



2017-12-15

前端面试汇总

2017 年版



C3gen

小菜鸟一枚 QQ 294027471

- 一、HTML和CSS 19
- 1、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?..... 19
- 2、每个HTML文件里开头都有个很重要的东西，Doctype，知道这是干什么的吗？ 19
- 3、Quirks 模式是什么？它和 Standards 模式有什么区别 19
- 4、div+css 的布局较 table 布局有什么优点？ 20
- 5、img 的 alt 与 title 有何异同？strong 与 em 的异同？ 20
- 6、你能描述一下渐进增强和优雅降级之间的不同吗?..... 20
- 7、为什么利用多个域名来存储网站资源会更有效？..... 21
- 8、请谈一下你对网页标准和标准制定机构重要性的理解。..... 21
- 9、请描述一下cookies，sessionStorage和localStorage的区别？..... 21
- 10、简述一下src与href的区别。.....22
- 11、知道的网页制作会用到的图片格式有哪些？..... 22
- 12、知道什么是微格式吗？谈谈理解。在前端构建中应该考虑微格式吗？..... 23
- 13、在css/js代码上线之后开发人员经常会优化性能，从用户刷新网页开始，一次js 请求一般情况下有哪些地方会有缓存处理？ 23
- 14、一个页面上有大量的图片（大型电商网站），加载很慢，你有哪些方法优化这些图片的加载，给用户更好的体验。.....23
- 15、你如何理解HTML结构的语义化？ 23
- 16、谈谈以前端角度出发做好SEO需要考虑什么？ 24

- 17、有哪项方式可以对一个DOM设置它的CSS样式？ 26
- 18、CSS都有哪些选择器？ 26
- 19、CSS中可以通过哪些属性定义，使得一个DOM元素不显示在浏览器可视范围内？ 27
- 20、超链接访问过后hover样式就不出现的问题是什么？如何解决？ 27
- 21、什么是Css Hack？ie6,7,8的hack分别是什么？ 27
- 22、请用Css写一个简单的幻灯片效果页面 28
- 24、行内元素和块级元素的具体区别是什么？行内元素的padding和margin可设置吗？
..... 29
- 25、什么是外边距重叠？重叠的结果是什么？ 29
- 26、rgba()和opacity的透明效果有什么不同？ 29
- 27、css中可以让文字在垂直和水平方向上重叠的两个属性是什么？ 30
- 28、如何垂直居中一个浮动元素？30
- 29、px和em的区别。 31
- 30、描述一个“reset”的CSS文件并如何使用它。知道normalize.css吗？你了解他们的不同之处？ 31
- 31、Sass、LESS是什么？大家为什么要使用他们？ 31
- 32、display:none与visibility:hidden的区别是什么？ 32
- 34、CSS中link和@import的区别是： 32

35、简介盒子模型 :	32
36、为什么要初始化样式 ?	33
37、BFC是什么?	33
38、html语义化是什么 ?	33
39、Doctype 的作用 ? 严格模式与混杂模式的区别 ?	34
40、IE的双边距BUG : 块级元素float后设置横向margin , ie6显示的margin比设置的较大。解决 : 加入_display : inline	34
41、HTML与XHTML——二者有什么区别 ?	34
42、html常见兼容性问题 ?	34
43、对WEB标准以及W3C的理解与认识	35
44、行内元素有哪些?块级元素有哪些?CSS 的盒模型?	35
45、前端页面有哪三层构成 , 分别是什么?作用是什么?.....	35
46、Doctype作用?严格模式与混杂模式-如何触发这两种模式 , 区分它们有何意义?	35
47、行内元素有哪些 ? 块级元素有哪些 ? 空(void)元素有那些 ?	35
48、CSS的盒子模型 ?	36
49、CSS 选择符有哪些 ? 哪些属性可以继承 ? 优先级算法如何计算 ? CSS3新增伪类有 那些 ?	36
50、如何居中 div,如何居中一个浮动元素?	36
51、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 ? 原因 , 解决方法是 什么 , 常用hack的技巧 ?	37
52、列出display的值 , 说明他们的作用。position的值 , relative和absolute定位原点是 ?	38
53、absolute的containing block计算方式跟正常流有什么不同 ?	38
54、position跟display、margin collapse、overflow、float这些特性相互叠加后 会怎么样 ?	38

- 55、对WEB标准以及W3C的理解与认识 38
- 56、css的基本语句构成是? 39
- 57、浏览器标准模式和怪异模式之间的区别是什么?..... 39
- 58、CSS中可以通过哪些属性定义，使得一个DOM元素不显示在浏览器可视范围内？39
- 59、超链接访问过后hover样式就不出现的问题是什么？如何解决？
39
- 60、什么是Css Hack？ie6,7,8的hack分别是什么？ 39
- 62、请用Css写一个简单的幻灯片效果页面 40
- 63、行内元素和块级元素的具体区别是什么？行内元素的padding和margin可设置吗？
.....
..... 41
- 64、什么是外边距重叠？重叠的结果是什么？..... 41
- 65、rgba()和opacity的透明效果有什么不同？ 41
- 66、css 中可以让文字在垂直和水平方向上重叠的两个属性是什么？ 42
- 67、如何垂直居中一个浮动元素？.....42

68、描述一个"reset"的CSS文件并如何使用它。知道normalize.css吗？你了解他们

的不同之

处？.....

43 69、说display属性有哪些？可以做什么？..... 43

70、哪些css属性可以继承？ 43

71、css优先级算法如何计算？ 43

72、b标签和strong标签,i标签和em标签的区别？ 43

73、有那些行内元素、有哪些块级元素、盒模型？..... 44

74、有哪些选择符,优先级的计算公式是什么？行内样式和 !important 哪个优先级高？

.....

... 45

75.我想让行内元素跟上面的元素距离10px，加margin-top和padding-top可以吗？45 76.CSS的盒模型由什么组成？ 45

77、.说说display属性有哪些？可以做什么？ 45

78、哪些css属性可以继承？ 45

79、css优先级算法如何计算？ 45

80、text-align:center和line-height有什么区别？ 46

81、前端页面由哪三层构成，分别是什么？作用是什么？..... 46

82、写一个表格以及对应的 CSS,使表格奇数行为白色背景，偶数行为灰色，鼠标一上去为黄色背景。 46

二、JS基础46

1、javascript的typeof返回哪些数据类型 46

2、例举3种强制类型转换和2种隐式类型转换?..... 46

3、split() join() 的区别 46

4、数组方法pop() push() unshift() shift()	47
5、事件绑定和普通事件有什么区别.....	47
6、IE和DOM事件流的区别	47
7、IE和标准下有哪些兼容性的写法	47
8、call和apply的区别	48
9、b继承a的方法	48
10、JavaScript this指针、闭包、作用域	48
11、事件委托是什么.....	48
12、闭包是什么，有什么特性，对页面有什么影响.....	48
13、如何阻止事件冒泡和默认事件.....	48
14、添加 删除 替换 插入到某个接点的方法.....	49
15、javascript的本地对象，内置对象和宿主对象	49
16、document load 和document ready的区别	49
17、“==” 和 “===” 的不同.....	49
18、javascript的同源策略	50
19、编写一个数组去重的方法.....	50

- 20、JavaScript 是一门什么样的语言，它有哪些特点？
50
- 21、JavaScript 的数据类型都有什么？ 51
- 22、已知ID的Input输入框，希望获取这个输入框的输入值，怎么做？(不使用第三方 框架) 52
- 23、希望获取到页面中所有的 checkbox 怎么做？(不使用第三方框架) 52
- 24、设置一个已知ID的DIV的html内容为xxxx，字体颜色设置为黑色(不使用第三方 框架) 52
- 25、当一个 DOM 节点被点击时候，我们希望能够执行一个函数，应该怎么做？ 52
- 26、看下列代码输出为何？解释原因。..... 53
- 27、看下列代码,输出什么？解释原因。..... 53
- 28、看下列代码,输出什么？解释原因。..... 53
- 29、看代码给答案。.....54
- 30、已知数组var stringArray = ["This" , "is" , "Baidu" , "Campus"] , Alert出" This is BaiduCampus" 。 54
- 31、var numberArray = [3,6,2,4,1,5]; (考察基础API) 55
- 32、输出今天的日期，以YYYY-MM-DD的方式，比如今天是2014年9月26日，则输出
2014-09-
26.....
55
- 33、将字符串" <tr><td>{\$id}</td><td>{\$name}</td></tr>" 中的{\$id}替换成 10 ,
{\$name}替换成Tony (使用正则表达式) 56

- 34、为了保证页面输出安全，我们经常需要对一些特殊的字符进行转义，请写一个函数 `escapeHtml`，将 `<`，`>`，`&`，“进行转义..... 56
- 35、`foo = foo||bar`，这行代码是什么意思？为什么要这样写？ 56
- 36、看下列代码，将会输出什么?(变量声明提升)..... 57
- 37、用js实现随机选取10-100之间的10个数字，存入一个数组，并排序。 57
- 38、把两个数组合并，并删除第二个元素。..... 57
- 39、怎样添加、移除、移动、复制、创建和查找节点（原生JS，实在基础，没细写每一步） 58
- 40、有这样一个URL：
`http://item.taobao.com/item.htm?a=1&b=2&c=&d=xxx&e`，请写一段JS程序提取URL中的各个GET参数(参数名和参数个数不确定)，将其按key-value形式返回到一个json结构中，如 `{a:'1', b:'2', c:'', d:'xxx', e:undefined}`。 58
- 41、正则表达式构造函数 `var reg=new RegExp("xxx")` 与正则表达式字面量 `var reg=//` 有什么不同？匹配邮箱的正则表达式？ 59
- 42、写一个function，清除字符串前后的空格。（兼容所有浏览器） 59
- 43、Javascript中 `callee` 和 `caller` 的作用？ 60
- 44、Javascript中，以下哪条语句一定会产生运行错误？答案(B) 61
- 45、以下两个变量a和b，a+b的哪个结果是NaN？ 答案(C) 61
- 46、`var a=10; b=20; c=4; ++b+c+a++` 以下哪个结果是正确的？答案(B) 61

47、下面的JavaScript语句中，(D)实现检索当前页面中的表单元素中的所有文本

框，并将它们全部清

空.....61

48、要将页面的状态栏中显示“已经选中该文本框”，下列 JavaScript 语句正确的是

(A) 62

49、以下哪条语句会产生运行错误：(A) A.var obj = ();

.var obj = []; Cvar obj = {};

D.ar obj = //; 62

50、以下哪个单词不属于javascript保留字：(B) A.with

.parent Cclass

D.oid 62

51、请选择结果为真的表达式：(C) A.null instanceof Object

.null === undefined Cnull === undefined

D.aN === NaN 62

52、Javascript中, 如果已知HTML页面中的某标签对象的id="username"，用

_____方法获得该标签对

_____document.getElementById('usern 象。62

ame')_____

53、typeof 运算符返回值中有一个跟 javascript 数据类型不一致，它是

_____ " function" ____。 62

54、定义了一个变量，但没有为该变量赋值，如果alert该变量，javascript弹出的

对话框中显示____undefined____。62

... 62

55、分析代码，得出正确的结

果。62

- 56、写出函数DateDemo的返回结果，系统时间假定为今天 63
- 57、写出程序运行的结果？64
- 58、阅读以下代码，请分析出结果：64
- 59、补充按钮事件的函数，确认用户是否退出当前页面，确认之后关闭窗口；
<html>64
- 60、写出简单描述 html 标签（不带属性的开始标签和结束标签）的正则表达式，并将以下字符串中的html标签去除掉 65
- 61、完成 foo()函数的内容，要求能够弹出对话框提示当前选中的是第几个单选框。 65
- 62、完成函数 showImg()，要求能够动态根据下拉列表的选项变化，更新图片的显示 66
- 63、截取字符串abcdefg的efg 66
- 64、列举浏览器对象模型BOM里常用的至少4个对象，并列举window对象的常用方法 至少5个 67
- 65、简述列举文档对象模型DOM里document的常用的查找访问节点的方法并做简单说明 67
- 66、希望获取到页面中所有的checkbox怎么做？(不使用第三方框架)
67
- 67、JavaScript的数据类型都有什么？ 67

- 68、javascript 中有哪几种数据类型，分别写出中文和英文。 68
- 69、javascript 中==和===的区别是什么？举例说明。 69
- 70、简述创建函数的几种方式.....69
- 71、Javascript如何实现继承？ 69
- 72、Javascript 创建对象的几种方式？ 69
- 73、把 Script 标签 放在页面的最底部的body封闭之前 和封闭之后有什么区别？浏览器会如何解析它们？ 69
- 74、iframe的优缺点？ 69
- 75、请你谈谈Cookie的弊端？ 70
- 76、DOM 操作——怎样添加、移除、移动、复制、创建和查找节点。 70
- 77、js延迟加载的方式有哪些？ 70
- 78、documen.write和 innerHTML 的区别？ 70
- 79、哪些操作会造成内存泄漏？.....71
- 80、javascript的typeof返回哪些数据类型？ 71
- 81、split() join() 的区别 71
- 82、数组方法pop() push() unshift() shift()各表示什么意思？ 71
- 83、判断一个字符串中出现次数最多的字符，统计这个次数..... 71
- 84、javascript的typeof返回哪些数据类型 72
- 85、例举3种强制类型转换和2种隐式类型转换?..... 72
- 86、split() join() 的区别 72
- 87、数组方法pop() push() unshift() shift() 72
- 88、事件绑定和普通事件有什么区别..... 错误！未定义书签。
- 89、IE和DOM事件流的区别 72

90、IE和标准下有哪些兼容性的写法	72
91、call和apply的区别	72
92、b继承a的方法.....	错误！未定义书签。
93、写一个获取非行间样式的函数.....	73
94、事件委托是什么.....	错误！未定义书签。
95、闭包是什么，有什么特性，对页面有什么影响.....	73
96、解释jsonp的原理，以及为什么不是真正的ajax	73
97、javascript 的本地对象，内置对象和宿主对象	73
98、document load 和document ready的区别	73
99、字符串反转，如将 '12345678' 变成 '87654321'.....	74
100、将数字 12345678 转化成 RMB形式 如：12,345,678	74
101、生成 5 个不同的随机数；.....	74
102、去掉数组中重复的数字方法一；.....	74
103、阶乘函数；.....	75
104、window.location.search()返回的是什么？	76

105、window.location.hash 返回的是什么？	76
106、window.location.reload() 作用？	76
107、阻止冒泡函数.....	76
108、什么是闭包？写一个简单的闭包？；	77
109、javascript 中的垃圾回收机制？	77
110、看题做答：	77
111、下面输出多少？	78
112、再来一个.....	78
113、	78
114、	79
115、JS的继承性	80
116、精度问题: JS 精度不能精确到 0.1 所以。。。。同时存在于值和差值中..	80
117、加减运算.....	80
118、什么是同源策略？	81
119、call和applay的区别是什么？	81
120、为什么不能定义1px左右的div容器？	81
121、结果是什么？	81
122、输出结果.....	81

123、计算字符串字节数：	82
124、结果是：	82
125、声明对象，添加属性，输出属性.....	82
126、匹配输入的字符：第一个必须是字母或下划线开头，长度 5-20.....	82
127、检测变量类型.....	83
128、如何在HTML中添加事件，几种方法？	83
129、BOM对象有哪些，列举window对象？	83
130、请问代码实现 outerHTML	83
131、JS中的简单继承 call方法！	84
132、bind(), live(), delegate()的区别	85
133、typeof 的返回类型有哪些？	85
134、简述link和import的区别？	86
135、window.onload 和 document.ready的区别？	86
136、解析URL成一个对象？	86
137、看下列代码输出什么？.....	86
138、看下列代码,输出什么？.....	87
139、已知数组 var stringArray = ["This" , "is" , "Baidu" , "Campus"] , Alert 出" This is Baidu Campus" 。	87
140、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"getElementById"。	87

- 141、怎样添加、移除、移动、复制、创建和查找节点..... 87
- 142、原生JS的window.onload与Jquery的\$(document).ready(function(){})有什么 不同？ 88
- 143、你如何优化自己的代码？88
- 144、请描述出下列代码运行的结果.....88
- 145、需要将变量e的值修改为 “a+b+c+d” ,请写出对应的代码 88
- 146、怎样实现两栏等高？89
- 147、使用js实现这样的效果：在文本域里输入文字时，当按下enter键时不换行，而 是替换成 “{{enter}}” ,(只需要考虑在行尾按下enter键的情况) 89
- 148、以下代码中end字符串什么时候输出 89
- 149、specify('hello,world')//=> ' h,e,l,l,o,w,o,r,l,d' 实现 specify 函数..89
- 150、请将一个URL的search部分参数与值转换成一个json对象 89
- 151、请用原生js实现jquery的get\post功能，以及跨域情况下 90
- 152、请简要描述web前端性能需要考虑哪方面，你的优化思路是什么？ 90
- 153、简述readonly与disabled的区别 91
- 154、判断一个字符串出现次数最多的字符，统计这个次数并输出..... 91
- 155、编写一个方法，去掉一个数组的复重元素..... 91
- 156、写出3个使用this的典型应用 91
- 157、请尽可能详尽的解释ajax的工作原理91
- 158、为什么扩展javascript内置对象不是好的做法？ 91
- 159、请解释一下javascript的同源策略 91

- 160、什么是三元表达式？“三元”表示什么意思？..... 91
- 161、浏览器标准模式和怪异模式之间的区别是什么？..... 91
- 162、如果设计中使用了非标准的字体，你该如何去实现？..... 92
- 163、用css分别实现某个div元素上下居中和左右居中 92
- 164、`modulo(12,5)//2` 实现满足这个结果的`modulo`函数 92
- 165、HTTP协议中，GET和POST有什么区别？分别适用什么场景？
92
- 166、HTTP状态消息200 302 304 403 404 500分别表示什么 93
- 167、HTTP 协议中，header 信息里面，怎么控制页面失效时间
(`last-modified`,`cache-control`,`Expires`分别代表什么) 93
- 168、HTTP雷锋议目前常用的有哪几个？KEEPALIVE从哪个版本开始出现的？93
- 169、业界常用的优化 WEB 页面加载速度的方法（可以分别从页面元素展现，请求连接，css,js,服务器等方面介绍） 93
- 170、列举常用的web页面开发，调试以及优化工具 93
- 171、解释什么是sql注入，xss漏洞 94
- 172、如何判断一个js变量是数组类型 94
- 173、请列举js数组类型中的常用方法 94
- 174、FF与IE中如何阻止事件冒泡，如何获取事件对象，以及如何获取触发事件的元

