Javascript Basic Quiz

**1、分析下段代码输出结果**

　　var arr = [2,3,4,5,6];

　　var sum =0;

　　for(var i=1;i < arr.length;i++) {

　　sum +=arr[i]    }

　　console.log(sum);

**2、以下关于 Array 数组对象的说法不正确的是（  ）**

　　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(  )**

　　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(   )**

　　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(   )**

　　A．1558     B．1552     C．15553     D．1553

**6、以下代码运行后弹出的结果是(   )**

　　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（    ）**

　　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（    ）**

　　A．!(3<=1)

　　B．(4>=4)&&(5<=2)

　　C．(“a”==“a”)&&(“c”!=“d”)

　　D．(2<3)||(3<2)

**9、下面代码，k的运行结果是（    ）**

　　var i = 0,j = 0;

　　for(;i<10,j<6;i++,j++){

　　k = i + j;

　　}

　　A．16     B．10     C．6     D．12

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

　　A．2     B．1     C．3     D．undefined

**11、[1,2,3,4].join('0').split('') 的执行结果是（    ）**

　　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（    ）第二次弹（    ）**

　　function fn1() {

　　alert(1);

　　}

　　alert( fn1() );

　　A．1

　　B．alert(1);

　　C．function fn1() { alert(1); }

　　D．undefined

**13，以下代码运行后，结果为（    ）**

　　fn1();

　　var fn1 = function(a){ alert(a); }

　　A．1     B．程序报错     C．alert(1);     D．undefined

**14，var n = "miao wei ke tang".indexOf("wei",6)；n的值为：（    ）**

　　A．-1     B．5     C．程序报错     D．-10

**15，下面对 substring() 方法描述不正确的是（    ）**

　　A．一共有两个参数，省略第二个参数表示从参数开始位置提、截取到字符串结束。

　　B．提取之前会比较两个参数的大小，并根据大小调整位置。

　　C．可以接收负数参数，负数表示从后往前数字符位置。

　　D．如果没有设置参数，直接返回整个字符串。

**16，alert( "12">"9" ) 的运行结果正确的是（    ）**

　　A．true     B．false

**17，下面的描述中不正确的是（    ）**

　　A．arguments 中保存了实际传入函数内的所有参数。

　　B．return 只能在函数内部使用。

　　C．setInterval(fn1,1000) 只会调用一次 fn1。

　　D．Date 对象的 getMonth() 获取到的值比实际月份小1。

**18，下面的等式成立的是（    ）**

　　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（    ）**

　　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。

　　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（    ），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（    ），第二次弹（    ）**

　　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（    ），arr2的结果为（    ）**

　　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．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：（    ），（    ），（    ）**

　　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（    ）**

　　var arr=new Array(5);

　　arr[1]=1;

　　arr[5]=2;

　　console.log(arr.length);

　　A．2     B．5     C．6     D．报错

**26，在JavaScript中，下列（    ）语句能正确获取系统当前时间的小时值。**

　　A．var date=new Date();  var hour=date.getHour();

　　B．var date=new Date();  var hour=date.gethours();

　　C．var date=new date();  var hour=date.getHours();

　　D．var date=new Date();  var hour=date.getHours();

**27，请选择结果为真的选项（    ）**

　　A．null == undefined

　　B．null === undefined

　　C．undefined == false

　　D．NaN == NaN

**28, Math.ceil(-3.14)的结果是（    ），Math.floor(-3.14) 的结果是（    ）**

　　A．-3.14     B．-3     C．-4     D．3.14

**29, 阅读以下代码，在页面中结果是（    ）**

　　var  s="abcdefg";

　　alert(s.substring(1,2));

　　A．a     B．b     C．bc     D．ab

**30,  以下 ECMAScript 变量命名格式正确的是（    ）**

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

**二、问答题（每题8分）**

　　1，找到数组 [-1,-2,1,10,4,5,8] 中的最大值，至少写出两种方法？

　　2，封装一个函数，将字符串 "miao-wei-ke-tang" 从第二个单词开始首字母大写，然后拼成字符串 miaoWeiKeTang，并返回。（注意：封装成一个函数）

　　3 , 封装一个函数，将字符串 "      miao  v        " 实现 trim 的功能，但不使用 trim 方法，去掉字符串前后空格，返回处理后的字符串。

　　4，写一个方法，找出字符串 "abcabcabcabcabcabda" 中 "ab" 出现的次数和位置。

　　5 ,请用 js 代码完成下列需求：body 里生成 100 个 div，每个 div 宽 100px，高 100px，内容是 1-100 自身序号，颜色按红、黄、蓝、绿交替变色，10行10列排列。

