

###4.说出下列语句的结果

console.log(1 + '1')//'11'

console.log('1'- 0)//1

console.log(2 \* '2')//4

console.log([1, 2] + [2, 1])//1,22,1

console.log(typeof console.log('1'- 0))// 1 undefined

### ###第一题：写出以下代码段的执行结果

>题目来源：腾讯外包前端笔试题

>难度系数：★★★

>考察点:

(1). 作用域与作用域链

(2). 变量查找

var x = 10

function fn() {

console.log(x)

}

function show(f) {

var x = 20

f();

}

show(fn) // 10

### ###第二题：写出执行结果

>题目来源：百度中级前端笔试题

>难度系数：★★★★

>考察点:

(1). 引用变量赋值

(2). 内存分析

(3). 函数调用传参

(4). 运算符的优先级

var a = {n: 1}

var b = a

a.x = a = {n: 2}

console.log(a.n, b.n) // 2 1

console.log(a.x, b.x) // undefined,{n:2}

### ###第三题：程序执行结果

>题目来源：360企业安全前端笔试题

>难度系数：★★★★

>考察点:

(1). 作用域

(2). 加法运算

(3). 数据类型

(4). js事件循环机制

function print(arr) {

var i = arr.length;

for (var j=0; j<i; j++ ){>题目来源：今日头条前端笔试题

>难度系数：★★★

>考察点:

(1). 箭头函数

(2). ES6对象简写方式

var f = x => x;

console.log(f(1)); //1

var f = x => {x};

console.log(f(1)); //undefined

var f = x => ({x});

console.log(f(1));//{x: 1}

setTimeout(()=>{

console.info(arr[j]+1)//三次NaN

},200)

}

}

print([1,2,3])

### ###第四题：程序执行结果

### ###第五题：控制台输出结果

>题目来源：未知

>难度系数：★

>考察点:

(1). Promise对象

(2). js事件循环机制

const promise = new Promise((resolve, reject) => {

console.log(1)

resolve()

console.log(2)

})

promise.then(() => {

console.log(3)

})

console.log(4)

//1

//2

//4

//3

### ###第一题：写出控制台输出结果

var obj = {

fn2: function () {

console.log(fn2)

// 如果想输出fn2属性的值，这么写：console.log(this.fn2) 或者 obj.fn2也行, 但比较死。

}

}

obj.fn2() //报错， fn2 is not defined

### ###第二题：写出控制台输出结果

function fn() {

console.log(a)

var a = 3

}

fn() // undefined

//解析：函数内部预处理，不去找全局的a，同时变量提升，打印之前a定义了，还未赋值，undefined

###第三题：写出控制台输出结果

function fn2() { // 预计处理:

console.log(a)

a = 3 // 此时的a是全局变量

}

fn2() // a is not defined

### ###第四题：你所知道的常见拷贝技术？都分别都有什么问题或特点？

1. arr.concat(): 数组浅拷贝

2. arr.slice(): 数组浅拷贝

3. JSON.parse(JSON.stringify(arr/obj)): 数组或对象深拷贝, 但不能处理函数数据

### ###第五题：请编写一个函数，完成深度复制对象类型的数据

//判断数据类型的方法

function getTYpe(target) {

return Object.prototype.toString.call(target).slice(8,-1)

}

function clone(target) {

let result

if(getTYpe(target) === 'Object'){

result = {}

}

else if(getTYpe(target) === 'Array'){

result = []

}

else{

return target

}

for (let key in target){

let item = target[key]

if(getTYpe(item) === 'Object'||'Array'){

result[key] = clone(item)

}

else{

result[key] = item

}

}

return result

}