

HTML+CSS+JS（面试题）

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](#))

3、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。

4、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。

5、什么是Css Hack？ie6,7,8的hack分别是什么？

针对不同的浏览器写不同的CSS code的过程，就是CSS hack。

<pre>#test{ width:300px; height:300px; /firefox/ background-color:red\9; /all ie/ /ie8/ +background-color:pink; /ie7/ _ /ie6/ } :root #test { background- color:purple\9; } /ie9/ @media all and (min-width:0px){ #test {} } /opera/ @media screen and (-webkit-min-device-pixel-ratio:0){ #test {background-color:gray;}} /chrome and safari/</pre>

6、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。

7、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.le 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 */
```

```
}
```

1. IE6不支持PNG透明背景，解决办法: IE6下使用gif图片

HTML5+CSS3

1、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？

新特性：

- \1. 拖拽释放(Drag and drop) API
- \2. 语义化更好的内容标签 (header,nav,footer,aside,article,section)
- \3. 音频、视频API(audio,video)
- \4. 画布(Canvas) API
- \5. 地理(Geolocation) API
- \6. 本地离线存储 localStorage 长期存储数据，浏览器关闭后数据不丢失；
- \7. sessionStorage 的数据在浏览器关闭后自动删除
- \8. 表单控件，calendar、date、time、email、url、search
- \9. 新的技术webworker, websocket, Geolocation

移除的元素：

- \1. 纯表现的元素：basefont，big，center，font，s，strike，tt，u；
- \2. 对可用性产生负面影响的元素：frame，frameset，noframes；

HTML5兼容性解决方案

\1. 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（当然最好的方式是直接使用成熟的框架、使用最多的是html5shim框架）：

```
<!--[if lt IE 9]>
```

```
<![endif]-->
```

2你如何对网站的文件和资源进行优化？****

文件合并

文件最小化/文件压缩

使用CDN托管

缓存的使用

3 CSS3新增伪类有那些？

p:first-of-type 选择属于其父元素的首个

元素的每个

元素。

p:last-of-type 选择属于其父元素的最后元素的每个元素。

p:only-of-type 选择属于其父元素唯一的元素的每个元素。

p:only-child 选择属于其父元素的唯一子元素的每个元素。

p:nth-child(2) 选择属于其父元素的第二个子元素的每个元素。

:enabled、:disabled 控制表单控件的禁用状态。

:checked, 单选框或复选框被选中。

4 H5+CSS3兼容性解决的方案

1 htmlshiv.js框架 HTML5shiv通过JavaScript 来创建HTML5元素(如 main, header, footer等)

```
<!--[if lt IE 9]>
```

```
<![endif]-->
```

2 selectivizr.js

selectivizr是一个JavaScript工具，使IE浏览器也可以支持CSS3伪类以及属性选择器

使用很简单，只要把js include到页面上，然后你就可以

