HTML5笔试题

HTML4

1. 元素的Alt和title有什么异同，选出正确的说法？不同的浏览器，表现一样 ( )

A、Alt和title同时设置的时候，Alt作为图片的代替文字出现title是图片的解释文字

B、Alt和title同时设置的时候，title作为图片的代替文字出现，Alt是图片的解释文字

C、以上说法都不正确

1. 想要找到表单中的hidden元素，下面哪个是正确的？（单选）

A、visible B、hidden C、visible() D、hidden()

1. 介绍一下你对浏览器内核的理解？
2. 说下行内元素和块级元素的区别？行内块元素的兼容性使用？

IE6/7的inline-block属性只对原生默认的内联元素(span, a, em......)有效，对于块级元素(div,p,ul......)无效。

div {display:inline-block;...}

div {display:inline;}

div {display:inline; zoom:1;...}

1. What is doctype?Why do you need it?
2. Can you apply css rule to a part of html document?
3. What are you the differences between div and span?
4. Does css properties are case sensitive ?
5. Does margin-top or margin-bottom has effect on inline element?
6. Does padding-top or padding-bottom has effect on inline element?
7. How absolute ,relative ,fixed and static position differ?
8. 列出清除浮动的几种方法(至少3种)？
9. B标签超过一定长度，以省略号显示？
10. 用css绘制一个箭头指向右的三角形？
11. iframe的使用场景有?( )

A、与第三方域名下的页面共享cookie;

B、上传图片，避免当前页刷新;

C、左边固定右边自适应的布局：

D、资源加载;

1. 通常我们在浏览器内容区看到的内容，都是放入在 ( ) 之内?

A、<!doctype html>

B、<html></html>

C、<head></head>

D、<body></body>

1. 在网站上添加链接使用的标签是( )

A、title标签;

B、img标签;

C、meta标签;

D、a标签;

1. 通过分析如下HTML代码，可以得出（）（多选）

<table border="10">

<tr>

<td colsapn="2" align="center">员工号</td>

</tr>

<tr>

<td colsapn="2" align="center">学历</td>

<td align="center">专业</td>

</tr>

<tr>

<td colsapn="2" align="center">毕业院校</td>

</tr>

</table>

A、该表格共有两行三列

B、该表格中文字均居中显示

C、该表格边框宽度为10毫米

D、“员工号”单元跨两列

1. CSS中的\_\_\_\_\_\_\_\_\_\_\_\_属性可谓元素设置外边距，改变元素的内填充用\_\_\_\_\_\_\_\_\_\_\_\_属性。
2. 对ul li的样式设成无，应该用什么属性\_\_\_\_\_\_\_\_\_\_\_\_。
3. 合理的页面布局中常听过结构与表现分离，那么结构是\_\_\_\_\_\_\_\_\_\_\_\_，表现是\_\_\_\_\_\_\_\_\_\_\_\_。
4. 在Table中，TR是\_\_\_\_\_\_\_\_\_\_\_\_，TD是\_\_\_\_\_\_\_\_\_\_\_\_。
5. 下面标签嵌套正确的是（D）

A. <ul><p>赶集网</p></ul>

B. <a href="#"><a href="#">赶集网</a></a>

C. <dl><li>赶集网</li></dl>

D. <ol><li>赶集网</li></ol>

1. 在HTML中，（C）可以在网页上通过链接打开邮件客户端发送邮件

A. <a href="telnet:zhou@126.com">发送邮件</a>

B. <a href="mail:zhou@126.com">发送邮件</a>

C. <a href="mailto:zhou@126.com">发送邮件</a>

D. <a href="ftp:zhou@126.com">发送邮件</a>

1. 请选择所有的置换元素（ABCD）

A. img B. input C.textarea D.select

1. 下面哪条声明能固定背景图片（A）

A. background-attachment:fixed;

B. background-attachment:scroll;

C. background-origin:initial;

D. background-clip:initial;

1. 下列说法正确的是（AB）（多选）

A. display:none; 不为被隐藏的对象保留其物理空间

B. visibility:hidden; 所占据的空间位置仍然存在，仅为视觉上的完全透明

C. visibility:hidden; 产生reflow和repaint（回流与重绘）

D. visibility:hidden; 与displa:none;两者没有本质上的区别

1. 以下关于盒子模型描述正确的是（A ）

A. 标准盒子模型中:盒子的总宽度=左右margin + 左右border + 左右padding + width

B. IE盒子模型中：盒子总宽度 = 左右margin + 左右border + width

C. 标准盒子模型中：盒子的总宽度 = 左右margin + 左右border + width

D. IE盒子模型中：盒子总宽度 = width

1. 英文字母全部转为大写正确的是（C）

A. text-transform:capitalize;

B. text-transform:lowercase;

C. text-transform:uppercase;

D. font-weight:bold;

1. 列举W3C推荐的属性标签，说一下p标签和img标签的特点。
2. 你做的页面在哪些浏览器测试过？这些浏览器的内核分别是什么？经常遇到的浏览器的兼容性有哪些？怎么会出现？解决的方法是什么？
3. 分别写出3个块级元素和内联元素。
4. 解释以下4个标签的含义strong em b i
5. Disabled与readonly的区别
6. 行内元素有哪些？块级元素有哪些？空元素有哪些？
7. 定义链接四种状态的伪类的正确书写顺序是?
8. 你知道的css选择器有哪些
9. 下面两个div的间距应该是多少？为什么？

<div style=”margin:5px”></div>

<div style=”margin:10px”></div>

1. png8、24的区别是？
2. Png和jpg这两种图片格式分别适用的场景是？
3. 请谈一下你对网页标准和标准制定机构重要性的理解
4. 简述一下src和href的区别

