**周考一专业题答案：**

1、px表示像素，%表示占整个页面宽度的百分比。

2,、DIV默认情况下是分行显示，SPAN在同行显示。

3、1）CSS+DIV，这个网页设计模式中，DIV承担了网页的内容，CSS承担了网页的样式。这样就使网页的内容和样式的分离开来。有利于页面的维护升级。

2）有助于提高搜索引擎亲和力（快速找到需要的数据，而不是像在TABLE中一层层的查找）

3）有助于页面的重构(换皮肤如blog，直接套用另外一套样式就可以实现，而不用改动网页脚本。)

4、 .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\*/ } （此题答案合理即可）

5、#content p{color:blue}

html>body #content p {color:red}

\* html #content p{color:green}

html>body #content p {\*color:black;}

6、1）position的五种值分别是static, relative, absolute, fixed, inherit

2）position的默认值是static

1. 被点击访问过的超链接样式不在具有hover和active了,很多人应该都遇到过这个问题，解决方法是改变CSS属性的排列顺序: L-V-H-A
2. 例如:

<style type="text/css">

body {margin:0}

div { float:left; margin-left:10px; width:200px; height:200px; border:1px solid red }

</style>浮动后本来外边距10px,但IE解释为20px,解决办法是加上display:inline

1. 标准浏览器中固定高度值的容器是不会象IE6里那样被撑开的,那我又想固定高度，又想能被撑开需要怎样设置呢？办法就是去掉he ight设置min-height:200px; 这里为了照顾不认识min-height的IE6 可以这样定义：

div { height:auto!important; height:200px; min-height:200px; }

1. 原来样式设置：

<style type=text/css>

body { scrollbar-face-color:#f6f6f6; scrollbar-highlight-color:#fff; scrollbar-shadow-color:#eeeeee; scrollbar-3dlight-color:#eeeeee; scrollbar-arrow-color:#000; scrollbar-track-color:#fff; scrollbar-darkshadow-color:#fff; }

</style>

解决办法是将body换成 html

ps: 当时被要求要重新定义滚动条的时候，我查到了两个方法，一种如上，一种是js仿制滚动条，我应用的是上面的这种，但是发现不兼容IE，后来不了了之了，再后来偶然间才发现了这个解决办法。

11、IE6下这个问题是因为默认的行高造成的，解决的方法也有很多，例如：overflow:hidden | zoom:0.08 | line-height:1px

12、解决的办法是给FLASH设置透明属性：

<param name="wmode" value="transparent" />或者<param name="wmode" value="opaque" />

13、就是使用百分比绝对定位，与外补丁负值的方法，负值的大小为其自身宽度高度除以二

14、答：假定有如下情况：

<div id="a">

    <div id="b"> </div>

</div>

如果要实现b在a中居中放置，一般只需用CSS设置a的text-align属性为center。这样的方法在IE里看起来一切正常；但是在Firefox中b却会是居左的。

解决办法就是设置b的横向margin为auto。例如设置b的CSS样式为：margin: 0 auto;。

15、不支持

16、 8.0版本 IE全称是Internet Explorer

17、CSS中每一条样式结束用分号 样式名与样式值之间用冒号

18、1） .（句号）后面跟一个名称表示文档中所有class属性值包含这个名称的应用其样式，

2） #（井号）后面跟个名称表示文档中ID为此名称的元素应用其样式。

3） CSS样式写在style属性内。

19、CSS就是一种叫做样式表（stylesheet）的技术。也有的人称之为层叠样式表（Cascading Stylesheet）

在主页制作时采用CSS技术，可以有效地对页面的布局、字体、颜色、背景和其它效果实现更加精确的控制。

只要对相应的代码做一些简单的修改，就可以改变同一页面的不同部分，或者页数不同的网页的外观和格式。

它的作用可以达到：

（1）在几乎所有的浏览器上都可以使用。

（2）以前一些非得通过图片转换实现的功能，现在只要用CSS就可以轻松实现，从而更快地下载页面。

（3）使页面的字体变得更漂亮，更容易编排，使页面真正赏心悦目。

（4）你可以轻松地控制页面的布局 。

（5）你可以将许多网页的风格格式同时更新，不用再一页一页地更新了。你可以将站点上所有的网页风格都使用一个CSS文件进行控制，只要修改这个CSS文件中相应的行，那么整个站点的所有页面都会随之发生变动。