```
<!--[if (gte IE 6)&!(lte IE 8)]>
```

```
<![endif]-->
```

3 条件注释

5 CSS3有哪些新特性？

\1. CSS3实现圆角 (border-radius) , 阴影 (box-shadow) ,

\2. 对文字加特效 (text-shadow、) , 线性渐变 (gradient) , 旋转 (transform)

3.transform:rotate(9deg) scale(0.85,0.90) translate(0px,-30px) skew(-9deg,0deg);// 旋转,缩放,定位,倾斜

\4. 增加了更多的CSS选择器 多背景 rgba

\5. 在CSS3中唯一引入的伪元素是 ::selection.

\6. 媒体查询，多栏布局

\7. border-image

JS+jQuery部分内容

1 数组和字符串**中常用的方法**

1) 数组中常用的方法

Push()向数组的末尾增加一项 unshift()向数组开头增加一项

pop()删除数组的末尾项 shift()删除数组开头项

Splice() 删除数组中的任意项 concat拼接

Join() 把数组中的每一项 按照指定的分隔符拼接成字符串

reverse:倒序数组 返回值倒序数组 原有数组改变 sort:根据匿名函数进行冒泡排序 b-a倒序 a-b升序

兼容性不好: indexOf:返回获取项在数组中的索引 lastIndexOf:返回获取项在数组中出现的最后一次索引 forEach: 循环遍历数组 参数是一个匿名函数 默认返回为undefined map: 循环遍历数组 参数是一个匿名函数

2) 字符串中常用的方法

substring方法返回位于String对象中指定位置的子字符串。

substr方法返回一个从指定位置开始的指定长度的子字符串。

slice方法返回字符串的片段。

split将一个字符串分割为子字符串,然后将结果作为字符串数组返回。

2 js兼容性问题

1 获取,设置标签的内容 IE和chrome浏览器支持innerText 火狐浏览器支持 textContent

2获取指定元素的第一个元素和最后一个元素 element.firstChild

IE679只能获取firstChild获取第一个字节节点,然后通过判断的节点的类型区的第一个子元素

3event.x 与 event.y, 在IE中, event 对象有x,y属性, 现代浏览器中, 与 event.x 等效的是 event.pageX, 但 event.pageY IE中没有

4问题说明: 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。

3说说你对闭包的理解

可以把闭包简单理解成“定义在一个函数内部的函数”

闭包有三个特性:

1.函数嵌套函数

2.函数内部可以引用外部的参数和变量

3.参数和变量不会被垃圾回收机制回收

由于闭包会使得函数中的变量都被保存在内存中, 内存消耗很大, 所以不能滥用闭包, 否则会造成网页的性能问题

4、javascript里面的继承怎么实现, 如何避免原型链上面的对象共享

用构造函数和原型链的混合模式去实现继承，避免对象共享可以参考经典的extend()函数，很多前端框架都有封装的，就是用一个空函数当做中间变量

5、JavaScript原型，原型链？有什么特点？

原型对象也是普通的对象，是对象一个自带隐式的 **proto** 属性，原型也有可能有自己的原型，如果一个原型对象的原型不为null的话，我们就称之为原型链。*

原型链是由一些用来继承和共享属性的对象组成的（有限的）对象链。

6请说出三种减少页面加载时间的方法。

1.优化图片

2.图像格式的选择（GIF：提供的颜色较少，可用在一些对颜色要求不高的地方）

3.优化CSS（压缩合并css，如margin-top,margin-left...）

4.网址后加斜杠（如www.campr.com/目录，会判断这个“目录是什么文件类型，或者是目录。）

5.标明高度和宽度（如果浏览器没有找到这两个参数，它需要一边下载图片一边计算大小，如果图片很多，浏览器需要不断地调整页面。这不但影响速度，也影响浏览体验。

当浏览器知道了高度和宽度参数后，即使图片暂时无法显示，页面上也会腾出图片的空位，然后继续加载后面的内容。从而加载时间快了，浏览体验也更好了。）

6.减少http请求（合并文件，合并图片）

7.你有哪些性能优化的方法？

（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)。

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、平时如何管理你的项目？

1)先期团队必须确定好全局样式（globe.css），编码模式(utf-8)等；

2)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）

3)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）

4)页面进行标注（例如 页面 模块 开始和结束）

5)CSS跟HTML分文件夹并行存放，命名都得统一（例如style.css）；

6)JS分文件夹存放 命名以该JS功能为准的英文翻译。

9、call和apply的区别

call和apply相同点：改变函数中this的指向

不同点：函数参数的传递形式

call将函数参数依次传入

apply将函数参数用一个数组的形式传入

10、原生JS的window.onload与jQuery 的\$(document).ready(function()), (function() {})有什么不同？

1)执行时间

window.onload必须等到页面内包括图片的所有元素加载完毕后才能执行。\$(document).ready()是DOM结构绘制完毕后就执行，不必等到加载完毕。

2)编写个数不同

window.onload不能同时编写多个，如果有多个window.onload方法，只会执行一个\$(document).ready()可以同时编写多个，并且都可以得到执行

3)简化写法 window.onload没有简化写法 \$(document).ready(function())可以简写成(function){};

11、split(),join()的区别？

split() 方法通过把字符串分割成子字符串来把一个 String 对象分割成一个字符串数组。

join() 方法将数组中的所有元素连接成一个字符串。

12、**谈一下jQuery中的bind(),live(),delegate(),on()的区别？**

bind：绑定事件，对新添加的事件不起作用，方法用于将一个处理程序附加到每个匹配元素的事件上并返回jQuery对象。

live：方法将一个事件处理程序附加到与当前选择器匹配的所有元素（包含现有的或将来添加的）的指定事件上并返回jQuery对象。

delegate：方法基于一组特定的根元素将处理程序附加到匹配选择器的所有元素（现有的或将来的）的一个或多个事件上。

On:支持直接在目标元素上绑定事件，也支持在目标元素的祖辈元素上委托绑定。在事件委托绑定模式下，即使是执行on()函数之后新添加的元素，只要它符合条件，绑定的事件处理函数也对其有效。

Ajax的面试题

1、解释jsonp的原理

动态创建script标签，回调函数

2、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。

3、Ajax的最大的特点是什么。

Ajax可以实现动态不刷新（局部刷新）

readyState属性 状态 有5个可取值：0=未初始化，1=启动 2=发送，3=接收，4=完成

4、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。
- 5)不容易调试。

5、get和post的区别？

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；

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。

也就是说Get是通过地址栏来传值，而Post是通过提交表单来传值。

6、**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

1.Get方式请求的跨域可以通过jsonp来解决。

<http://www.tuicool.com/articles/QBbuiaR>

2.Post方式的请求 cors解决跨域

Cors解释

http://www.cnblogs.com/Darren_code/p/cors.html

JSONP实现跨域

常用的jquery实现跨域调用

```
$.ajax({  
    url: "http://127.0.0.1/~chenjiebin/mycode/php/crossdomain/index.php",  
    dataType: "jsonp",  
    jsonp: "callback",  
    context: document.body,  
    success: function(data) {  
        console.log(data);  
    }  
});
```


这个调用实际上的实现原理是

在网页中构造一个script标签，将src设置为对应的url，并增加上相应的callback参数，形如如下格式：

请求的服务端代码如下：