HTML5

1. 请描述一下cookies,sessionStorage和localStorage的区别？
2. html5有哪些新特性？如何处理HTML5新标签的浏览器兼容性问题？如何区分HTML和HTML5？
3. 简述一下你对HTML语义化的理解？
4. HTML5的离线缓存，描述一下cookies。
5. 简述Doctype的作用；区分严格模式与混杂模式有何意义？如何触发这两种模式？
6. HTML5为什么只需要写<Doctype html>？
7. html5新增哪些标签？

header、footer、article、section、nav、hgroup、datalist、output、mark、audio、video

1. html5新增表单元素有哪些(至少5个)？

时间类型：date、dataTime、dataTime-local、month、time、week

非时间类型：email、color、range、file、number、tel

1. 有没有关注HTML5和CSS3？如有请简单说一些您对他们的了解情况！
2. H5新增的标签？
3. H5新增的标签类元素标签？
4. 描述一下常见存储机制的区别？
5. 使用HTML5中的本地存储保存一个json对象？

CSS2

1. 下列哪个样式定以后，内联（非块状）元素可以定义宽度和高度（）

A、display:inline

B、display:none

C、display:block

D、display:inherit

1. CSS单位中px和em分别代表什么？其中em是如何计算的？
2. 请用DIV+CSS实现一个3列布局，要求：左右两列宽度固定，为100像素，中间列宽度为页面剩余宽度。即：中间列宽度=页面宽度-左侧宽度-右侧宽度。
3. 简述CSS优先级。
4. Position 的值有哪些？Relative和absolute的定位原理是什么？
5. 简述下为什么要用css reset。
6. 列出几个常用的meta。
7. 请写出必要的HTML/CSS，B元素水平垂直居中，A不固定（A是B的父元素）。
8. [填空]为div同时设置两个样式a和b，应如何编写HTML代码\_\_\_\_\_\_\_\_\_\_\_\_\_
9. [填空]请在样式表中填写内容，使该元素字体大小为20像素

<div class=”title” style=”font-size:12px;”>ASCII</div>

.title{ \_\_\_\_\_\_\_\_\_\_\_\_\_}

1. Position的值relative和absolute定位原点是？
2. 经常遇到的浏览器的兼容性有哪些(列举3-4个)？原来，解决方法是什么？常用的hack的技巧。
3. 用CSS如何实现下图的布局？

<div class=”first”>

<div class=”second”>

<div class=”third”>

1. 请指出以下结构中A标签内的字体颜色值？

<style>

a{color:#ccc;}

#container a{color:#f60;}

h1 a{color:#eee}

</style>

<div id="container">

<h1 class="link"><a href="http://itouzi.com">爱投资</a></h1>

</div>

1. 在空白处加行css，让这个div在页面居中（上下左右都居中）

<div style="border:1px solid red;width:200px;height:200px;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_”>Hello 爱投资！！</div>

1. 页面导入样式时，使用link和@import有什么区别？
2. 常用哪几种浏览器测试？有哪些内核（Layout Engline）？
3. 清除浮动有哪些方式？比较好的是哪一种？
4. 使用CSS来格式化网页，共有三种方式，一下哪一种不是? ( )

A、在HEAD中引用;

B、作为标记来使用;

C、在BODY中引用;

D、作为文件来引用;

1. 下述有关border : none以及border : 0的区别，描述错误的是? ( )(多选)

A、border:none表示边框样式无;

B、border:0 表示边框宽度为0;

C、当定义了border : none，即隐藏了边框的显示，实际就是边框宽度为0;

D、当定义边框时，仅设置边框宽度也可以达到显示的效果;

1. 清除浮动的常用方法？zoom实现的原理？
2. 如何让img标签在div中上下居中？
3. CSS选择符有哪些？那些属性可以继承？优先级算法如何计算？内联和 important 那个优先级高？
4. css引入的方式有哪些？link和@import的区别是？
5. 介绍一下css的盒子模型
6. 解释css sprites,它的作用是什么？
7. display: none和 visibility: hidden的区别？
8. css中line-height：2和line-height：200%的区别？
9. 用css绘制一个箭头向右的三角形？
10. 如何居中div，如何居中一个浮动元素？
11. 你怎么实现页面设计图？你认为前端应如何高质量完成工作？一个满屏品字布局如何设计？
12. Absolute的containing block计算方式与正常流有什么不同？
13. 简述你对BFC规范的理解。
14. 什么叫”优雅降级”和”渐进增强”
15. 写出几种IE6的BUG的解决方法？
16. 如何用纯css创建一个三角形？
17. :before和::before的区别？
18. 一个高度位置的图片如何在一个比它大容器内水平垂直居中

CSS3

1. 简单描述响应式页面的理解以及在css中如何使用。
2. 怎么让Chorme支持小于12px的文字?
3. box-sizing 常用的属性有哪些？分别有什么作用？
4. 现有一个表格，如果想要匹配表格的偶数行，用\_\_\_\_\_\_\_实现，奇数行用\_\_\_\_\_实现。
5. 简述一下rem和em的区别
6. 请简述一下你对响应式布局的理解

原生JavaScript

1. 放在HTML里的哪一部分javascript会在页面加载的时候被执行了?( )

A .文件头部位置 B.文件尾 C.<head>标签部分 D.<body>标签部分

1. 在以下的HTML中，哪个是外部样式或js文件的正确引用方法？（）

A、<style src=”mystyle.css”>

B、<link type=”text/css” rel=”stylesheet” src=” mystyle.css”/>

C、@import url(myscript.js);

D、<script type=”text/javascript” src=”myscript.js”></script>

1. 声明一个变量,给它加上name属性和show方法显示其name的值,以下代码正确的是()

A、var obj={name:”zhangsan”,show: “alert(this.name);”};

B、var obj={name:”zhangsan”,show:function(){alert(name);}};

C、var obj={name:”zhangsan”,show:function(){alert(this.name);}};

D、var obj=[name:”zhangsan”,show:function(){alert(name);}];

1. 分析下面的js代码块

var a=new Array(2,4,4,5,6);

var sum=0;

