HTML+CSS+CSS3 综合面试题

1. <img> 的 title 和 alt 有什么区别

- 通常当鼠标滑动到元素上的时候显示
- alt 是<img> 的特有属性，是图片内容的等价描述，用于图片无法加载显示、读屏器阅读图片。可提图片高可访问性，除了纯装饰图片外都必须设置有意义的值，搜索引擎会重点分析。

2. html5 有哪些新特性、移除了那些元素？

- HTML5 现在已经不是 SGML 的子集，主要是关于图像，位置，储，多任务等功能的增加
  - 绘画 canvas
  - 用于媒介回放的 video 和 audio 元素
  - 本地离线存储 localStorage 长期存储数据，浏览器关闭后数据不丢失
  - sessionStorage 的数据在浏览器关闭后自动删除
  - 语意化更好的内容元素，比如 article、footer、header、nav、section
  - 表单控件，calendar、date、time、email、url、search
  - 新的技术 webworker、websocket、Geolocation
- 移除的元素：
  - 纯表现的元素：basefont、big、center、font、s、strike、tt、u
  - 对可用性产生负面影响的元素：frame 、frameset 、noframes
- 支持 HTML5 新标签：
  - IE8/IE7/IE6 支持通过 document.createElement 方法产生的标签
  - 可以利用这一特性让这些浏览器支持 HTML5 新标签
  - 浏览器支持新标签后，还需要添加标签默认的样式
- 当然也可以直接使用成熟的框架、比如 html5shim

3. iframe 标签有那些缺点？

- iframe 会阻塞主页面的 Onload 事件
- 搜索引擎的检索程序无法解读这种页面，不利于 SEO
- iframe 和主页面共享连接池，而浏览器对相同域的连接有限制，所以会影响页面的并行加载
- 使用 iframe 之前需要考虑这两个缺点。如果需要使用 iframe ，最好是通过 javascript 动态给 iframe 添加 src 属性值，这样可以绕开以上两个问题

4. Xhtml 与 Html 有什么区别?

- 一个是功能上的差别
  - 主要是 XHTML 可兼容各大浏览器、手机以及 PDA ，并且浏览器也能快速正确地编译网页
- 另外是书写习惯的差别
  - XHTML 元素必须被正确地嵌套，闭合，区分大小写，文档必须拥有根素

5. 新的 html5 文档类型和字符集是什么?

- HTML5 文档类型很简单：HTML 使用的是 UTF-8

6. Html5 应用程序员缓存和浏览器缓存有什么区别 ?

- 应用程序缓存是 HTML5 的重要特新之一，提供了离线使用的功能，让应用程序可以获取本地的网站内容，例如 html、css、图片以及 JavaScript。这个特性可以提高网站性能，它的实现借助与 manifest。文件。

7. Html5 存储类型有什么区别?

- html5 能够本地存储数据，在之前都是 cookies 的。Html5 提供了两种本地存储方案：localStorage 持久性的本地存储，关闭浏览器也不会丢失。sessionStorage 关闭浏览器窗口数据也随之销毁。

8. HTML5 的离线储存怎么使用，工作原理能不能解释一下?

- 在用户没有与因特网连接时，可以正常访问站点或应用，在用户与因特连接时，更新用户机器上的缓存文件
- 原理：HTML5 的离线存储是基于一个新建的.appcache 文件的缓存机制(不是存储技术)，通过这个文件上的解析清单离线存储资源，这些资源就会像 cookie 一样被存储了下来。之后当网络在处于离线状态下时，浏览器会通过被离线存储的数据进行页面展示
- 如何使用：
  - 页面头部像下面一样加入一个 manifest 的属性；
  - 在 cache.manifest 文件的编写离线存储的资源
  - 在离线状态时，操作 window.applicationCache 进行需求实现

9. 语义化的理解

- 用正确的标签做正确的事情!
- HTML 语义化就是让页面的内容结构化，便于对浏览器、搜索引擎解析;
- 在没有样式 CSS 情况下也以一种文档格式显示，并且是容易阅读的。
- 搜索引擎的爬虫依赖于标记来确定上下文和各个关键字的权重，利 SEO 。
- 使阅读源代码的人对网站更容易将网站分块，便于阅读维护理解