素.....	
..... 94	175、列举常用的js框架以及分别适用的领域..... 94
176、js中如何实现一个map	95
177、js可否实现面向对象编程，如果可以如何实现js对象的继承	96
178、约瑟夫环—已知n个人（以编号1，2，3...分别表示）围坐在一张圆桌周围。从编号为k的人开始报数，数到m的那个人出列；他的下一个人又从1开始报数，数到m的那个人又出列；依此规律重复下去，直到圆桌周围的人全部出列。	96
179、有1到10w这个10w个数，去除2个并打乱次序，如何找出那两个数？	96
180、如何获取对象 a 拥有的所有属性（可枚举的、不可枚举的，不包括继承来的属性）	
.....	
....	96
181、有下面这样一段HTML结构，使用css实现这样的效果：	96
182、下面这段代码想要循环输出结果 01234，请问输出结果是否正确，如果不正确，请说明为什么，并修改循环内的代码使其输出正确结果.....	97
183、解释下这个css选择器什么发生什么？	97
184、JavaScript以下哪条语句会产生运行错误	97
185、以下哪些是javascript的全局函数：（ABC）	97
186、关于IE的window对象表述正确的有：（ACD）	97
187、描述错误的是.....	98
188、关于link和@import的区别正确的是 A	98
189、下面正确的是 A	98
188、错误的是.....	99
189、不用任何插件，如何实现一个tab栏切换？	99

- 190、基本数据类型的专业术语以及单词拼写..... 99
- 191、变量的命名规范以及命名推荐.....99
- 192、三种弹窗的单词以及三种弹窗的功能..... 99
- 193、`console.log(8 | 1);` 输出值是多少？ 99
- 194、只允许使用 `+` `-` `*` `/` 和 `Math.*`，求一个函数 `y = f(x, a, b)`;当 `x > 100` 时 返回 `a` 的值，否则返回 `b` 的值，不能使用 `if else` 等条件语句，也不能使用`|,?;` 数组。 100
- 195、`JavaScriptalert(0.4*0.2)`;结果是多少？和你预期的一样吗？如果不一样该如何处理？ 100
- 196、一个 `div`，有几种方式得到这个 `div` 的 `jQuery` 对象？`<div class='aabbcc' id='nodesView'></div>`想直接获取这个 `div` 的 `dom` 对象，如何获取？`dom` 对象如何转 化为 `jQuery` 对象？ 100
- 197、主流浏览器内核.....100
- 198、如何显示/隐藏一个 `dom` 元素？请用原生的 `JavaScript` 方法实现 100
- 199、`JavaScript` 有哪几种数据类型 101
- 200、`jQuery` 框架中 `$.ajax()` 的常用参数有哪些？写一个 `post` 请求并带有发送数据和

返回数据的样例.....	
101	
201、JavaScript数据元素添加、删除、排序等方法有哪些？	103
202、如何添加html元素的事件，有几种方法？请列举	104
203、JavaScript的循环语句有哪些？	104
204、作用域-编译期执行期以及全局局部作用域问题.....	104
205、闭包：下面这个ul，如何点击每一列的时候alert其index？	104
206、列出3条以上ff和IE的脚本兼容问题	106
207、列举可以哪些方面对前端开发进行优化.....	106
208、至少列出一种JavaScript继承的实现方式	106
209、如现在有一个效果，有显示用户头像、用户昵称、用户其他信息；当用户鼠标移 到头像上时，会弹出用户的所有信息；如果是你，你会如何实现这个功能，请用代码实现？	106
210、call与apply有什么作用？又有什么什么区别？用callee属性实现函数递归？	106
211、用正则表达式，写出由字母开头，其余由数字、字母、下划线组成的 6~30 的字符串？	107
212、列举浏览器对象模型BOM里常用的至少4个对象，并列举window对象的常用方法 至少5个（10分）	107
213、Javascript中callee和caller的作用？	107
214、对于apply和call两者在作用上是相同的，即是调用一个对象的一个方法，以另一个对象替换当前对象。将一个函数的对象上下文从初始的上下文改变为由thisObj 指定的新对象。	107
215、在Javascript中什么是伪数组？如何将伪数组转化为标准数组？	107
216、写一个函数可以计算 sum(5,0,-5);输出0; sum(1,2,3,4);输出10;	108

- 217、事件代理怎么实现？.....108
- 218、《正则》写出正确的正则表达式匹配固话号，区号3-4位，第一位为0，中横线，7-8位数字，中横线，3-4位分机号格式的固话号..... 108
- 219、《算法》一下A,B可任选一题作答，两题全答加分 109
- 220、请写出一张图片的HTML代码，已知道图片地址为“images/abc.jpg”，宽100px，高50px 109
- 221、请写一个正则表达式：要求最短6位数，最长20位，阿拉伯数和英文字母（不区分大小写）组成..... 109
- 222、统计1到400亿之间的自然数中含有多少个1？比如1-21中，有1、10、11、21
这四个自然数有5个
1.....109
- 223、删除与某个字符相邻且相同的字符，比如fdaffdaaklfjklja字符串处理之后成为“fdafdaklfjklja” 109
- 224、请写出三种以上的 Firefox 有但，InternetExplorer 没有的属性或函数 109
- 225、请写出一个程序，在页面加载完成后动态创建一个form表单，并在里面添加一个input对象并给它任意赋值后以post方式提交到：
http://127.0.0.1/save.php 110

226、用JavaScript实现冒泡排序。数据为23、45、18、37、92、13、24	110
227、解释一下什么叫闭包，并实现一段闭包代码.....	110
228、简述一下什么叫事件委托以及其原理.....	110
229、前端代码优化的方法.....	110
230、下列JavaScript代码执行后，依次alert的结果是	111
231、下列JavaScript代码执行后，iNum的值是	111
232、输出结果是多少？	112
233、	114
234、下列JavaScript代码执行后，运行的结果是	115
235、下列JavaScript代码执行后，依次alert的结果是	115
236、下列JavaScript代码执行后的效果是	116
237、下列JavaScript代码执行后的li元素的数量是	116
238、程序中捕获异常的方法？	117
239、将字符串“<tr><td>{\$id}</td><td>{\$name}</td></tr>”中的{\$id}替换成 10， {\$name}替换成Tony（使用正则表达式）	117
240、给String对象添加一个方法，传入一个string类型的参数，然后将string的每个字符间价格空格返回，例如：	117
241、写出函数DateDemo的返回结果，系统时间假定为今天	117
242、输出今天的日期，以YYYY-MM-DD的方式，比如今天是2014年9月26日，则输出 2014-09- 26.....	

118	
243、已知数组	
var?stringArray=?["This" ,? "is" ,? "Baidu" ,? "Campus"] , Alert	
出" This?is?Baidu?Campus" 。	118
244、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" getElementById" 。	118
245、.varnumberArray=[3,6,2,4,1,5]; (考察基础API)	119
246、把两个数组合并，并删除第二个元	
素。	119
247、如何消除一个数组里面重复的元	
素？	119
248、用js实现随机选取10–100之间的10个数字，存入一个数组，并排序。	
120	
249、正则表达式构造函数varreg=newRegExp("xxx")与正则表达字面	
量varreg=// 有什么不同？匹配邮箱的正则表达式？	120
250、1var regMail =	
/^([a-zA-Z0-9_-])+@([a-zA-Z0-9_-])+((.[a-zA-Z0-9_-]{2,3}){1,2})\$/;	
120	
251、数组和字符	
串.....	121
252、下列控制台都输出什	
么.....	122
253、第 2	
题：	122
254、第 3	
题：	122
255、第 4	
题：	122
256、第 5	
题：	123

257、第 6	
题 :	123
258、第 7	
题 :	123
259、第 8	
题 :	123
260、第 9	
题 :	123
261、第10	
题 :	124
262、第11题 : 考点 : 函数声明提 前.....	124
263、第12	
题 :	124
264、第13	
题 :	124
265、第14	
题 :	125
266、第15	
题.....	125
267、第16	
题.....	125

三、Jquery 126

1、jQuery 的 slideUp 动画 , 如果目标元素是被外部事件驱动, 当鼠标快速地连续触 发外部元素事件, 动画会滞后的反复执行 , 该如何处理呢? 126

四、HTML5 CSS3 126

1、CSS3 有哪些新特性 ? 126

2、html5有哪些新特性、移除了那些元素 ? 如何处理HTML5新标签的浏览器兼容问题 ? 如何区分 HTML 和

HTML5 ? 126

3、本地存储 (Local Storage) 和cookies (储存在用户本地终端上的数据) 之间的 区别是什么 ? 127

4、如何实现浏览器内多个标签页之间的通信?..... 127

5、你如何对网站的文件和资源进行优化 ? 127

6、什么是响应式设计 ? 127

7、新的 HTML5 文档类型和字符集是 ? 127

8、HTML5 Canvas 元素有什么用 ? 128

9、HTML5 存储类型有什么区别 ? 128

10、用 H5+CSS3 解决下导航栏最后一项掉下来的问题 128

11、CSS3 新增伪类有那些 ? 128

12、请用 CSS 实现：一个矩形内容，有投影，有圆角，hover 状态慢慢变透明。 128

13、描述下 CSS3 里实现元素动画的方法 128

14、html5\CSS3有哪些新特性、移除了那些元素？如何处理HTML5新标签的浏览器兼容问题？如何区分 HTML 和 HTML5 ? 128

15、你怎么来实现页面设计图，你认为前端应该如何高质量完成工作？一个满屏品字布局如何设计?..... 129

16、你能描述一下渐进增强和优雅降级之间的不同吗?..... 129

17、为什么利用多个域名来存储网站资源会更有效 ? 130

18、请谈一下你对网页标准和标准制定机构重要性的理

解。 130 19、请描述一下cookies , sessionStorage

和localStorage的区别？ 130

20、知道 css 有个 content 属性吗？有什么作用？有什么应用？	131
21、如何在 HTML5 页面中嵌入音频？	131
22、如何在 HTML5 页面中嵌入视频？	132
23、HTML5 引入什么新的表单属性？	132
24、CSS3 新增伪类有那些？	132
25、(写)描述一段语义的 html 代码吧。	132
26.cookie在浏览器和服务端间来回传递。sessionStorage和 localStorage区别	133
27、html5有哪些新特性、移除了那些元素？如何处理HTML5新标签的浏览器兼容问题？如何区分 HTML 和 HTML5？	133
28、如何区分： DOCTYPE 声明\新增的结构元素\功能元素	134
29、语义化的理解？	134
30、HTML5 的离线储存？	134
31、写出 HTML5 的文档声明方式	134
32、HTML5 和 CSS3 的新标签	134
33、自己对标签语义化的理解	134
五、移动web开发	134
1、移动端常用类库及优缺点	134
2、Zepto 库和 JQ 区别	134
六、Ajax	135
1、Ajax 是什么？如何创建一个 Ajax？	135
2、同步和异步的区别？	135
3、如何解决跨域问题？	135

- 4、页面编码和被请求的资源编码如果不一致如何处理？ 136
- 5、简述ajax 的过程。 136
- 6、阐述一下异步加载。136
- 7、请解释一下 JavaScript 的同源策略。 136
- 8、GET和POST的区别，何时使用POST？ 136
- 9、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? 137
- 10、Ajax的最大的特点是什么。 137
- 11、ajax的缺点 137
- 12、ajax请求的时候get 和post方式的区别 137
- 13、解释jsonp的原理，以及为什么不是真正的ajax 137
- 14、什么是Ajax和JSON，它们的优缺点。 137
- 15、http 常见的状态码有那些？分别代表是什么意思？ 138
- 16、一个页面从输入 URL 到页面加载显示完成，这个过程中都发生了什么？ 138
- 17、ajax请求的时候get 和post方式的区别 138
- 18、ajax请求时，如何解释json数据 138
- 19、.javascript的本地对象，内置对象和宿主对象 139

20、为什么利用多个域名来存储网站资源会更有 效？.....	139
21、请说出三种减低页面加载时间的方 法.....	139
22 、 HTTP 状 态 码 都 有 那 些 。	139
七、JS高级.....	139
1 、 JQuery 一个对象可以同时绑定多个事件，这是如何实现的？	139
2、知道什么是webkit么？知道怎么用浏览器的各种工具来调试和debug代码么？	139
3、如何测试前端代码么？知道 BDD, TDD, Unit Test 么？知道怎么测试你的前端工程 么 (mocha, sinon, jasmine, qUnit..)?	140
4、 前端 templating(Mustache, underscore, handlebars)是干嘛的，怎么用？	140
5 、 简 述 一 下 Handlebars 的 基 本 用 法 ？	140
6、简述一下 Handlerbars 的对模板的基本处理流程，如何编译的？如何缓存的？	140
7 、 用 js 实 现 千 位 分 隔 符 ？	140
8、检测浏览器版本有哪些方式？.....	140
9、我们给一个 dom 同时绑定两个点击事件，一个用捕获，一个用冒泡，你来说下会 执 行 几 次 事 件 ， 然 后 会 先 执 行 冒 泡 还 是 捕 获	141
10、实现一个函数clone，可以对JavaScript中的5种主要的数据类型（包括Number、 String、Object、Array、Boolean）进行值复制.....	141

11、如何消除一个数组里面重复的元素？.....	142
12、小贤是一条可爱的小狗(Dog)，它的叫声很好听(wow)，每次看到主人的时候就会乖乖叫一声(yelp)。从这段描述可以得到以下对象：	14
2	
13、下面这个 ul，如何点击每一列的时候 alert 其 index?（闭包）	14
3	
14、编写一个JavaScript函数，输入指定类型的选择器(仅需支持id,class,tagName三种简单CSS选择器，无需兼容组合选择器)可以返回匹配的DOM节点，需考虑浏览器兼容性和性能。	14
4	
15、请评价以下代码并给出改进意见。.....	145
16、给String对象添加一个方法，传入一个string类型的参数，然后将string的每个字符间价格空格返回，例如：	14
6	
17、定义一个 log 方法，让它可以代理 console.log 的方法。	14
7	
18、在 Javascript 中什么是伪数组？如何将伪数组转化为标准数组？	14
7	
19、对作用域上下文和 this 的理解，看下列代码：	14
8	
20、原生JS的window.onload与Jquery的\$(document).ready(function){})有什么不同？如何用原生JS实现Jq的ready方法？	14
8	
21、（设计题）想实现一个对页面某个节点的拖曳？如何做？（使用原生JS）	15
0	

22、	151
23、说出以下函数的作用是？空白区域应该填写什么？	152
24 、 Javascript 作 用 链 域 ？	153
3	
25 、 谈 谈 This 对 象 的 理 解 。	153
3	
26 、 eval 是 做 什 么 的 ？	153
3	
27、关于事件，IE与火狐的事件机制有什么区别？如何阻止冒泡？	153
28、什么是闭包（closure），为什么要用它？	153
29、javascript 代码中的"use strict";是什么意思？使用它区别是什么？	154
30、如何判断一个对象是否属于某个类？	155
31、new操作符具体干了什么呢？	155
32、用原生JavaScript的实现过什么功能吗？	155
33、Javascript 中，有一个函数，执行时对象查找时，永远不会去查找原型，这个函数是？	155
34、对JSON的了解？	155
35、js延迟加载的方式有哪些？	155
36、模块化开发怎么做？	156
37、AMD（Modules/Asynchronous-Definition）、CMD（CommonModule Definition）规范区别？	156
38、requireJS的核心原理是什么？（如何动态加载的？如何避免多次加载的？如何缓存的？）	156
39、让你自己设计实现一个 requireJS，你会怎么做？	156
40、谈一谈你对ECMAScript6的了解？	156
41、ECMAScript6 怎么写class么，为什么会出现class这种东西？	157
42、异步加载的方式有哪些？	157
43、document.write和innerHTML的区别？	157
44、DOM 操作——怎样添加、移除、移动、复制、创建和查找节点？	157
45、call() 和 .apply() 的含义和区别？	158

46、数组和对象有哪些原生方法，列举一下？	158
47、JS怎么实现一个类。怎么实例化这个类	158
48、JavaScript 中的作用域与变量声明提升？	159
49、如何编写高性能的Javascript？	159
50、那些操作会造成内存泄漏？	159
51、javascript 对象的几种创建方式？	159
52、javascript继承的 6 种方法？	160
53、eval是做什么的？	160
54、JavaScript 原型，原型链？有什么特点？	160
55、事件、IE与火狐的事件机制有什么区别？如何阻止冒泡？	160
56、简述一下 Sass、Less，且说明区别？	160
57、关于javascript中apply()和call()方法的区别？	161
58、简述一下JS中的闭包？	161
59、说说你对this的理解？	161
60、分别阐述 split(),slice(),splice(),join()？	161
61、事件委托是什么？	161
62、如何阻止事件冒泡和默认事件？	162
63、添加 删除 替换 插入到某个接点的方法？	162
64、你用过require.js吗？它有什么特性？	162
65、谈一下JS中的递归函数，并且用递归简单实现阶乘？	162
66、请用正则表达式写一个简单的邮箱验证。	162
67、简述一下你对web性能优化的方案？	162
68、在JS中有哪些会被隐式转换为false.....	163
69、定时器setInterval有一个有名函数fn1，setInterval(fn1,500)与setInterval (fn1(),500) 有什么区别？	163
70、外部JS文件出现中文字符，会出现什么问题，怎么解决？	163
71、谈谈浏览器的内核，并且说一下什么是内核？	163
72、JavaScript原型，原型链？有什么特点？	163
73、写一个通用的事件侦听器函数.....	164
74、事件、IE与火狐的事件机制有什么区别？如何阻止冒泡？	166
75、什么是闭包（closure），为什么要用？	166
76、如何判断一个对象是否属于某个类？	166
77、new操作符具体干了什么呢？	167