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

sum+=a[i];

document.write(sum); 输出的结果是（）

A、21 B、24456 C、2,4,4,5,6 D、19

1. 请选择结果为真的表达式（）

A、null instanceof Object B、null==undefined

C、null===undefined D、NaN==NaN

1. 在js里，下列选项中不属于数组方法的是（）

A、sort() B、length() C、concat() D、reverse()

1. var emp=new Array(3); For(var i in emp)

以下代码中能与for循环代码互换的是（）

A、for(var i=0;i<emp;i++)

B、for(var i=0;i<Array(3);i++)

C、for(var i=0;i<emp.length();i++)

D、for(var i=0;i<emp.length;i++)

1. 如要将各项成绩之和放到总分框中，关键的JavaScript代码应该是？（）

A、zongfen.value = yuwen.value + shuxue.value

B、document.form1.zhongfen.value = document.from1.yuwen.value + document.form1.shuxue.value

C、document.form[1].zhongfen.value = document.from[1].yuwen.value + document.form[1].shuxue.value

D、document.zhongfen.value = document.yuwen.value + document.shuxue.value

1. javascript中表达式parseInt(“8x8”)+parseFloat(“8”)的结果是什么？（）

A、88 B、96 C、16 D、8X88

1. 在表单（form1）中有一个文本框元素（fname），用于输入电话号码，格式如：010-82668155，要求前3位是010紧接着一个“-”，后面是8位数字，要求在提交表单时，根据上述条件验证该文本框中输入内容的有效性，下列语句中，（ ）能正确实现以上功能

A、var str = form1.fname.value;

if(str.substr(0,4)!=”010-” &&str.substr(4).length!=8 && isNaN(parseFloat(str.substr(4)))) alert(“无效的电话号码“)；

B、var str = form1.fname.value;

if(str.substr(0,4)!=”010-” || str.substr(4).length!=8 || isNaN(parseFloat(str.substr(4)))) alert(“无效的电话号码“)；

C、var str = form1.fname.value;

if(str.substr(0,3)!=”010-” || str.substr(3).length!=8 || isNaN(parseFloat(str.substr(3)))) alert(“无效的电话号码“)；

D、var str = form1.fname.value;

if(str.substr(0,3)!=”010-” && str.substr(3).length!=8 && isNaN(parseFloat(str.substr(3)))) alert(“无效的电话号码“)；

1. 下列JS可以让一个input的背景变成红色的是？（）

A、inputElement.style.backgroundColor=”red”;

B、inputElement.backgroundColor=”red”;

C、inputElement.style.backgroundColor=”#0000”;

D、inputElement.backgroundColor=”#0000”;

1. 页面有一个按钮button，id为button1，通过原生js如何禁用？(多选)

A、document.getElementById(“button1”).readonly = true;

B、document.getElementById(“button1”).setAttribute(“readonly”,”true”);

C、document.getElementById(“button1”).disabled = true;

D、document.getElementById(“button1”).setAttribute(“disabled”,”true”);

1. 下面有关javaScript中call和apply的描述，错误的是？（）

A、call与apply都属于Function.prototype的一个方法，所以每个Function实例都有call，apply属性

B、两者传递的参数不同，call函数第一个参数都是要输入给当前对象的对象，apply不是

C、Apply传入的是一个参数数组，也就是将多个参数组合成为一个数组传入

D、Call传入的则是直接的参数列表。Call方法可将一个函数的对象上下文从初始的上下文改变为由thisObj制定的新对象

1. 创建一个对象obj，该对象包含一个名为”name”的属性，其值为”value”.以下哪一段JavaScript代码无法得到上述的结果？（单选）

A、var obj = new Object();

obj[“name”]=”value”;

B、var obj ={name:”value”};

C、var obj = new function(){

This.name = “value”;

}

D、var obj = new Object();

obj.prototype.name = “value”;

1. 正则表达式匹配，开头为“010-”，后面是7-8个数字的电话号码。

function isPhone(num){

var validPhone = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

var flag = num.match(validPhone)?true:false;

return flag

}

1. 请写出下面代码运行后的结果

var obj = {a:1,b:function(){console.log(this.a)}}

var a = 2;var objb = obj.b;

obj.b();

objb();

obj.b.call(window);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. 判断浏览器类型及设备的属性是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. 解释一下alert(Function instanceof Object);和alert(Object instanceof Function);的结果。
3. 写一段脚本，实现：当页面上任意一个链接被点击的时候，alert出这个链接在页面上的顺序号，如第一个链接则alert(1),依次类推。
4. 以下代码输出什么，为什么？(function(){var a=b=5;})();console.log(b);
5. 创建”内置”方法：给String对象定义一个repeatify方法，该方法接收一个整数参数，作为字符串重复的次数，最后返回重复指定次数的字符串。
6. 请简述javascript中如何避免全局变量污染？请给出简短的代码片段。
7. 已定义类SuperType如下，现在需要通过继承的方式定义SuperType的子类SubType，同时要求为SubType新增一个属性age，新增一个方法sayAge，请给出类SubType的实现。
8. 请简述setTimeout在web UI开发中的作用，以及如何使用setTimeout模拟setInterval。
9. 请为字符串增加trinerse函数，实现将字符串中数字删除后将剩余字符串逆序。
10. 写一个函数：统计从1到10亿之间的自然数中含有多少个2？例如1-20中，有2,12,20这三个自然数，有3个2。
11. 常见的排序都有那些？写出一种你常用的排序方法。
12. 写出下面代码的结果  
    foo() //?  
    function foo(){

Console.log(‘foo’) ;

}  
 -----------------------------  
 var foo = function bar(){  
 console.log(‘foo’);

}  
 foo(); //?  
 bar(); //?

