# 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  
  
2、以下关于 Array 数组对象的说法不正确的是（B  ）  
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（  A CED ）  
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（ D   ）  
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（ B   ）  
    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   ）  
A．2     B．1     C．3     D．undefined  
  
11、[1,2,3,4].join('0').split('') 的执行结果是（  D  ）  
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() );  
A．1       
B．alert(1);       
C．function fn1() { alert(1); }        
D．undefined  
  
13，以下代码运行后，结果为（  D  ）  
    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（ C   ）  
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（ B   ）  
A．true     B．false  
  
17，下面的描述中不正确的是（  C  ）  
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  ）  
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（   A ），第二次弹（ A   ），第三次弹（ B   ）  
    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（    ）  
    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  ），（   A ），（  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  
  
26，在JavaScript中，下列（  D  ）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（ c   ）  
A．null == undefined  
B．null === undefined  
C．undefined == false  
D．NaN == NaN   
  
28, Math.ceil(-3.14)的结果是（ c   ），Math.floor(-3.14) 的结果是（  B  ）  
A．-3.14     B．-3     C．-4     D．3.14  
  
29, 阅读以下代码，在页面中结果是（  c  ）  
    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（  c  ）  
A．\_125dollor     B．1207A     C．-dollor     D．this

# 二 填空题

1 js中强制类型转换为number类型的方法有:\_\_\_toNumber\_\_\_\_. \_\_\_\_parseInt()\_\_\_\_.\_\_\_\_\_使用运算符减、乘、除\_\_\_\_\_\_\_.

2 获取所有子元素节点: \_\_document.getElementById().children\_\_\_\_\_\_.获取父节点\_\_\_\_\_\_document.getElementById().parentNode\_\_\_\_\_\_\_.

3 把字符串按照指定字符分割并返回数组的方法为: \_\_split()\_\_\_\_.查询字符a在字符串‘fsdfsadf’中位置索引为：\_\_\_\_\_5\_\_\_\_\_\_\_.

4 有x = 10,y=49，写出生成x到y直接随机数的方法：\_\_\_\_\_\_\_parseInt(Math.random()\*(49-10+1) + 10);\_\_\_\_\_­­\_\_\_\_\_\_\_\_.

5 js中的多次执行的定时器为：\_\_setInterval()\_\_\_\_\_\_\_\_\_\_.

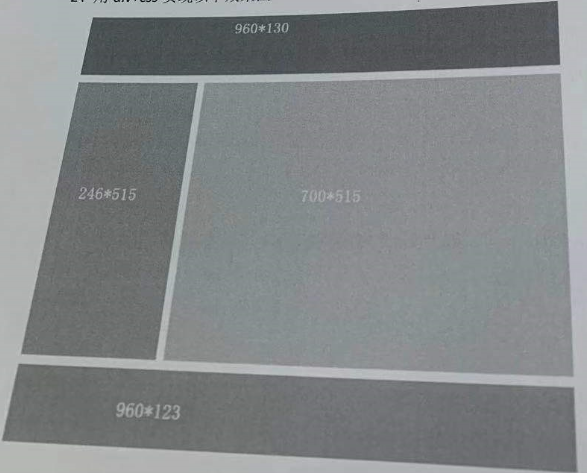
只执行一次的定时器为：\_\_setTimeout()\_\_\_

清除定时器的方法：\_\_\_\_\_clearInterval();\_\_\_clearTimeout()\_\_\_\_.

6 js获取焦点事件为：\_\_\_\_onfocus()\_\_\_\_\_\_\_\_.失去焦点事件为：\_\_onblur()\_\_\_\_\_\_\_\_\_\_.

# 三 代码题

1 用div + CSS 实现以下效果



<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        body{

            position: relative;

        }

        .a{

            width: 960px;

            height: 130px;

            background-color: red;

        }

        .b{

            width: 246px;

            height: 515px;

            background-color: aqua;

            margin-top: 10px;

        }

        .c{

            position: absolute;

            width: 700px;

            height: 515px;

            background-color: blueviolet;

            top: 140px;

            left: 260px;

        }

        .d{

            width: 960px;

            height: 123px;

            background-color: brown;

            margin-top: 10px;

        }

    </style>

</head>

<body>

    <div class="a"></div>

    <div class="b"></div>

    <div class="c"></div>

    <div class="d"></div>

</body>

</html>

2 现有数组 var arr = [0,2,3,6,6,0,2],请编写一个方法，去掉其中的重复元素。

function fn( arr) {

return arr.filter(function(item,index,arr){

return arr.indexOf(item,0)===index

})

}

Var Arr = [0,2,3,6,6,0,2]

Console.log(fn(Arr))