10. 浏览器是怎么对 HTML5 是的离线存储资源管理和加载呢

- 在线的情况下，浏览器发现 html 头部有 manifest 属性，它会请求 manifest 文件，如果是第一次访问 app ，那么浏览器就会根据 manife 文件的内容下载相应的资源并且进行离线存储。如果已经访问过 app 并且资源已经离线存储了，那么浏览器就会使用离线的资源加载页面，然后浏览器会对比新的 manifest 文件与旧的 manifest 文件，如果文件没有发生改变，就不做任何操作，如果文件改变了，那么就会重新下载文件中的资源并进行离线存储。

- 离线的情况下，浏览器就直接使用离线存储的资源。

11. Doctype 作用？严格模式与混杂模式如何区分？它们有何意义?

- 页面被加载的时，link 会同时被加载，而@imort 页面被加载的时，link 会同时被加载，而@import 引用的 CSS 会等到页面被加载完再加载 import 只在 IE5 以上才能识别，而 link 是 XHTML 标签，无兼容问题 link 方式的样式的权重 高于@import 的权重。
- <!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严格模式的排版和 JS 运作模式是 以该浏览器持的最高标准运行在混杂模式中，页面以宽松的向后兼容的方式显示。模拟老式浏览器的行以防止站点无法工作。 DOCTYPE 不存在或格式不正确会导致文档以混杂模式呈现。

12. 行内元素有哪些？块级元素有哪些？

- 行内元素有： a b span img input select strong
- 块级元素有： div ul ol li dl dt dd h1 h2 h3 h4… p

13. 空(void)元素有那些？行内元 素和块级元素有什么区别？

- 空元素：
 <hr> <img> <input> <link> <meta>
- 行内元素不可以设置宽高，不独占一行
- 块级元素可以设置宽高，独占一行

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

- cookie 是网站为了标示用户身份而储存在用户本地终端（Client Side）上的数据（通常经过加密）
- cookie 数据始终在同源的 http 请求中携带（即使不需要），记会在浏览器和服务器间来回传递
- sessionStorage 和 localStorage 不会自动把数据发给服务器，仅在本地保存
- 存储大小：
  - cookie 数据大小不能超过 4k
  - sessionStorage 和 localStorage 虽然也有存储大小的限制，但比 cookie 大得多，可以达到 5M 或更大
- 有期时间：
  - localStorage 存储持久数据，浏览器关闭后数据不丢失除非主动删除数据
  - sessionStorage 数据在当前浏览器窗口关闭后自动删除
  - cookie 设置的 cookie 过期时间之前一直有效，即使窗口或浏览器关闭

15. 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
- 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

16. 请列举几种隐藏元素的方法

- visibility:hidden,这个属性只是简单隐藏，但是元素暂用的空间任然存在。
- opacity:0,一个 CSS3 属性，设置 0 为透明，它可以被 transition 和 animate。
- position:absolute, 元素脱离文档流，处于普通文档之上，给它设置一个很大的 left 负值定位，使元素定位在可见区域之外。
- display:none, 元素不可见，不占用文档空间。
- transform:scale(0), 将一个元素设置无限小，这个元素将不可见。、
- html5 hidden attribute:hidden, 属性效果和 display:none 一样，记录一个元素的状态。

height:0;overflow:hidden，将元素在垂直方向上收缩为 0，使元素消失。

- filter:blur(0)，将一个元素的模糊度设置为 0

17. WEB 标准和 W3C 标准是什么?

- 标签闭合、标签小写、不乱嵌套、使用外链 css 和 js 结构行为表现的分离

18. HTML 全局属性有哪些?

- class :为元素设置类标识
- data-*：为元素增加自定义属性
- draggable：设置元素是否可拖拽
- id：元素 id，文档内唯一
- lang：元素内容的的语言
- style：行内 css 样式
- title：元素相关的建议信息

