**·**

**前端面试题汇总**

[一、HTML和CSS 21](#_Toc23047)

[1. 你做的页面在哪些流览器测试过？这些浏览器的内核分别是什么? 21](#_Toc26235)

[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？ 21](#_Toc2440)

[3. Quirks模式是什么？它和Standards模式有什么区别 21](#_Toc17105)

[4. div+css的布局较table布局有什么优点？ 22](#_Toc31898)

[5. img的alt与title有何异同？ strong与em的异同？ 22](#_Toc30593)

[6. 你能描述一下渐进增强和优雅降级之间的不同吗? 23](#_Toc17013)

[7. 为什么利用多个域名来存储网站资源会更有效？ 23](#_Toc20815)

[8. 请谈一下你对网页标准和标准制定机构重要性的理解。 24](#_Toc11664)

[9. 请描述一下cookies，sessionStorage和localStorage的区别？ 24](#_Toc13965)

[10. 简述一下src与href的区别。 24](#_Toc22985)

[11. 知道的网页制作会用到的图片格式有哪些？ 25](#_Toc12076)

[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？ 25](#_Toc17581)

[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？ 25](#_Toc2657)

[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。 25](#_Toc4364)

[15. 你如何理解HTML结构的语义化？ 26](#_Toc30307)

[16. 谈谈以前端角度出发做好SEO需要考虑什么？ 27](#_Toc6218)

[17. 有哪项方式可以对一个DOM设置它的CSS样式？ 28](#_Toc17154)

[18. CSS都有哪些选择器？ 28](#_Toc16130)

[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？ 29](#_Toc3636)

[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？ 29](#_Toc28665)

[21. 什么是Css Hack？ie6,7,8的hack分别是什么？ 30](#_Toc20936)

[22. 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？ 30](#_Toc30581)

[23. 什么是外边距重叠？重叠的结果是什么？ 31](#_Toc17365)

[24. rgba()和opacity的透明效果有什么不同？ 31](#_Toc2668)

[25. 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？ 31](#_Toc18353)

[26. 如何垂直居中一个浮动元素？ 31](#_Toc24628)

[27. px和em的区别。 32](#_Toc23397)

[28. 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？ 33](#_Toc14995)

[29. Sass、LESS是什么？大家为什么要使用他们？ 33](#_Toc3578)

[30. display:none与visibility:hidden的区别是什么？ 33](#_Toc22364)

[31. CSS中link和@import的区别是： 34](#_Toc11553)

[32. 简介盒子模型： 34](#_Toc24581)

[33. 为什么要初始化样式？ 34](#_Toc3313)

[34. BFC是什么? 35](#_Toc20182)

[35. html语义化是什么？ 35](#_Toc8922)

[36. Doctype的作用？严格模式与混杂模式的区别？ 35](#_Toc16033)

[37. 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。 35](#_Toc13573)

[38. HTML与XHTML——二者有什么区别？ 35](#_Toc289)

[39. html常见兼容性问题？ 36](#_Toc10663)

[40. 对WEB标准以及W3C的理解与认识 36](#_Toc31093)

[41. 行内元素有哪些?块级元素有哪些?CSS的盒模型? 36](#_Toc7711)

[42. 前端页面有哪三层构成，分别是什么?作用是什么? 37](#_Toc25426)

[43. 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? 37](#_Toc27607)

[44. 行内元素有哪些？块级元素有哪些？ 空(void)元素有那些？ 37](#_Toc13535)

[45. CSS的盒子模型？ 37](#_Toc12597)

[46. 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？ 37](#_Toc30809)

[47. 如何居中div,如何居中一个浮动元素? 38](#_Toc26409)

[48. 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 ？ 39](#_Toc9201)

[49. 列出display的值，说明他们的作用。position的值， relative和absolute定位原点是？ 40](#_Toc4124)

[50. absolute的containing block计算方式跟正常流有什么不同？ 40](#_Toc27013)

[51. 对WEB标准以及W3C的理解与认识 41](#_Toc13120)

[52. css的基本语句构成是? 41](#_Toc1306)

[53. 浏览器标准模式和怪异模式之间的区别是什么? 41](#_Toc12189)

