CSS

**1、介绍一下标准的CSS的盒子模型？低版本IE的盒子模型有什么不同的？**

（1）有两种， IE 盒子模型、W3C 盒子模型；

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

（3）区  别： IE的content部分把 border 和 padding计算了进去;

**2、CSS选择符有哪些？哪些属性可以继承？**

（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）

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

\*   不可继承的样式：borderpadding margin width height ;

**3、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:

同权重: 内联样式表（标签内部）> 嵌入样式表（当前文件中）> 外部样式表（外部文件中）。

!important >  id > class > tag

 important 比 内联优先级高

**4、CSS3新增伪类有那些？**

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。

   :after          在元素之前添加内容,也可以用来做清除浮动。

   :before         在元素之后添加内容

   :enabled

   :disabled       控制表单控件的禁用状态。

   :checked        单选框或复选框被选中。

**5、如何居中div？**

水平居中：给div设置一个宽度，然后添加margin:0 auto属性

div{

   width:200px;

   margin:0 auto;

 }

让绝对定位的div居中

div {

   position: absolute;

 width: 300px;

height: 300px;

 margin: auto;

 top:0;

left: 0;

bottom: 0;

 right: 0;

/\* 方便看效果 \*/

}

水平垂直居中一

确定容器的宽高宽500 高 300 的层

设置层的外边距

div {

   position: relative;     /\* 相对定位或绝对定位均可\*/

   width:500px;

   height:300px;

   top: 50%;

   left: 50%;

   margin: -150px 0 0 -250px;      /\*外边距为自身宽高的一半 \*/

        /\* 方便看效果 \*/

 }

水平垂直居中二

未知容器的宽高，利用 `transform` 属性

div {

   position: absolute;     /\* 相对定位或绝对定位均可\*/

   width:500px;

   height:300px;

   top: 50%;

   left: 50%;

   transform: translate(-50%, -50%);

        /\* 方便看效果 \*/

}

水平垂直居中三

利用 flex 布局

实际使用时应考虑兼容性

.container {

   display: flex;

   align-items: center;        /\* 垂直居中 \*/

   justify-content: center;    /\* 水平居中 \*/

}

.container div {

   width: 100px;

   height: 100px;

        /\* 方便看效果 \*/

}

**6、display有哪些值？说明他们的作用。**

 block        块类型。默认宽度为父元素宽度，可设置宽高，换行显示。

 none          缺省值。象行内元素类型一样显示。

 inline        行内元素类型。默认宽度为内容宽度，不可设置宽高，同行显示。

 inline-block  默认宽度为内容宽度，可以设置宽高，同行显示。

 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。

**7、position的值relative和absolute定位原点是？**

 absolute

    生成绝对定位的元素，相对于值不为 static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。

**8、CSS3有哪些新特性？**

  新增各种CSS选择器  （: not(.input)：所有 class 不是“input”的节点）

  圆角            （border-radius:8px）

  多列布局      （multi-column layout）

  阴影和反射   （Shadow\Reflect）

  文字特效      （text-shadow、）

  文字渲染      （Text-decoration）

  线性渐变      （gradient）

  旋转            （transform）

  缩放,定位,倾斜,动画,多背景

例如:

transform:\scale(0.85,0.90)\

translate(0px,-30px)\

skew(-9deg,0deg)\Animation:

**9、请解释一下CSS3的Flexbox（弹性盒布局模型）,以及适用场景？**

 一个用于页面布局的全新CSS3功能，Flexbox可以把列表放在同一个方向（从上到下排列，从左到右），并让列表能延伸到占用可用的空间。

 较为复杂的布局还可以通过嵌套一个伸缩容器（flex container）来实现。

 采用Flex布局的元素，称为Flex容器（flexcontainer），简称"容器"。

 它的所有子元素自动成为容器成员，称为Flex项目（flex item），简称"项目"。

 常规布局是基于块和内联流方向，而Flex布局是基于flex-flow流可以很方便的用来做局中，能对不同屏幕大小自适应。

 在布局上有了比以前更加灵活的空间。

**10、用纯CSS创建一个三角形的原理是什么？**

把上、左、右三条边隐藏掉（颜色设为 transparent）

#demo {

 width: 0;

 height: 0;

 border-width: 20px;

 border-style: solid;

 border-color: transparent transparent red transparent;