1. 请描述以下两段代码在1000ms内共执行了多少次，并且描述setTimeout与setInterval的运行机制

setTimeout (function(){

setTimeout(argument.callee,100);

/\*代码块，执行时间150ms\*/

},100)  
 setInterval(function(){

/\*代码块，执行时间150ms\*/

},100)

1. 请写出下面代码执行后的结果

var a = 100;

function testResult(){

var b = 2 \* a;

var a = 200;

var c = a/2;

alert(b);

alert(c)

}

1. 请写出下面代码执行的结果

代码1：

for(var i=1;i<=5;i++){

setTimeout(function timer(){

console.log(j)

},j\*1000)

}

结果：undefined

代码2：

for(var i=1;i<=5;i++){

(function(j){

setTimeout(function timer(){

console.log(j);

},j\*1000);

})(i);

}

结果：1 2 3 4 5

代码3：

function foo(num){

console.log("foo:"+num);

this.count++;

}

foo.count=0;

var i;

for(i=0;i<10;i++){

if(i>5){

foo.call(foo,i);

}

}

console.log(foo.count)

结果：foo:6 foo:7 foo:8 foo:9

1. 请写出下面代码执行后的结果：

var a=100;

function testResult(){

var b=2\*a;

var a=200;

var c=a/2;

alert(b);

alert(c);

}

testResult();

答案：NaN 100

1. for..of和for..in的区别？
2. 完成一个函数，接受数组作为参数，数组元素为整数或者数组，数组元素包含整数或数组，函数返回扁平化后的数组，如：[1,[2,[[3,4],5],6]] = > [1,2,3,4,5,6]
3. 下面代码运行的结果是什么？

function printing(){

console.log(1)

setTimeout(function(){console.log(2);},1000);

setTimeout(function(){console.log(3);}，0)；

console.log(4)

}

printing();

1. 请问控制台上回输出什么？

(function(){var a = b = 5})()

console.log(b)

1. 什么是闭包，闭包的作用？
2. 有一个数组[3,4,1,2,5,6,6,5,4,3,3]请用javascript写一个函数，找出该数组中没有重复的数的总和。（上面的数据的没有重复的总和为1+2=3）
3. 实现id为div的元素以每秒20px 的速度右移100px，并且20px和100px可以调整。
4. 从0-99这100个数中随机取出10个，要求不能有重复，可以自己设计数据结构。
5. 跨域是什么？有什么常用的解决方法。
6. 以下代码alert出来是多少？

alert(1&&2);

alert(1||2);

1 && 2 : 2;

1 || 2 : 1;

js中&& 和 || 遵循”短路”原理, 在||中 只要遇到true的分项就停止处理,并返回该分项 而在&& 中 遇到false的分项就停止处理,并返回改分项 如果没有遇到false 则返回最后分项

1. 验证字符串的组成规则，第一个需为数字，后面可以是字母、数字，下划线总长度为3~50位。

/^[0-9]\w{3,50}$/

1. 以下代码alert出来的值是多少?

window.val = 1;

var json = {

val:10,

dbl:function(){

this.val+=2;

}

};

json.dbl();

var del = json.dbl;

dbl();

json.dbl.call(window);

alert(window.val+json.val);

报错 dbl is not defind();

dbl();直接执行的时候,意思是指在window对象下的方法 可是dbl既没有声明也没有function 因此汇报 dbl is not defind();

1. 以下代码alert出来的值是多少?

(function(){

var val = 1;

var json = {

val:10,

dbl:function(){

val\*=2

};

};

json.dbl();

alert(json.val+val);

}());

存在语法错误,在声明json对象的时候 对象内的属性值dbl之后不应接; 倘若去掉;则值为12,按照作用域链寻找规则 dbl方法内的val 从内而外,寻找最近作用域内的val 为自运行函数内的val声明 则为 1 \* 2 最后结果为 10 + 2 结果12

1. 以下代码alert的值

function C1(name){

if(name) this.name=name

}

function C2(name){

this.name=name

}

function C3(name){

this.name=name||"John"

}

C1.prototype.name='Tom';

C2.prototype.name='Tom';

C3.prototype.name='Tom';

alert((new C1().name)+(new C2().name)+(new C3().name))

C1函数的构造方法当中有if判断 new的时候没有传参 则值不会变 依然为字符串”Tom”

C2函数构造方法在new的时候没有传参,参数值为undefined 则c2的name值为undefined

C3函数的构造方法中,name赋值过程有||符 按照短路原理 第一个分项的值为undefined 相当于false 跳过,继续匹配 而第二个分项的值为有值的字符串 返回第二个分项 所以name值为”John”

最后在alert中进行字符串拼接 输出结果为 TomundefinedJohn

1. 以下代码，alert出的值是多少?

var test=(function(i){

return function(){

alert(i+2)

}

}(2));

test(5);

值为4 ;

变量test 后面赋值的最外层实际上是一个自运行函数 函数内return一个方法

最后test实际上是一个不接收参数的函数 只是该函数是接收的函数内部return出来的函数 故可以访问自运行函数传进来的参数i 即 2 最后结果为 2 + 2 结果为4 典型的闭包结构

1. 写一段代码，让id为seconds的容器里面的数字每秒少一，一直变到0

var secDom = document.getElementById(‘seconds’);

var timeObj = setInterval(function(){

var num = secDom.innerHTML\*1;

if(num != 0){

secDom.innerHTML = — num;

}else{

clearInterval(timeObj);

}

},1000);

1. 请写出以下输出结果

var Foo=function(){};

var foo=new Foo();

delete Foo.prototype

Foo.prototype.aaa=function(){alert("原型方法aaa")};

Foo.prototype={

"aaa":function(){alert("重写的aaa")},

"bbb":function(){alert("重写的bbb")}

}

foo.aaa();// 原型方法aaa

var foo2=new Foo();

foo2.aaa();// 重写的aaa

foo.constructor===Foo;//true

foo2.constructor===Foo;//false

foo2.constructor===Object;//true

Foo.prototype={

"constructor":Foo,

"aaa":function(){alert("第二次重写的aaa")},

"bbb":function(){alert("第二人重写的bbb")}

}

var foo3=new Foo();

foo3.aaa();//第二次重写的aaa

foo.constructor===Foo;//true

foo2.constructor===Foo;//false

foo3.constructor===Foo;//ture

1. 编写一个方法 去掉一个数组的重复元素

Array.prototype.unique2 = function(){

this.sort(); //先排序

var res = [this[0]];