```
$data = json_encode(array("id" => "1", "name" => "tom"));
```

```
callback = _GET["callback"];
```

```
echo callback . " (" . $data . ")";
```

实际上最后返回的内容就是一段js代码：

```
jQuery211018970995225637144_1465350372062({"id":"1","name":"tom"})
```

当浏览器获取到该段js代码后就会执行这个函数，从而实现回调ajax请求时设置的success方法。

jsonp实现的缺点

了解了原理后，就知道jsonp实现的跨域方式不支持post请求，只能支持get请求。但是如果需要支持post请求该怎么办呢？下面谈下服务器端设置的方式。

服务端设置支持跨域

主要是Access-Control-Allow-Origin头参数，该参数用来指定允许哪个来源的域请求。服务端代码如下：

```
// 表示支持所有来源的域进行请求
```

```
// 实际在操作过程中可以设置为指定域
```

```
header('Access-Control-Allow-Origin: *');
```

```
$data = json_encode(array("id" => "1", "name" => "tom"));
```

```
echo $data;
```

对应的js代码：

```
$.ajax({
```

```
  type: "POST",
```

```
  url: "http://127.0.0.1/~chenjiebin/mycode/php/crossdomain/header.php",
```

```
  dataType: "json",
```

```
  success: function(data) {
```

```
    console.log(data);
```

```
  }
```

```
});
```

可以支持post请求

移动web开发

1、px，em，rem的区别？

1) px像素 (Pixel)。绝对单位。像素px是相对于显示器屏幕分辨率而言的，是一个虚拟长度单位，是计算机系统的数字化图像长度单位，如果px要换算成物理长度，需要指定精度DPI。

2) em是相对长度单位，相对于当前对象内文本的字体尺寸。如当前对行内文本的字体尺寸未被人为设置，则相对于浏览器的默认字体尺寸。它会继承父级元素的字体大小，因此并不是一个固定的值。

3) rem是CSS3新增的一个相对单位 (rootem，根em)，使用rem为元素设定字体大小时，仍然是相对大小，但相对的只是HTML根元素。

2、**jQuery和Zepto的区别？各自的使用场景？Zepto比jQuery轻量的原因？**

jQuery主要用于PC端开发，Zepto比jQuery更加小巧，主要用于移动端，Zepto是不兼容低版本的浏览器原因：

1) Zepto轻量的原因是没有像jQuery一样处理移动端的兼容性问题，

2) Zepto的模块是放到不同的模块中，根据业务需要引入模块，减少代码量，提高网页的加载速度

3、bootstrap框架中的栅格系统的本质是什么？以及bootstrap框架的优缺点？

bootstrap的栅格系统的本质是通过css3的媒体查询实现的

优点:bootstrap框架便于快速布局，而且添加了更加灵活的栅格系统，增加了响应式设计，来适配不同的设备

缺点:对ie678的兼容性可能不太好，而且bootstrap是依赖于jQuery，代码的体积大，会影响网页的加载速度，不适合移动端。

4、如何解决移动端的click事件的延迟响应？

移动端click屏幕产生200-300 ms的延迟响应，往往会造成按钮点击延迟甚至是点击失效。

解决方案

1) github上有一个叫做fastclick的库，它也能规避移动设备上click事件的延迟响应，<https://github.com/ftlabs/fastclick>

2)zepto的touch模块，tap事件也是为了解决在click的延迟问题

5、什么是mvc开发模式？

Mvc是一种组织代码结构的方式，便于代码的管理和维护。1、标准盒模型和低版本IE盒模型（怪异模式）有什么区别？

标准盒模型：内容的宽度 (content) +border+padding+margin；

IE低版本盒模型：内容的宽度 (content+border+padding) +margin；

最主要的区别在于盒模型的宽度；

box-sizing属性是用来控制元素的盒子模型的解析模式，默认为content-box

content-box：w3c标准盒子模型，设置元素的height/width属性指的是content部分的高宽

border-box：IE传统盒子模型，设置元素的height/width属性指的是border+padding+content部分的高宽

2、利用CSS3属性写出一个三角形

3、怎样理解HTML5？

(1)、在前端领域H5是一个技术集合（技术栈），而不是单纯的一个技术点，所以不能理解是一个html的规范。

(2)、我们可以从html、css、js三个方面进行梳理

html:语义化标签、新增表单类型、新增表单属性

css：新增选择器、过渡、转换、动画、媒体查询

js:canvas绘图、ES6

(3)、从职能理解H5的开发

移动web的开发

响应式开发

单页面应用开发

混合APP开发

微信小程序

微信公众号开发

H5开发泛指对H5技术栈（HTML的提升、css的提升、JavaScript的提升）的综合使用开发网页应用程序

4、CSS3有哪些新的特性？

(1)、RGBA和透明度

(2)、background-image、background-origin、background-size、background-repeat

(3)、word-wrap(对长的不可分割的单词进行换行)word-wrap:break-word;

(4)、文字阴影：text-shadow：5px 5px 5px #ccc;（水平阴影，垂直阴影，模糊距离，阴影颜色）

(5)、font-face：自定义自己的字体

(6)、圆角（边框半径）：border-radius 属性用于创建圆角

(7)、盒阴影box-shadow:5px 5px 5px #ccc;

(8)、媒体查询：定义两套css，当浏览器的尺寸变化时会采用不同的属性

5、为什么移动端项目要使用box-sizing：border-box?

box-sizing:border-box;可以避免宽度溢出，造成横向滚动条（移动端项目都是非固定式宽度）

6、display:none与visibility：hidden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）

重绘：当render-tree中的一些元素需要更新属性，而这些属性只影响元素的外观、风格，而不影响布局，比如是background-color,则就称为重绘。

