# 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（ C ）  
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)  
  
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（ D）  
A．2     B．1     C．3     D．undefined  
  
11、[1,2,3,4].join('0').split('') 的执行结果是（  C  ）  
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（  B  ）  
    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 ）  
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（  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（  A ，B ）  
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（ 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（  C  ）  
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（   A ）  
A．null == undefined  
B．null === undefined  
C．undefined == false  
D．NaN == NaN   
  
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  
  
29, 阅读以下代码，在页面中结果是（   B ）  
    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  ）  
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:Number（）

\_\_. parseInt(). parseFloat().

2 获取所有子元素节点:parentNode.children().获取父节点 node. parents().

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：\_\_\_indexof(‘a’)\_\_\_\_\_\_\_\_\_.

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

5 js中的多次执行的定时器为：\_\_\_\_setInterval(()=>{},1000)\_\_\_\_\_\_\_\_.

只执行一次的定时器为：\_\_\_\_setTimeout(()=>{},1000)\_\_\_\_\_\_\_\_.

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>

      .main {

        width: 960px;

      }

      .head {

        width: 100%;

        height: 130px;

        background-color: rgb(180, 181, 182);

      }

      .body {

        margin: 14px 0;

        display: flex;

        height: 515px;

      }

      .body1 {

        width: 246px;

        margin-right: 14px;

        background-color: rgb(130, 131, 133);

      }

      .body2 {

        width: 700px;

        background-color: rgb(120, 120, 121);

      }

      .foot {

        width: 100%;

        height: 123px;

        background-color: rgb(42, 42, 43);

      }

    </style>

  </head>

  <body>

    <div class="main">

      <div class="head">1</div>

      <div class="body">

        <div class="body1">2</div>

        <div class="body2">3</div>

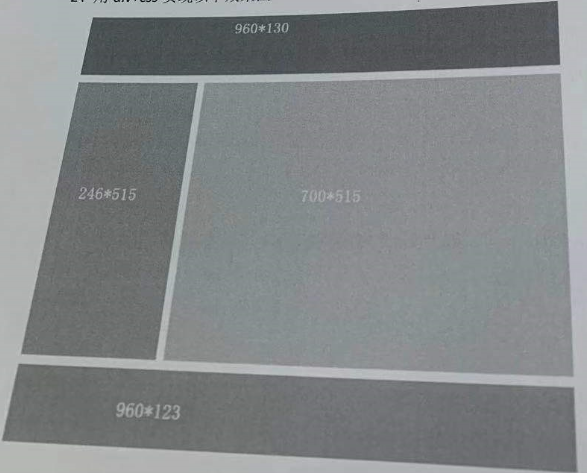
      </div>

      <div class="foot">4</div>

    </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。

arr = new Set(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。

arr.sort(function (a, b) {

        return a - b

      })

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>

 <script>

      var div = document.getElementById('a')

      var p = document.getElementsByTagName('p')

      var name5 = document.createElement('p')

      name5.innerHTML = '姓名5'

      div.insertBefore(name5, p[4])

      div.removeChild(p[p.length - 1])

    </script>

# 四 简答题

1 .call()和.apply()的含义和区别?

call()和apply()都会改变函数的this指向，通常用在改变函数this的指向，call传递参数使用逗号隔开,apply使用数组传递

call 经常做继承.

apply经常跟数组有关系. 比如借助于数学对象实现数组最大值最小值

2 谈谈对原型和闭包的理解？

原型：每一个实例对象上有一个**proto**属性，指向的构造函数的原型对象，构造函数的原型

闭包：当一个函数访问另外一个函数的变量时，就会形成闭包，闭包的作用是延长变量的生命

3 简单介绍一下promise,他解决了什么问题？

promise解决了回调地狱的问题，promise是用于解决异步执行程序的，通常搭配.then来使用，promise会返回一个异步函数

4 普通函数和箭头函数this指向说明？

普通函数的this指向window，

箭头函数里没有this指向，通常指向上下文

5 let var const的区别是什么？

Let和const具备块级作用域，不具备变量提升，const不能修改值，let可以修改

Var具备变量提升，可以修改值，没有块级作用域

6 说一下es6的新特性和es6新增的数组方法不少于6个？

ES6是2015年推出的一个新的版本、这个版本相对于ES5的语法做了很多的优化、例如：新增了let、const

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。

数组新方法：

1. forEach，用来遍历数组，，不会反悔一个新数组
2. Map,遍历数组会得到一个新的数组
3. Filter，遍历数组，返回一个新数组，一般用于条件赛选
4. Some ，遍历数组，返回一个布尔值，一般用于赛选条件
5. Every，遍历数组，赛选的条件都满足时，返回true，只要一个不满足，返回false
6. ...，把伪数组编程数组，
7. set数组去重

7 ==和===有什么区别？JavaScript基本数据类型有哪些？

==：当等号两边的类型相同时，直接比较值是否相等，若不相同，则先转化为类型相同的值，再进行比较；

===：称为等同符，当两边值的类型相同时，直接比较值，若类型不相同，直接返回false；

基本数据类型：number，boolean ，null，undefined，string

8 什么是事件委托？请写出一个事件委托？

子级委托给父级处理事件，利用事件冒泡的原理，一般称为事件委托

子级li委托给父节点ul处理点击事件

$(‘ul’).on(‘click’,’li’,function(){

Alert(1)

})

9 null 和 undefined的区别？

Null为定义变量且值为null

undefined为定义变量未赋值

10 HTML5有哪些新特性？

（一）  语义标签

（二）增强型表单

（三）视频和音频

（四）Canvas绘图

（五）SVG绘图

11 跨越问题如何解决？

1. jsonp利用script标签不受跨域的限制，简单兼容性好，可以解决，缺点不能使用post请求
2. Cors，后端在请求头时，Access-Control-Allow-Origin，设置一些数据

12 ajax工作流程以及原理？

Ajax的工作原理相当于在用户和服务器之间加了—个中间层(AJAX引擎)，使用户操作与服务器响应异步化。并不是所有的用户请求都提交给服务器。像—些数据验证和数据处理等都交给Ajax引擎自己来做,，只有确定需要从服务器读取新数据时再由Ajax引擎代为向服务器提交请求。

var xhr = new XMLHttpRequest()

xhr.open('get', 'http://www.liulongbin.top:3006/api/getbooks')

xhr.send()

xhr.onreadystatechange = function () {

if (xhr.readyState == 4 && xhr.status == 200) {

console.log(xhr)

console.log(xhr.responseText)｝｝