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

if(this[i] !== res[res.length - 1]){

res.push(this[i]);

}

}

return res;

}

var arr = [1, 'a', 'a', 'b', 'd', 'e', 'e', 1, 0]

alert(arr.unique2());

1. 简答

Is ‘false’ is false?

no

Is ‘ ‘ is false?

no

What is typeof []?

object

What is the value of ‘6’+9

69

What is the value of 4+3+2+’1’

91

What is the value of ‘1’+2+4

124

1. 已知 arr1 = [‘a’,’b’,'c','d'],arr2=['x','b','c','y'],用js取两个数组相同的元素

var newList = [];

for(var i = 0; i < arr1; i++){

for(var j = 0; j < arr2; j ++){

if(arr1[i] == arr2[j]){

newList.push(arr1[i]);

}

}

}

1. 用正则表达式将下面字符串中的html标签去除

<span style='dispaly:none;'>This is test</span><img src=“”>ss</img><strong></strong>

var text = “<span style='dispaly:none;'>This is test</span><img src=‘’>ss</img><strong></strong>

”

var reg = />([^<]\*)/ig;

var result = reg.exec();

var text = “”;

while(result == null){

text += result[1];

result = reg.exec();

}

console.log(text);

1. JS是什么，JS和HTML的开发如何结合？

JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能。

通过在<script></script>标签中使用或者<script src=“XX.js”></script>引入

1. 怎样添加，移除，移动，复制，创建，查找节点？

（1）创建新节点

createDocumentFragment() //创建一个DOM片段

createElement() //创建一个具体的元素

createTextNode() //创建一个文本节点

（2）添加、移除、替换、插入

appendChild()

removeChild()

replaceChild()

insertBefore()

（3）查找

getElementsByTagName() //通过标签名称

getElementsByName() //通过元素的Name属性的值

getElementById() //通过元素Id，唯一性

1.创建元素节点

document.createElement() 方法 用于创建元素，接受一个参数，即要创建元素的标签名，返回创建的元素节点

1 var div = document.createElement("div"); //创建一个div元素

2 div.id = "myDiv"; //设置div的id

3 div.className = "box"; //设置div的class

创建元素后还要把元素添加到文档树中

2.添加元素节点

appendChild() 方法 用于向childNodes列表的末尾添加一个节点，返回要添加的元素节点

1 var ul = document.getElementByIdx("myList"); //获得ul

2 var li = document.createElement("li"); //创建li

3 li.innerHTML = "项目四"; //向li内添加文本

4 ul.appendChild(li); //把li 添加到ul子节点的末尾

appendChild() 方法还可以添加已经存在的元素，会将元素从原来的位置移到新的位置

1 var ul = document.getElementById("myList"); //获得ul

2 ul.appendChild(ul.firstChild); //把ul的第一个元素节点移到ul子节点的末尾

insertBefore() 方法，如果不是在末尾插入节点，而是想放在特定的位置上，用这个方法，该方法接受2个参数，第一个是要插入的节点，第二个是参照节点，返回要添加的元素节点

1 var ul = document.getElementById("myList"); //获得ul

2 var li = document.createElement("li"); //创建li

3 li.innerHTML= "项目四"; //向li内添加文本

4 ul.insertBefore(li,ul.firstChild); //把li添加到ul的第一个子节点前

1 var ul = document.getElementById("myList"); //获得ul

2 var li = document.createElement("li"); //创建li

3 li.innerHTML= "项目四"; //向li内添加文本

4 ul.insertBefore(li,ul.lastChild); //把li添加到ul的最后一个子节点（包括文本节点）之前

1 var ul = document.getElementById("myList"); //获得ul

2 var li = document.createElement("li"); //创建li

3 li.innerHTML= "项目四"; //向li内添加文本

4 var lis = ul.getElementsByTagName("li") //获取ul中所有li的集合

5 ul.insertBefore(li,lis[1]); 　　　　//把li添加到ul中的第二个li节点前

添加后:

3.移除元素节点

removeChild() 方法，用于移除节点，接受一个参数，即要移除的节点，返回被移除的节点，注意被移除的节点仍然在文档中，不过文档中已没有其位置了

1 var ul = document.getElementById("myList"); //获得ul

2 var fromFirstChild = ul.removeChild(ul.firstChild); //移除ul第一个子节点

1 var ul = document.getElementById("myList"); //获得ul

2 var lis = ul.getElementsByTagName("li") //获取ul中所有li的集合

3 ul.removeChild(lis[0]); 　　　　　　//移除第一个li，与上面不同，要考虑浏览器之间的差异

4.替换元素节点

replaceChild() 方法，用于替换节点，接受两个参数，第一参数是要插入的节点，第二个是要替换的节点，返回被替换的节点

1 var ul = document.getElementById("myList"); //获得ul

2 var fromFirstChild = ul.replaceChild(ul.firstChild); //替换ul第一个子节点

1 var ul = document.getElementById("myList"); //获得ul;

2 var li = document.createElement("li"); //创建li

3 li.innerHTML= "项目四"; //向li内添加文本

4 var lis = ul.getElementsByTagName("li") //获取ul中所有li的集合

5 var returnNode = ul.replaceChild(li,lis[1]); //用创建的li替换原来的第二个li

5.复制节点

cloneNode() 方法，用于复制节点， 接受一个布尔值参数， true 表示深复制（复制节点及其所有子节点）， false 表示浅复制（复制节点本身，不复制子节点）

1 var ul = document.getElementById("myList"); //获得ul

2 var deepList = ul.cloneNode(true); //深复制

3 var shallowList = ul.cloneNode(false); //浅复制

1. 截取字符串abcdefg中的efg?

str = “abcdefg”

str.substr(-1,3);