回流：当页面中的布局和几何属性改变时就需要回流，比如是：

<1>、添加或删除可见的DOM元素

<2>、元素位置的改变

<3>、元素尺寸的改变（边框、尺寸、填充、宽度、高度）

<4>、内容的改变（比如文本的改变和图片大小的改变而引起的计算值宽度和高度的改变）

<5>、页面渲染初始化

<6>、浏览器窗口尺寸的改变-resize事件发生时

回流必将引起重绘，重绘不一定会引起回流

7、对BFC（块级格式化上下文block formatting context）的理解？

简单的来说BFC是一种属性，这种属性会影响着元素的定位以及与其兄弟元素之间的相互作用。

9、position的值？

static（默认）：按照正常文档流进行排列

relative（相对定位）不脱离文档流，参考自身的top、right、bottom、left进行定位

absolute（绝对定位）参考其最近的一个非static的父级元素通过top、right、bottom、left进行定位

fixed（固定定位）所固定的参照对象是可视窗口的位置

10、常见的兼容性问题

<1>不同浏览器标签默认的padding和margin不一样，*{padding:0;margin:0}

<2>chrome浏览器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;

11、为什么会出现浮动和什么时候需要清除浮动？清除浮动的方式？

由于浮动元素不在文档流中，所以文档流的块框表现得就像浮动框不存在一样。浮动元素会漂浮在文档流的块框上。

浮动带来的问题：

<1>父元素的高度无法被撑开

<2>与浮动元素同级的非浮动元素（内联元素）会跟随其后

<3>若非第一个元素浮动，则该元素之前的元素也需要浮动，否则会影响页面显示的结构。

清除浮动的方式：

<1> 父级div定义高度

<2> 最后一个浮动元素后加空div标签，并添加样式clear：both

<3> 包含浮动元素的父标签添加样式overflow为hidden和auto

<4> 父级定义zoom

1、标准盒模型和低版本IE盒模型（怪异模式）有什么区别？

标准盒模型：内容的宽度（content）+border+padding+margin；

IE低版本盒模型：内容的宽度（content+border+padding）+margin；

最主要的区别在于盒模型的宽度；

box-sizing属性是用来控制元素的盒子模型的解析模式，默认为content-box

content-box：w3c标准盒子模型，设置元素的height/width属性指的是content部分的高宽

border-box：IE传统盒子模型，设置元素的height/width属性指的是border+padding+content部分的高宽

3、怎样理解HTML5？

（1）、在前端领域H5是一个技术集合（技术栈），而不是单纯的一个技术点，所以不能理解是一个html的规范。

（2）、我们可以从html、css、js三个方面进行梳理

html:语义化标签、新增表单类型、新增表单属性

css：新增选择器、过渡、转换、动画、媒体查询

js:canvas绘图、ES6

（3）、从职能理解H5的开发

移动web的开发

响应式开发

单页面应用开发

混合APP开发

微信小程序

微信公众号开发

H5开发泛指对H5技术栈（HTML的提升、css的提升、JavaScript的提升）的综合使用开发网页应用程序

4、CSS3有哪些新的特性？

（1）、RGBA和透明度

（2）、background-image、background-origin、background-size、background-repeat

（3）、word-wrap(对长的不可分割的单词进行换行)word-wrap:break-word;

（4）、文字阴影：text-shadow：5px 5px 5px #ccc;（水平阴影，垂直阴影，模糊距离，阴影颜色）

（5）、font-face：自定义自己的字体

（6）、圆角（边框半径）：border-radius 属性用于创建圆角

(7)、盒阴影box-shadow:5px 5px 5px #ccc;

(8)、媒体查询：定义两套css，当浏览器的尺寸变化时会采用不同的属性

5、为什么移动端项目要使用box-sizing：border-box?

box-sizing:border-box;可以避免宽度溢出，造成横向滚动条（移动端项目都是非固定式宽度）

6、display:none与visibility：hidden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）

重绘：当render-tree中的一些元素需要更新属性，而这些属性只影响元素的外观、风格，而不影响布局，比如是background-color,则就称为重绘。

回流：当页面中的布局和几何属性改变时就需要回流，比如是：

<1>、添加或删除可见的DOM元素

<2>、元素位置的改变

<3>、元素尺寸的改变（边框、尺寸、填充、宽度、高度）

<4>、内容的改变（比如文本的改变和图片大小的改变而引起的计算值宽度和高度的改变）

<5>、页面渲染初始化

<6>、浏览器窗口尺寸的改变-resize事件发生时

回流必将引起重绘，重绘不一定会引起回流

7、对BFC（块级格式化上下文block formatting context）的理解？

简单的来说BFC是一种属性，这种属性会影响着元素的定位以及与其兄弟元素之间的相互作用。

9、position的值？

static（默认）：按照正常文档流进行排列

relative（相对定位）不脱离文档流，参考自身的top、right、bottom、left进行定位

absolute（绝对定位）参考其最近的一个非static的父级元素通过top、right、bottom、left进行定位

fixed（固定定位）所固定的参照对象是可视窗口的位置

10、常见的兼容性问题

<1>不同浏览器标签默认的padding和margin不一样，*{padding:0;margin:0}

<2>chrome浏览器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;

11、为什么会出现浮动和什么时候需要清除浮动？清除浮动的方式？

由于浮动元素不在文档流中，所以文档流的块框表现得就像浮动框不存在一样。浮动元素会漂浮在文档流的块框上。

浮动带来的问题：

<1>父元素的高度无法被撑开

<2>与浮动元素同级的非浮动元素（内联元素）会跟随其后

<3>若非第一个元素浮动，则该元素之前的元素也需要浮动，否则会影响页面显示的结构。

清除浮动的方式：

<1> 父级div定义高度

<2> 最后一个浮动元素后加空div标签，并添加样式clear：both

<3> 包含浮动元素的父标签添加样式overflow为hidden和auto

<4> 父级定义zoom

5、移动端适应（响应式）