}

**11、一个满屏 品 字布局 如何设计?**

简单的方式：

    上面的div宽100%，

    下面的两个div分别宽50%，

    然后用float或者inline使其不换行即可

**12、css多列等高如何实现？**

利用padding-bottom|margin-bottom正负值相抵；

设置父容器设置超出隐藏（overflow:hidden），这样子父容器的高度就还是它里面的列没有设定padding-bottom时的高度，

当它里面的任一列高度增加了，则父容器的高度被撑到里面最高那列的高度，

其他比这列矮的列会用它们的padding-bottom补偿这部分高度差。

**13、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？**

\* png24位的图片在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{

         /\*所有识别\*/

          /\*IE6、7、8识别\*/

          +/\*IE6、7识别\*/

         \_/\*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。

\*  解决方法：（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 {}

**14、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。

**15、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。

- 当然，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。

最简单的初始化方法： \* {padding: 0; margin: 0;} （强烈不建议）

淘宝的样式初始化代码：

body, h1, h2, h3, h4, h5, h6, hr, p,blockquote, dl, dt, dd, ul, ol, li, pre, form, fieldset, legend, button, input,textarea, th, td { margin:0; padding:0; }

body, button, input, select, textarea {font:12px/1.5tahoma, arial, \5b8b\4f53; }

h1, h2, h3, h4, h5, h6{ font-size:100%; }

address, cite, dfn, em, var {font-style:normal; }

code, kbd, pre, samp {font-family:couriernew, courier, monospace; }

small{ font-size:12px; }

ul, ol { list-style:none; }

a { text-decoration:none; }

a:hover { text-decoration:underline; }

sup { vertical-align:text-top; }

sub{ vertical-align:text-bottom; }

legend { color:#000; }

fieldset, img { border:0; }

button, input, select, textarea {font-size:100%; }

table { border-collapse:collapse;border-spacing:0; }

**16、absolute的containingblock(容器块)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

**17、对BFC规范(块级格式化上下文：blockformatting context)的理解？**

（W3C CSS 2.1 规范中的一个概念,它是一个独立容器，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。）

 一个页面是由很多个 Box组成的,元素的类型和 display 属性,决定了这个 Box 的类型。

 不同类型的 Box,会参与不同的Formatting Context（决定如何渲染文档的容器）,因此Box内的元素会以不同的方式渲染,也就是说BFC内部的元素和外部的元素不会互相影响。

**18、css定义的权重**

以下是权重的规则：标签的权重为1，class的权重为10，id的权重为100，以下例子是演示各种定义的权重值：

/\*权重为1\*/

div{

}

/\*权重为10\*/

.class1{

}

/\*权重为100\*/

#id1{

}

/\*权重为100+1=101\*/

#id1 div{

}

/\*权重为10+1=11\*/

.class1 div{

}

/\*权重为10+10+1=21\*/

.class1 .class2 div{

}

如果权重相同，则最后定义的样式会起作用，但是应该避免这种情况出现

**19、请解释一下为什么需要清除浮动？清除浮动的方式**

清除浮动是为了清除使用浮动元素产生的影响。浮动的元素，高度会塌陷，而高度的塌陷使我们页面后面的布局不能正常显示。

（1）父级div定义height；

（2）父级div 也一起浮动；

（3）常规的使用一个class；

   .clearfix:before, .clearfix:after {

        content: " ";

       display: table;

    }

   .clearfix:after {

       clear: both;