23. viewport 详解

```
<meta name="viewport"
    content="width=devicewidth,
    userscalable=no,
    initialscale=1.0,
    maximumscale=1.0,
    minimum-scale=1.0"
>
//width 设置 viewport 宽度，为一个正整数，或字符串‘device-width’
// device-width 设备宽度
// height 设置 viewport 高度，一般设置了宽度，会自动解析出高度，可以不用设置
// initial-scale 默认缩放比例（初始缩放比例），为一个数字，可以带小数
// minimum-scale 允许用户最小缩放比例，为一个数字，可以带小数
// maximum-scale 允许用户最大缩放比例，为一个数字，可以带小数
// user-scalable 是否允许手动缩放
```

- 怎样处理移动端 1px 被渲染成 2px 问题?
  - mate 标签中的 viewport 属性，initial-scale 设置为 1rem 按照设计稿标准走，外加利用 transfrome 的 scale(0.5)缩小一倍即可;
  - mate 标签中的 viewport 属性，initial-scale 设置为 0.5,rem 按照设计稿标准走即可

24. meta 相关

- <DOCTYPE html> <!--H5 标准声明，使用 HTML5 doctype，不区分大小写-->
- <head lang=”en”> <!--标准的 lang 属性写法-->

- <meta charset='utf-8'><!--声明文档使用的字符编码-->
- <meta name="description" content="不超过 150 个字符"/><!--页面描述-->
- <meta name="keywords" content=""/><!-- 页面关键词-->
- <meta name="author" content="name, email@gmail.com"/><!--网页作者-->
- <meta name="robots" content="index,follow"/><!--搜索引擎抓取-->
- <meta name="apple-mobile-web-app-title" content="标题"><!--iOS 设备 begin-->
- <meta name="apple-mobile-web-app-capable" content="yes"/><!--添加到主屏后的标题（i 是否启用 WebApp 全屏模式，删除苹果默认的工具栏和菜单栏-->
- <meta name="renderer" content="webkit"><!-- 启用 360 浏览器的极速模式(webkit)-->
- <meta name="x5-orientation" content="portrait"> <!--QQ 强制竖屏-->
  <!--设置页面不缓存-->
- <meta http-equiv="pragma" content="no-cache">
- <meta http-equiv="cache-control" content="no-cache">
- <meta http-equiv="expires" content="0">

25. 渐进增强和优雅降级之间的不同?

- 渐进增强：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：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。

26. div+css 的布局较 table 布局有什么优点?

- 改版的时候更方便 只要 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。

27. 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)

28. 页面导入样式时，使用 link 和 @import 有什么区别?

- link 属于 XHTML 标签，import 是 CSS 提供的方式。link 方式除了 CSS，还可以定 RSS，定义 rel 连接属性等，而 import 只能加载 CSS
- link 是页面加载时同时执行的，而 import 是在页面加载完之后，才会执行的页面加载速更快、结构化清晰、页面显示简洁
- link 支持使用 Javascript 控制 DOM 去改变样式；而 @import 支持
- 页 link 是 XHTML 标签，无兼容问题；@import 是在 CSS2.1 提出的，低版本(IE5 及以下的浏览器不支持。

29. 介绍一下你对浏览器内核的理解?

- 主要分成两部分：渲染引擎(layout engineer 或 linkRendering Engine)和 JS 引擎。
  - 渲染引擎：用于获取 html、css 和图片,然后会输出至显示器或打印机。
  - JS 引擎：解析和执行 javascript 来实现网页的动态效果。

30. HTML5 的工作原理能不能解释一下?

- HTML5 的离线存储是基于一个新建的.appcache 文件的缓存机制(不是存储技术)，通过这个文件上的解析清单离线存储资源，这些资源就会像 cookie 一样被存储了下来。之后当网络在处于离线状态下时，浏览器会通过被离线存储的数据进行页面展示。

31. Label 的作用是什么？是怎么用的？（加 for 或 包裹）？

- Label 义表单控制间的关系,当用户选择该标签时，浏览器会自动将焦点转到和标签相关的单控件上。

32. title 与 h1 的区别、b 与 strong 的区别、i 与 em 的区别?