1. 写出程序的运行结果。

for(i=0,j=0;i<10,j<6;i++,j++){

k=i+j

}

k == 10

1. 实现一个遍历数组或者对象里所有成员的迭代器。

for(var i in obj){

console.log(obj[i]);

}

1. 写出三个javascript中使用this关键字的典型应用场景,谈谈对This对象的理解

情况一：纯粹的函数调用

这是函数的最通常用法，属于全局性调用，因此this就代表全局对象Global。

请看下面这段代码，它的运行结果是1。

复制代码代码如下:

　　function test(){

　　　　this.x = 1;

　　　　alert(this.x);

　　}

　　test(); // 1

为了证明this就是全局对象，我对代码做一些改变：

复制代码代码如下:

　　var x = 1;

　　function test(){

　　　　alert(this.x);

　　}

　　test(); // 1

运行结果还是1。再变一下：

复制代码代码如下:

　　var x = 1;

　　function test(){

　　　　this.x = 0;

　　}

　　test();

　　alert(x); //0

情况二：作为对象方法的调用

函数还可以作为某个对象的方法调用，这时this就指这个上级对象。

复制代码代码如下:

　　function test(){

　　　　alert(this.x);

　　}

　　var o = {};

　　o.x = 1;

　　o.m = test;

　　o.m(); // 1

情况三 作为构造函数调用

所谓构造函数，就是通过这个函数生成一个新对象（object）。这时，this就指这个新对象。

复制代码代码如下:

　　function test(){

　　　　this.x = 1;

　　}

　　var o = new test();

　　alert(o.x); // 1

运行结果为1。为了表明这时this不是全局对象，我对代码做一些改变：

复制代码代码如下:

　　var x = 2;

　　function test(){

　　　　this.x = 1;

　　}

　　var o = new test();

　　alert(x); //2

运行结果为2，表明全局变量x的值根本没变。

情况四 apply调用

apply()是函数对象的一个方法，它的作用是改变函数的调用对象，它的第一个参数就表示改变后的调用这个函数的对象。因此，this指的就是这第一个参数。

复制代码代码如下:

　　var x = 0;

　　function test(){

　　　　alert(this.x);

　　}

　　var o={};

　　o.x = 1;

　　o.m = test;

　　o.m.apply(); //0

apply()的参数为空时，默认调用全局对象。因此，这时的运行结果为0，证明this指的是全局对象。

1. 写一段JS代码：判断字符串是否是这样组成的，第一个必须是字母，后面可以是字母、数字、下划线、总长度为5-20

var text = “abc”;

var reg = /^[a-zA-Z]\w{5,20}$/;

reg.test(text);

1. Javascript中callee和caller的作用？

callee是对象的一个属性，该属性是一个指针，指向参数arguments对象的函数

caller是函数对象的一个属性，该属性保存着调用当前函数的函数的引用（指向当前函数的直接父函数）

返回一个对函数的引用，该函数调用了当前函数。

注意：

对于函数来说，caller 属性只有在函数执行时才有定义。 如果函数是由 Javascript 程序的顶层调用的，那么 caller 包含的就是 null 。

1. 前端页面有那三层构成，分别是什么？作用是什么？

分成：结构层、表示层、行为层。

结构层（structural layer）

由 HTML 或 XHTML之类的标记语言负责创建。标签，也就是那些出现在尖括号里的单词，对网页内容的语义含义做出了描述，但这些标签不包含任何关于如何显示有关内容的信息。例如，P标签表达了这样一种语义：“这是一个文本段。”

表示层（presentation layer）

由 CSS 负责创建。 CSS对“如何显示有关内容”的问题做出了回答。

行为层（behaviorlayer）

负责回答“内容应该如何对事件做出反应”这一问题。这是 Javascript 语言和 DOM主宰的领域。

1. 前端页面由哪三层构成，分别是什么作用？

同上

1. JS语句var a1=10;var a2=20;alert(“a1+a2=”a1+a2) (D )

A、a1+a2=30 B、a1+a2=1020 C、a1+a2=a1+a2 D、显示错误

1. 将字符串s中的所有字母变为小写字母的方法（B）

A、s.toSmallCase() B、s.toLowerCase() C、s.toUpperCase() D、s.toUpperCase()

1. 以下（B）表达式产生一个0-7之间（含0.7）的随机整数

A、Math.floor(Math.random()\*6)

B、Math.floor(Math.random()\*7)

C、Math.floor(Math.random()\*8)

D、Math.ceil(Math.random()\*8)

1. 什么是JavaScripts原型、原型链?有什么特点?

在JavaScript中,一共有两种类型的值,原始值和对象值.每个对象都有一个内部属性[[prototype]],我们通常称之为原型.原型的值可以是一个对象,也可以是null.如果它的值是一个对象,则这个对象也一定有自己的原型.这样就形成了一条线性的链,我们称之为原型链.

访问一个对象的原型可以使用ES5中的Object.getPrototypeOf方法,或者ES6中的\_\_proto\_\_属性.

原型链的作用是用来实现继承,比如我们新建一个数组,数组的方法就是从数组的原型上继承而来的.

var arr = [];

arr.map === Array.prototype.map //arr.map是从arr.\_\_proto\_\_上继承下来的,arr.\_\_proto\_\_也就是Array.prototype

图形化原型链

虽然我们都说原型链,但实际上,在不考虑网页中frame的情况,js引擎在执行期间的某一时刻,所有存在的对象组成的是一棵原型树.默认情况下,只有一棵树.根节点可以说是Object.prototype,也可以说是null.

但我们可以再建立一棵原型树,通过使用Object.create方法

var foo = Object.create(null); //foo是一个对象,但它是游离的,不属于已有的那棵原型树

var bar = Object.create(foo); //bar的原型是foo

var baz = Object.create(foo); //baz的原型是foo

这样我们有了第二棵原型树

遍历原型链

我们没有办法遍历到所有以某个对象为原型的对象,但我们可以向上遍历,获取到一个对象所有的上层原型,这个原型链必定是线性的,尽头是null.

function getPrototypeChain(object) {

var protoChain = [];

while (object = object.\_\_proto\_\_) {

protoChain.push(object);

}

protoChain.push(null);

return protoChain;

}

试验一下,不同的环境实现不同,显示形式也不同.下面是在chrome控制台中的显示.