[54. 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？ 41](#_Toc2123)

[55. 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？ 42](#_Toc3153)

[56. 什么是外边距重叠？重叠的结果是什么？ 42](#_Toc20995)

[58、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？ 42](#_Toc22476)

[57. 说display属性有哪些？可以做什么？ 43](#_Toc7491)

[58. 哪些css属性可以继承？ 43](#_Toc30239)

[59. css优先级算法如何计算？ 43](#_Toc9912)

[60. b标签和strong标签,i标签和em标签的区别？ 43](#_Toc32405)

[61. 有那些行内元素、有哪些块级元素、盒模型？ 43](#_Toc10355)

[62. 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](#_Toc2857)

[63. 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](#_Toc9224)

[64. CSS的盒模型由什么组成？ 45](#_Toc12440)

[65. 说说display属性有哪些？可以做什么？ 46](#_Toc30416)

[66. 哪些css属性可以继承？ 46](#_Toc19489)

[67. css优先级算法如何计算？ 46](#_Toc20782)

# 一、HTML和CSS

## 你做的页面在哪些流览器测试过？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](http://baike.baidu.com/view/1369399.htm" \l "1_5" \t "_blank))

## 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）

## Quirks(怪癖）模式是什么？它和Standards（标准）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，然而这时的问题就来了，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（文档类型定义）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。

[](http://jbcdn2.b0.upaiyun.com/2014/10/511748c935cab6e8c46d56b2e6e9c342.png)

设置行内元素的高宽：在Standards模式下，给<span>等行内元素设置wdith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

用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。）

## 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。

## 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

## 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？

## 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

## 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。

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

sessionStorage用于本地存储一个会话（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，Web Storage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。

## 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Hypertext Reference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。

## 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。（是否有关注新技术，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](http://baike.baidu.com/view/4447461.htm)格式图像的体积要比JPEG格式图像小40%

## 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）

## 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（域名服务器）缓存，cdn（）缓存，浏览器缓存，服务器缓存。

## 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。

## 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；<strong>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”,但现在它们他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。

便于团队开发和维护

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](http://www.cssforest.org/blog/index.php?id=134)。

## 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。如果你的是商业网站，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。

合理的标签使用

## 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

## 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么样定义的？

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。

## 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-1000em

## 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）

## 什么是Css Hack？ie6,7,8的hack分别是什么？

答案：针对不同的浏览器写不同的CSS code的过程，就是CSS hack。

示例如下：

#test{

        width:300px;

        height:300px;

        background-color:blue;      /\*firefox\*/

        background-color:red\9;      /\*all ie\*/

        background-color:yellow;    /\*ie8\*/

        +background-color:pink;        /\*ie7\*/

        \_background-color:orange;       /\*ie6\*/    }

        :root #test { background-color:purple\9; }  /\*ie9\*/

@media all and (min-width:0px)

{ #test {background-color:black;} }  /\*opera\*/

@media screen and (-webkit-min-device-pixel-ratio:0)

{ #test {background-color:gray;} }       /\*chrome and safari\*/

## 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> 、<img> 、<button> 、<texterea> 、<label>。

## 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。

## 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！）

## 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。

## 如何垂直居中一个浮动元素？

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | // 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      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<img>?（用更简便的方法。）

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #container     //<img>的容器设置如下  {      display:table-cell;      text-align:center;      vertical-align:middle;  } |

## 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。

## 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](http://necolas.github.io/normalize.css/)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。

## 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](http://www.lesscss.org/)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。

## 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CSS display: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。

## 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）

## 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



## 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

## 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

## 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。

## 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

## 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

## HTML与XHTML——二者有什么区别？

1. 所有的标记都必须要有一个相应的结束标记

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

## 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dislpay: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 IE 8\*/

}

10. IE6不支持PNG透明背景，解决办法: IE6下使用gif图片

## 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。

## 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

## 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。

## 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。

## 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 b span 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：

<br><hr><img><input><link><meta>鲜为人知的是： <area><base><col><command>

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

## 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 pading;

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

## 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。

## 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%;

}

## 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游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,even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 : a:link {} a:visited {} a:hover {} a:active {}