```
<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
```

使用 flexbox 布局

使用百分比布局

使用 rem、em

媒体查询实现响应式

1. Doctype作用，HTML5 为什么只需要写 <!DOCTYPE HTML>

doctype是一种标准通用标记语言的文档类型声明，目的是告诉标准通用标记语言解析器要使用什么样的文档类型定义（DTD）来解析文档。

```
1 <!DOCTYPE>
```

声明必须是HTML文档的第一行，位于html标签之前

HTML5不基于SGML，所以不需要引用DTD。在HTML5中<!DOCTYPE>只有一种

SGML: 标准通用标记语言,是现时常用的超文本格式的最高层次标准

2. 行内元素有哪些，块级元素有哪些，空(void)元素有那些

行内元素：a span i img input select b 等

块级元素：div ul ol li h1~h6 p table 等

空元素：br hr link 等

5. html5有哪些新特性

- 语义化标签: header footer nav section article aside 等

- 增强型表单：date (从一个日期选择器选择一个日期) email (包含 e-mail 地址的输入域) number (数值的输入域) range (一定范围内数字值的输入域) search (用于搜索域) tel (定义输入电话号码字段) 等
- 视频和音频：audio video
- Canvas绘图 SVG绘图
- 地理定位：Geolocation
- 拖放API：drag
- web worker：是运行在后台的 JavaScript，独立于其他脚本，不会影响页面的性能
- web storage: localStorage sessionStorage
- WebSocket: HTML5开始提供的一种在单个 TCP 连接上进行全双工通讯的协议

• 6. 描述一下 cookie，sessionStorage 和 localStorage 的区别

特性	Cookie	localStorage	sessionStorage
生命周期	可设置失效时间，没有设置的话，默认是关闭浏览器后失效	除非被手动清除，否则将会永久保存	仅在当前网页会话下有效，关闭页面或浏览器后就会被清除
存放数据大小	4KB左右	可以保存5MB的信息	
http 请求	每次都会携带在HTTP头中，如果使用cookie保存过多数据会带来性能问题	仅在客户端（即浏览器）中保存，不参与和服务器的通信	

7. 如何实现浏览器内多个标签页之间的通信

- 使用localStorage: localStorage.setItem(key,value)、localStorage.getItem(key)
- websocket协议
- webworker
-

9. src与href的区别

区别：src用于替代这个元素，而href用于建立这个标签与外部资源之间的关系

`<link href="style.css" rel="stylesheet" />` 浏览器加载到这里的时候，html的渲染和解析不会暂停，css文件的加载是同时进行的

`<script src="script.js"></script>` 当浏览器解析到这句代码时，页面的加载和解析都会暂停直到浏览器拿到并执行完这个js文件

10. 表单提交中Get和Post方式的区别

- Get一般用于从服务器上获取数据，Post向服务器传送数据
- Get传输的数据是拼接在Url之后的，对用户是可见的；Post的传输数据对用户是不可见的
- Get传送的数据量较小，不能大于2KB。Post传送的数据量较大，一般被默认为不受限制

- Get安全性非常低，Post安全性较高
- 在FORM提交的时候，如果不指定Method，则默认为Get请求

2. 清除浮动，什么时候需要清除浮动，清除浮动都有哪些方法

浮动的元素是脱离文档标准流的，如果我们不清楚浮动，那么就会造成父元素高度塌陷，影响页面布局。

清除浮动的方式：

- 为父元素设置高度
- 为父元素添加 `overflow:hidden`
- 伪元素

4. px和em和rem的区别

1 px

: 像素，相对长度单位。像素

1 px

是相对于显示器屏幕分辨率而言的

`em` 的值并不是固定的，会继承父级元素的字体大小，代表倍数

`rem` 的值并不是固定的，始终是基于根元素 `<html>` 的，也代表倍数

5. position的值有哪些

`static`：默认值。没有定位，元素出现在正常的流中

`relative`（相对定位）：生成相对定位的元素,相对于其正常（原先本身）位置进行定位

`absolute`（绝对定位）：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`（固定定位）：生成绝对定位的元素，相对于浏览器窗口进行定位

6. display:none与visibility : hidden的区别

区别	<code>display:none</code>	<code>visibility : hidden</code> 的
是否占据空间	不占据任何空间，在文档渲染时，该元素如同不存在（但依然存在文档对象模型树中）	该元素空间依旧存在
是否渲染	会触发 <code>reflow</code> （回流），进行渲染	只会触发 <code>repaint</code> （重绘），因为没有发现位置变化，不进行渲染
是否是继承属性	不是继承属性，元素及其子元素都会消失	是继承属性，若子元素使用了 <code>visibility:visible</code> ，则不继承，这个子孙元素又会显现出

7. CSS中link 和@import的区别

link属于XHTML标签，@import完全是CSS提供的一种方式,只能加载CSS

加载顺序的差别，当一个页面被加载的时候，link引用的CSS会同时被加载，而@import引用的CSS 会等到页面全部被下载完再被加载

兼容性的差别。由于@import是CSS2.1提出的所以老的浏览器不支持，而link标签无此问题

当使用javascript控制dom去改变样式的时候，只能使用link标签，因为@import不是dom可以控制的