- title 属性没有明确意义只表示是个标题，h1 则表示层次明确的标题，对页面信息的抓取也有很大的影响。
- strong 是标明重点内容，有语气加强的含义，使用阅读设备阅读网络时：< strong >会重读，而< B >是展示强调内容。
- i 内容展示为斜体，em 表示强调的文本。

33. 解释一下 utf-8 和 GBK 和 ISISO8859-2 字符集?

- charset 属性规定 HTML 文档的字符
- UTF-8 是一种针对 Unicode 的可变长度字符编码。
- GBK 是汉字编码，是双字节码，可表示繁体字和简体字。
- ISO8859-2 字符集，也称为 Latin-2，收集了 东欧 字符。

34. 放在 HTML 里的哪一部分 JavaScript 会在页面加载的时候被执行?

- 在 body 部分中的 JavaScript 会在页面加载的时候被执行
- 在 head 部分中的 JavaScript 会在被调用的时候才执行。

35. Canvas 和 SVG 有什么区别?

- svg 绘制出来的每一个图形的元素都是独立的 DOM 节点，能够方便的绑定事件或用来修改。 canvas 输出的是一整幅画布
- svg 输出的图形是矢量图形，后期可以修改参数来自由放大缩小，不会失真和锯齿。而 canvas 输出标量画布，就像一张图片一样，放大会失真或者锯齿

36. 如何在页面上实现一个圆形的可点击区域?

- svg
- border-radius 纯 js 实现 需要求一个点在不在圆上简单算法、获取鼠标坐标等等

37. HTML5 为什么只需要写 <!DOCTYPE HTML5>?

- HTML5 不基于 SGML ，因此不需要对 DTD 进行引用，但是需要 doctype 来规范浏览器的行为
- HTML4.01 基于 SGML ,所以需要对 DTD 进行引用，才能告知浏览器文档所使用的文档类型

38. SEO 优化

- 合理的 title 、description 、keywords ： 搜索对着三项的权重逐个减小，titl 值强调重点即可，重要关键词出现不要超过 2 次，而且要靠前，不同页面 title 要有所不同；description 把页面内容高度概括，长度合适，不可过分堆砌关键词，不同页 description 有所不同； keywords 列举出重要关键词即可。
- 语义化的 HTML 代码，符合 W3C 规范：语义化代码让搜索引擎容易理解网页
- 重要内容 HTML 代码放在最前：搜索引擎抓取 HTML 顺序是从上到下，有的搜索引擎对抓取长度有限制，保证重要内容一定会被抓取。
- 重要内容不要用 js 输出：爬虫不会执行 js 取内容
- 少用 iframe ： 搜索引擎不会抓取 iframe 中的内容
- 非装饰性图片必须加 alt
- 提高网站速度：网站速度是搜索引擎排序的一个重要指标

39. 渲染优化

- 禁止使用 iframe 阻塞父文档 onload 事件）
  - iframe 会阻塞主页面的 Onload 事件

- ✧ 搜索引擎的检索程序无法解读这种页面，不利于 SEO
- ✧ iframe 和主页面共享连接池，而浏览器对相同域的连接有限制，所以会影响页面的并行加载
- ✧ 使用 iframe 之前需要考虑这两个缺点。如果需要使用 iframe ，最好是通过 javascript
- ✧ 动态给 iframe 添加 src 属性值，这样可以绕开以上两个问题

- 禁止使用 gif 图片实现 loading 效果（降低 CPU 消耗，提升渲染性能）
- 使用 CSS3 代码代替 JS 动画（尽可能避免重绘重排以及回流）
- 对于一些小图标，可以使用 base64 位编码，以减少网络请求。但不建议大图使用，比较耗费 CPU
- 页面头部的<style></style><script></script>会阻塞页面；（因为 Renderer 进程中 JS 线程和渲染线程是互斥的）
- 页面中空的 href 和 src 会阻塞页面其他资源的加载（阻塞下载进程）
- 网页 gzip ， CDN 托管， data 缓存 ， 图片服务器
- 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
- 用 innerHTML 代替 DOM 操作，减少 DOM 操作次数，优化 javascript 性能
- 当需要设置的样式很多时设置 className 而不是直接操作 style
- 少用全局变量、缓存 DOM 节点查找的结果。减少 IO 读取操作
- 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
- 对普通的网站有一个统一的思路，就是尽量向前端优化、减少数据库操作、减少磁盘 IO