**- 答案如下 -**

**一、选择题答案：（每题2分，多选题错选、少选不得分）**

　　1、B

　　2、CD

　　3、B

　　4、B

　　5、C

　　6、B

　　7、ABCDE

　　8、B

　　9、B

　　10、D

　　11、C

　　12、第一次弹（ A ）第二次弹（ D ）

　　13、B

　　14、A

　　15、C

　　16、B

　　17、C

　　18、D

　　19、ABC

　　20、第一次弹（ A ），第二次弹（ A ），第三次弹（ B ）

　　21、第一次弹（ D ），第二次弹（ D ）

　　22、arr 的结果为（ A ），arr2 的结果为（ C ）

　　23、A

　　24、按顺序分别弹出（ B ），（ A ），（ B ）

　　25、C

　　26、D

　　27、A

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

　　29、B

　　30、A

**二、问答题答案：（每题8分）**

　　1、找到数组 [-1, -2, 1, 10, 4, 5, 8] 中的最大值，至少写出两种方法？

　　var arr = [-1, -2, 1, 10, 4, 5, 8];

　　// 第一种

　　var max1 = Math.max.apply(null, arr);

　　// 第二种

　　var max2 = arr.sort(function(a, b){

　　return b - a;

　　})[0];

　　// 第三种

　　var max3 = -Infinity;

　　for (var i = 0; i < arr.length; i++) {

　　if (max3 < arr[i]) {

　　max3 = arr[i];

　　}

　　}

　　2、封装一个函数，将字符串 "miao-wei-ke-tang" 从第二个单词开始首字母大写，然后拼成字符串 miaoWeiKeTang，并返回。（注意：封装成一个函数）

　　var str = 'miao-wei-ke-tang';

　　function toTuoFeng(str) {

　　var arrStr = str.split('-');

　　for (var i = 1; i < arrStr.length; i++) {

　　arrStr[i] = arrStr[i].substr(0, 1).toUpperCase() + arrStr[i].substr(1);

　　}

　　return arrStr.join('');

　　}

　　console.log(toTuoFeng(str));

　　3、封装一个函数，将字符串 "      miao  v        " 实现 trim 的功能，但不使用 trim 方法，去掉字符串前后空格，返回处理后的字符串。

　　var str = '   miao v         ';

　　// 第一种方法

　　function trim(str) {

　　var start, end;

　　for (var i=0; i < str.length; i++) {

　　if (str[i] !== '') {

　　start = i;

　　break;

　　}

　　}

　　for (var i = str.length-1; i >= 0; i--) {

　　if (str[i] !== '') {

　　end = i;

　　break;

　　}

　　}

　　return str.substring(start, end + 1);

　　}

　　// 第二种方法

　　function trim(str) {

　　var start = 0, end = str.length - 1;

　　while(start < end && str[start] == '') {

　　start++;

　　}

　　while(start < end && str[end] == '') {

　　end--;

　　}

　　return str.substring(start, end + 1);

　　}

　　console.log(trim(str));

　　4、写一个方法，找出字符串 "abcabcabcabcabcabda" 中 "ab" 出现的次数和位置。

　　var str = 'abcabcabcabcabcabda';

　　var arr = [];

　　var n = 0;

　　while(str.indexOf('ab',n) != -1 && n < str.length) {

　　arr.push(str.indexOf('ab', n));

　　n = str.indexOf('ab', n) + 2;

　　}

　　console.log(arr);

　　5、请用 JS 代码完成下列需求：body 里生成 100 个 div，每个 div 宽 100px，高 100px，内容是 1-100 自身序号，颜色按红、黄、蓝、绿交替变色，10 行 10 列排列。

　　- CSS -

　　.box { position:relative; }

　　.box div { width: 60px; height: 60px; position: absolute; }

　　- HTML -

　　< div class="box"></div>

　　- JavaScript -

　　var box = document.getElementsByTagName('div')[0];

　　var str = '';

　　var arr = ['red', 'yellow', 'blue', 'green'];

　　for (var i = 0; i < 100; i++) {

　　str += ' < div style="left:' + i%10\*60 + 'px; top:' + Math.floor(i/10)\*60 + 'px; background:' + arr[i%arr.length] + ';">' + (i+1) + '</div>';

　　}

　　box.innerHTML += str;

　　当然答案还有很多，我们只是提供了其中几种