3 请我JS代码把数组 arr = [1,23,12,7,3,28,2,14]按正序的方式排序。

function sort(arr){

for(var i=0;i<arr.length-1;i++){

for(var j=0;j<arr.length-1-i;j++){

if(arr[j] > arr[j+1]){

var swap=arr[j];

arr[j]=arr[j+1];

arr[j+1]=swap;

}

}

}

}

var arr=[1,23,12,7,3,28,2,14]

4 请使用原生DOM对象的方法，实现如下操作，创建一个内容为“姓名5”的p标签并插入到“姓名6”的p标签上面，同时删除最后一个p标签（注：不确定有多少个p标签）

<div id="a">

    <p>姓名 1</p >

    <p>姓名 2</p >

    <p>姓名 3</p >

    <p>姓名 4</p >

    <p>姓名 6</p >

<p>姓名 7</p >

……………

  </div>

# 四 简答题

1 .call()和.apply()的含义和区别?

* apply：调用一个对象的一个方法，用另一个对象替换当前对象。例如：B.apply(A, arguments);即A对象应用B对象的方法。
* call：调用一个对象的一个方法，用另一个对象替换当前对象。例如：B.call(A, args1,args2);即A对象调用B对象的方法。
* call可以传入多个参数；
* apply只能传入两个参数，所以其第二个参数往往是作为数组形式传入

2 谈谈对原型和闭包的理解？

原型是一个对象，其他对象可以通过它实现属性继承。

闭包就是能够读取其他函数内部变量的函数。  
有两种存在形式：函数作为返回值时 和 函数作为参数传递时  
要理解闭包，首先理解javascript特殊的变量作用域，变量的作用于无非就是两种：全局变量，局部变量。而javascript语言的特殊处就是函数内部可以读取全局变量。

3 简单介绍一下promise,他解决了什么问题？

Promise 解决了地狱回调问题

4 普通函数和箭头函数this指向说明？

普通函数中this指向window 箭头函数没有this指向 一般是指向上下文

5 let var const的区别是什么？

Let Const 是ES6 中的语法Let是定义变量的 const是定义常量的 Var 是ES6之前的语法定义变量的

6 说一下es6的新特性和es6新增的数组方法不少于6个？

let和const具有块级作用域，不存在变量提升的问题。新增了箭头函数，简化了定义函数的写法，同时可以巧用箭头函数的this、（注意箭头函数本身没有this,它的this取决于外部的环境），新增了promise解决了回调地域的问题，新增了模块化、利用import 、export来实现导入、导出。新增了结构赋值，ES6 允许按照一定模式，从数组和对象中提取值，对变量进行赋值，这被称为解构（Destructuring）。新增了class类的概念，它类似于对象。

7 ==和===有什么区别？JavaScript基本数据类型有哪些？

全等和半等的区别在于 除了值相等还需要值的类型相等

基本数据类型有 字符串 数字 布尔值 undefined null

8 什么是事件委托？请写出一个事件委托？

事件委托又叫事件代理，事件委托就是利用事件冒泡，只指定一个事件处理程序，就可以管理某一类型的所有事件。

window.onload = function(){

var oul = document.getElementById("ul1");

oul.onclick = function(ev){

var ev = ev || window.event;

var target = ev.target || ev.srcElement;

if(target.nodeName.toLowerCase()=="li"){

console.log(target.innerHTML);

}

}

}

9 null 和 undefined的区别？

Null表示此处没有对象，就是不应该有值

Undefined表示此处应该有值但是值没有定义

10 HTML5有哪些新特性？

语义化标签、增强型表单、增加了视频和音频等等

11 跨越问题如何解决？

jsonp: 利用 <script> 标签没有跨域限制的漏洞，网页可以得到从其他来源动态产生的 JSON 数据。JSONP请求一定需要对方的服务器做支持才可以。JSONP优点是简单兼容性好，可用于解决主流浏览器的跨域数据访问的问题。缺点是仅支持get方法具有局限性,不安全可能会遭受XSS攻击。

第二种就是通过CORS：跨域资源共享（CORS）是一种机制；当一个资源访问到另外一个资源(这个资源放在不同的域名或者不同的协议或者端口)，资源就会发起一个跨域的HTTP请求需要浏览器和服务器同时支持

12 ajax工作流程以及原理？

**Ajax的原理简单来说通过XmlHttpRequest对象来向服务器发异步请求，从服务器获得数据，然后用javascript来操作DOM而更新页面。这其中最关键的一步就是从服务器获得请求数据**

**Ajax的实现步骤：**

①：浏览器让xhr去跟服务器要点儿数据

②：浏览器接着干别的事情

③：xhr去向服务器请求数据

④：服务器返回数据给xhr

⑤：xhr通知浏览器数据回来了

⑥：浏览器收到xhr返回的数据渲染页面