78、JSON 的了解.....	167
79、js延迟加载的方式有哪些	167
80、模块化怎么 做？	167
81、异步加载的方 式.....	168
82、告诉我答案是多 少？	168
83、JS中的call()和apply()方法的区别？	168
84、Jquery与jQuery UI 有啥区别？	168
85、jquery 中如何将数组转化为json字符串，然后再转化回来？	169
86、JavaScript 中的作用域与变量声明提升？	169
87、前端开发的优化问题（看雅虎14条性能优化原 则）。	169
88、http 状态码有那些？分别代表是什么意思？	170
89、一个页面从输入 URL 到页面加载显示完成，这个过程中都发生了什 么？（流程说的越详细越好）	170
八、流行框 架.....	
170	
1、JQuery 的源码看过吗？能不能简单概况一下它的实现原理？	170
2、jQuery.fn的init方法返回的this指的是什么对象？为什么要返回this？	170
3、jquery中如何将数组转化为json字符串，然后再转化回来？	170
4、jQuery的属性拷贝(extend)的实现原理是什么，如何实现深拷贝？	170
5、jquery.extend 与 jQuery.fn.extend的区别？	170
6、谈一下Jquery中的bind(),live(),delegate(),on()的区别？	171

7、JQuery 一个对象可以同时绑定多个事件，这是如何实现的？ 171

8、Jquery	与	jQuery	UI	有啥区别？	171
9、jQuery	和	Zepto		的区别？各自的使用场景？	171
10、针对		jQuery		的优化方法？	171
11、Zepto				的点透问题如何解决？	171
12、知道各种JS框架(Angular, Backbone, Ember, React, Meteor, Knockout...)么？能讲出他们各自的优点和缺点么？					172
13、Underscore 对哪些 JS 原生对象进行了扩展以及提供了哪些好用的函数方法？					172
14、使用过 angular 吗？angular 中的过滤器是干什么用的					172
九、移动APP开发					172
1、移动端最小触控区域是多					
大？					172 十、
NodeJs.....					
.....					172 1、对Node的优点和缺点提出了自己的看
法：					172
2、需求：实现一个页面操作不会整页刷新的网站，并且能在浏览器前进、后退时正确 响应。给出你的技术实现方					
案？					172
3、Node.js				的适用场景？	173
4、(如果会用node)知道route, middleware, cluster, nodemon, pm2, server-side rendering				么？	173
5、解释一下 Backbone 的 MVC 实现方式？					173
6、什么是“前端路由”？什么时候适合使用“前端路由”？“前端路由”有哪些优点和 缺					
点？					
173 7、对Node的优点和缺点提出了自己的看					
法？					173 十一、前端概括性问

题.....	174
1、常使用的库有哪些？常用的前端开发工具？开发过什么应用或组件？.....	174
2、对BFC规范的理解？（W3CCSS2.1 规范中的一个概念,它决定了元素如何对其内容 进行定位,以及与其他元素的关系和相互作用。）.....	174
3、99%的网站都需要被重构是那本书上写的？.....	174
4、WEB 应用从服务器主动推送 Data 到客户端有那些方式？.....	174
5、加班的看 法.....	174
6、平时如何管理你的项目，如何设计突发大规模并发架构？.....	174
7、那些操作会造成内存泄漏？.....	175
8、你说你热爱前端，那么应该WEB行业的发展很关注吧？说说最近最流行的一些东西吧？.....	175
9、你了解我们公司吗？说说你的认识？.....	175
10、移动端（比如：Android IOS）怎么做好用户体验？.....	175
11、你所知道的页面性能优化方法有那些？.....	175
12、除了前端以外还了解什么其它技术么？你最最厉害的技能是什么？.....	175
13、AMD（Modules/Asynchronous-Definition）、CMD（Common Module Definition）规范区别？.....	175

- 14、谈谈你认为怎样做能是项目做的更好？..... 175
- 15、你对前端界面工程师这个职位是怎么样理解的？它的前景会怎么样？..... 176
- 16、php 中下面哪个函数可以打开一个文件，以对文件进行读和写操作？ 176
- 17、php中rmdir可以直接删除文件夹吗？该目录必须是空的，而且要有相应的权限-- 来自 api 176
- 18、phpinset 和 empty 的区别，举例说明 176
- 19、php 中\$_SERVER 变量中如何得到当前执行脚本路劲 176
- 20、写一个php函数，要求两个日期字符串的天数差，如2012-02-05~2012-03-06的日期差数 177
- 21、一个衣柜中放了许多杂乱的衬衫，如果让你去整理一下，使得更容易找到你想要的衣服；你会怎么做？请写出你的做法和思路？..... 177
- 22、如何优化网页加载速度？.....177
- 23、工作流程，你怎么来实现页面设计图，你认为前端应该如何高质量完成工作?..177
- 24、介绍项目经验、合作开发、独立开发。..... 177
- 25、开发过程中遇到困难，如何解决。..... 177
- 26、对前端界面工程师这个职位是怎么样理解的？它的前景会怎么样？..... 177

一、HTML和CSS

1、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？

IE: trident 内核 Firefox : gecko内核 Safari:webkit内核

Opera:以前是presto内核，Opera现已改用Google Chrome的Blink内核

Chrome:Blink(基于webkit，[Google与Opera Software共同开发](#))

2、每个 HTML 文件里开头都有个很重要的东西，Doctype，知道这是干什么的吗？

<!DOCTYPE> 声明位于文档中的最前面的位置，处于 <html> 标签之前。此标签可告知浏览器文档使用哪种 HTML 或 XHTML 规范。（重点：告诉浏览器按照何种规范解析页面）

3、Quirks 模式是什么？它和 Standards 模式有什么区别

从IE6开始，引入了Standards模式，标准模式中，浏览器尝试给符合标准的文档在规范上的正确处理达到在指定浏览器中的程度。

在IE6之前CSS还不够成熟，所以IE5等之前的浏览器对CSS的支持很差，IE6将对CSS提供更好的支持，然而这时的问题来了，因为有很多页面是基于旧的布局方式写的，而如果IE6支持CSS则将令这些页面显示不正常，如何在即保证不破坏现有页面，又提供新的渲染机制呢？

在写程序时我们也会经常遇到这样的问题，如何保证原来的接口不变，又提供更强大的功能，尤其是新功能不兼容旧功能时。遇到这种问题时的一个常见做法是增加参数和分支，即当某个参数为真时，我们就使用新功能，而如果这个参数不为真时，就使用旧功能，这样就能不破坏原有的程序，又提供新功能。IE6也是类似这样做的，它将DTD当成了这个“参数”，因为以

前的页面大家都不会去写DTD，所以IE6就假定 如果写了DTD，就意味着这个页面 将采用对CSS支持更好的布局，而如果没有，则采用兼容之前的布局方式。这就是 Quirks 模式（怪癖模式，诡异模式，怪异模式）。

区别：总体会有布局、样式解析和脚本执行三个方面的区别：

盒模型：在 W3C 标准中，如果设置一个元素的宽度和高度，指的是元素内容的宽度和高度，而在Quirks 模式下，IE的宽度和高度还包含了padding和border。



设置行内元素的高宽：在Standards模式下，给等行内元素设置width和height都不会生效，而在quirks模式下，则会生效。

设置百分比的高度：在 standards 模式下，一个元素的高度是由其包含的内容来决定的，如果父元素没有设置百分比的高度，子元素设置一个百分比的高度是无效的

margin:0 auto设置水平居中：使用margin:0 auto在standards模式下可以使元素水平居中，但在quirks模式下却会失效。

（还有很多，答出什么不重要，关键是看他答出的这些是不是自己经验遇到的，还是说都是看文章看的，甚至完全不知道。）

4、div+css的布局较table布局有什么优点？

改版的时候更方便 只要改css文件。 页面加载速度更快、结构化清晰、页面显示简洁。 表现与结构相分离。 易于优化（seo）搜索引擎更友好，排名更容易靠前。

5、img的alt与title有何异同？strong与em的异同？

a:alt(alt text):为不能显示图像、窗体或applets的用户代理（UA），alt属性用来指定替换文字。替换文字的语言由lang属性指定。（在IE浏览器下会在没有title时把alt当成 tool tip显示）

title(tool tip):该属性为设置该属性的元素提供建议性的信息。 strong:粗体

强调标签，强调，表示内容的重要性

em:斜体强调标签，更强烈强调，表示内容的强调点

6、你能描述一下渐进增强和优雅降级之间的不同吗？

渐进增强 progressive enhancement : 针对低版本浏览器进行构建页面，保证最基本的功能，然后再针对高级浏览器进行效果、交互等改进和追加功能达到更好的用户体验。

优雅降级 graceful degradation : 一开始就构建完整的功能，然后再针对低版本浏览器进行兼容。

区别：优雅降级是从复杂的现状开始，并试图减少用户体验的供给，而渐进增强则是从一个非常基础的，能够起作用的版本开始，并不断扩充，以适应未来环境的需要。降级（功能衰减）意味着往回看；而渐进增强则意味着朝前看，同时保证其根基处于安全地带。

“优雅降级”观点

“优雅降级”观点认为应该针对那些最高级、最完善的浏览器来设计网站。而将那些被认为“过时”或有功能缺失的浏览器下的测试工作安排在开发周期的最后阶段，并把测试对象限定为主流浏览器（如 IE、Mozilla 等）的前一个版本。

在这种设计范例下，旧版的浏览器被认为仅能提供“简陋却无妨 (poor, but passable)” 的浏览体验。你可以做一些小的调整来适应某个特定的浏览器。但由于它们并非我们所关注的焦点，因此除了修复较大的错误之外，其它的差异将被直接忽略。

“渐进增强”观点 “渐进增强”观点则认为应关注于内容本身。

内容是我们建立网站的诱因。有的网站展示它，有的则收集它，有的寻求，有的操作，还有的网站甚至会包含以上的种种，但相同点是它们全都涉及到内容。这使得“渐进增强”成为一种更为合理的设计范例。这也是它立即被 Yahoo! 所采纳并用以构建其“分级式浏览器支持 (Graded Browser Support)”策略的原因所在。

那么问题来了。现在产品经理看到IE6,7,8网页效果相对高版本现代浏览器少了很多圆角，阴影（CSS3），要求兼容（使用图片背景，放弃CSS3），你会如何说服他？

7、为什么利用多个域名来存储网站资源会更有效？

CDN 缓存更方便 突破浏览器并发限制

节约cookie带宽 节约主域名的连接数，优化页面响应速度 防止不必要的安全问题

8、请谈一下你对网页标准和标准制定机构重要性的理解。

网页标准和标准制定机构都是为了能让web发展的更‘健康’，开发者遵循统一的标准，降低开发难度，开发成本，SEO也会更好做，也不会因为滥用代码导致各种BUG、安全问题，最终提高网站易用性。

9、请描述一下cookies，sessionStorage和localStorage的区别？

sessionStorage（session）中的数据，这些数据只有在同一个会话中的页面才能访问并且当会话结束后数据也随之销毁。因此 sessionStorage 不是一种持久化的本地存储，仅仅是会话级别的存储。而 localStorage 用于持久化的本地存储，除非主动删除数据，否则数据是永远不会过期的。

web storage和cookie的区别

Web Storage的概念和cookie相似，区别是它是为了更大容量存储设计的。Cookie的大小是受限的，并且每次你请求一个新的页面的时候Cookie都会被发送过去，这样无形中浪费了带宽，另外cookie还需要指定作用域，不可以跨域调用。

除此之外，WebStorage拥有setItem,getItem,removeItem,clear等方法，

不像cookie 需要前端开发者自己封装setCookie , getCookie。但是Cookie也是不可以或缺的：Cookie 的作用是与服务器进行交互，作为HTTP规范的一部分而存在，而Web Storage仅仅是为了在本地“存储”数据而生。

10、简述一下src与href的区别。

src用于替换当前元素，href用于在当前文档和引用资源之间确立联系。

src是source的缩写，指向外部资源的位置，指向的内容将会嵌入到文档中当前标签所在 位置；在请求src资源时会将其指向的资源下载并应用到文档内，例如js脚本，img图片 和frame等元素。

`<script src = " js.js" ></script>` 当浏览器解析到该元素时，会暂停其他资源的下载和处理，直到将该资源加载、编译、执行

完毕，图片和框架等元素也如此，类似于将所指向资源嵌入当前标签内。这也是为什么将

js脚本放在底部而不是头部。

href是HypertextReference 的缩写，指向网络资源所在位置，建立和当前元素（锚点）或当前文档（链接）之间的链接，如果我们在文档中添加

`<link href=" common.css" rel=" stylesheet" />`

那么浏览器会识别该文档为 css 文件，就会并行下载资源并且不会停止对当前文档的处理。这也是为什么建议使用link方式来加载css，而不是使用@import方式。

11、知道的网页制作会用到的图片格式有哪些？

png-8，png-24，jpeg，gif，svg。但是上面的那些都不是面试官想要的最后答案。面试官希望听到是 Webp。（是否有关新技术，新鲜事物）科普一下 Webp：WebP 格式，谷歌（google）开发的一种旨在加快图片加载速度的图片格式。

图片压缩体积大约只有JPEG的2/3，并能节省大量的服务器带宽资源和数据空间。Facebook Ebay等知名网站已经开始测试并使用WebP格式。

在质量相同的情况下，WebP格式图像的体积要比JPEG格式图像小40%

12、知道什么是微格式吗？谈谈理解。在前端构建中应该考虑微格式吗？

微格式 (Microformats) 是一种让机器可读的语义化 XHTML 词汇的集合，是结构化数据的开放标准。是为特殊应用而制定的特殊格式。

优点：将智能数据添加到网页上，让网站内容在搜索引擎结果界面可以显示额外的提示。（应用范例：豆瓣，有兴趣自行google）

13、在css/js代码上线之后开发人员经常会优化性能，从用户刷新网页开始，一次js请求一般情况下有哪些地方会有缓存处理？

答案：dns缓存，cdn缓存，浏览器缓存，服务器缓存。

14、一个页面上有大量的图片（大型电商网站），加载很慢，你有哪些方法优化这些图片的加载，给用户更好的体验。

图片懒加载，在页面上的未可视区域可以添加一个滚动条事件，判断图片位置与浏览器顶端的距离与页面的距离，如果前者小于后者，优先加载。

如果为幻灯片、相册等，可以使用图片预加载技术，将当前展示图片的前一张和后一张优先下载。

如果图片为css图片，可以使用CSSsprite，SVGsprite，Iconfont、Base64等技术。如果图片过大，可以使用特殊编码的图片，加载时会先加载一张压缩的特别厉害的缩略图，以提高用户体验。

如果图片展示区域小于图片的真实大小，则因在服务器端根据业务需要先行进行图片压缩，图片压缩后大小与展示一致。

15、你如何理解HTML结构的语义化？

去掉或样式丢失的时候能让页面呈现清晰的结构：

html本身是没有表现的，我们看到例如<h1>是粗体，字体大小2em，加粗；是加粗 的，不要认为这是html的表现，这些其实html默认的css样式在起作用，所以去掉或样式 丢失的时候能让页面呈现清晰的结构不是语义化的HTML结构的优点，但是浏览器都有有默认样式，默认样式的目的也是为了更好的表达html的语义，可以说浏览器的默认样式和语义化的HTML结构是不可分割的。

屏幕阅读器（如果访客有视障）会完全根据你的标记来“读”你的网页。

例如,如果你使用的含语义的标记,屏幕阅读器就会“逐个拼出”你的单词,而不是试着去对它完整发音。

PDA、手机等设备可能无法像普通电脑的浏览器一样来渲染网页(通常是因为这些设备对CSS的支持较弱)

使用语义标记可以确保这些设备以一种有意义的方式来渲染网页.理想情况下,观看设备的任务是符合设备本身的条件来渲染网页。

语义标记为设备提供了所需的相关信息,就省去了你自己去考虑所有可能的显示情况(包括现有的或者将来新的设备)。例如,一部手机可以选择使一段标记了标题的文字以粗体显示.而掌上电脑可能会以比较大的字体来显示.无论哪种方式一旦你对文本标记为标题,您就可以确信读取设备将根据其自身的条件来合适地显示页面。

搜索引擎的爬虫也依赖于标记来确定上下文和各个关键字的权重 过去你可能还没有考虑搜索引擎的爬虫也是网站的“访客”,但现在它们实际上是极其宝贵的用户.没有他们的话,搜索引擎将无法索引你的网站,然后一般用户将很难过来访问。

你的页面是否对爬虫容易理解非常重要,因为爬虫很大程度上会忽略用于表现的标记,而只注重语义标记。

因此,如果页面文件的标题被标记,而不是,那么这个页面在搜索结果的位置可能会比较靠后.除了提升易用性外,语义标记有利于正确使用 CSS 和 JavaScript,因为其本身提供了许多“钩钩”来应用页面的样式与行为。

SEO 主要还是靠你网站的内容和外部链接的。便于团队开发和护

W3C 给我们定了一个很好的标准，在团队中大家都遵循这个标准，可以减少很多差异化的东西，方便开发和维护，提高开发效率，甚至实现模块化开发。

16、谈谈以前端角度出发做好SEO需要考虑什么？

了解搜索引擎如何抓取网页和如何索引网页 你需要知道一些搜索引擎的基本工作原理，各个搜索引擎之间的区别，搜索机器人

(SE robot或叫 web crawler)如何进行工作，搜索引擎如何对搜索结果进行排序等

等。

Meta标签优化主要包括主题 (Title) , 网站描述(Description) , 和关键词 (Keywords) 。
还有一些其它

的隐藏文字比如 Author (作者) , Category (目录) , Language (编码语种) 等。

如何选取关键词并在网页中放置关键词

搜索就得用关键词。关键词分析和选择是SEO最重要的工作之一。首先要给网站确定主关键词 (一般在5个上下) , 然后针对这些关键词进行优化, 包括关键词密度 (Density) 相关度 (Relavancy) , 突出性 (Prominency) 等等。

了解主要的搜索引擎 虽然搜索引擎有很多, 但是对网站流量起决定作用的就那么几个。比如英文的主要有

Google , Yahoo , Bing 等; 中文的有百度, 搜狗, 有道等。不同的搜索引擎对页面的抓取和

索引、排序的规则都不一样。还要了解各搜索门户和搜索引擎之间的关系, 比如AOL网页搜索用的是Google的搜索技术, MSN用的是Bing的技术。

主要的互联网目录

Open Directory自身不是搜索引擎, 而是一个大型的网站目录, 他和搜索引擎的主要区别是网站内容的收集方式不同。目录是人工编辑的, 主要收录网站主页; 搜索引擎是自动收集的, 除了主页外还抓取大量的内容页面。