>getPrototypeChain(new String(""))

[String, Object, null] //依次是String.prototype,Object.prototype,null

>getPrototypeChain(function(){})

内置类型的对象的原型链并不长,下面试试宿主对象.

>getPrototypeChain(document.createElement("div"))

[HTMLDivElement, HTMLElement, Element, Node, Object, null]

这个就长多了.

超长原型链

可以看出来,我们平时使用的对象并没有很长的原型链.但可以自己构造一个.

function Foo() {}

for (var i = 0; i < 100; i++) {

Foo.prototype["foo" + i] = i;

Foo.prototype = new Foo;

}

console.dir(getPrototypeChain(new Foo));

1. 简述 eval的作用

eval() 函数可计算某个字符串，并执行其中的的 JavaScript 代码。

该方法只接受原始字符串作为参数，如果 string 参数不是原始字符串，那么该方法将不作任何改变地返回。因此不要为 eval() 函数传递 String 对象来作为参数。

如果试图覆盖 eval 属性或把 eval() 方法赋予另一个属性，并通过该属性调用它，则 ECMAScript 实现允许抛出一个 EvalError 异常。

1. 写一段代码创建一个对象

var obj = {

name : “小明”,

age : 18

}

1. 什么是事件？Ie与火狐的事件机制有何区别？如何阻止冒泡？

JavaScript 使我们有能力创建动态页面。事件是可以被 JavaScript 侦测到的行为。

网页中的每个元素都可以产生某些可以触发 JavaScript 函数的事件。比方说，我们可以在用户点击某按钮时产生一个 onClick 事件来触发某个函数。事件在 HTML 页面中定义。

事件举例：

鼠标点击

页面或图像载入

鼠标悬浮于页面的某个热点之上

在表单中选取输入框

确认表单

键盘按键写一个通用的事件侦听函数

区别参考<http://www.cnblogs.com/zhangtao/articles/1490432.html>

阻止事件冒泡

function stopBubble(e){

if(e&&e.stopPropagation){//非IE

e.stopPropagation();

}

else{//IE

window.event.cancelBubble=true;

}

}

1. 什么是闭包(closure),为何要使用？

闭包是指有权访问另一个函数作用域中变量的函数，创建闭包的最常见的方式就是在一个函数内创建另一个函数，通过另一个函数访问这个函数的局部变量,利用闭包可以突破作用链域，将函数内部的变量和方法传递到外部。

闭包的特性：

1.函数内再嵌套函数

2.内部函数可以引用外层的参数和变量

3.参数和变量不会被垃圾回收机制回收

1. 如何判断一个对象是否属于某个类？

var obj = new Abc();

alert(obj instanceof Abc);

1. 说说new操作符具体做了什么？

举例: var obj = new Base();

new操作符具体干了三件事情。

一，我们创建了一个空对象obj

二，我们将这个空对象的\_\_proto\_\_成员指向了Base函数对象prototype成员对象

三，我们将Base函数对象的this指针替换成obj，然后再调用Base函数，于是我们就给obj对象赋值了一个id成员变量，这个成员变量的值是”base”

1. 上面代码执行的输出结果

var z = 10;

function foo(){

console.log(z);

}

(function(funArg){

var z = 20

funArg();

})(foo)

10

1. 上面代码执行后输出什么？

var data = [];

for(var k = 0;k<3;k++){

data[k] = function(){

console.log(k);

};

};

data[0]()

data[1]()

data[2]()

3 3 3

1. 假设现有一篇文章 var content = “…大量文字”，文章触及到一些敏感词[“习近平”，“周永康”，“中共”，“6.4”]等，如何在文章中发现这些敏感词，并将背景置为红色或改变字体颜色标识出来。

css : .red{

color : red;

}

js : var content = “…大量文字”,

var sen = [“习近平”，“周永康”，“中共”，“6.4”];

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

content.replace(sen[i],”<span class=‘red’>”+sen[i]+”</span>”);

}

1. Label标签的作用是?

可以使标签内的区域指向label标签for属性指代的对象的事件。

无label标签:<input id="ye" type="checkbox" /> 文本

有label标签:<label for="ye"><input id="ye" type="checkbox" />

文本</label>

有label标签的这一段，点击标签中的文本，可使多选框聚焦。

1. 已知有字符串foo=”get-element-by-id”,写一个function将其转化成驼峰表示法”。

function combo(msg){

var arr=msg.split("-"); //split("-")以-为分隔符截取字符串，返回数组

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

arr[i]=arr[i].charAt(0).toUpperCase()+arr[i].substr(1,arr[i].length-1);

}

msg=arr.join(""); //join()加入什么，连接数组元素，返回字符串

return msg;

}

1. 用js/jquery写出数组中去除重复元素的方法（写清大概思路即可，能想到几种写几种）

方法一:

1).构建一个新的数组存放结果

2.)for循环中每次从原数组中取出一个元素，用这个元素循环与结果数组对比

3).若结果数组中没有该元素，则存到结果数组中

方法二:

1).先将原数组进行排序

2).检查原数组中的第i个元素 与 结果数组中的最后一个元素是否相同，因为已经排序，所以重复元素会在相邻位置

3).如果不相同，则将该元素存入结果数组中

方法三:

1.创建一个新的数组存放结果

2.创建一个空对象

3.for循环时，每次取出一个元素与对象进行对比，如果这个元素不重复，则把它存放到结果数组中，同时把这个元素的内容作为对象的一个属性，并赋值为1，存入到第2步建立的对象中。

说明：至于如何对比，就是每次从原数组中取出一个元素，然后到对象中去访问这个属性，如果能访问到值，则说明重复。

1. 回答下列问题

Function Foo () {

getName = function() { alert (1);};

return this;

}

Foo.getName = function () { alert(2);};

Foo.prototype.getName = function(){alert ( 3 );};

Var getName = function (){alert (4);};

Function getName() {alert (5);}

//请写出以下输出结果：

Foo.getName() //2

getName() // 4

new Foo.getName(); //2

new Foo().getName();//3

new new Foo().getName();//3

new Foo().getName();//3

1. 下面代码会输出什么?

var test =(function(a){

this.a=a;

return function(b){

return this.a+b;

}

}(function(a,b){

return a;

}(1,2)))

console.log(test(1));//2

console.log(test(4));//5

1. 请把

<ul>

<li>第一行</li>

<li>第二行</li>

<li>第三行</li>

.......