8. 什么是响应式设计，响应式设计的基本原理是什么

响应式网站设计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

9. 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

10. CSS3有哪些新特性

- 实现圆角 border-radius，阴影 box-shadow，边框图片 border-image
- 对文字加特效 text-shadow，强制文本换行 word-wrap，线性渐变 linear-gradient
- 实现旋转 transform:rotate(90deg),缩放 scale(0.85,0.90),translate(0px,-30px) 定位,倾斜 skew(-9deg,0deg);
- 增加了更多的CSS选择器、多背景、rgba()
- 唯一引入的伪元素是 ::selection；
- 实现媒体查询 @media，多栏布局 flex
- 过渡 transition 动画 animation
-

11. ::before 和 :after中双冒号和单冒号有什么区别？解释一下这2个伪元素的作用

单冒号(:)用于CSS3伪类，双冒号(::)用于CSS3伪元素。（伪元素由双冒号和伪元素名称组成），双冒号是在当前规范中引入的，用于区分伪类和伪元素

12. CSS优化、提高性能的方法有哪些

- 移除空的css规则 (Remove empty rules)
- 正确使用display的属性
- 不滥用浮动、web字体
- 不声明过多的font-size
- 不在选择符中使用ID标识符
- 遵守盒模型规则
- 尽量减少页面重排、重绘

- 抽象提取公共样式，减少代码量

BFC:块级格式上下文。

定义：浮动元素和绝对定位元素，非块级盒子的块级容器（例如inline-blocks, table-cells, 和table-captions），以及overflow值不为“visible”的块级盒子，都会为他们的内容创建新的块级格式化上下文。在一个块级格式化上下文里，盒子从包含块的顶端开始垂直地一个接一个地排列，两个盒子之间的垂直的间隙是由他们的margin值所决定的。两个相邻的块级盒子的垂直外边距会发生叠加。在块级格式化上下文中，每一个盒子的左外边缘（margin-left）会触碰到容器的左边缘(border-left)（对于从右到左的格式来说，则触碰到右边缘），即使存在浮动也是如此，除非这个盒子创建一个新的块级格式化上下文。

1 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

2 box-sizing属性？

用来控制元素的盒子模型的解析模式，默认为content-box context-box：W3C的标准盒子模型，设置元素的height/width 属性指的是content部分的高/宽 border-box：IE传统盒子模型。设置元素的height/width属性指的是border + padding + content部分的高/宽

3 CSS选择器有哪些？哪些属性可以继承？

CSS选择符：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-family, color

不可继承的样式：border, padding, margin, width, height

优先级（就近原则）：!important > [id > class > tag] !important 比内联优先级高

4 CSS优先级算法如何计算？

元素选择符：1 class选择符：10 id选择符：100 元素标签：1000

!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。 5 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 :disabled 表单控件的禁用状态。 :checked 单选框或复选框被选中。

7 display有哪些值？说明他们的作用？

inline（默认）-内联 none-隐藏 block-块显示 table-表格显示 list-item-项目列表 inline-block

8 position的值？

static（默认）：按照正常文档流进行排列；relative（相对定位）：不脱离文档流，参考自身静态位置通过 top, bottom, left, right 定位；absolute(绝对定位)：参考距其最近一个不为static的父级元素通过top, bottom, left, right 定位；fixed(固定定位)：所固定的参照对象是可视窗口。

9 CSS3有哪些新特性？

RGBA和透明度 background-image background-origin(content-box/padding-box/border-box) background-size background-repeat word-wrap (对长的不可分割单词换行) word-wrap: break-word 文字阴影: text-shadow: 5px 5px 5px #FF0000; (水平阴影, 垂直阴影, 模糊距离, 阴影颜色) font-face属性: 定义自己的字体 圆角(边框半径): border-radius 属性用于创建圆角 边框图片: border-image: url(border.png) 30 30 round 盒阴影: box-shadow: 10px 10px 5px #888888 媒体查询: 定义两套css, 当浏览器的尺寸变化时会采用不同的属性 10 请解释一下CSS3的flexbox(弹性盒布局模型), 以及适用场景?

该布局模型的目的是提供一种更加高效的方式来对容器中的条目进行布局、对齐和分配空间。在传统的布局方式中, block 布局是把块在垂直方向从上到下依次排列的; 而 inline 布局则是在水平方向来排列。弹性盒布局并没有这样内在的方向限制, 可以由开发人员自由操作。 试用场景: 弹性布局适合于移动前端开发, 在Android和ios上也完美支持。

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。