按点击付费的搜索引擎 搜索引擎也需要生存, 随着互联网商务的越来越成熟, 收费的搜索引擎也开始大行其道。最

典型的有Overture和百度, 当然也包括Google的广告项目Google Adwords。越来越多的

人通过搜索引擎的点击广告来定位商业网站, 这里面也大有优化和排名的学问, 你得学会用最少的广告投入获得最多的点击。

搜索引擎登录 网站做完了以后, 别躺在那里等着客人从天而降。要让别人找到你, 最简单的办法就是将网

站提交 (submit) 到搜索引擎。如果你的商业网站，主要的搜索引擎和目录都会要求你付费来获得收录 (比如Yahoo要299美元)，但是好消息是 (至少到目前为止) 最大的搜索引擎Google目前还是免费，而且它主宰着60%以上的搜索市场。

链接交换和链接广泛度 (Link Popularity) 网页内容都是以超文本 (Hypertext) 的方式来互相链接的，网站之间也是如此。除了搜索

引擎以外，人们也每天通过不同网站之间的链接来 Surfing (“冲浪”)。其它网站到你的网站的链接越多，你也就会获得更多的访问量。更重要的是，你的网站的外部链接数越多，会被搜索引擎认为它的重要性越大，从而给你更高的排名。

17、有哪项方式可以对一个DOM设置它的CSS样式？

外部样式表，引入一个外部 css 文件 内部样式表，将 css 代码放在 <head> 标签内部 内联样式，将 css 样式直接定义在 HTML 元素内部

18、CSS都有哪些选择器？

派生选择器 (用HTML标签申明) id选择器 (用DOM的ID申明) 类选择器 (用一个样式类名申明)

属性选择器 (用DOM的属性申明，属于CSS2，IE6不支持，不常用，不知道就算了)

除了前3种基本选择器，还有一些扩展选择器，包括

后代选择器 (利用空格间隔，比如div .a{ }) 群组选择器 (利用逗号间隔，比如p,div,#a{ }) 那么问题来了，CSS 选择器的优先级是怎样定义的？

基本原则：

一般而言，选择器越特殊，它的优先级越高。也就是选择器指向的越准确，它的优先级就越高。

复杂的计算方法：

用1表示派生选择器的优先级

用10表示类选择器的优先级

用100标示ID选择器的优先级

div.test1 .span var 优先级 1+10 +10 +1

span#xxx .songs li 优先级 1+100 + 10 + 1

#xxx li 优先级 100 +1 那么问题来了，看下列代码，<p>标签内的文字是什么颜色的？

```
<style>
.classA{ color:blue;}
.classB{ color:red;}
</style>
<body>
<p class='classB classA'> 123 </p></body>
```

答案：red。与样式定义在文件中的先后顺序有关，即是后面的覆盖前面的，与在<p class=classB classA'>中的先后关系无关。

19、CSS中可以通过哪些属性定义，使得一个DOM元素不显示在浏览器可视范围内？

最基本的：

设置display属性为none，或者设置visibility属性为hidden 技巧性：

设置宽高为0，设置透明度为0，设置z-index位置在-1000

20、超链接访问过后hover样式就不出现的问题是什么？如何解决？

答案：被点击访问过的超链接样式不在具有hover和active了,解决方法是改变CSS属性的排列顺序：L-V-H-A (link,visited,hover,active)

21、什么是Css Hack？ie6,7,8的hack分别是什么？


```

1  /* Opera */
2  html:first-child #opera
3  {
4    display: block;
5  }
6  /* IE 7 */
7  html > body #ie7
8  {
9    *display: block;
10 }
11 /* IE 6 */
12 body #ie6
13 {
14   _display: block;
15 }
16 /* Firefox 1 - 2 */
17 body:empty #firefox12
18 {
19   display: block;
20 }
21 /* Firefox */
22 @-moz-document url-prefix()
23 {
24   #firefox { display: block; }
25 }
26 /* Safari */
27 @media screen and (-webkit-min-device-pixel-ratio:0)
28 {
29   #safari { display: block; }
30 }
31 /* Opera */
32 @media all and (-webkit-min-device-pixel-ratio:10000), not all and (-webkit-min-device-pixel-ratio:0)
33 {
34   head~body #opera { display: block; }
35 }
36

```

22、请用Css写一个简单的幻灯片效果页面

答案：知道是要用css3。使用animation动画实现一个简单的幻灯片效果。

```

/*定义动画*/
@keyframes marginLeft{
  0%{margin-left: 0;}
  28.5%{margin-left: 0;}
  33.3%{margin-left: -600px;}
  62%{margin-left: -600px;}
  66.7%{margin-left: -1200px;}
  95.2%{margin-left: -1200px;}
  100%{margin-left: 0;}
}

```

```

#slideInput1:checked ~ ul{ margin-left: 0;}
#slideInput2:checked ~ ul{ margin-left: -600px;}
#slideInput3:checked ~ ul{ margin-left: -1200px;}

```

```
<style>
  body {margin: 0}
  .slide{
    position: relative;
    margin:auto;
    width: 600px;
    height: 200px;

    /*整屏的效果*/
    /*width: 300vw;*/
    /*height: 100vh;*/

    text-align: center;
    font-family: Arial;
    color: #fff8f9;
    overflow: hidden;
  }
  .slide ul{
    margin: 0;
    padding:0;
    /*width: 9999px;*/
    transition:all 0.5s;
  }
}
```

```
<!-- 点击轮播 -->
<div class="slide">

  <input type="radio" name="slideInput" value="0" id="slideInput1" checked hidden>
  <label for="slideInput1">1</label>

  <input type="radio" name="slideInput" value="1" id="slideInput2" hidden>
  <label for="slideInput2">2</label>

  <input type="radio" name="slideInput" value="1" id="slideInput3" hidden>
  <label for="slideInput3">3</label>

  <ul>
    <li>one-大家好我是第一个ppt页面,</li>
    <li>two-哈哈我是第二个~~~</li>
    <li>three-第三个就是我</li>
  </ul>

</div>
```

23、行内元素和块级元素的具体区别是什么？行内元素的padding和margin可设置吗？

块级元素(block)特性：总是独占一行，表现为另起一行开始，而且其后的元素也必须另起一行显示；

宽度(width)、高度(height)、内边距(padding)和外边距(margin)都可控制；
内联元素(inline)特性：

和相邻的内联元素在同一行；

宽度(width)、高度(height)、内边距的 top/bottom(padding-top/padding-bottom)和外边距的top/bottom(margin-top/margin-bottom)都不可改变（也就是padding和margin的left和right是可以设置的），就是里面文字或图片的大小。那么问题来了，浏览器还有默认的天生 inline-block 元素（拥有内在尺寸，可设置高宽，但不会自动换行），有哪些？

答案：<input>、、<button>、<textarea>、<label>。

24、什么是外边距重叠？重叠的结果是什么？

外边距重叠就是margin-collapse。

在CSS当中，相邻的两个盒子（可能是兄弟关系也可能是祖先关系）的外边距可以结合成一个单独的外边距。这种合并外边距的方式被称为折叠，并且因而所结合成的外边距称为折叠外边距。

折叠结果遵循下列计算规则：两个相邻的外边距都是正数时，折叠结果是它们两者之间较大的值。

两个相邻的外边距都是负数时，折叠结果是两者绝对值的较大值。两个外边距一正一负时，折叠结果是两者的相加的和。

25、rgba()和opacity的透明效果有什么不同？

rgba()和opacity都能实现透明效果，但最大的不同是opacity作用于元素，以及元素内的所有内容的透明度，

而`rgba()`只作用于元素的颜色或其背景色。（设置`rgba`透明的元素的子元素不会继承透明

效果！）

26、css中可以让文字在垂直和水平方向上重叠的两个属性是什么？

垂直方向：`line-height` 水平方向：`letter-spacing`

那么问题来了，关于`letter-spacing`的妙用知道有哪些么？答案:可以用于消除`inline-block`元素间的换行符空格间隙问题。

27、如何垂直居中一个浮动元素？

```
// 方法一：已知元素的高宽
#div1{
  background-color:#6699FF;
  width:200px;
  height:200px;
  position: absolute; //父元素需要相对定位 7   top: 50%;
  left: 50%;
  margin-top:-100px ; //二分之一的 height, width;
  margin-left: -100px;
}
```

```
/方法二:未知元素的高宽

#div1{
  width: 200px;
  height: 200px;
  background-color: #6699FF;
  margin:auto;
  position: absolute; //父元素需要相对定位;
  left: 0;
  top: 0;
  right: 0;
  bottom: 0;
}
```

那么问题来了，如何垂直居中一个? (用更简便的方法。)

```
#container //<img>的容器设置如下
{
  display:table-cell;
  text-align:center;
  vertical-align:middle;
}
```

28、px和em的区别。

px和em都是长度单位，区别是，px的值是固定的，指定是多少就是多少，计算比较容易。em得值不是固定的，并且em会继承父级元素的字体大小。

浏览器的默认字体高都是 16px。所以未经调整的浏览器都符合：1em=16px。那么 12px=0.75em, 10px=0.625em。

29、描述一个“reset”的CSS文件并如何使用它。知道normalize.css吗？你了解他们的不同之处？

重置样式非常多，凡是一个前端开发人员肯定有一个常用的重置CSS文件并知道如何使用它们。他们是盲目的在做还是知道为什么这么做呢？原因是不同的浏览器对一些元素有不同的默认样式，如果你不处理，在不同的浏览器下会存在必要的风险，或者更有戏剧性的性发生。

你可能会用 Normalize 来代替你的重置样式文件。它没有重置所有的样式风格，但仅提供了一套合理的默认样式值。既能让众多浏览器达到一致和合理，但又不扰乱其他的东西（如粗体的标题）。在这一方面，无法做每一个复位重置。它也确实有些超过一个重置，它处理了你永远都不用考虑的怪癖，像HTML的audio元素不一致或line-height不一致。

30、Sass、LESS是什么？大家为什么要使用他们？

他们是CSS预处理器。他是CSS上的一种抽象层。他们是一种特殊的语法/语言编译成CSS。例如Less是一种动态样式语言。将CSS赋予了动态语言的特性，如变量，继承，运算，函

数. LESS 既可以在客户端上运行 (支持IE 6+, Webkit, Firefox) , 也可一在
服务端运行

(借助 Node.js)。为什么要使用它们？

结构清晰，便于扩展。

可以方便地屏蔽浏览器私有语法差异。这个不用多说，封装对浏览器语法差异的重复处理，减少无意义的机械劳动。

可以轻松实现多重继承。

完全兼容 CSS 代码，可以方便地应用到老项目中。LESS 只是在 CSS 语法上做了扩展，所以老的 CSS 代码也可以与 LESS 代码一同编译。

Ps:关于 sass 和 less 可以查找我写的简单教程

31、display:none与visibility:hidden的区别是什么？

display : 隐藏对应的元素但不挤占该元素原来的空间。 visibility: 隐藏对应的元素并且挤占该元素原来的空间。

即是，使用CSSdisplay:none属性后，HTML元素（对象）的宽度、高度等各种属性值都将“丢失”；而使用visibility:hidden属性后，HTML元素（对象）仅仅是在视觉上看不见（完全透明），而它所占据的空间位置仍然存在。

32、CSS中link和@import的区别是：

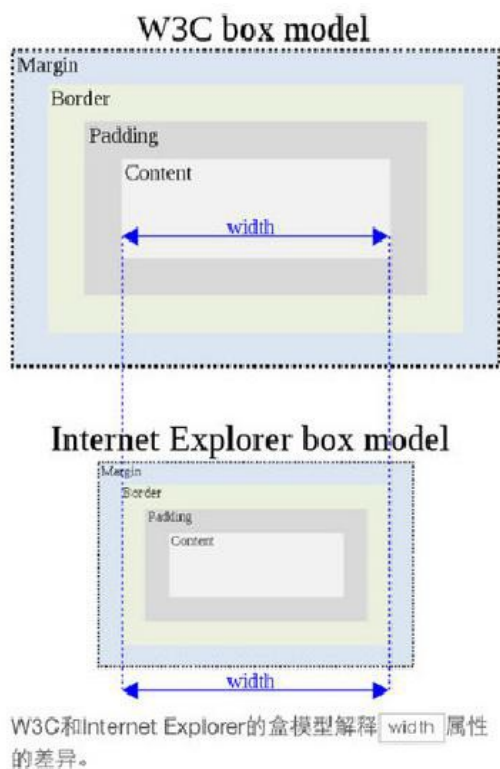
Link 属于html 标签，而@import 是CSS 中提供的

在页面加载的时候，link会同时被加载，而@import引用的CSS会在页面加载完成后才会加载引用的CSS

@import只有在ie5以上才可以被识别，而link是html标签，不存在浏览器兼容性问题 Link引入样式的权重大于@import的引用（@import是将引用的样式导入到当前的页面中）

33、简介盒子模型：

CSS 的盒子模型有两种：IE 盒子模型、标准的 W3C 盒子模型模型 盒模型：内容、内边距、外边距（一般不计入盒子实际宽度）、边框



34、为什么要初始化样式？

由于浏览器兼容的问题，不同的浏览器对标签的默认样式值不同，若不初始化会造成不同浏览器之间的显示差异

但是初始化CSS会对搜索引擎优化造成小影响

35、BFC是什么？

BFC（块级格式化上下文），一个创建了新的BFC的盒子是独立布局的，盒子内元素的布局不会影响到盒子外面的元素。在同一个BFC中的两个相邻的盒子在垂直方向发生margin重叠的问题

BFC 是指浏览器中创建了一个独立的渲染区域，该区域内所有元素的布局不会影响到区域外元素的布局，这个渲染区域只对块级元素起作用

36、html语义化是什么？

当页面样式加载失败的时候能够让页面呈现出清晰的结构

有利于 seo 优化，利于被搜索引擎收录（更便于搜索引擎的爬虫程序来识别）
便于项目的开发及维护，使html代码更具有可读性，便于其他设备解析。

37、Doctype的作用？严格模式与混杂模式的区别？

<!DOCTYPE>用于告知浏览器该以何种模式来渲染文档 严格模式下：页面排版及JS解析是以该浏览器支持的最高标准来执行 混杂模式：不严格按照标准执行，主要用来兼容旧的浏览器，向后兼容

38、IE的双边距BUG：块级元素float后设置横向margin，ie6显示的margin 比设置的较大。

解决：加入_display：inline

39、HTML与XHTML——二者有什么区别？

- 1.所有的标记都必须要有个相应的结束标记
- 2.所有标签的元素和属性的名字都必须使用小写
- 3.所有的 XML 标记都必须合理嵌套
- 4.所有的属性必须用引号 "" 括起来
- 5.把所有 < 和 & 特殊符号用编码表示
- 6.给所有属性赋一个值
- 7.不要在注释内容中使用 "--"
- 8.图片必须有说明文字

40、html常见兼容性问题？

- 1.双边距BUG float引起的 使用display
- 2.3像素问题 使用float引起的 使用display:inline -3px
- 3.超链接hover 点击后失效 使用正确的书写顺序 link visited hover active
- 4.IE z-index问题 给父级添加position:relative
- 5.Png 透明 使用js代码 改
- 6.Min-height 最小高度 ! Important 解决'
- 7.select 在ie6下遮盖 使用

iframe嵌套

8.为什么没有办法定义 1px 左右的宽度容器（IE6 默认的行高造成的，使用 over:hidden,zoom:0.08 line-height:1px）

9.IE5-8不支持opacity，解决办法：

```
.opacity {
  opacity: 0.4;
  filter: alpha(opacity=60); /* for IE5-7 */
  -ms-filter: "progid:DXImageTransform.Microsoft.Alpha(Opacity=60)"; /* for IE8 */
}
```

10. IE6不支持PNG透明背景，解决办法: IE6下使用gif图片

41、对WEB标准以及W3C的理解与认识

答：标签闭合、标签小写、不乱嵌套、提高搜索机器人搜索几率、使用外链css和js脚本、结构行为表现的分离、文件下载与页面速度更快、内容能被更多的用户所访问、内容能被更广泛的设备所访问、更少的代码和组件，容易维护、改版方便，不需要变动页面内容、提供打印版本而不需要复制内容、提高网站易用性。

42、行内元素有哪些?块级元素有哪些?CSS的盒模型?

答：块级元素：div p h1 h2 h3 h4 form ul 行内元素: a b br i span input select

Css盒模型:内容，border,margin，padding

43、前端页面有哪三层构成，分别是什么?作用是什么?

答：结构层 Html 表示层 CSS 行为层 js。

44、Doctype作用? 严格模式与混杂模式-如何触发这两种模式，区分它们有何意义?

(1)、<!DOCTYPE> 声明位于文档中的最前面，处于 <html> 标签之前。告知浏览器的解析器，用什么文档类型规范来解析这个文档。

(2)、严格模式的排版和 JS 运作模式是以该浏览器支持的最高标准运行。

(3)、在混杂模式中，页面以宽松的向后兼容的方式显示。模拟老式浏览器的行为以防止站点无法工作。

(4)、DOCTYPE 不存在或格式不正确会导致文档以混杂模式呈现。

45、行内元素有哪些？块级元素有哪些？空(void)元素有那些？

(1) CSS规范规定，每个元素都有display属性，确定该元素的类型，每个元素都有默认的display值，比如div默认display属性值为“block”，成为“块级”元素；span默认display属性值为“inline”，是“行内”元素。

(2) 行内元素有：a bspan img input select strong (强调的语气) 块级元素有：div ul ol li dl dt dd h1 h2 h3 h4...p

(3) 知名的空元素：

 <hr> <input> <link> <meta> 鲜为人知的是：

<area> <base> <col> <command>

<embed> <keygen> <param> <source> <track> <wbr>

46、CSS的盒子模型？

(1) 两种，IE盒子模型、标准W3C盒子模型；IE的content部分包含了border和padding;

(2) 盒模型：内容(content)、填充(padding)、边界(margin)、边框(border).