## 列出display的值，说明他们的作用。position的值， relative和absolute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。

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

lock-level boxes  
一个 block-level element ('display' 属性值为 'block', 'list-item' 或是 ‘table’) 会生成一个 block-level box，这样的盒子会参与到 block-formatting context (一种布局的方式) 中。  
block formatting context  
在这种布局方式下，盒子们自所在的 containing block 顶部起一个接一个垂直排列，水平方向上撑满整个宽度 (除非内部的盒子自己内部建立了新的 BFC)。  
containing block  
一般来说，盒子本身就为其子孙建立了 containing block，用来计算内部盒子的位置、大小，而对内部的盒子，具体采用哪个 containing block 来计算，需要分情况来讨论：  
  
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；  
否则则由这个祖先元素的 padding box 构成。  
根元素所在的 containing block 被称为 initial containing block，在我们常用的浏览器环境下，指的是原点与 canvas 重合，大小和 viewport 相同的矩形；  
对于 position 为 static 或 relative 的元素，其 containing block 为祖先元素中最近的 block container box 的 content box (除 margin, border, padding 外的区域)；  
对于 position:fixed 的元素，其 containing block 由 viewport 建立；  
对于 position:absolute 的元素，则是先找到其祖先元素中最近的 position 属性非 static 的元素，然后判断：  
如果都找不到，则为 initial containing block。

## 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；

## css的基本语句构成是?

选择器{属性1:值1;属性2:值2;……}

## 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

## 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

## 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> 、<img> 、<button> 、<textarea> 、<label>

## 什么是外边距重叠？重叠的结果是什么？

　　答案：

　　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：

1. 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。
2. 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。
3. 两个外边距一正一负时，折叠结果是两者的相加的和。

## **58、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](http://necolas.github.io/normalize.css/" \t "http://www.cnblogs.com/coco1s/p/_blank)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。

## 说display属性有哪些？可以做什么？

display:block行内元素转换为块级元素

display:inline块级元素转换为行内元素

display:inline-block转为内联元素

## 哪些css属性可以继承？

可继承： font-size font-family color, ul li dl dd dt;

不可继承 ：border padding margin width height ;

## 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;

## 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。

## 有那些行内元素、有哪些块级元素、盒模型？

1.内联元素(inline element)

a – 锚点

abbr – 缩写

acronym – 首字

b – 粗体(不推荐)

big – 大字体

br – 换行

em – 强调

font – 字体设定(不推荐)

i – 斜体

img – 图片

input – 输入框

label – 表格标签

s – 中划线(不推荐)

select – 项目选择

small – 小字体文本

span – 常用内联容器，定义文本内区块

strike – 中划线

strong – 粗体强调

sub – 下标

sup – 上标

textarea – 多行文本输入框

tt – 电传文本

u – 下划线

var – 定义变量

2、块级元素

address – 地址

blockquote – 块引用

center – 举中对齐块

dir – 目录列表

div – 常用块级容易，也是css layout的主要标签

dl – 定义列表

fieldset – form控制组

form – 交互表单

h1 – 大标题

h2 – 副标题

h3 – 3级标题

h4 – 4级标题

h5 – 5级标题

h6 – 6级标题

hr – 水平分隔线

isindex – input prompt

menu – 菜单列表

noframes – frames可选内容，（对于不支持frame的浏览器显示此区块内容）

noscript – ）可选脚本内容（对于不支持script的浏览器显示此内容）

ol – 排序表单

p – 段落

pre – 格式化文本

table – 表格

ul – 非排序列表

3.CSS盒子模型包含四个部分组成：

内容、填充（padding）、边框（border）、外边界（margin）。

## 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？

#ID > .class > 标签选择符 !important优先级高

## 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

## CSS的盒模型由什么组成？

内容，border ,margin，padding

## 说说display属性有哪些？可以做什么？

display:block行内元素转换为块级元素

display:inline块级元素转换为行内元素

display:inline-block转为内联元素

## 哪些css属性可以继承？

可继承： font-size font-family color, ul li dl dd dt;

不可继承 ：border padding margin width height ;

## 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;