15 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:

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。 如果都找不到, 则为 initial containing block。

补充:

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) absolute: 向上找最近的定位为absolute/relative的元素 fixed: 它的containing block一律为根元素(html/body) 16 CSS里的visibility属性有个collapse属性值? 在不同浏览器下以后什么区别?

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display: none没有什么区别。 17 display:none与visibility: hidden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)

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

display属性规定元素应该生成的框的类型; position属性规定元素的定位类型; float属性是一种布局方式, 定义元素在哪个方向浮动。 类似于优先级机制: position: absolute/fixed优先级最高, 有他们在时, float不起作用, display值需要调整。 float 或者absolute定位的元素, 只能是块元素或表格。

19 对BFC规范(块级格式化上下文: block formatting context)的理解?

BFC规定了内部的Block Box如何布局。 定位方案:

内部的Box会在垂直方向上一个接一个放置。 Box垂直方向的距离由margin决定, 属于同一个BFC的两个相邻Box的margin会发生重叠。 每个元素的margin box 的左边, 与包含块border box的左边相接触。 BFC的区域不会与float box重叠。 BFC是页面上的一个隔离的独立容器, 容器里面的子元素不会影响到外面的元素。 计算BFC的高度时, 浮动元素也会参与计算。 满足下列条件之一就可触发BFC

根元素，即html float的值不为none（默认）overflow的值不为visible（默认）display的值为inline-block、table-cell、table-caption position的值为absolute或fixed 20 为什么会出现浮动和什么时候需要清除浮动？清除浮动的方式？

浮动元素碰到包含它的边框或者浮动元素的边框停留。由于浮动元素不在文档流中，所以文档流的块框表现得就像浮动框不存在一样。浮动元素会漂浮在文档流的块框上。浮动带来的问题：

父元素的高度无法被撑开，影响与父元素同级的元素 与浮动元素同级的非浮动元素（内联元素）会跟随其后 若非第一个元素浮动，则该元素之前的元素也需要浮动，否则会影响页面显示的结构。清除浮动的方式：

父级div定义height 最后一个浮动元素后加空div标签 并添加样式clear:both。 包含浮动元素的父标签添加样式overflow为hidden或auto。 父级div定义zoom

22设置元素浮动后，该元素的display值是多少？

自动变成display:block

23 移动端的布局用过媒体查询吗？

通过媒体查询可以为不同大小和尺寸的媒体定义不同的css，适应相应的设备的显示。

里边 CSS : @media only screen and (max-device-width:480px) {/css样式/} 24 使用 CSS 预处理器吗？ Less sass

25 CSS优化、提高性能的方法有哪些？

避免过度约束 避免后代选择符 避免链式选择符 使用紧凑的语法 避免不必要的命名空间 避免不必要的重复 最好使用表示语义的名字。一个好的类名应该是描述他是什么而不是像什么 避免！important，可以选择其他选择器 尽可能的精简规则，你可以合并不同类里的重复规则 26 浏览器是怎样解析CSS选择器的？

CSS选择器的解析是从右向左解析的。若从左向右的匹配，发现不符合规则，需要进行回溯，会损失很多性能。若从右向左匹配，先找到所有的最右节点，对于每一个节点，向上寻找其父节点直到找到根元素或满足条件的匹配规则，则结束这个分支的遍历。两种匹配规则的性能差别很大，是因为从右向左的匹配在第一步就筛选掉了大量的不符合条件的最右节点（叶子节点），而从左向右的匹配规则的性能都浪费在了失败的查找上面。而在 CSS 解析完毕后，需要将解析的结果与 DOM Tree 的内容一起进行分析建立一棵 Render Tree，最终用来进行绘图。在建立 Render Tree 时（WebKit 中的「Attachment」过程），浏览器就要为每个 DOM Tree 中的元素根据 CSS 的解析结果（Style Rules）来确定生成怎样的 Render Tree。

27 在网页中的应该使用奇数还是偶数的字体？为什么呢？

使用偶数字体。偶数字号相对更容易和 web 设计的其他部分构成比例关系。Windows 自带的点阵宋体（中易宋体）从 Vista 开始只提供 12、14、16 px 这三个大小的点阵，而 13、15、17 px 时用的是小一号的点。（即每个字占的空间大了 1 px，但点阵没变），于是略显稀疏。

28 margin和padding分别适合什么场景使用？

何时使用margin：

需要在border外侧添加空白 空白处不需要背景色 上下相连的两个盒子之间的空白，需要相互抵消时。 何时使用padding：

需要在border内侧添加空白 空白处需要背景颜色 上下相连的两个盒子的空白，希望为两者之和。兼容性的问题：在IE5 IE6中，为float的盒子指定margin时，左侧的margin可能会变成两倍的宽度。通过改变padding或者指定盒子的display：inline解决。

29 元素竖向的百分比设定是相对于容器的高度吗？