47、CSS选择符有哪些？哪些属性可以继承？优先级算法如何计算？CSS3新增伪类有那些？

1.id选择器 (# myid)

2.类选择器 (.myclassname)

3.标签选择器 (div, h1, p)

4.相邻选择器 (h1 + p)

5.子选择器 (ul < li)

6.后代选择器 (li a)

7.通配符选择器 (*)

8.属性选择器 (a[rel = "external"])

9.伪类选择器 (a: hover, li: nth - child)

* 可继承 : font-size font-family color, UL LI DL DD DT;

* 不可继承 : border padding margin width height ;

* 优先级就近原则 , 样式定义最近者为准;

* 载入样式以最后载入的定位为准; 优先级为:

!important > id > class > tag important 比 内联优先级高

CSS3新增伪类举例 :

p:first-of-type 选择属于其父元素的首个 <p> 元素的每个 <p> 元素。

p:last-of-type 选择属于其父元素的最后 <p> 元素的每个 <p> 元素。

p:only-of-type 选择属于其父元素唯一的 <p> 元素的每个 <p> 元素。

p:only-child 选择属于其父元素的唯一子元素的每个 <p> 元素。

p:nth-child(2) 选择属于其父元素的第二个子元素的每个 <p> 元素。

:enabled、:disabled控制表单控件的禁用状态。

:checked , 单选框或复选框被选中。

48、如何居中div,如何居中一个浮动元素?

给div设置一个宽度 , 然后添加margin:0 auto属性

```
div{  
width:200px; margin:0 auto;  
}
```

居中一个浮动元素

确定容器的宽高 宽500 高 300 的层 设置层的外边距

```
.div {  
Width:500px ; height:300px;//高度可以不设 Margin: -150px 0 0 -  
250px; position:relative;相对定位  
background-color:pink;//方便看效果 left:50%;
```

```
top:50%;  
}
```

49、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?原因,解决方法是什么,常用hack的技巧?

- * IE浏览器的内核Trident、Mozilla的Gecko、google的WebKit、Opera内核Presto;

- * png24为的图片在ie6浏览器上出现背景,解决方案是做成PNG8.

- * 浏览器默认的 margin 和 padding 不同。解决方案是加一个全局的

- * {margin:0;padding:0;}来统一。

- * IE6双边距bug:块属性标签float后,又有横行的margin情况下,在ie6显示margin 比设置的大。

浮动ie产生的双倍距离 #box{ float:left; width:10px; margin:0 0 0 100px;}
这种情况之下IE会产生20px的距离,解决方案是在float的标签样式控制中加入 —

—_display:inline;将其转化为行内属性。(_这个符号只有ie6会识别) 渐进识别的方式,从总体中逐渐排除局部。 首先,巧妙的使用 “\9” 这一标记,将IE浏览器从所有情况中分离出来。接着,再次使用 “+” 将IE8和IE7、IE6分离开来,这样IE8已经独立识别。 css

```
.bb{  
background-color:#f1ee18;/*所有识别*/  
.background-color:#00deff\9;/*IE6、7、8识别*/  
+background-color:#a200ff;/*IE6、7 识别*/  
_background-color:#1e0bd1;/*IE6 识别*/  
}
```

- * IE 下,可以使用获取常规属性的方法来获取自定义属性,也可以使用getAttribute()获取自定义属性;

Firefox下,只能使用getAttribute()获取自定义属性. 解决方法:统一通过

getAttribute()获取自定义属性.

- * IE下,event对象有x,y属性,但是没有pageX,pageY属性;
Firefox下,event对象有pageX,pageY属性,但是没有x,y属性.
- * (条件注释) 缺点是在IE浏览器下可能会增加额外的HTTP请求数。
- * Chrome 中文界面下默认会将小于 12px 的文本强制按照 12px 显示,
可通过加入 CSS 属性 -webkit-text-size-adjust: none; 解决.
- 超链接访问过后hover样式就不出现了 被点击访问过的超链接样式不在
active了解决方法是改变CSS属性的排列顺序:
L-V-H-A : a:link {} a:visited {} a:hover {} a:active {}

50、列出display的值,说明他们的作用。position的值,relative 定位原点是？

1. block 象块类型元素一样显示。

none缺省值。向行内元素类型一样显示。

inline-block象行内元素一样显示，但其内容象块类型元素一样显示。

list-item象块类型元素一样显示，并添加样式列表标记。

2. position 的值

*absolute

生成绝对定位的元素，相对于 static 定位以外的第一个父元素进行定位。

*fixed （老IE 不支持）

生成绝对定位的元素，相对于浏览器窗口进行定位。

* relative

生成相对定位的元素，相对于其正常位置进行定位。

* static 默认值。没有定位，元素出现在正常的流中

* (忽略 top, bottom, left, right z-index 声明) 。

* inherit 规定从父元素继承 position 属性的值。

51、absolute的containing block计算方式跟正常流有什么不同？

无论属于哪种,都要先找到其祖先元素中最近的 position 值不为 static 的元素,然后再判断：

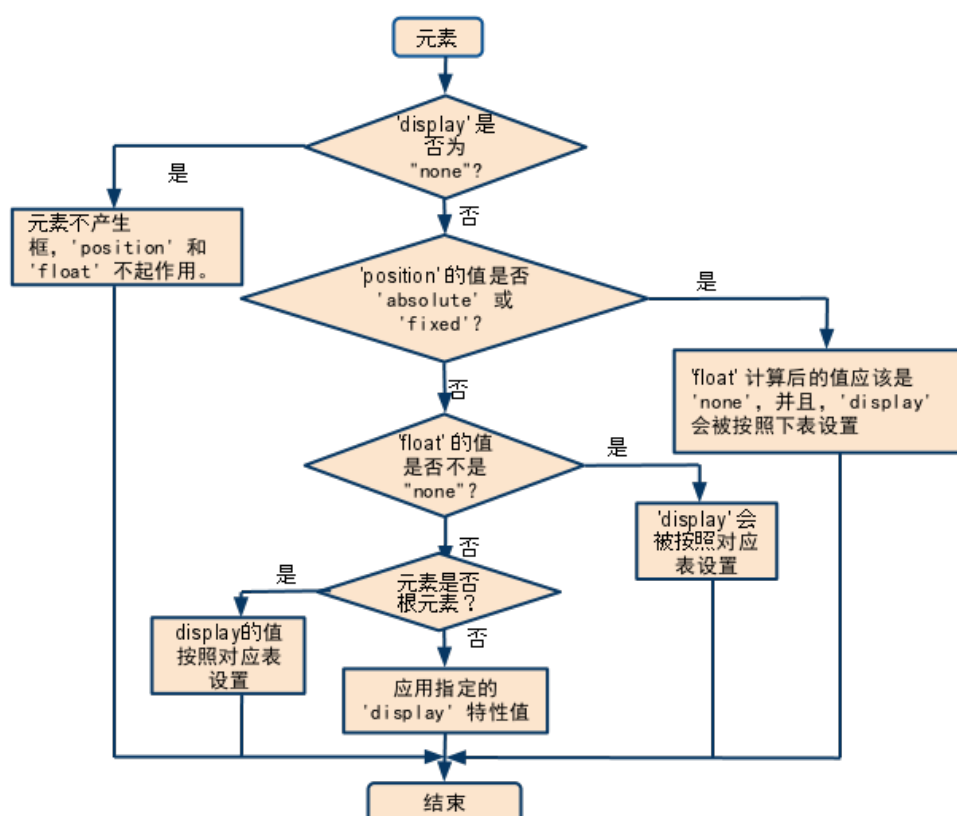
- 1、若此元素为 inline 元素,则 containing block 为能够包含这个元素生成的第一个和最后一个 inline box 的 padding box (除 margin, border 外的区域) 的最小矩形；
- 2、否则,则由这个祖先元素的 padding box 构成。如果都找不到,则为 initial containing block。

补充：

1. static(默认的)/relative：简单说就是它的父元素的内容框(即去掉 padding 的部分)
2. absolute: 向上找最近的定位为 absolute/relative 的元素
3. fixed: 它的 containing block 一律为根元素(html/body),根元素也是 initialcontaining block

52、position 跟 display、margin collapse、overflow、float 这些特性相互叠加后会怎么样？

我画一个图，也许你看了就懂



53、对WEB标准以及W3C的理解与认识

标签闭合、标签小写、不乱嵌套、提高搜索机器人搜索几率、使用外链css和js脚本、结构行为表现的分离、文件下载与页面速度更快、内容能被更多的用户所访问、内容能被更广泛的设备所访问、更少的代码和组件，容易维护、改版方便，不需要变动页面内容、提供打印版本而不需要复制内容、提高网站易用性；

54、css的基本语句构成是？

选择器{属性1:值1;属性2:值2;.....}

55、浏览器标准模式和怪异模式之间的区别是什么？

盒子模型 渲染模式的不同

使用 window.top.document.compatMode 可显示为什么模式

56、css优先级算法如何计算？

!important > id > class > 标签

!important 比 内联优先级高

*优先级就近原则，样式定义最近者为准;

*以最后载入的样式为准;

57、b标签和strong标签,i标签和em标签的区别？

后者有语义，前者则无。

58、我想让行内元素跟上面的元素距离 10px，加margin-top和padding-top可以吗？

margin-top,padding-top无效

59、text-align:center和line-height有什么区别？

text-align是水平对齐，line-height是行间。

二、JS 基础

1、javascript的typeof返回哪些数据类型

```

object number function boolean underfind
string typeof null;//
object typeof isNaN;//
typeof isNaN(123) typeof [];//
object Array.isArray(); es5
toString.call([]);//
"[object Array]" var arr=[];
arr.constructor;//Array

```

2、例举3种强制类型转换和2种隐式类型转换？

强制 (parseInt,parseFloat,Number()) 隐式 (==)

1==" 1" //true null==undefined//true

3、split() join() 的区别

前者是切割成数组的形式，后者是将数组转换成字符串

4、数组方法pop() push() unshift() shift()

Push()尾部添加 pop()尾部删除 Unshift()头部添加 shift()头部删除

5、事件绑定和普通事件有什么区别

传统事件绑定和符合W3C 标准的事件绑定有什么区别？

```
div1.onclick=function(){};
```

```
<button onmouseover=" " ></button>
```

1、如果说给同一个元素绑定了两次或者多次相同类型的事件，那么后面的绑定会覆盖前面 的绑定

2、不支持**DOM 事件流** 事件捕获阶段□ 目标元素阶段=>事件冒泡阶段

addEventListener

1、如果说给同一个元素绑定了两次或者多次相同类型的事件，所有的绑定将会依次触发

2、支持DOM 事件流的

3、进行事件绑定传参不需要on 前缀

`addEventListener("click" ,function(){},true);` //此时的事件就是在事件冒泡阶段执行

ie9 开始 , ie11 edge : `addEventListener` ie9 以前 :

`attachEvent/detachEvent`

- 1、进行事件类型传参需要带上 `on` 前缀
- 2、这种方式只支持事件冒泡，不支持事件捕获

事件绑定是指把事件注册到具体的元素之上，普通事件指的是可以用来注册的事件

6、IE 和DOM 事件流的区别

- 1.执行顺序不一样、
- 2.参数不一样
- 3.事件加不加 `on` 4.this指向问题

IE9 以前 : `attachEvent("onclick")`、`detachEvent("onclick")` **IE9开始** 跟DOM事件流是一样的，都是`addEventListener`

```

IE
    在IE中，要阻止事件的进一步冒泡，必须设置cancelBubble属性为true
    oEvent.cancelBubble=true;

DOM
    在mozilla中，只需要调用stopPropagation()方法即可
    oEvent.stopPropagation()

<html onclick="alert('html')">
  <body onclick="alert('body')">
    <input type=button value="click" onclick="alert('button')">

  </body>
</html>
    点击button会依次出现 button,body,html的提示
    如果增加如下函数，则可阻止事件的复制(冒泡)

function handleClick(oEvent)
{
    alert('button');
    if(window.event)
    {
        oEvent=window.event;
        oEvent.cancelBubble=true;
    }
    else
    {
        oEvent.stopPropagation();
    }
}

<input type=button value="click" onclick="handleClick(event)">
    则只会出现 button的提示

```

7、IE和标准下有哪些兼容性的写法

```
var ev = ev || window.event
```

```
document.documentElement.clientWidth ||
```

```
document.body.clientWidth
```

```
var target = ev.srcElement||ev.target
```

8、call 和apply 的区别

call和apply相同点：都是为了用一个本不属于一个对象的方法，让这个对象去执行

```
toString.call([],1,2,3)
```

```
toString.apply([], [1,2,3]) Object.call(this,obj1,obj2,obj3)
```

```
Object.apply(this,arguments)
```

9、b 继承a 的方法

考点：继承的多种方式 `function b(){ } b.prototype=new a;`

10、JavaScript this 指针、闭包、作用域

this：指向调用上下文 闭包：内层作用域可以访问外层作用域的变量

作用域：定义一个函数就开辟了一个局部作用域，整个 js 执行环境有一个全局作用域

11、事件委托是什么

符合W3C 标准的事件绑定 `addEventListener /attachEvent`

利用事件冒泡的原理，让自己的所触发的事件，让他的父元素代替执行！

12、闭包是什么，有什么特性，对页面有什么影响

闭包就是能够读取其他函数内部变量的函数。 闭包的缺点：滥用闭包函数会造成内存泄露，因为闭包中引用到的包裹函数中定义的变量都 永远不会被释放，所以我们应该在必要的时候，及时释放这个闭包函数

13、如何阻止事件冒泡和默认事件

`e.stopPropagation();`//标准浏览器

`event.canceBubble=true;`//ie9之前

阻止默认事件：

为了不让a点击之后跳转，我们就要给他的点击事件进行阻止 `return false`

`e.preventDefault();`

14、添加 删除 替换 插入到某个接点的方法

obj.appendChild()

obj.insertBefore() //原生的js中不提供insertAfter();

obj.replaceChild()//替换

obj.removeChild()//删除

15、javascript的本地对象，内置对象和宿主对象

本地对象为array obj regexp等可以new实例化 内置对象为global Math等不可以实例化的 宿主为浏览器自带的document>window等

16、document load 和document ready的区别

Document.onload 是在结构和样式加载完才执行js window.onload：不仅仅要在结构和样式加载完，还要执行完所有的样式、图片这些资源文件，全部加载完才会触发window.onload 事件

Document.ready原生中没有这个方法，jquery中有\$.ready(function)

17、“==” 和 “===” 的不同

前者会自动转换类型 后者不会

1==" 1"

null==undefined

===先判断左右两边的数据类型，如果数据类型不一致，直接返回 false 之后才会进行两边值的判断

18、javascript的同源策略

一段脚本只能读取来自于同一样源的窗口和文档的属性，这里的同一样源指的是主机名、协议和端口号的组合

http,ftp:协议 主机名 ; localhost

端口名 : 80

同源策略带来的麻烦：ajax在不0:http协议的默认端口 https:默认端口是8083

同域名下的请求无法实现，

如果说想要请求其他来源的js文件，或者json数据，那么可以通过jsonp来解决

19、编写一个数组去重的方法

```
var arr=[1,1,3,4,2,4,7];
```

```
=>[1,3,4,2,7]
```

一个比较简单的实现就是：

- 1、先创建一个空数组，用来保存最终的结果
- 2、循环原数组中的每个元素
- 3、再对每个元素进行二次循环，判断是否有与之相同的元素，如果没有，将把这个元素放 到新数组中
- 4、返回这个新数组

```
function oSort(arr) { var result = {};  
var newArr=[];  
for(var i=0;i<arr.length;i++){ if(!result[arr]) { newArr.push(arr)  
result[arr]=1  
}  
}  
return newArr
```



```
}</arr.length;i++)
```

20、JavaScript是一门什么样的语言，它有哪些特点？

没有标准答案。

运行环境：浏览器中的JS引擎（v8。。）

语言特性：面向对象，动态语言：

//动态语言的特性

var num=10;//num 是一个数字类型

num="jim";//此时 num 又变成一个字符串类型

//我们把一个变量用来保存不同数据类型的语言称之为一个动态语言

//静态语言：c# java c c++

//静态语言在声明一个变量就已经确定了这个变量的数据类型，

// 而且在任何时候都不可以改变他的数据类型

21、JavaScript 的数据类型都有什么？

基本数据类型：String,Boolean,number,undefined,object,Null 引用数据类型：Object(Array,Date,RegExp,Function) 那么问题来了，如何判断某变量是否为数组数据类型？

方法一.判断其是否具有“数组性质”，如slice()方法。可自己给该变量定义slice方法，故有时会失效

方法二.obj instanceof Array 在某些IE版本中不正确

方法三.方法一二皆有漏洞，在ECMAScript5中定义了新方法

Array.isArray(), 保证其兼容性，最好的方法如下：

toString.call(18);// "[object Number]" toString.call("");// "[object String]" 解析这种简单的数据类型直接通过typeof就可以直接判断

toString.call 常用于判断数组、正则这些复杂类型

```
toString.call(/[0-9]{10}/)///" [object RegExp]"
```

```
if(typeof Array.isArray=== "undefined"){ Array.isArray = function(arg){  
return Object.prototype.toString.call(arg)=== "[object Array]"  
};  
}
```

22、已知ID的Input输入框，希望获取这个输入框的输入值，怎么做？(不使用第三方框架)

```
document.getElementById( "ID" ).value
```

23、希望获取到页面中所有的checkbox怎么做？(不使用第三方框架)

```
var domList = document.getElementsByTagName( 'input' ) var  
checkBoxList = []; //返回的所有的checkbox  
var len = domList.length; //缓存到局部变量  
while (len--) { //使用while的效率会比for循环更高 if (domList[len].type ==  
    'checkbox' ) {  
    checkBoxList.push(domList[len]);  
    }  
}
```

24、设置一个已知ID的DIV的html内容为xxxx，字体颜色设置为黑色(不使用第三方框架)

```
var dom = document.getElementById( "ID" ); dom.innerHTML = "xxxx"  
dom.style.color = "#000"
```

25、当一个DOM节点被点击时候，我们希望能够执行一个函数，应该怎么做？

直接在DOM里绑定事件：`<div onclick=" test()" ></div>` 在JS里通过
onclick绑定：`xxx.onclick = test`

通过事件添加进行绑定：`addEventListener(xxx, 'click', test)` 那么问题来了，
Javascript 的事件流模型都有什么？

“事件冒泡”：事件开始由最具体的元素接受，然后逐级向上传播

“事件捕捉”：事件由最不具体的节点先接收，然后逐级向下，一直到最具体的

“DOM 事件流”：三个阶段：事件捕捉，目标阶段，事件冒泡

26、看下列代码输出为何？解释原因。

```
var a;  
alert(typeof a); // "undefined"  
//alert(b); // 报错 b=10;  
alert(typeof b);// " number"
```

