一、基础知识填空

JS中的数据类型包含哪些(要求详细)以及基本数据类型和引用数据类型的本质区别：\_\_\_\_基本数据类型（number,string,null,Boolean,undefined）\_\_和引用数据类型\_\_（object对象数据类型［对象，数组，正则］\_\_，function函数数据类型）区别：基本数据类型是直接的操作值的，引用数据类型操作的是对应空间的引用地址，它存储的是一个地址。\_\_\_\_\_\_\_\_

JS中如何的检测数据类型(写出四种方法)：\_typeof, instanceof,\_\_constructor,\_\_ Object.prototype.toString.call()\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ JS中判断某个值是否是一个有效数字的方法是：\_\_\_\_isNaN\_\_\_\_\_\_\_\_\_\_\_

想要接收函数执行的时候传递的参数，都可以如何的处理：\_\_arguments\_\_\_\_\_\_\_\_\_\_\_\_ 将其他的数据类型转化为number类型的方式有那些：\_\_Nubmber,parseInt,parseFloat\_\_\_\_\_\_\_\_\_\_\_\_\_\_

二、写出DOM中常用的方法和属性

有一个div:<div class=”w”name=”h”id=”div1”></div>我想获取这一个div你有几种解决办法(不考虑兼容)：getElementsByClassName(“w”)\_\_,getElementById(“div1”)\_\_\_\_，getElementsByTagName(“div”)\_\_ ,get ElementsByName(“h”)

获取当前浏览器屏幕的宽度和高度(兼容所有的浏览器)：\_ **document.documentElement.clientWidth||document.body.clientWidth** \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 获取上一个哥哥元素节点（兼容所有的浏览器）: \_previousSibling\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

动态创建一个div标签，并且添加到body的最后面位置：\_document.createElement(“div”) body.appendChild(“div”)\_\_\_\_\_\_\_\_\_\_\_\_\_ 继续上一题，把刚才创建的克隆一份一模一样的，添加到刚才创建的div前面：\_cloneNode(“div”).insertBefore(“div”)\_\_\_\_\_\_\_\_\_\_\_\_

三、数组的基础知识和应用

我想删除数组中的最后一项，至少写出两种办法：ary.pop() ,**splice(ary.length-1,1)\_\_\_,ary.length--**\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 我想在数组的末尾增加一个元素，至少写出两种办法：\_\_\_push(); ary[ary.length]\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

基于原来的数组复制一份一模一样新的数组，不考虑原有数组是否改变的情况，请写出至少三种方案： \_slice(0),splice(0)\_\_\_concat()\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

数组从小到大排序(可以处理10以上的数字)：\_ **sort(function(a,b){return a-b;})**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 实现找到第n项到第m项的内容，返回一个新的数组(原有数组不变)：slice(n-1,m)\_\_\_\_\_\_\_\_\_\_\_\_\_\_

四、写出下面表达式的运算结果

[]==false：\_\_\_true\_\_\_\_\_\_\_\_\_ ![]==false：true\_\_\_\_\_\_\_\_\_\_\_\_ []==![]：\_\_\_\_\_\_\_\_false\_\_\_\_\_\_ Number(“13px”)==NaN：\_\_\_false\_\_\_\_\_\_\_\_ 5+”3”-2：\_\_\_\_\_\_\_51\_\_\_\_\_\_\_\_\_\_\_\_

If(“3px”\*3){alert(“true”)}else{alert(“false”)}：\_\_\_false\_\_\_\_\_\_\_\_\_ If(“3px”+3){alert(“true”)}else{alert(“false”)}：\_\_\_true\_\_\_\_\_\_\_\_

If（document.body）{alert(“ok”);}else{alert(“no”)}：\_ok\_\_\_\_\_\_\_\_\_\_\_\_\_\_ alert([])：\_\_\_“”\_\_\_\_\_\_\_\_\_\_\_ null==undefined：\_\_\_\_\_true\_\_\_\_\_\_\_\_\_\_\_

1. <input type="button" value="按钮1"/><input type="button" value="按钮2"/><input type="button" value="按钮3"/><input type="button" value="按钮4"/><input type="button" value="按钮5"/>

var ary=[“red”,”yellow”,”pink”,”green”,”blue”];

以上是页面中的5个按钮，和一个存储五个颜色值的数组，要求实现一个方法bindEvent，循环给5个按钮绑定点击事件，当点击的时候，整个body的背景颜色正好可以和数组中的指定项颜色对应(例如：点击第三个按钮，body的背景颜色”pink”)，请按照你的思路来实现?