40. a：img 的 alt 与 title 有何异同？b：strong 与 em 的异同？

- 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

41. 网页制作会用到的图片格式有哪些？

- png-8 、png-24 、jpeg 、gif 、svg
- 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% 。

- Apng：全称是 “Animated Portable Network Graphics”，是 PNG 的位图动画扩展，可实现 png 格式的动态图片效果。04 年诞生，但一直得不到各大浏览器厂商的支持，直到日前得到 iOS safari 8 的支持，有望代替 GIF 成为下一代动态图标准

42. 从用户刷新网页，js 请求一般情况下有哪些地方会有缓存处理?

- alt(alt text) :为 dns 缓存，cdn 缓存，浏览器缓存，服务器缓存

43. 介绍一下标准的 CSS 的盒子模型？与低版本 IE 的盒子模型有什么不同的?

- 标准盒子模型：宽度=内容的宽度（content）+ border + padding + margin。
- 低版本 IE 盒子模型：宽度=内容宽度（content + border + padding）+ margin。

44. stylus/sass/less 区别?

- 均具有“变量”、“混合”、“嵌套”、“继承”、“颜色混合”五大基本特性。
- Scss 和 LESS 语法较为严谨，LESS 要求一定要使用大括号“{}”，Scss 和 Stylus 可以通过缩进表示层次与嵌套关系。
- Scss 无全局变量的概念，LESS 和 Stylus 有类似于其它语言的作用域概念。
- Sass 是基于 Ruby 语言的，而 LESS 和 Stylus 可以基于 NodeJS NPM 下载相应库进行编译。

45. CSS 优先级算法如何计算?

- 声明的元素选择符：1
- class 选择符：10
- id 符：100
- !important 声明的样式优先级最高，如果冲突再进行计算。
- 如果优先级相同，则选择最后出现的样式。
- 继承得到的样式的优先级最低。

46. CSS 选择器有哪些？哪些属性可以继承?

- id 选择器(#myid)
- 类选择器(.myclassname)
- 标签选择器(div, h1, p)
- 相邻选择器(h1 + p)
- 子选择器（ul > li）

- 后代选择器（li a）
- 通配符选择器（*）
- 属性选择器（a[rel=”external”]）
- 伪类选择器（a>:hover,li:nth-child）
- 可继承的属性：font-size font-faminly color
- 不可继承的样式：border,padding,margin,width,height

47. CSS3 新增伪类有那些?

- p:first-of-type 选择属于其父元素的首个元素
- p:last-of-type 选择属于其父元素的最后元素
- p:only-of-type 选择属于其父元素唯一的元素
- p:only-child 选择属于其父元素的唯一子元素
- p:nth-child(2) 选择属于其父元素的第二个子元素
- :enabled :disable 表单控件的禁用状态
- :checked 单选框或复选框被选中

48. CSS 合并方法?

- 避免使用 @import 引入多个 css 文件，可以使用 CSS 工具将 CSS 合并为一个 CSS 文件，例如使用 Sass\Compass 等

49. ::before 和:after 中双冒号和单冒号 有什么区别？以及作用

- 单冒号(:)用于 CSS3 伪类，双冒号(::)用于 CSS3 伪元素
- 用于区分伪类和伪元素

50. PNG /GIF /JPG 的区别及如何选

- GIF
  - 8 位像素，256 色
  - 无损压缩
  - 支持简单动画
  - 支持 boolean 透明
  - 适合简单动画

- JPEG
  - 颜色限于 256
  - 有损压缩

- 可控制压缩质量
- 支持透明
- 适合照片

- PNG
  - 有 PNG8 和 truecolor PNG
  - PNG8 类似 GIF 颜色上限为 256，文件小，支持 alpha 透明度，无动画
  - 适合图标、背景、按钮

51. display:?nline-block 什么时候不会显示间隙?