当按百分比设定一个元素的宽度时，它是相对于父容器的宽度计算的，但是，对于一些表示竖向距离的属性，例如padding-top, padding-bottom, margin-top, margin-bottom等，当按百分比设定它们时，依据的也是父容器的宽度，而不是高度。

30 全屏滚动的原理是什么？用到了CSS的哪些属性？

原理：有点类似于轮播，整体的元素一直排列下去，假设有5个需要展示的全屏页面，那么高度是500%，只是展示100%，剩下的可以通过transform进行y轴定位，也可以通过margin-top实现 overflow: hidden; transition: all 1000ms ease; 31 什么是响应式设计？响应式设计的基本原理是什么？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。

1 32 视差滚动效果？

视差滚动 (Parallax Scrolling) 通过在网页向下滚动的时候，控制背景的移动速度比前景的移动速度慢来创建出令人惊叹的3D效果。

CSS3实现 优点：开发时间短、性能和开发效率比较好，缺点是不能兼容到低版本的浏览器 jQuery实现 通过控制不同层滚动速度，计算每一层的时间，控制滚动效果。 优点：能兼容到各个版本的，效果可控性好 缺点：开发起来对制作者要求高 插件实现方式 例如：parallax-scrolling，兼容性十分好 33 ::before 和 :after中双冒号和单冒号有什么区别？解释一下这2个伪元素的作用

单冒号(:)用于CSS3伪类，双冒号(::)用于CSS3伪元素。::before就是以子元素的存在，定义在元素主体内容之前的一个伪元素。并不存在于dom之中，只存在于页面之中。:before 和 :after 这两个伪元素，是在CSS2.1里新出现的。起初，伪元素的前缀使用的是单冒号语法，但随着Web的进化，在CSS3的规范里，伪元素的语法被修改成使用双冒号，成为::before ::after

34 你对line-height是如何理解的？

行高是指一行文字的高度，具体说是两行文字间基线的距离。CSS中起高度作用的是height和line-height，没有定义height属性，最终其表现作用一定是line-height。单行文本垂直居中：把line-height值设置为height一样大小的值可以实现单行文字的垂直居中，其实也可以把height删除。多行文本垂直居中：需要设置display属性为inline-block。

35 怎么让Chrome支持小于12px 的文字？

1 p{font-size:10px;-webkit-transform:scale(0.8);} //0.8是缩放比例 36 让页面里的字体变清晰，变细用CSS怎么做？

-webkit-font-smoothing在window系统下没有起作用，但是在IOS设备上起作用-webkit-font-smoothing: antialiased是最佳的，灰度平滑。

37 position:fixed;在android下无效怎么处理？

1 38 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 $1/60 * 1000ms = 16.7ms$ 。

39 li与li之间有看不见的空白间隔是什么原因引起的？有什么解决办法？

行框的排列会受到中间空白（回车空格）等的影响，因为空格也属于字符，这些空白也会被应用样式，占据空间，所以会有间隔，把字符大小设为0，就没有空格了。解决方法：

可以将

- 代码全部写在一排
- 浮动li中float: left

在ul中用font-size : 0 (谷歌不支持) ; 可以使用letter-space : -3px
40 display:inline-block 什么时候会显示间隙 ?

有空格时候会有间隙 解决 : 移除空格 margin正值的时候 解决 : margin使用负值 使用font-size时候 解决 : font-size:0、letter-spacing、word-spacing 41 有一个高度自适应的div , 里面有两个div , 一个高度100px , 希望另一个填满剩下的高度

外层div使用position : relative ; 高度要求自适应的div使用position: absolute; top: 100px; bottom: 0; left: 0

42 png、jpg、gif 这些图片格式解释一下 , 分别什么时候用。有没有了解过webp ?

png是便携式网络图片 (Portable Network Graphics) 是一种无损数据压缩位图文件格式.优点是 : 压缩比高 , 色彩好。大多数地方都可以用。jpg是一种针对相片使用的一种失真压缩方法 , 是一种破坏性的压缩 , 在色调及颜色平滑变化做的不错。在www上 , 被用来储存和传输照片的格式。gif是一种位图文件格式 , 以8位色重现真色彩的图像。可以实现动画效果。webp格式是谷歌在2010年推出的图片格式 , 压缩率只有jpg的2/3 , 大小比png小了45%。缺点是压缩的时间更久了 , 兼容性不好 , 目前谷歌和opera支持。 43 style标签写在body后与body前有什么区别 ?

页面加载自上而下 当然是先加载样式。 写在body标签后由于浏览器以逐行方式对HTML文档进行解析 , 当解析到写在尾部的样式表 (外联或写在style标签) 会导致浏览器停止之前的渲染 , 等待加载且解析样式表完成之后重新渲染 , 在windows的IE下可能会出现FOUC现象 (即样式失效导致的页面闪烁问题)

44 CSS属性overflow属性定义溢出元素内容区的内容会如何处理?

参数是scroll时候 , 必会出现滚动条。 参数是auto时候 , 子元素内容大于父元素时出现滚动条。 参数是visible时候 , 溢出的内容出现在父元素之外。 参数是hidden时候 , 溢出隐藏。

45 阐述一下CSS Sprites

将一个页面涉及到的所有图片都包含到一张大图中去 , 然后利用CSS的 background-image , background-repeat , background-position 的组合进行背景定位。利用CSS Sprites能很好地减少网页的http请求 , 从而大大的提高页面的性能 ; CSS Sprites能减少图片的字节。