解释：Undefined 是一个只有一个值的数据类型，这个值就是“undefined”，在使用 var 声明变量但并未对其赋值进行初始化时，这个变量的值就是 undefined。而b由于未声明将 报错。注意未声明的变量和声明了未赋值的是不一样的。

undefined 会在以下三种情况下产生：1、 一个变量定义了却没有被赋值

2、 想要获取一个对象上不存在的属性或者方法:

3、 一个数组中没有被赋值的元素

注意区分undefined跟not defnied(语法错误)是不一样的

27、看下列代码,输出什么？解释原因。

```
var a = null; alert(typeof a); //object
```

解释：null 是一个只有一个值的数据类型，这个值就是 null。表示一个空指针对象，所以用typeof检测会返回“object”。

28、看下列代码,输出什么？解释原因。

```
var undefined;//此时undefined这个变量的值是undefined
```

```
undefined == null; // true
```

```
1 == true; // true 此时会把布尔类型的值转换为数字类型 true=1 false=0
```

```
2 == true; // false
```

```
0 == false; // true
```

```
0 == ""; // true NaN == NaN; // false [] == false; // true
```

```
[] == ![]; // true
```

undefined与null相等，但不恒等（===）

一个是number一个是string时，会尝试将string转换为number 尝试将boolean转换为number，0或1

尝试将Object转换成number或string，取决于另外一个对比量的类型

所以，对于0、空字符串的判断，建议使用“===”。 “===”会先判断两边的值类型，类型不匹配时为false。

那么问题来了，看下面的代码，输出什么，foo 的值为什么？

```
var foo = "11"+2-"1";
```

```
console.log(foo); //111
```

```
console.log(typeof foo);
```

执行完后foo的值为111，foo的类型为number。

29、看代码给答案。

```
var a = new Object();  
a.value = 1;  
b = a;  
{value:1} b.value = 2;  
alert(a.value); //2
```

答案：2（考察引用数据类型细节）

30、已知数组如下：

```
var strArrar = ['This','is','Baidu','Campus'];  
alert('This is Baidu Campus');
```

该如何实现 alert 的结果：

答案：alert(stringArray.join(""))

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" getElementByld" 。

```
function combo(msg){  
  var arr=msg.split("-");//[get,element,by,id] for(var i=1;i<arr.length;i++){  
    arr[i]=arr[i].charAt(0).toUpperCase()+arr[i].substr(1,arr[i].length  
-1);//Element  
  }  
  msg=arr.join(""); //msg=" getElementByld" return msg;  
}
```

(考察基础API)

31、var numberArray = [3,6,2,4,1,5];

- 1) 实现对该数组的倒排，输出[5,1,4,2,6,3]
- 2) 实现对该数组的降序排列，输出[6,5,4,3,2,1]

```
function combo(msg){  
  var arr=msg.split("-");  
  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("");  
  return msg;  
}
```

32、输出今天的日期，以YYYY-MM-DD的方式，比如今天是2017年9月26日，则输出2017-09-26

```
var d = new Date();
// 获取年，getFullYear()返回 4 位的数字 var year = d.getFullYear();
// 获取月，月份比较特殊，0 是 1 月，11 是 12 月
var month = d.getMonth() + 1;
// 变成两位
month = month < 10 ? '0' + month : month;
// 获取日
var day = d.getDate();
day = day < 10 ? '0' + day : day;

alert(year + '-' + month + '-' + day);
```

33、将字符串“<tr><td>{\$id}</td><td>{\$name}</td></tr>”中的{\$id}替换成10，{\$name}替换成Tony（使用正则表达式）

答案：“<tr><td>{\$id}</td><td>{\$id}_{\$name}</td></tr>”.replace(/{\\$id}/g, '10').replace(/{\\$name}/g, 'Tony');

34、为了保证页面输出安全，我们经常需要对一些特殊的字符进行转义
请写一个函数escapeHtml，将<, >, &, “进行转义

```
function escapeHtml(str) {
  // [ < > " & ] : 中括号中字符只要其中的一个出现就代表满足条件
  // 给 replace 第二个参数传递一个回调函数，回调函数中参数就是匹配结果，如果匹配不到就是 null
  return str.replace(/[<>"&]/g, function(match) {
    switch (match) {
      case '<':
        return '&lt;';
      case '>':
        return '&gt;';
      case '&':
        return '&amp;';
      case '"':
        return '&quot;';
    }
  });
}
```

35、foo = foo||bar，这行代码是什么意思？为什么要这样写？

这种写法称之为短路表达式

答案：if(!foo) foo = bar; //如果 foo 存在，值不变，否则把 bar 的值赋给 foo。

短路表达式：作为“&&”和“||”操作符的操作数表达式，这些表达式在进行求值时，只要最终的结果已经可以确定是真或假，求值过程便告终止，这称之为短路求值。

注意 if 条件的真假判定，记住以下是 false 的情况：空字符串、false、undefined、null、0

36、看下列代码，将会输出什么?(变量声明提升)

```
var foo = 1;
function(){
  console.log(foo);
  var foo = 2;
  console.log(foo);
}
```

答案：输出undefined 和 2。上面代码相当于：

```
var foo = 1;
function(){
  var foo;
  console.log(foo); //undefined
  var foo = 2;
  console.log(foo); // 2;
}
```

函数声明与变量声明会被 JavaScript 引擎隐式地提升到当前作用域的顶部，但是只提升名称不会提升赋值部分。

37、用js实现随机选取 10–100 之间的 10 个数字，存入一个数组，并排序

```
var iArray = [];
function getRandom(istart, iend){
  var iChoice = istart - iend + 1;
  return Math.floor(Math.random() * iChoice + istart);
}
//Math.random()就是获取 0-1 之间的随机数 (永远获取不到 1)
for(var i=0; i<10; i++){
  var result= getRandom(10,100);
  iArray.push(result);
}
iArray.sort();
```

38、把两个数组合并，并删除第二个元素。

```
var array1 = ['a','b','c'];
var bArray = ['d','e','f'];
var cArray = array1.concat(bArray);
cArray.splice(1,1);
```

39、怎样添加、移除、移动、复制、创建和查找节点（原生JS，实在基础没细写每一步）

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()    //添加

removeChild()    //移除

replaceChild()    //替换

insertBefore()    //插入 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
```

40、有这样一个URL：

<http://item.taobao.com/item.htm?a=1&b=2&c=&d=xxx&e>，请写一段JS程序提取URL中的各个GET参数(参数名和参数个数不确定)，将其按key-value形式返回到一个json结构中,如{a:' 1', b:' 2', c:', d:' xxx' , e:undefined}。

答案：

```

> function splitUrl(url){
    var str=url.split("?")[1];
    var items=str.split("&");
    var arr={};
    var json={};
    for(var i=0;i<items.length;i++){
        arr=items[i].split("=");
        json[arr[0]]=arr[1];
    }
    return json;
}
splitUrl("http://www.baidu.com/index.php?a=0&b=1&c=2&d=3");
< ▶ Object {a: "0", b: "1", c: "2", d: "3"}
> |

```

41、正则表达式构造函数var reg=new RegExp(“xxx”)与正则表达式字面量var reg=//有什么不同？匹配邮箱的正则表达式？

答案：当使用 RegExp()构造函数的时候，不仅需要转义引号（即\"表示”），并且还需要双反斜杠（即\\表示一个\）。使用正则表达式字面量的效率更高。

邮箱的正则匹配：

```
var regMail = /^[a-zA-Z0-9_-]+@[a-zA-Z0-9_-]+((.[a-zA-Z0-9_-]{2,3}){1,2})$/;
```

****思考：看下面代码，给出输出结果。

```

for(var i=1;i<=3;i++){
    setTimeout(function(){
        console.log(i);
    },0);
};

```

答案：4 4 4。

原因：Javascript事件处理器在线程空闲之前不会运行。追问，如何让上述代码输出123？

```

for(var i=1;i<=3;i++){
    setTimeout((function(a){ //改成立即执行函数 console.log(a);
    })(i),0);
};

```

//输出 什么？

42、写一个function，清除字符串前后的空格。（兼容所有浏览器）

使用自带接口trim()，考虑兼容性：

```

if (!String.prototype.trim) {
  String.prototype.trim = function() {
    return this.replace(/^\s+/, "").replace(/\s+$/, "");
    // \s 匹配空白字符：回车、换行、制表符 tab 空格
  }
}
// test the function
var str = " \t\n test string ".trim(); alert(str == "test string"); // alert "true"

```

43、Javascript中callee和caller的作用？

arguments.callee：获得当前函数的引用

caller 是返回一个对函数的引用，该函数调用了当前函数；

callee是返回正在被执行的function函数，也就是所指定的function对象的正文。

看一串代码：

```

var a = function() {
  alert(a.caller);
}
var b = function() {
  a();
}
b();

```

上面的代码中，b调用了a，那么a.caller返回的是b的引用，结果如下：

```

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

```

如果直接调用a(即a在任何函数中被调用，也就是顶层调用),返回 null:

```

var a = function() {
  alert(a.caller);
}
var b = function() {
  a();
}
//b();
a();

```

那么下面的代码会执行出什么结果？

```
var a = function() {  
  alert(arguments.callee);  
}  
var b = function() {  
  a();  
}  
b();
```

44、列举浏览器对象模型BOM里常用的至少4个对象，并列举window对象的常用方法至少5个

对象：Window document location screen history navigator 方法：

Alert() confirm() prompt() open() close()

45、简述列举文档对象模型DOM里document的常用的查找访问节点的方法并做简单说明

Document.getElementById 根据元素id 查找元素

Document.getElementsByTagName 根据元素name查找元素

Document.getElementById根据指定的元素名查找元素

46、希望获取到页面中所有的checkbox怎么做？(不使用第三方框架)

```
<input class="1" type="checkbox">  
<input class="2" type="text">  
<input class="3" type="checkbox">  
<input class="4" type="textare">  
<input class="5" type="checkbox">  
<input class="6" type="button">
```

```
var domList = document.getElementsByTagName('input')
var checkBoxList = [];
var len = domList.length;    //缓存到局部变量
while (len--) { //使用 while 的效率会比 for 循环高
  if (domList[len].type == 'checkbox') {
    checkBoxList.push(domList[len]);
  }
}
console.log(checkBoxList)
```

47、JavaScript的数据类型都有什么？

基本数据类型：String,Boolean,Number,Undefined,Null

引用数据类型：Object(Array,Date,RegExp,Function)

48、javascript中==和===的区别是什么？举例说明。

===会自动进行类型转换，==不会

49、简述创建函数的几种方式

第一种（函数声明）：

```
function sum1(num1,num2){ return num1+num2;
}
```

第二种（函数表达式）：

```
var sum2 = function(num1,num2){ return num1+num2;
}
```

匿名函数：

function(){}:只能自己执行自己

第三种（函数对象方式）：

```
var sum3 = new Function("num1","num2","return num1+num2");
```

50、Javascript如何实现继承？

原型链继承，借用构造函数继承，组合继承，寄生式继承，寄生组合继承

51、Javascript创建对象的几种方式？

工厂方式，构造函数方式，原型模式，混合构造函数原型模式，动态原型方式

一对大括号括起来创建：JS 对象是一种复合类型，它允许你通过变量名存储和访问，换一种思路，对象是一个无序的属性集合，集合中的每一项都由名称和值组成

```
var emptyObj = {};  
var myObj =  
{  
  'id': 1,           //属性名用引号括起来，属性间由逗号隔开  
  'name': 'myName'  
};  
//var m = new myObj(); //不支持
```

在 function 中用 this 引用当前对象，通过对属性的赋值来声明属性。如果用 var 声明变量，则该变量为局部变量，只允许在类定义中调用

```
function myClass() {  
  this.id = 5;  
  this.name = 'myclass';  
  this.getName = function() {  
    return this.name;  
  }  
}  
var my = new myClass();  
alert(my.id);  
alert(my.getName());  
// 结果  
// 5  
// myclass
```

在函数体中创建对象可利用第一点的方法，或先 new Object(); 再为各属性赋值

```
function myClass() {
    var obj =
    {
        'id':2,
        'name':'myclass'
    };
    return obj;
}
function _myClass() {
    var obj = new Object();
    obj.id = 1;
    obj.name = '_myclass';
    return obj;
}
var my = new myClass();
var _my = new _myClass();
alert(my.id);
alert(my.name);
alert(_my.id);
alert(_my.name);

// 结果
// 2
// myclass
// 1
// _myclass
```

52、把 Script 标签 放在页面的最底部的body封闭之前 和封闭之后有什么区别？浏览器会如何解析它们？

如果说放在body 的封闭之前，将会阻塞其他资源的加载 如果放在body 封闭之后，不会影响body 内元素的加载

53、iframe的优缺点？

优点：

- 1.解决加载缓慢的第三方内容如图标和广告等的加载问题
2. Security sandbox
3. 并行加载脚本 缺点：

1. iframe会阻塞主页面的Onload事件

2. 即时内容为空，加载也需要时间
3. 没有语意

54、请你谈谈Cookie的弊端？

缺点：

1. `Cookie`数量和长度的限制。每个 domain 最多只能有 20 条 cookie，每个 cookie 长度不能超过 4KB，否则会被截掉。
2. 安全性问题。如果 cookie 被人拦截了，那人就可以取得所有的 session 信息。即使加密 也与事无补，因为拦截者并不需要知道 cookie 的意义，他只要原样转发 cookie 就可以达到 目的了。
3. 有些状态不可能保存在客户端。例如，为了防止重复提交表单，我们需要在服务器端保存 一个计数器。如果我们把这个计数器保存在客户端，那么它起不到任何作用。

55、js延迟加载的方式有哪些？

1. defer和async
2. 动态创建DOM方式（创建script，插入到DOM中，加载完毕后callBack）
3. 按需异步载入js

56、document.write和innerHTML的区别？

document.write 只能重绘整个页面 innerHTML 可以重绘页面的一部分

57、哪些操作会造成内存泄漏？

内存泄漏指任何对象在您不再拥有或需要它之后仍然存在。垃圾回收器定期扫描对象，并计算引用了每个对象的其他对象的数量。如果一个对象的引用数量为 0（没有其他对象引用过该对象），或对该对象的惟一引用是循环的，那么该对象的 内存即可回收。

1. setTimeout 的第一个参数使用字符串而非函数的话，会引发内存泄漏。

2. 闭包

3. 控制台日志

4. 循环（在两个对象彼此引用且彼此保留时，就会产生一个循环）

58、split() join() 的区别

答：前者是切割成数组的形式，后者是将数组转换成字符串

59、数组方法 pop() push() unshift() shift()各表示什么意思？

答：Push()尾部添加、pop()尾部删除、Unshift()头部添加、shift()头部删除

60、判断一个字符串中出现次数最多的字符，统计这个次数

```
var str = "abcaaaaacdef";
//console.log(str.length);
var obj = {};
for(var i = 0; i < str.length; i++){
    if( !obj[str.charAt(i)]){
        obj[str.charAt(i)] = 1;
    }else{
        obj[str.charAt(i)]++;
    }
}
//存出现次数最多的值和次数
var number = '';
var num = 0;
console.log("obj为:",obj); //Object {a: 6, b: 1, c: 2, d: 1, e: 1...}
//i代表每一项
for(var i in obj){
    //console.log("i为:",i);
    //obj[i] 为每一项的值
    console.log("obj",obj[i]);
    if(obj[i]>num){
        num = obj[i];
        number = i;
    }
}
console.log("最终出现最多的字母是：" + number + "，次数为：" + num); //最终出现最多的字母是：a，次数为：6
```

61、例举3种强制类型转换和2种隐式类型转换？

强制 (parseInt,parseFloat,number) 隐式 (== - ===)

```
var toString = Object.prototype.toString;

toString.call(new Date); // [object Date]
toString.call(new String); // [object String]
toString.call(Math); // [object Math]

// Since JavaScript 1.8.5
toString.call(undefined); // [object Undefined]
toString.call(null); // [object Null]

// 自定义类型
toString.call(new MyClass); // [object Object]
```

```
Boolean(0) // => false - 零
Boolean(new object()) // => true - 对象
Number(undefined) // => NaN
Number(null) // => 0
String(null) // => "null"

var iNum1 = parseInt("10", 2); //返回 2
var iNum2 = parseInt("10", 8); //返回 8
var iNum3 = parseInt("10", 10); //返回 10

var iNum1 = parseInt("12345red"); //返回 12345
var iNum1 = parseInt("0xA"); //返回 10
var iNum1 = parseInt("56.9"); //返回 56
var iNum1 = parseInt("red"); //返回 NaN
var fNum4 = parseFloat("11.22.33"); //返回 11.22
```

62、写一个获取非行间样式的函数

```
function getStyle(obj,attr,value)
{
    if(!value)
    {
        if(obj.currentStyle)//ie
        {
            return obj.currentStyle(attr);
        }
        else{//标准浏览器
            obj.getComputedStyle(attr,false);
        }
    }
    else
    {
        obj.style[attr] = value;
    }
}
```

63、解释jsonp的原理，以及为什么不是真正的ajax

动态创建script标签，回调函数 Ajax是页面无刷新请求数据操作

64、字符串反转，如将 '12345678' 变成 '87654321'

//思路：先将字符串转换为数组 split()，利用数组的反序函数
reverse()颠倒数组，再利用 join() 为字符串

```
var str = '12345678';
str = str.split('').reverse().join('');
```

65、将数字 12345678 转化成 RMB形式 如：12,345,678

```
var num=123456789;
//转化成字符串
var str=num+"";
//封装一个反转数组
function rever(str){
    return str=str.split("").reverse().join("");
}
//翻转一个数组
str=rever(str)

//给定一个空字符串 为了放最中的结果
var result="";
for(var i=1;i<=str.length;i++){
    result=result+str[i-1]; //把字符串的每一项重新放到一个新的字符串里
    if(i%3==0 && i!=str.length){ //字符串的索引是3 并且不是最后一项的时候添加，
        result+=","
    }
}
//将最终的结果进行反转
result=rever(result)

console.log(result)
```

66、生成5个不同的随机数；

```
function randomNumber(){
    var str = '';
    for(var i = 0; i < 5; i += 1){
        str += Math.floor(Math.random() * 10);
    }
    return str;
}
```

67、去掉数组中重复的数字 方法一；