</ul>

(ul之间有10个li元素)插入body里面，注意：需要考虑到性能问题。

var bodyDom = document.body;

var bHtml = “<ul><li>第一行</li>”;

var list = [‘一’,’二’,’三’…];

for(var i = 0;i < 9; i ++){

bHtml += “<li>第”+list[i+1]+”行</li>”;

}

bHtml += “</ul>”;

bodyDom.innerHtml += bHtml;

1. 有一个长度为100的数组，不适用于for循环自出该数组的前10个元素之和。

jQuery

1. 下面有关jquery事件的响应，描述错误的是? ()

A、onclick 鼠标点击某个事件

B、onfocus 元素失去焦点

C、onload 是某个页面的css js html 文档结构和图像被完成加载

1. 如果需要匹配包含文本的元素，用下面哪种函数来实现？

A、text()

B、contains()

C、input()

D、attr(name)

1. 如果想要找到一个表格的指定行数的元素，可以使用下面哪个函数？（单选）

A、text()

B、get()

C、eq()

D、contents()

1. 在一个表单里，想要找到指定元素的第一个元素用\_\_\_\_\_实现，第二个元素用\_\_\_\_实现。
2. 下面哪种不属于jQuery的筛选？（单选）

A、过滤 B、自动 C、查找 D、串联

1. 下面哪几种属于jQuery文档处理？（多选）

A、包裹 B、替换 C、删除 D、内部和外部插入

1. 如果想在一个指定的元素前添加内容，下面哪几个可以实现？（单选）

A、append(content) B、appendTo(content) C、insertAfter(content) D、after(content)

1. 在jQuery中，用一个表达式来检查当前选择的元素集合，使用\_\_\_\_来实现，如果这个表达式失效，则返回\_\_\_\_值。
2. 在jQuery中，如果想从DOM中删除所有匹配的元素，下面哪一个正确？（单选）

A、delete() B、empty() C、remove() D、removeAll()

1. 请用js实现全选和取消全选的功能，只需要写出js代码(可以使用jquery框架)
2. JavaScript 中typeof的返回值有哪些？
3. 问三行a,b,c的输出分别是什么？

function fun(n,o){

console.log(o)

return {

fun:function(m){

return fun(m,n);

}

};

}

var a = fun(0);a.fun(1);a.fun(2);a.fun(3);

var b = fun(0);b.fun(1);b.fun(2);b.fun(3);

var c = fun(0);c.fun(1);c.fun(2);c.fun(3);

1. var a = [],在单页面中如何判断一个a是数组？
2. 想在JQuery中，如果想要获取当前窗口的宽度值，下面那个是实现...（）

A.width() B.width(avl) C.width D.innerWidth()

1. 1+2+3+”4”-1 输出结果是什么？
2. 原生js如何给id为hello的元素绑定一个点击事件(兼容IE8)？
3. 将当前日期输出如以下格式2013/03/08，并且，定义一个函数输出当前是星期几？
4. 用户A QQ：5201234， 用户B QQ：520696969， 捣乱用户 C QQ:520 捣乱用户：D QQ：122233333333333333333 问：一般QQ号码是5~12位的，请通过正则表达式过滤出以上正常用户的Q[Q
5. $(this)和this关键字在jQuery中有何不同？
6. jQuery的作用以及优点
7. 请写出jquery绑定事件的方法，不少于两种
8. 请使用jQuery编写doQuery方法满足一下要求：

1）doQuery接收字符串参数作为url\_1

2) 使用HTTP GET异步请求 url\_1, 获取返回数据中的url\_2

3) 使用HTTP GET 异步请求 url\_2, 并使用控制台输出返回数据

4）url\_1定义如下：

var url\_1 = “/url1.do”;

doQuery(url\_1);

请求/url1.do返回JSON数据：

{

“url”: “/url2.do”

}

5）

HTTP

1. HTTP协议返回的状态码：200,304,404,500分别代表什么含义？
2. 一次完整的HTTP事务是怎样的一个过程？

前后端合作

1. 关于json和jsonp的以下说法，错误的是（）
2. 什么是cookie？什么是session？他们之间有什么关系？假设客户端不支持cookie，是否可实现session功能？
3. 如何实现跨域？JSONP实现原理？
4. Ajax是什么？ajax的交互模型？同步和异步的区别？如何解决跨域问题？
5. 简述Ajax过程
6. 异步加载和延迟加载
7. 简述同步和异步的区别

模块化开发

1. require.js的具体用法

AJAX

1. 获取的数据编码不一样，怎么进行转码？
2. AJAX术语是由哪家公司或组织最先提出的？（单选）

A 、Google B、IBM C、Adaptive Path D、进一步选择器

1. 以下哪个技术不是Ajax技术体系的组成部分？（单选）

A、XMLHttpRequest B、CSS C、DHEML D、DOM

1. XMLHttpRequest 对象有几个返回状态值？

A、3 B 、4 C 、5 D、6

1. 下面不属于AJAX事件的是哪个？（单选）

A、ajaxComplete(callback)

B、ajaxSuccess(callpack)

C、$.post(url)

D、ajaxSend(callback)

1. 写出ajax获取数据的常用写法，并做相应注释(可以使用jquery框架)
2. 用js写个简单原生的ajax过程，关键部分。