- 移除空格
- 使用 margin 负值
- 使用 font-size:0
- letter-spacing
- word-spacing

52. position 的值，relative 和 absolute 的区别

- absolut：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 ：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
- inherit 规定从父元素继承 position 属性的值

53. CSS 优先级算法如何计算?

- 优先级就近原则，同权重情况下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

54. display 有哪些值? 说明他们的作用?

- block 转换成块状元素。
- inline 转换成行内元素。
- none 设置元素不可见。
- 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。
- table 此元素会作为块级表格来显示

- `inherit` 规定应该从父元素继承 `display` 属性的值

55. 谈谈浮动和清除浮动?

- 动的框可以向左或向右移动，直到他的外边缘碰到包含框或另一个浮动框的边框为止。由于浮动框不在文档的普通流中，所以文档的普通流的块框表现得就像浮动框不存在一样。浮动的块框会漂浮在文档普通流的块框上

56. 对 BFC 规范的理解?

- 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

57. CSS3 有哪些新特性?

- 新增各种 `css` 选择器
- 圆角 `border-radius`
- 多列布局
- 阴影和反射
- 文字特效 `text-shadow`
- 线性渐变
- 旋转 `transform`

58. 为什么要初始化 CSS 样式?

- 因为浏览器的兼容问题，不同浏览器对有些标签的默认值是不同的，如果没对 `CSS` 初始化往往会出现浏览器之间的页面显示差异。
- 初始化样式会对 `SEO` 有一定的影响，但鱼和熊掌不可兼得，但力求影响最小的情况下初始化

59. visibility 属性有个 collapse 属性值在不同浏览器下以后什别?

- 当一个元素的 visibility 属性被设置成 collapse 值后，对于一般的元素，它的表现跟 hidden 是一样的
  - ✧ `chrome` 中，使用 `collapse` 值和使用 `hidden` 没有区别
  - ✧ `firefox`，`opera` 和 `IE`，使用 `collapse` 值和使用 `displat:none` 没有什么区别

60. display:none 与 visibility:hideen 的区别?

- `display: none` 不显示对应的元素，在文档布局中不再分配空间（回流+重绘）。
- `visibility: hidden` 隐藏对应元素，在文档布局中仍保留原来的空间（重绘）
- `display:none` ;会让元素完全从渲染树中消失，渲染的时候不占据任何空间，`sibility: hidden` ，会让元素从渲染树消失，渲染师元素继续占据空间，只是内容不可见

61. display、float、position 的关系?

- 如果 display 取值为 none，那么 position 和 float 都不起作用，这种情况下元不产生框
- 否则，如果 position 取值为 absolute 或者 fixed 框就是绝对定位的，float 计算值为 none ，display 根据下面的表格进行调整。
- 否则，如果 float 不是 none，框是浮动的，display 根据下表进行调整
- 否则，如果元素是根元素，display 根据下表进行调整
- 其他情况下 display 的值为指定值

  总结起来：**绝对定位、浮动、根元素都需要调整 display**

62. link 与 @import 的区别?

- link 是 HTML 方式，@import 是 CSS 方式
- link 最大限度支持并行下载，@import 过多嵌套导致串行下载，出现 FOUC (文档样式短暂失效)
- link 可以通过 rel="alternate stylesheet" 指定候选样式
- 浏览器对 link 支持早于@import，可以使用@import 对老浏览器隐藏样式
- @import 必须在样式规则之前，可以在 css 文件中引用其他文件
- 总体来说：link 优于@import

63. css sprite 是什么,有什么优缺点?

- 概念：将多个小图片拼接到一个图片中。通过 background-position 和元素尺寸调节需要显示的背景图案。
- 优点
  - 减少 HTTP 请求数，极大地提高页面加载速度
  - 增加图片信息重复度，提高压缩比，减少图片大小
  - 更换风格方便，只需在一张或几张图片上修改颜色或样式即可实现
- 缺点
  - 图片合并麻烦
  - 维护麻烦，修改一个图片可能需要从新布局整个图片，样式

64. 什么是 FOUC?如何避免?