```
function unique(data){
  data = data || [];
  var a = {};
  for (var i=0; i<data.length; i++) {
    var v = data[i];
    if (typeof(a[v]) == 'undefined'){
      a[v] = 1;
    }
  };
  data.length=0;
  for (var i in a){
    data[data.length] = i;
  }
  return data;
}
function test(){
  var arr = [9,1,3,8,7,7,6,6,5,7,8,8,7,4,3,1];
  var arr1 = unique(arr);
  alert(arr1.join(","));
}
test();
```

68、阶乘函数； $9*8*7*6*5...*1$

```
function factorial (num) {
  var result = num;
  if (num < 0) {
    return -1;
  } else if (num == 0 || num == 1) {
    return 1;
  } else {
    while (num > 1) {
      num--; result *= num;
    }
  }
  return result;
}
```

69、window.location.search返回的是什么？

答：查询(参数)部分。除了给动态语言赋值以外，我们同样可以给静态页面，并使用javascript来获得相信应的参数值

返回值：?ver=1.0&id=timlq 也就是问号后面的！

70、window.location.hash 返回的是什么？

答：锚点

71、window.location.reload() 作用？

答：刷新当前页面。

72、javascript 中的垃圾回收机制？

答：在Javascript中，如果一个对象不再被引用，那么这个对象就会被GC回收。如果两个对象互相引用，而不再被第3者所引用，那么这两个互相引用的对象也会被回收。因为函数a被b引用，b又被a外的c引用，这就是为什么 函数a执行后不会被回收的原因。

测试代码如下：

```
//猫
function cat(name)
{
    var zhuren ;
    this.name = name;

    //设置主人
    this.addZhuRen = function(zr){
        zhuren = zr;
    }

    this.getZhuRen = function(){
        return zhuren;
    }
}

//主人
function zhuren(name){
    this.name = name;
}

//创建主人：
var zr = new zhuren("zhangsan");
//创建猫
var cat1 = new cat("asan");
//设置该猫的主人
cat1.addZhuRen(zr);
//释放主人
zr = null ;
//此处还存在对主人对象的引用
alert(cat1.getZhuRen().name)
```

73、你如何优化自己的代码？

代码重用 避免全局变量（命名空间，封闭空间，模块化 mvc..）拆分函数

避免函数过于臃肿：单一职责原则

适当的注释，尤其是一些复杂的业务逻辑或者是计算逻辑，都应该写出这个业务逻辑的具体过程

内存管理，尤其是闭包中的变量释放

74、请用原生js 实现jquery 的get\post 功能，以及跨域情况下

完整的 ajax 原生写法如下：

```
function ajax(option) {
    // 创建一个 XMLHttpRequest 对象
    var xhr = window.XMLHttpRequest ? new XMLHttpRequest() : new ActiveXObject("Microsoft.XMLHTTP"),
        requestData = option.data,
        requestUrl = option.url,
        requestMethod = option.method;
    // 如果是GET请求，需要将option中的参数拼接到URL后面
    if ('POST' != requestMethod && requestData) {
        var query_string = '';
        // 遍历option.data对象，构建GET查询参数
        for(var item in requestData) {
            query_string += item + '=' + requestData[item] + '&';
        }
        // 注意这儿拼接的时候，需要判断是否已经有？
        requestUrl.indexOf('?') > -1
            ? requestUrl = requestUrl + '&' + query_string
            : requestUrl = requestUrl + '?' + query_string;
        // GET 请求参数放在URL中，将requestData置为空
        requestData = null;
    }
    // ajax 请求成功之后的回调函数
    xhr.onreadystatechange = function () {
        // readyState=4表示接受响应完毕
        if (xhr.readyState == ("number" == typeof XMLHttpRequest.DONE ? XMLHttpRequest.DONE : 4)) {
            if (200 == xhr.status) { // 判断状态码
                var response = xhr.response || xhr.responseText || {}; // 获取返回值
                // if define success callback, call it, if response is string, convert it to json object
                console.log(response);
                option.success && option.success(response); // 调用回调函数处理返回数据
                // 可以判断返回数据类型，对数据进行JSON解析或者XML解析
                // option.success && option.success('string' == typeof response ? JSON.parse(response) : response);
            } else {
                // if define error callback, call it
                option.error && option.error(xhr, xhr.statusText);
            }
        }
    };
    // 发送ajax请求
    xhr.open(requestMethod, requestUrl, true);
    // 请求超时的回调
    xhr.ontimeout = function () {
        option.timeout && option.timeout(xhr, xhr.statusText);
    };
    // 定义超时时间
    xhr.timeout = option.timeout || 0;
    // 设置响应头，这儿默认设置为json格式，可以定义为其他格式，修改头部即可
    xhr.setRequestHeader && xhr.setRequestHeader('Content-Type', 'application/json;charset=utf-8');
    xhr.withCredentials = (option.xhrFields || {}).withCredentials;
    // 这儿主要用于发送POST请求的数据
    xhr.send(requestData);
}
```


简写如下：

```
function easyAjax(url){
var xhr = new XMLHttpRequest();
    xhr.open("get",url,true);
    xhr.send();
    xhr.onreadystatechange = function(){
if (xhr.readyState == 4) {
    if (xhr.status>=200 && xhr.status<300) {
        alert(xhr.responseText);
    };
};
}}
```

但是上面的代码其实还不能解决 跨域问题

有个最笨的办法：

通过 script 标签引用，写死你需要的 src 的 url 地址

```
<script>
var callbackfunction = function(data) {
    console.log('我是跨域请求来的数据-->' + data);
};
</script>
<script src="xxxxx?callback=callbackfunction"></script>
```

拿到 data 后再解析出 URL 就可以了

75、请尽可能详尽的解释ajax的工作原理

思路：先解释异步，再解释ajax 如何使用

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。要清楚这个过程和原理，我们必须对XMLHttpRequest有所了解。

XMLHttpRequest 是ajax 的核心机制，它是在IE5 中首先引入的，是一种支持异步请求的技术。简单的说，也就是 javascript 可以及时向服务器提出请求和处理响应，而不阻塞用户。

达到无刷新的效果。

76、为什么扩展javascript内置对象不是好的做法？

因为扩展内置对象会影响整个程序中所使用到的该内置对象的原型属性

77、浏览器标准模式和怪异模式之间的区别是什么？

标准模式是指，浏览器按W3C标准解析执行代码；怪异模式则是使用浏览器自己的方式解析执行代码，因为不同浏览器解析执行的方式不一样，所以我们称之为怪异模式。浏览器解析时到底使用标准模式还是怪异模式，与你网页中的DTD声明直接相关，DTD声明定义了标准文档的类型（标准模式解析）文档类型，会使浏览器使用相应的方式加载网页并显示，忽略DTD声明，将使网页进入怪异模式

78、HTTP 协议中，GET 和POST 有什么区别？分别适用什么场景？

get 传送的数据长度有限制，post 没有

get 通过url 传递，在浏览器地址栏可见，post 是在报文中传递

适用场景：

post 一般用于表单提交

get 一般用于简单的数据查询，严格要求不是那么高的场景

79、HTTP 状态消息 200 302 304 403 404 500 分别表示什么

200：请求已成功，请求所希望的响应头或数据体将随此响应返回。

302：请求的资源临时从不同的 URI 响应请求。由于这样的重定向是临时的，客户端应当继续向原有地址发送以后的请求。只有在 Cache-Control 或 Expires 中进行了指定的情况下，这个响应才是可缓存的

304：如果客户端发送了一个带条件的 GET 请求且该请求已被允许，而文档的内容（自上次访问以来或者根据请求的条件）并没有改变，则服务器应当返回这个状态码。304 响应禁止包含消息体，因此始终以消息头后的第一个空行结尾。403：服务器已经理解请求，但是拒绝执行它。

404：请求失败，请求所希望得到的资源未被在服务器上发现。

500：服务器遇到了一个未曾预料的状态，导致了它无法完成对请求的处理。一般来说，这个问题都会在服务器端的源代码出现错误时出现。

80、HTTP 协议中，header 信息里面，怎么控制页面失效时间

（last-modified,cache-control,Expires 分别代表什么）

Last-Modified	文档的最后改动时间。客户可以通过 If-Modified-Since 请求头提供一个日期，该请求将被视为一个条件 GET，只有改动时间迟于指定时间的文档才会返回，否则返回一个 304 (Not Modified) 状态。Last-Modified 也可用 setDateHeader 方法来设置。
Expires	应该在什么时候认为文档已经过期，从而不再缓存它？

81、HTTP 协议目前常用的有哪几个？KEEPALIVE 从哪个版本开始出现的？

http1.0

http1.1 keepalive

176、js 中如何实现一个map

```
function myMap()
{
    /** Map 大小 */
    var size = 0;
    /** 对象 */
    var entry = new Object();
    /** 存 */
    this.put = function (key , value)
    {
        if(!this.containsKey(key))
        {
            size ++ ;
        }
        entry[key] = value;
    }
    /** 取 */
    this.get = function (key)
    {
        if( this.containsKey(key) )
        {
            return entry[key];
        }
        else
        {
            return null;
        }
    }
    /** 删除 */
    this.remove = function ( key )
    {
        if( delete entry[key] )
        {
            size --;
        }
    }
}
```

```
/** 是否包含 Value */  
this.containsValue = function ( value )  
{  
    for(var prop in entry)  
    {  
        if(entry[prop] == value)  
        {  
            return true;  
        }  
    }  
    return false;  
}  
  
/** 所有 Value */  
this.values = function ()  
{  
    var values = new Array(size);  
    for(var prop in entry)  
    {  
        values.push(entry[prop]);  
    }  
    return values;  
}  
  
/** 所有 Key */  
this.keys = function ()  
{  
    var keys = new Array(size);  
    for(var prop in entry)  
    {  
        keys.push(prop);  
    }  
    return keys;  
}  
  
/** Map Size */  
this.size = function ()  
{  
    return size;  
}  
}
```

82、js可否实现面向对象编程，如果可以如何实现js对象的继承

创建对象的几种方式 实现继承的几种方式 **原型**

83、不用任何插件，如何实现一个tab 栏切换？

通过改变不同层的css 设置层的显示和隐藏

84、js 三种弹窗以及三种弹窗的功能

alert confirm prompt

警告框 alert()

alert 是警告框，只有一个按钮“确定”无返回值，警告框经常用于确保用户可以得到某些信息。当警告框出现后，用户需要点击确定按钮才能继续进行操作。语法：alert("文本")

确认框 confirm()

confirm 是确认框，两个按钮，确定或者取消，返回 true 或 false。确认框用于使用户可以验证或者接受某些信息。当确认框出现后，用户需要点击确定或者取消按钮才能继续进行操作。如果用户点击确认，那么返回值为 true。如果用户点击取消，那么返回值为 false。语法：confirm("文本")

提示框 prompt()

prompt 是提示框，返回输入的消息，或者其默认值提示框经常用于提示用户在进入页面输入某个值。当提示框出现后，用户需要输入某个值，然后点击确认或取消按钮才能继续操纵。如果用户点击确认，那么返回值为输入的值。如果用户点击取消，那么返回值为 null。语法：prompt("文本","默认值")

85、JavaScript alert(0.1*0.2);结果是多少？和你预期的一样吗？如果不一样该如何处理？

有误差，应该比准确结果偏大。

有种最简单的解决方案，就是给出明确的精度要求，在返回值的过程中，计算机会自动四舍五入，比如：

```
var numA = 0.1;
var numB = 0.2;

alert( parseFloat((numA + numB).toFixed(2)) === 0.3 );
```

但是这只是一种简单的做法。看看下面的做法：

```
Math.formatFloat = function(f, digit) {
    var m = Math.pow(10, digit);
    return parseInt(f * m, 10) / m;
}

var numA = 0.1;
var numB = 0.2;

alert(Math.formatFloat(numA + numB, 1) === 0.3);
```

这个方法是什么意思呢？为了避免产生精度差异，我们要把需要计算的数字乘以 10 的 n 次幂，换算成计算机能够精确识别的整数，然后再除以 10 的 n 次幂，大部分编程语言都是这样处理精度差异的，我们就借用过来处理一下 JS 中的浮点数精度误差。

86、列出3条以上ff和IE的脚本兼容问题

1、在IE下可通过document.frames["id"];得到该IFRAME对象，而在火狐下则是通过

document.getElementById("content_panel_if").contentWindow;

2、IE的写法：_tbody=_table.childNodes[0]

在FF中，firefox会在子节点中包含空白则第一个子节点为空白""，而ie不会返回空白可以通过if("" != node.nodeName)过滤掉空白子对象

3、模拟点击事件

```
if(document.all){//ie 下
document.getElementById("a3").click();
}
else{//非IE
var evt = document.createEvent("MouseEvents");
evt.initEvent("click", true, true);
document.getElementById("a3").dispatchEvent(evt);
}
```

4、事件注册

```
if (isIE){window.attachEvent("onload",
init);}else{window.addEventListener("load", init, false);}
```

87、在Javascript中什么是伪数组？如何将伪数组转化为标准数组？

伪数组（类数组）：无法直接调用数组方法或期望 `length` 属性有什么特殊的行为，但仍可以对真正数组遍历方法来遍历它们。典型的是函数的 `argument` 参数，还有像调用 `getElementsByTagName`, `document.childNodes` 之类的，它们都返回 `NodeList` 对象都属于伪数组。

可以使用 `Array.prototype.slice.call(fakeArray)` 将数组转化为真正的 `Array` 对象。

88、事件代理怎么实现？

在元素的父节点注册事件，通过事件冒泡，在父节点捕获事件

三、Jquery 部分

1、jQuery 的美元符号\$有什么作用？

其实美元符号\$只是"jQuery"的别名，它是jQuery的选择器，如下代码：

```
$(document).ready(function(){  
});  
当然你也可以用jQuery来代替$，如下代码：  
jQuery(document).ready(function(){  
});
```

2. body 中的 onload()函数和 jQuery 中 document.ready() 有什么区别？

onload()和 document.ready()的区别有以下两点：

1、我们可以在页面中使用多个 document.ready()，但只能使用一次 onload()。

2、document.ready()函数在页面 DOM 元素加载完以后就会被调用，而 onload()函数则要在所有的关联资源（包括图像、音频）加载完毕后会调用。

3. jQuery 中有哪几种类型的选择器

有 3 种类型的选择器，如下：

1、基本选择器：直接根据 id、css 类名、元素名返回匹配的 dom 元素。

2、层次选择器：也叫做路径选择器，可以根据路径层次来选择相应的 DOM 元素。

3、过滤选择器：在前面的基础上过滤相关条件，得到匹配的 dom 元素。

4. 请使用 jQuery 将页面上的所有元素边框设置为 2px 宽的虚

线？

```
$(".*").css("border", "2px dotted red");
```

4. 当 CDN 上的 jQuery 文件不可用时，该怎么办？

为了节省带宽和脚本引用的稳定性，我们会使用 CDN 上的 jQuery 文件，例如 google 的 jquery cdn 服务。但是如果这些 CDN 上的 jQuery 服务不可用，我们还可以通过以下代码来切换到本地服务器的 jQuery 版本

```
if (typeof jQuery == 'undefined') {  
    document.write(unescape("%3Cscript src='/Script/jquery-1.4.1.min.js' type='text/javascript' %3E%3C/script%3E"));  
}
```

5. 怎样用 jQuery 编码和解码 URL

encodeURIComponent(url) and decodeURIComponent(url)

6. 针对 jQuery 的优化方法？

- a、优先使用ID选择器
- b、jquery获取到的DOM元素如果需要多次使用，建议使用一个变量将其保存起来，因为操作DOM的过程是非常耗费性能的
- c、在class前使用tag(标签名) d、给选择器一个上下文
- e、慎用.live()方法（应该说尽量不要使用） f、使用data()方法存储临时变量

四、Ajax 部分

1、Ajax 是什么？如何创建一个 Ajax ？

Ajax并不算是一种新的技术，全称是asynchronousjavascriptandxml，可以说是已有技术的组合，主要用来实现客户端与服务器端的异步通信效果，实现页面的局部刷新，早期的浏览器并不能原生支持 ajax，可以使用隐藏帧（iframe）方式变相实现异步效果，后来的浏览器提供了对ajax的原生支持使用 ajax 原生方式发送请求主要通过 XMLHttpRequest(标准浏览器)、ActiveXObject(IE 浏览器)对象实现异步通信效果

基本步骤：