六、

var ary=[1,2,3,4];

function fn(ary){

ary[0]=0; ary=[0]; ary[0]=100; return ary;

}

var res=fn(ary); console.log(ary); console.log(res); //此题偏难，认真思考(360面试题难度降低版本代码)

### 二、选择题：（要求写出分析步骤/画图）

**1、** console.log(a); B

var a=12;

function fn(){

console.log(a);

var a=13;

}

fn();

console.log(a);

输出的三次分别是多少

A、undefined 12 13 B、undefined undefined 12

C、undefined undefined 13 D、有程序报错

1. console.log(a); A

var a=12;

function fn(){

console.log(a);

a=13;

}

fn();

console.log(a);

A、undefined 12 13 B、undefined undefined 12

C、undefined undefined 13 D、有程序报错

1. console.log(a);A

a=12;

function fn(){

console.log(a);

a=13;

}

fn();

console.log(a); 输出的三次分别是多少

A、undefined 12 13 B、undefined undefined 12

C、undefined undefined 13 D、有程序报错

1. var foo=1;   
    function bar(){

if(!foo){

var foo=10;

}  
 console.log(foo);

}  
 bar(); 输出的结果是多少

A、1 B、10 C、undefined D、报错

1. var n=0;

function a(){

var n=10;

function b(){

n++;

alert(n);

}

b();

return b;

}

var c=a();

c();

alert(n);

弹出三次的结果分别是什么？

A、1 1 1 B、11 11 0 C、11 12 0 D、11 12 12

1. var a=10,b=11,c=12;

function test(a){

a=1;var b=2;c=3;

}

test(10);

alert(a); alert(b); alert(c); 弹出的三个值分别是多少？

A、1 11 3 B、10 11 12 C、1 2 3 D、10 11 3

7、if(!("a" in window)){

var a=1;

}

alert(a);

弹出的a的值是？

A、1 B、undefined C、报错 D、以上答案都不对

8、var a=4;

function b(x,y,a) {

alert(a);

arguments[2]=10;

alert(a);

}

a=b(1,2,3); alert(a); 三次弹出的结果是

A、3 3 4 B、3 10 4 C、3 10 10 D、3 10 undefined

9、var foo='hello';

(function(foo){

console.log(foo);

var foo=foo||'world';

console.log(foo);

})(foo);

console.log(foo); 三次分别输出什么？

A、hello hello hello B、undefined world hello C、hello world world D、以上答案都不正确

10、var a=9;

function fn(){

a=0;

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

}

var f=fn()

var m=f(5);alert(m);

var n=fn()(5);alert(n);

var x=f(5);alert(x);

alert(a); 弹出的四次结果？

A、6 6 7 2 B、5 6 7 3 C、5 5 6 3 D、以上答案都不正确

### 三、思考题：

1. document.parentNode的值是什么，为什么 null，因为document.parentNode是最顶级的。
2. document.parentnode的值是什么，为什么 。Undefined，因为document.parentnode不是私有属性也不是公有属性，所以找不到
3. prototype是谁的属性，它起什么作用？\_\_proto\_\_是谁的属性，它有什么作用？prototype和\_\_proto\_\_有什么联系？

Prototype是函数数据类型的属性，作用是通过他们之间的关系，找到公共属性。 \_\_proto\_\_是对象数据类型的属性， \_\_proto\_\_的值是当前实例所属类的原型prototype

1. hasOwnProperty和in都是用来判断对象的属性的，这两个是怎么用的，有什么区别？

hasOwnProperty只用来判断一个属性是不是某个对象的私有属性，是就返回true不是则是false，in是用来判断一个属性是不是某个对象里的属性，而判断这个属性不管是公有属性还是私有属性只要是就返回true,否则false.

5、什么是闭包，它有什么作用？

函数形成一个新的私有的作用域保护了里面的私有变量不受外界的干扰(外面修改不了私有的,私有的也修改不了外面的)-->"闭包"