    }

   .clearfix {

       \*zoom: 1;

    }

（4）SASS编译的时候，浮动元素的父级div定义伪类:after

   &:after,&:before{

       content: " ";

       visibility: hidden;

       display: block;

       height: 0;

       clear: both;

    }

解析原理：

1) display:block 使生成的元素以块级元素显示,占满剩余空间;

2) height:0 避免生成内容破坏原有布局的高度。

3) visibility:hidden 使生成的内容不可见，并允许可能被生成内容盖住的内容可以进行点击和交互;

4）通过content:"."生成内容作为最后一个元素，至于content里面是点还是其他都是可以的，例如oocss里面就有经典的 content:".",有些版本可能content 里面内容为空,一丝冰凉是不推荐这样做的,firefox直到7.0content:”" 仍然会产生额外的空隙；

5）zoom：1 触发IEhasLayout。

通过分析发现，除了clear：both用来闭合浮动的，其他代码无非都是为了隐藏掉content生成的内容，这也就是其他版本的闭合浮动为什么会有font-size：0，line-height：0。

**20、什么是外边距合并？**

外边距合并指的是，当两个垂直外边距相遇时，它们将形成一个外边距。

合并后的外边距的高度等于两个发生合并的外边距的高度中的较大者。

**21、zoom:1的清除浮动原理?**

清除浮动，触发hasLayout；

Zoom属性是IE浏览器的专有属性，它可以设置或检索对象的缩放比例。解决ie下比较奇葩的bug。

譬如外边距（margin）的重叠，浮动清除，触发ie的haslayout属性等。

来龙去脉大概如下：

当设置了zoom的值之后，所设置的元素就会就会扩大或者缩小，高度宽度就会重新计算了，这里一旦改变zoom值时其实也会发生重新渲染，运用这个原理，也就解决了ie下子元素浮动时候父元素不随着自动扩大的问题。

Zoom属是IE浏览器的专有属性，火狐和老版本的webkit核心的浏览器都不支持这个属性。然而，zoom现在已经被逐步标准化，出现在 CSS 3.0 规范草案中。

目前非ie由于不支持这个属性，它们又是通过什么属性来实现元素的缩放呢？

可以通过css3里面的动画属性scale进行缩放。

**22、移动端的布局用过媒体查询吗？**

假设你现在正用一台显示设备来阅读这篇文章，同时你也想把它投影到屏幕上，或者打印出来，而显示设备、屏幕投影和打印等这些媒介都有自己的特点，CSS就是为文档提供在不同媒介上展示的适配方法

当媒体查询为真时，相关的样式表或样式规则会按照正常的级联规被应用。当媒体查询返回假，标签上带有媒体查询的样式表仍将被下载 （只不过不会被应用）。

包含了一个媒体类型和至少一个使用宽度、高度和颜色等媒体属性来限制样式表范围的表达式。 CSS3加入的媒体查询使得无需修改内容便可以使样式应用于某些特定的设备范围。

@media (min-width: 700px) and (orientation:landscape){ .sidebar { display: none; } }

**23、使用 CSS 预处理器吗？喜欢那个？**

SASS (SASS、LESS没有本质区别，只因为团队前端都是用的SASS)

**24、CSS优化、提高性能的方法有哪些？**

关键选择器（key selector）。选择器的最后面的部分为关键选择器（即用来匹配目标元素的部分）；

如果规则拥有 ID 选择器作为其关键选择器，则不要为规则增加标签。过滤掉无关的规则（这样样式系统就不会浪费时间去匹配它们了）；

提取项目的通用公有样式，增强可复用性，按模块编写组件；增强项目的协同开发性、可维护性和可扩展性;