```
var xhr = null; //创建对象 if(window.XMLHttpRequest){
xhr = new XMLHttpRequest();
}else
{
    xhr = new ActiveXObject("Microsoft.XMLHTTP");
}

xhr.open( "方式" , " 地址" , " 标志位" ); //初始化请求
xhr.setRequestHeader( "" , " " ); //设置http头信息
xhr.onreadystatechange = function(){} //指定回调函数 xhr.send(); //发送请求
```

js框架 (jQuery/EXTJS等) 提供的ajax API对原生的ajax进行了封装，熟悉了基础理论，再学习别的框架就会得心应手，好多都是换汤不换药的内容

2、同步和异步的区别？

同步：阻塞的

-张三叫李四去吃饭，李四一直忙得不停，张三一直等着，直到李四忙完两个人一块去吃饭

=浏览器向服务器请求数据，服务器比较忙，浏览器一直等着（页面白屏），直到服务器返回数据，浏览器才能显示页面

异步：非阻塞的

-张三叫李四去吃饭，李四在忙，张三说了一声然后自己就去吃饭了，李四忙完后自己去吃

=浏览器向服务器请求数据，服务器比较忙，浏览器可以自如的干原来的事情（显示页面），服务器返回数据的时候通知浏览器一声，浏览器把返回的数据再渲染到页面，局部更新

3、如何解决跨域问题？

理解跨域的概念：协议、域名、端口都相同才同域，否则都是跨域 出于安全考虑，服务器不允许 ajax 跨域获取数据，但是可以跨域获取文件内容，所

以

基于这一点，可以动态创建script标签，使用标签的src属性访问js文件的形式获取js

脚本，并且这个 js 脚本中的内容是函数调用，该函数调用的参数是服务器返回的数据，为

了获取这里的参数数据，需要事先在页面中定义回调函数，在回调函数中处理服务器返回的数据，这就是解决跨域问题的主流解决方案

4、页面编码和被请求的资源编码如果不一致如何处理？

对于 ajax 请求传递的参数，如果是 get 请求方式，参数如果传递中文，在有些浏览器会乱码，不同的浏览器对参数编码的处理方式不同，所以对于 get 请求的参数需要使用 encodeURIComponent 函数对参数进行编码处理，后台开发语言都有相应的解码 api。对于 post 请求不需要进行编码

5、简述 ajax 的过程。

1. 创建XMLHttpRequest对象,也就是创建一个异步调用对象
2. 创建一个新的HTTP请求,并指定该HTTP请求的方法、URL及验证信息
3. 设置响应HTTP请求状态变化的函数
4. 发送HTTP请求
5. 获取异步调用返回的数据
6. 使用JavaScript和DOM实现局部刷新

6、阐述一下异步加载。

1. 异步加载的方案：动态插入 script 标签
2. 通过 ajax 去获取 js 代码，然后通过 eval 执行
3. script 标签上添加 defer 或者 async 属性
4. 创建并插入 iframe，让它异步执行 js

7、请解释一下 JavaScript 的同源策略。

同源策略是客户端脚本（尤其是 Javascript）的重要的安全度量标准。它最早出自 Netscape Navigator 2.0，其目的是防止某个文档或脚本从多个

不同源装载。所谓同源指的是：协议，域名，端口相同，同源策略是一种安全协议，指一段脚本只能读取来自同一起来源的窗口和文档的属性。

8、GET和POST的区别，何时使用POST？

GET：一般用于信息获取，使用 URL 传递参数，对所发送信息的数量也有限制，一般在 2000个字符，有的浏览器是8000个字符 POST：一般用于修改服务器上的资源，对所发送的信息没有限制

在以下情况中，请使用 POST 请求：

1. 无法使用缓存文件（更新服务器上的文件或数据库）
2. 向服务器发送大量数据（POST 没有数据量限制）
3. 发送包含未知字符的用户输入时，POST 比 GET 更稳定也更可靠

9、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?

- 1.通过异步模式，提升了用户体验
2. 优化了浏览器和服务器之间的传输，减少不必要的数据往返，减少了带宽占用
3. Ajax 在客户端运行，承担了一部分本来由服务器承担的工作，减少了大用户量下的服务器负载。

10、Ajax的最大的特点是什么。

Ajax 可以实现异步通信效果，实现页面局部刷新，带来更好的用户体验；按需获取数据，节约带宽资源；

11、ajax的缺点

- 1、ajax不支持浏览器back按钮。
- 2、安全问题 AJAX暴露了与服务器交互的细节。
- 3、对搜索引擎的支持比较弱。
- 4、破坏了程序的异常机制。

12、ajax请求的时候get 和post方式的区别

get一般用来进行查询操作，url地址有长度限制，请求的参数都暴露在url地址当中，如果传递中文参数，需要自己进行编码操作，安全性较低。

post 请求方式主要用来提交数据，没有数据长度的限制，提交的数据内容存在于 http 请求体中，数据不会暴露url地址中。

13、解释jsonp的原理，以及为什么不是真正的ajax

Jsonp并不是一种数据格式，而json是一种数据格式，jsonp是用来解决跨域获取数据的一种解决方案，具体是通过动态创建script标签，然后通过标签的src属性获取js文件中的js脚本，该脚本的内容是一个函数调用，参数就是服务器返回的数据，为了处理这些返回的数据，需要事先在页面定义好回调函数，本质上使用的并不是ajax技术

14、什么是Ajax和JSON，它们的优缺点。

Ajax是全称是asynchronous JavaScript andXML，即异步JavaScript和xml，用于在 Web 页面中实现异步数据交互，实现页面局部刷新。

优点：可以使得页面不重载全部内容的情况下加载局部内容，降低数据传输量，避免用户不断刷新或者跳转页面，提高用户体验

缺点：对搜索引擎不友好；要实现 ajax 下的前后退功能成本较大；可能造成请求数的

增加跨域问题限制；

JSON是一种轻量级的数据交换格式，ECMA的一个子集 优点：轻量级、易于人的阅读和编写，便于机器（JavaScript）解析，支持复合数据类型（数组、对象、字符串、数字）

15、http常见的状态码有那些？分别代表是什么意思？

200 - 请求成功

301 - 资源（网页等）被永久转移到其它URL

404 - 请求的资源（网页等）不存在

500 - 内部服务器错误

16、一个页面从输入 URL 到页面加载显示完成，这个过程中都发生了什么？

分为4个步骤：

1. 当发送一个 URL 请求时，不管这个 URL 是 Web 页面的 URL 还是 Web 页面上每个资源的 URL，浏览器都会开启一个线程来处理这个请求，同时在远程 DNS 服务器上启动一个 DNS 查询。这能使浏览器获得请求对应的 IP 地址。
2. 浏览器与远程 Web 服务器通过 TCP 三次握手协商来建立一个 TCP/IP 连接。该握手包括一个同步报文，一个同步-应答报文和一个应答报文，这三个报文在浏览器和服务器之间传递。该握手首先由客户端尝试建立起通信，而后服务器应答并接受客户端的请求，最后由客户端发出该请求已经被接受的报文。
3. 一旦 TCP/IP 连接建立，浏览器会通过该连接向远程服务器发送 HTTP 的 GET 请求。远程服务器找到资源并使用 HTTP 响应返回该资源，值为 200 的 HTTP 响应状态表示一个正确的响应。
4. 此时，Web 服务器提供资源服务，客户端开始下载资源。

17、javascript的本地对象，内置对象和宿主对象

本地对象为独立于宿主环境的ECMAScript提供的对象，包括Array Object RegExp等可以new实例化的对象

内置对象为Global，Math等不可以实例化的(他们也是本地对象，内置对象是本地对象的一个子集)

宿主对象为所有的非本地对象，所有的BOM和DOM对象都是宿主对象，如浏览器自带的 document,window 等对象

18、为什么利用多个域名来存储网站资源会更有效？

确保用户在不同地区能用最快的速度打开网站，其中某个域名崩溃用户也能通

过其他郁闷访问网站，并且不同的资源放到不同的服务器上有利于减轻单台服务器的压力。

19、请说出三种减低页面加载时间的方法

- 1、压缩css、js文件
- 2、合并js、css文件，减少http请求
- 3、外部js、css文件放在最底下
- 4、减少dom操作，尽可能用变量替代不必要的dom操作

五、移动 APP 开发

1、移动端最小触控区域是多大？

移动端的点击事件的有延迟，时间是多久，为什么会有？怎么解决这个延时？（click有300ms延迟,为了实现safari的双击事件的设计，浏览器要知道你是不是要双击操作。）

2. seajs 和 requirejs 的区别

answer:seajs 是 CMD 规范，requirejs 是 AMD 规范，一个是迟加载一个是预加载。

3. 常用的 app 端框架

jquery mobile、ioinc、onsen

简单说明：

jQuery Mobile 是用于创建移动 Web 应用的前端开发框架。

jQuery Mobile 可以应用于智能手机与平板电脑

jQuery Mobile 使用 HTML5 & CSS3 最小的脚本来布局网页

缺点就是性能差。

Ionic:

- 1.ionic 基于 Angular 语法，简单易学。
- 2.ionic 是一个轻量级框架。
- 3.ionic 完美的融合下一代移动框架，支持 Angularjs 的特性，MVC，代码易维护。

4.ionic 提供了漂亮的设计，通过 SASS 构建应用程序，它提供了很多 UI 组件来帮助开发者开发强大的应用。

5.ionic 专注原生，让你看不出混合应用和原生的区别

6.ionic 提供了强大的命令行工具。

7.ionic 性能优越，运行速度快。

缺点：IOS6 和 Android4.1 以下的版本不支持

六、NodeJs

1、对Node的优点和缺点提出了自己的看法：

* (优点) 因为 Node 是基于事件驱动和无阻塞的，所以非常适合处理并发请求，因此构建在Node上的代理服务器相比其他技术实现（如Ruby）的服务器表现要好得多。此外，与Node代理服务器交互的客户端代码是由 javascript语言编写的，因此客户端和服务端都用同一种语言编写，这是非常美妙的事情。

* (缺点) Node 是一个相对新的开源项目，所以不太稳定，它总是一直在变，而且缺少足够多的第三方库支持。看起来，就像是 Ruby/Rails 当年的样子。

2、需求：实现一个页面操作不会整页刷新的网站，并且能在浏览器前进、后退时正确响应。给出你的技术实现方案？

至少给出自己的思路（url-hash,可以使用已有的一些框架history.js等）

3、Node.js 的适用场景？

- 1)、实时应用：如在线聊天，实时通知推送等等（如 socket.io）
- 2)、分布式应用：通过高效的并行I/O使用已有的数据
- 3)、工具类应用：海量的工具，小到前端压缩部署（如 grunt），大到桌面图形界面应用程序
- 4)、游戏类应用：游戏领域对实时和并发有很高的要求（如网易的 pomelo 框架）
- 5)、利用稳定接口提升Web渲染能力
- 6)、前后端编程语言环境统一：前端开发人员可以非常快速地切入到服务器端的开发（如著名的纯Javascript全栈式MEAN架构）

4、解释一下 Backbone 的 MVC 实现方式？

Backbone 是一个轻量级的前端 MVC 框架，用于结构化管理页面中的大量 JS，建立与服务器、视图间的无缝连接，为构建复杂的应用提供基础框架

5、什么是“前端路由”？什么时候适合使用“前端路由”？ “前端路由”有哪些优点和 缺点？

熟悉前后端通信相关知识 前端路由就是在不进行后端请求的情况下对页面进行跳转

6、对Node 的优点和缺点提出了自己的看法？

优点：

1. 因为Node是基于事件驱动和无阻塞的，所以非常适合处理并发请求，因此构建在Node 上的代理服务器相比其他技术实现（如 Ruby）的服务器表现要好得多。
2. 与Node代理服务器交互的客户端代码是由javascript语言编写的，因此客户端和服务 器端都用同一种语言编写，这是非常美妙的事情。

缺点：

1. Node是一个相对新的开源项目，所以不太稳定，它总是一直在变。
2. 缺少足够多的第三方库支持。看起来，就像是Ruby/Rails当年的样子（第三方库现在已经很丰富了，所以这个缺点可以说不存在了）。

七、前端概括性问题

1、常使用的库有哪些？常用的前端开发工具？开发过什么应用或组件？

使用率较高的框架有 jQuery、YUI、Prototype、Dojo、Ext.js、Mootools 等。尤其是 jQuery，超过91%。

轻量级框架有 Modernizr、underscore.js、backbone.js、Raphael.js 等。
（理解这些框架的功能、性能、设计原理）

前端开发工具：SublimeText、Eclipse、Notepad、Firebug、HttpWatch、Yslow。开发过的插件：城市选择插件，汽车型号选择插件、幻灯片插件。弹出层。（写过开源程序，加载器，js引擎更好）

2、对BFC规范的理解？

Formatting Context：指页面中的一个渲染区域，并且拥有一套渲染规则，他决定了其子元素如何定位，以及与其他元素的相互关系和作用。

3、WEB应用从服务器主动推送Data到客户端有那些方式？

html5 websocket WebSocket通过Flash XHR长时间连接
XHR Multipart Streaming 不可见的Iframe
<script>标签的长时间连接(可跨域)

4、加班的看法

加班就像借钱，原则应当是-----救急不救穷

个人能够接受加班的情况下：我能够接受加班，在工作需要的情况下，我会义不容辞地接受加班，我现在刚从学校毕业，个人时间比较多，我认为加班会给我带来更多的机会，我愿意用我的努力为公司发展贡献一份力量，也为我自己提供一个舞台

5、平时如何管理你的项目，如何设计突发大规模并发架构？

先期团队必须确定好全局样式（`globe.css`），编码模式(utf-8)等 编写习惯必须一致（例如都是采用继承式的写法，单样式都写成一行）；标注样式编写人，各模块都及时标注（标注关键样式调用的地方）；页面进行标注（例如 页面 模块 开始和结束）；

CSS跟HTML分文件夹并行存放，命名都得统一（例如`style.css`）JS分文件夹存放 命名以该JS功能为准英文翻译；图片采用整合的`images.png` png8 格式文件使用 尽量整合在一起使用方便将来的管理

6、那些操作会造成内存泄漏？

内存泄漏指任何对象在您不再拥有或需要它之后仍然存在。垃圾回收器定期扫描对象，并计算引用了每个对象的其他对象的数量。如果一个对象的引用数量为 0（没有其他对象引用过该对象），或对该对象的惟一引用是循环的，那么该对象的内存即可回收。

`setTimeout`的第一个参数使用字符串而非函数的话，会引发内存泄漏。闭包、控制台日志、循环（在两个对象彼此引用且彼此保留时，就会产生一个循环）

7、你说你热爱前端，那么应该WEB行业的发展很关注吧？说说最近最流行的一些东西吧？

Node.js、Mongodb、npm、MVVM、MEAN、react、angularjs、browserify、webpack

8、你有了解我们公司吗？说说你的认识？

比如我想去阿里，所以我针对阿里的说

最羡慕就是在双十一购物节，350.19亿元，每分钟支付79万笔。海量数据，居然无一漏单、无一故障。太厉害了。

携程、去哪儿：一个主要从事在线旅游 O2O，当然也可以订酒店，买车票，买机票

一句话，事先在去公司面试前上网查一下该公司主营业务是你面试成功的一个加分项。

9、移动端（比如：Android IOS）怎么做好用户体验？

融入自己的设计理念，注重用户体验，选择合适的技术

10、你所知道的页面性能优化方法有那些？

压缩、合并，减少请求，代码层析优化。。。

具体：

（1）减少 http 请求次数：CSS Sprites, JS、CSS 源码压缩、图片大小控制合适；网页 Gzip，CDN 托管，data 缓存，图片服务器。

（2）前端模板 JS+数据，减少由于 HTML 标签导致的带宽浪费，前端用变量保存 AJAX 请求结果，每次操作本地变量，不用请求，减少请求次数

（3）用 innerHTML 代替 DOM 操作，减少 DOM 操作次数，优化 javascript 性能。

（4）当需要设置的样式很多时设置 className 而不是直接操作 style。

（5）少用全局变量、缓存 DOM 节点查找的结果。减少 IO 读取操作。

（6）避免使用 CSS Expression (css 表达式) 又称 Dynamic properties(动态属性)。

（7）图片预加载，将样式表放在顶部，将脚本放在底部 加上时间戳。

(8) 避免在页面的主体布局中使用 table，table 要等其中的内容完全下载之后才会显示出来，显示比 div+css 布局慢。

11、除了前端以外还了解什么其它技术么？你最最厉害的技能是什么？

知识面宽度，最好熟悉一些后台语言，比如 数据库 mysql,java 等

12、AMD(Modules/Asynchronous-Definition)、CMD (Common Module Definition) 规范区别？

1.AMD 是提前执行，CMD 是延迟执行

2. CMD 推崇依赖就近，AMD 推崇依赖前置。

13、谈谈你认为怎样做能使项目做的更好？

考虑问题的深入，不仅仅停留在完成任务上，要精益求精

比如：针对某些问题，我会更深入的去了解它为什么会那样，对于某 bug，下次我会做好预先防范。

14、你对前端界面工程师这个职位是怎么样理解的？它的前景会怎么样？

表现出对前端的认同与兴趣，关注相关技术前沿

15、谈谈你对 Web 标准的理解，以及对 w3c 组织的认识

(1)Web 标准规范要求，书写标签必须闭合、标签小写、不乱嵌套，可提高搜索机器人对网页内容的搜索几率。

(2)建议使用外链 css 和 js 脚本，从而达到结构与行为、结构与表现的分离，提高页面的渲染速度，能更快地显示页面的内容。

(3)样式与标签的分离，更合理的语义化标签，使内容能被更多的用户所访问、内容能被更广泛的设备所访问、更少的代码和组件，从而降低维护成本、改版更方便

(4)不需要变动页面内容，便可提供打印版本而不需要复制内容，提高网站易用性

一句话总结：遵循 w3c 制定的 Web 标准，能够使用户浏览者更方便的阅读，使网页开发者之间更好的交流。

16、最能概括你自己的三个词是什么？

我经常用的三个词是：适应能力强，有责任心和做事有始终，结合具体例子向主考官解释

17、你为什么要离开原来的公司？

提示：

1.回答这个问题时一定要小心，就算在前一个工作受到再大的委屈，对公司有多少的怨言，都千万不要表现出来，尤其要避免对公司本身主管的批评，避免面试官的负面情绪及印象。建议此时最好的回答方式是将问题归咎在自己身上，例如觉得工作没有学习发展的空间，自己想面试工作的相关产业中多加学习，或是前一份工作与自己的生涯规划不合等等，回答的答案最好是积极正面的。

2.我希望能获得一份更好的工作，如果机会来临，我会抓住。我觉得目前的工作，已经达到顶峰，即没有升迁机会。

18、你为什么愿意到我们公司来工作？

对于这个问题，你要格外小心，如果你已经对该单位作了研究，你可以回答一些详细的原因，像“公司本身的高技术开发环境很吸引我。”、“我同公司出生在同样的时代，我希望能够进入一家与我共同成长的公司。”、“你们公司一直都稳定发展，在近几年来在市场上很有竞争力。”、“我认为贵公司能够给我提供一个与众不同的发展道路。”这都显示出你已经做了一些调查，也说明你对自己的未来有了较为具体的远景规划。

19、如果录用了你，你将怎样开展工作？

尝试采用迂回战术来回答，如“首先听取领导的指示和要求，然后就有关情况进行了解和熟悉，接下来制定一份近期的工作计划并报领导批准，最后根据计划开展工作。”。

听话的员工永远是领导喜欢的。

20、你希望与什么样的上级共事？

这个问题就是个坑，你别傻乎乎的说自己真的希望什么什么之类的，避对上级具体的希望，多谈对自己的要求，一句话，领导不是你想要什么样的就是什么样的，你只能适应，最虚伪的回答：希望我的上级能够在工作中对我多指导，对我工作中的错误能够立即指出。

好了，祝你面试成功，和小胖一起学习进步吧。。加油~