- Flash Of Unstyled Content：用户定义样式表加载之前浏览器使用默认样式显示文档，用户样式加载渲染之后再从新显示文档，造成页面闪烁。
- **解决方法**：把样式表放到文档的<head>

65. 如何创建块级格式化上下文(block formatting context),BFC 有什么用?

- 创建规则：1. 根元素 2. 浮动元素（float 不取值为 none）3. 绝对定位元素（position 为 absolute 或 fixed ）4. display 取值为 inline-block、table-cell、tablecaption、flex、inline-flex 之一的元素 5. overflow 不取值为 visible 的元素。
- 作用：1. 可以包含浮动元素 2. 不被浮动元素覆盖 3. 阻止父子元素的 margin 折叠

66. 清除浮动的几种方式，各自的优缺点?

- 父级 div 定义 height
- 结尾处加空 div 标签 clear:both
- 父级 div 定义伪类 :after 和 zoom
- 父级 div 定义 overflow:hidden
- 父级 div 也浮动，需要定义宽度
- 结尾处加 br 标签 clear:both
- 比较好的是第 3 种方式，好多网站都这么用

67. 行内元素 float:left 后是否变为块级元素?

- 行内元素设置成浮动之后变得更加像是 inline-block（行内块级元素，设置成这个属性的元素会同时拥有行内和块级的特性，最明显的不同是它的默认宽度不是 100%），这时候给行内元素设置 padding-top 和 padding-bottom 者 width、height 都是有效果的

68. 如果需要手动写动画，你认为最小时间间隔是多久，为什么?

- 多数显示器默认频率是 60Hz，即 1 秒刷新 60 次，所以理论上最小间隔为 1/60*1000ms = 16.7ms

69. 列出你所知道可以改变页面布局的属性?

- position、display、float、width、height、margin、padding、top、left、right

70. CSS 在性能优化方面的实践?

- css 压缩与合并、Gzip 压缩
- css 文件放在 head 里、不要用@import
- 尽量用缩写、避免用滤镜、合理使用选择器

71. base64 的原理及优缺点

- 优点可以加密，减少了 HTTP 请求
- 缺点是需要消耗 CPU 进行编解码

72. CSS 不同选择器的权重

- ! important 规则最重要，大于其它规则
- 行内样式规则，加 1000
- 对于选择器中给定的各个 ID 属性值，加 100
- 对于选择器中给定的各个类属性、属性选择器或者伪类选择器，加 10
- 对于选择其中给定的各个元素标签选择器，加 1
- 如果权值一样，则按照样式规则的先后顺序来应用，顺序靠后的覆盖靠前的规则

73. CSS3 动画

- 依靠 CSS3 中提出的三个属性：transition、transform、animation
- transition：定义了元素在变化过程中是怎么样的，包含 transition-property、transition-duration、transition-timing-function、transition-delay。
- transform：定义元素的变化结果，包含 rotate、scale、skew、translate。
- animation：动画定义了动作的每一帧（@keyframes）有什么效果，包括 animation-name ，animation-duration、animation-timing-function、animation-delay、animation-iteration-count、animation-direction

74. postcss 的作用

- 可以直观的理解为：它就是一个平台。为什么说它是一个平台呢？因为我们直接用它，感觉不能干什么事情，但是如果让一些插件在它上面跑，那么将会很强大
- PostCSS 提供了一个解析器，它能够将 CSS 解析成抽象语法树通过在 PostCSS 这个平台上我们能够开发一些插件，来处理我们的 CSS，比如热门的：autoprefixer
- postcss 可以对 sass 处理过后的 css 再处理 最常见的就是 autoprefixer

75. 伪类和伪元素的区别

- 伪类表状态
- 伪元素是真的有元素
- 前者单冒号，后者双冒号

76. 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，
- 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！）

77. CSS 中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

78. CSS 自适应布局?

- 左侧浮动或者绝对定位，然后右侧 margin 撑开
- 使用 包含，然后靠负 margin 形成 bfc
- 使用 flex 弹性布局

79. 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

80. 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 。

81. 使用 css 预处理器吗?

- Less 和 Sass

82. 怎么让 Chrome 支持小于 12px 的文字?