使用预处理工具或构建工具（gulp对css进行语法检查、自动补前缀、打包压缩、自动优雅降级）；

**25、浏览器是怎样解析CSS选择器的？**

样式系统从关键选择器开始匹配，然后左移查找规则选择器的祖先元素。

只要选择器的子树一直在工作，样式系统就会持续左移，直到和规则匹配，或者是因为不匹配而放弃该规则。

**­­**

**26、­margin和padding分别适合什么场景使用？**

margin是用来隔开元素与元素的间距；padding是用来隔开元素与内容的间隔。

margin用于布局分开元素使元素与元素互不相干；

padding用于元素与内容之间的间隔，让内容（文字）与（包裹）元素之间有一段

**27、::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。不过浏览器需要同时支持旧的已经存在的伪元素写法，

比如:first-line、:first-letter、:before、:after等，

而新的在CSS3中引入的伪元素则不允许再支持旧的单冒号的写法。

想让插入的内容出现在其它内容前，使用::before，否者，使用::after；

在代码顺序上，::after生成的内容也比::before生成的内容靠后。

如果按堆栈视角，::after生成的内容会在::before生成的内容之上

**28、如何修改chrome记住密码后自动填充表单的黄色背景 ？**

input:-webkit-autofill,textarea:-webkit-autofill, select:-webkit-autofill {

  /\* #FAFFBD; \*/

 background-image: none;

 color: rgb(0, 0, 0);

}

**29、设置元素浮动后，该元素的display值是多少？**

自动变成了 display:block

**30、怎么让Chrome支持小于12px 的文字？**

（1）用图片：如果是内容固定不变情况下，使用将小于12px文字内容切出做图片，这样不影响兼容也不影响美观。

（2）使用12px及12px以上字体大小：为了兼容各大主流浏览器，建议设计美工图时候设置大于或等于12px的字体大小，如果是接单的这个时候就需要给客户讲解小于12px浏览器不兼容等事宜。

（3）继续使用小于12px字体大小样式设置：如果不考虑chrome可以不用考虑兼容，同时在设置小于12px对象设置-webkit-text-size-adjust:none，做到最大兼容考虑。

（4）使用12px以上字体：为了兼容、为了代码更简单 从新考虑权重下兼容性。

**31、让页面里的字体变清晰，变细用CSS怎么做？**

-webkit-font-smoothing: antialiased;

**32、font-style属性可以让它赋值为“oblique” oblique是什么意思？**

倾斜的字体样式

**33、position:fixed;在android下无效怎么处理？**

fixed的元素是相对整个页面固定位置的，你在屏幕上滑动只是在移动这个所谓的viewport，

原来的网页还好好的在那，fixed的内容也没有变过位置，

所以说并不是iOS不支持fixed，只是fixed的元素不是相对手机屏幕固定的。

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

**34、如果需要手动写动画，你认为最小时间间隔是多久，为什么？（阿里）**

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

**35、display:inline-block什么时候会显示间隙？(携程)**

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

**36、什么是Cookie 隔离？（或者说：请求资源的时候不要让它带cookie怎么做）**

如果静态文件都放在主域名下，那静态文件请求的时候都带有的cookie的数据提交给server的，非常浪费流量，

所以不如隔离开。

因为cookie有域的限制，因此不能跨域提交请求，故使用非主要域名的时候，请求头中就不会带有cookie数据，

这样可以降低请求头的大小，降低请求时间，从而达到降低整体请求延时的目的。

同时这种方式不会将cookie传入Web Server，也减少了Web Server对cookie的处理分析环节，

提高了webserver的http请求的解析速度。

**37、什么是CSS 预处理器 / 后处理器？**

- 预处理器例如：LESS、Sass、Stylus，用来预编译Sass或less，增强了css代码的复用性，

  还有层级、mixin、变量、循环、函数等，具有很方便的UI组件模块化开发能力，极大的提高工作效率。

- 后处理器例如：PostCSS，通常被视为在完成的样式表中根据CSS规范处理CSS，让其更有效；目前最常做的

  是给CSS属性添加浏览器私有前缀，实现跨浏览器兼容性的问题。