# 一、选择题

## 1、分析下段代码输出结果是（ b ）

　　var arr = [2,3,4,5,6];

　　var sum =0;

　　for(var i=1;i < arr.length;i++) {

　　sum +=arr[i] }

　　console.log(sum);

　　A．20 B．18 C．14 D．12

## 以下关于 Array 数组对象的说法不正确的是（CD

## ）

　　 A．对数组里数据的排序可以用 sort 函数，如果排序效果非预期，可以给 sort 函数加一个排序函数的参数

　　B．reverse 用于对数组数据的倒序排列

　　C．向数组的最后位置加一个新元素，可以用 pop 方法

　　D．unshift 方法用于向数组删除第一个元素

## 3、以下代码运行的结果是输出( B )

　　var a = b = 10;

　　(function(){

　　var a=b=20

　　})();

console.log(b);

A．10 B．20 C．报错 D．undefined

## **4**、**以下代码运行后的结果是输出( B )**

var a=[1, 2, 3];

　　console.log(a.join());

**A．123 B．1,2,3 C．1 2 3 D．[1,2,3]**

## **5、在 JS 中，’1555’+3 的运行结果是( C )**

A．1558 B．1552 C．15553 D．1553

## **6、以下代码运行后弹出的结果是( B )**

　　var a = 888;

　　++a;

　　alert(a++);

　　A．888 B．889 C．890 D．891

## **7、关于变量的命名规则，下列说法正确的是（ ABCDE ）**

　　A．首字符必须是大写或小写的字母，下划线（\_）或美元符（$）

　　B．除首字母的字符可以是字母，数字，下划线或美元符

　　C．变量名称不能是保留字

　　D．长度是任意的

　　E．区分大小写

## **8、下列的哪一个表达式将返回值为假（ B ）**

　　A．!(3<=1)

　　B．(4>=4)&&(5<=2)

　　C．(“a”==“a”)&&(“c”!=“d”)

　　D．(2<3)||(3<2)

## 10、var x = 1; function fn(n){n = n+1}; y = fn(x); y 的值为（ D ）

　　A．2 B．1 C．3 D．undefined

## [1,2,3,4].join('0').split('') 的执行结果是（ C ）split是切割的意思

　　A．'1,2,3,4'

　　B．[1,2,3,4]

　　C．[“1”,“0”,“2”,“0”,“3”,“0”,“4”]

　　D．'1,0,2,0,3,0,4'

## 12、下面代码的运行结果是：第一次弹（ A ）第二次弹（ D ）

　　function fn1() {

　　alert(1);

　　}

　　alert( fn1() );

函数再打印里面什么都 没有

　　A．1

　　B．alert(1);

　　C．function fn1() { alert(1); }

　　D．Undefined

## 13，以下代码运行后，结果为（ B ）

　　fn1();

　　var fn1 = function(a){ alert(a); }

函数不会提升， 变量提升

　　A．1 B．程序报错 C．alert(1); D．undefined

## **18，下面的等式成立的是（ D ）**

　　A．parseInt(12.5) == parseFloat(12.5)

　　B．Number('') == parseFloat('')

　　C．isNaN('abc') == NaN

　　D．typeof NaN === 'number'

## 19，下面的描述中不正确的是（AC ）

　　A．'==' 在比较过程中，不但会比较两边的值，还会比较两边的数据类型。相等不比较类型，全等才比较类型

　　B．NaN == NaN 的结果是 true。

　　C．isNaN，判断传入的参数是否为数字，为数字返回true，否则返回false

　　D．字符串的 length 只可以获取，不可以设置。

## 20，以下代码中，会出现什么结果：第一次弹（ A ），第二次弹（ ），第三次弹（ ）

　　function fn1() {

　　var a = 0;

　　function fn2() { ++a; alert(a); }

　　return fn2;

　　}

　　fn1()();

　　var newFn = fn1();

　　newFn();

　　newFn();

　　A．1 B．2 C．0 D．3

## 21，下面代码的运行结果是：第一次弹（ D ），第二次弹（ D ）

　　var a = 100;

　　function fn1() {

　　alert(a);

　　var a = 10;

　　}

　　alert( fn1() );

　　A．100

　　B．10

　　C．function fn1() { alert(1); }

　　D．undefined

## 22，以下代码运行后，arr的结果为（ A ），arr2的结果为（ C ）

　　var arr = [1,2];

　　var arr2 = arr.concat();

　　arr2.push( arr.splice(1,0) );

A．[1,2] B．[1,2,[2]] C．[1,2,[]] D．[1,2,3]

## 23，下面关于数组的描述正确的是（ A ）

　　A．数组的 length 既可以获取，也可以修改。

　　B．调用 pop() 方法，不会修改原数组中的值。

　　C．shift() 方法的返回值是新数组的长度。

　　D．调用 concat() 方法，会修改原数组的值。

## 24，下列程序中alert按顺序分别弹出：（ B ），（ B ），（ B ）

　　var a = 10;

　　function test() {

　　a = 100;

　　alert(a);

　　alert(this.a);

　　var a;

　　alert(a);

　　}

　　test();

　　A．10 B．100 c．undefined D．程序报错

## 25，分析下面的代码，输出的结果是（ C ）

　　var arr=new Array(5);

　　arr[1]=1;

　　arr[5]=2;

　　console.log(arr.length);

　　A．2 B．5 C．6 D．报错

## 28, Math.ceil(-3.14)的结果是（ B ），Math.floor(-3.14) 的结果是（ C ）

　　A．-3.14 B．-3 C．-4 D．3.14

## 30, 以下 ECMAScript 变量命名格式正确的是（

## A ）

A.\_125dollor B．1207A C．-dollor D．This

1.

var a = {n: 1}

var b = a;

a.x = a = {n: 2}

console.log(a.x);

console.log(b.x)

对象属性赋值优先于对象赋值

a undnfined

N:2

6、写出下面代码的输出结果，并说明原因

var x= 0;

var foo = {

x:1,

bar:{

x:2,

baz: function () {

console. log(this.x)

}

}

}