- transform:scale() 这个属性只可以缩放可以定义宽高的元素，而行内元素是没有宽高的，我们可以加上一个 display:inline-block ；
- p{font-size:10px;-webkit-transform:scale(0.8);} //0.8 是缩放比例
- css 的属性，可以缩放大小

83. 两种以上方式实现已知或者未知宽度的垂直水平居中?

- 第一种:

```
.wraper {
  position: relative;
  .box {
    position: absolute;
    top: 50%;
    left: 50%;
    width: 100px;
    height: 100px;
    margin: -50px 0 0 -50px;
  }
}
```

- 第二种:

```
.wraper {
  position: relative;
  .box {
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
  }
}
```

-
- 第三种:

```
.wraper {
  .box {
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100px;
  }
}
```

- 第四种:

```
.wraper {
  display: table;
  .box {
    display: table-cell;
    vertical-align: middle;
  }
}
```

84. 说一说 css3 的 animation

- css3 的 `animation` 是 css3 新增的动画属性，这个 css3 动画的每一帧是通过`@keyframes` 来声明的，`keyframes` 声明了动画的名称，通过 `from`、`to` 或者是百分比来定义
- 每一帧动画元素的状态，通过 `animation-name` 来引用这个动画，同时 css3 动画也可以义动画运行的时长、动画开始时间、动画播放方向、动画循环次数、动画播放的方式，
- 相关的动画子属性有：`animation-name` 定义动画名、`animation-duration` 定义动画播放的时长、`animation-delay` 定义动画延迟播放的时间、`animation-direction` 定义动画的播放方向、`animation-iteration-count` 定义播放次数、`animation-fill-mode` 定义动画播放之后的状态、`animation-play-state` 定义播放状态，如暂停运行等、`animation-timing-function`
- 定义播放的方式，如恒速播放、艰涩播放等。

85. 重绘和回流（重排）是什么，如何避免?

- DOM 的变化影响到了元素的几何属性（宽高）,浏览器重新计算元素的几何属性，其他元素的几何
- 属性和位置也会受到影响，浏览器需要重新构造渲染树，这个过程称为重排，浏览器将受到影响的部分
- 重新绘制到屏幕上的过程称为重绘。引起重排的原因有：
  - ✧ 添加或者删除可见的 DOM 元素
  - ✧ 元素位置、尺寸、内容改变
  - ✧ 浏览器页面初始化
  - ✧ 浏览器窗口尺寸改变，重排一定重绘，重绘不一定重排
- 减少重绘和重排的方法：
  - ✧ 不在布局信息改变时做 `DOM` 查询
  - ✧ 使用 `cssText` 或者 `className` 一次性改变属性
  - ✧ 使用 `fragment`
  - ✧ 对于多次重排的元素，如动画，使用绝对定位脱离文档流，让他的改变不影响到其他元素

86. 水平居中的方法?

- 元素为行内元素，设置父元素 `text-align:center`
- 如果元素宽度固定，可以设置左右 `margin` 为 `auto`；
- 如果元素为绝对定位，设置父元素 `position` 为 `relative`，元素 `left:0;right:0;margin:auto;`
- 使用 `flex-box` 布局，指定 `justify-content` 属性为 center
- `splay` 设置为 `tabel-ceil`

87. css 有个 content 属性吗？有什么作用？有什么应用

- css 的 `content` 的属性专门应用在 `before/after` 伪元素上，用于来插入生成内容。最常见的应用是利用伪类清除浮动。

```
/**一种常见利用伪类清除浮动的代码**/
.clearfix:after {
  content: "."; //这里利用到了 content 属性
  display: block;
  height: 0;
  visibility: hidden;
  clear: both;
}
.clearfix {
  *zoom: 1;
}
```

88. 什么是响应式设计，其原理是什么？如何兼容低版本的 IE?

- 响应式网站设计(Responsive Web design)是一个网站能够兼容多个终端，而不是为每一个终端做一个特定的版本。
- 基本原理是通过媒体查询检测不同的设备屏幕尺寸做处理。
- 页面头部必须有 meta 声明的 viewport。