想一想，没有使用CSS前我们是如何控制字体的颜色和大小以及所使用的字体的？我们一般使用HTML标签来实现，代码非常烦琐。

很难想象，如果在一个页面里需要频繁地更替字体的颜色大小，最终生成的HTML代码的长度一定臃肿不堪。

说实话，CSS就是为了简化这样的工作诞生的，当然其功能决非这么简单。

CSS是通过对页面结构的风格控制的思想，控制整个页面的风格的。

式样单放在页面中，通过浏览器的解释执行，是完全的文本，任何懂得HTML的人都可以掌握，非常的容易。生殖对一些非常老的浏览器，页不会产生页面混乱的现象。

1. CSS(Cascading Style Sheets)也叫层叠样式表，它是用来控制网页中一个或多个元素的外观。在网页制作中，使用样式表可以减少很多具有相同外观元素的重复设置工作，从而给网页的设计和维护带来了很大的方便;具体应用是：改变样式表中元素显示样式的定义，则在使用该样式表的所有网页中相应元素的显示样式都会自动发生改变。
2. 声明位于文档中的最前面的位置，处于 标签之前。此标签可告知浏览器文档使用哪种 HTML 或 XHTML 规范。

标准模式和混杂模式（quirks mode）。在标准模式中，浏览器根据规范呈现页面；在混杂模式中，页面以一种比较宽松的向后兼容的方式显示。混杂模式通常模拟老式浏览器（比如Microsoft IE 4和Netscape Navigator 4）的行为以防止老站点无法工作。

在IE 6出现时，在标准模式中使用的是正确的盒模型，在混杂模式中使用的则是老式的专有盒模型。为了维持对IE 5和更低版本的向后兼容性，Opera 7和更高版本也在混杂模式中使用有缺点的IE盒模型。

周考二专业题答案：

1. 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

2、 block 象块类型元素一样显示。

none 缺省值。象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。

\*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。

3、答案：1.使用空标签清除浮动。

这种方法是在所有浮动标签后面添加一个空标签 定义css clear:both. 弊端就是增加了无意义标签。

2.使用overflow。

给包含浮动元素的父标签添加css属性 overflow:auto; zoom:1; zoom:1用于兼容IE6。

1. 原理：任何元素 element 都可以被浮动。段落、div、list、tables,以及图像都可以被浮动，事实上即使是像 span 和 strong 这样的行内置元素也可以很好地进行浮动。任何申明为 float 的元素自动被设置为一个"块级元素", 这表示它可以具有申明的"width"和"height"属性。事实上，floats当前被要求具有一个申明的宽度，但这不是现代浏览器制造者的思 路，W3C以及开始同意这样的作法。现在大多数人的意见是没有指定宽度的float应当伸缩包装到浮动内容的宽度。因此，内部带有图片的一个float将 和图片一样宽，带有文本的一个浮动将与该浮动内的最长文本行一样宽。

CSS2.1的浮动规则中这样讲： "如果 width 是以 auto 方式计算得到，使用的值是 shrink-tofit 伸缩到适合的宽度。" 所有现代的浏览器已经这样做，除了IE5/Mac。除非浮动具有某种指定的宽度，否则该浏览器会变得错误百出。现在大多数聪明的编码人员会让浏览器伸缩其 没有宽度属性的floats， 并使用一个隐藏的专门给IE5/Mac的宽度知识。这可能不算很巧妙，但是它是现在不能满足该规范的极少数的浏览器，同时Mac用户有几种更好的选择。 MS停止了对IE5/Mac的支持，因此寻求像伸缩包装式浮动这样有用特性被避免的代价太高了，这样只有IE5/Mac用户不会看到错误的浮动。

使用after伪对象清除浮动。

该方法只适用于非IE浏览器。具体写法可参照以下示例。使用中需注意以下几点。一、该方法中必须为需要清除浮动元素的伪对象中设置 height:0，否则该元素会比实际高出若干像素；二、content属性是必须的，但其值可以为空，蓝色理想讨论该方法的时候content属性的值 设为”.”，但我发现为空亦是可以的。

<style type=”text/css”>

<!–

\*{margin:0;padding:0;}

body{font:36px bold; color:#F00; text-align:center;}

#layout{background:#FF9;}

#layout:after{display:block;clear:both;content:”";visibility:hidden;height:0;} #left{float:left;width:20%;height:200px;background:#DDD;line-height:200px;} #right{float:right;width:30%;height:80px;background:#DDD;line-height:80px;}

–>

</style>

<div id=”layout”>

<div id=”left”>Left</div>

<div id=”right”>Right</div>

</div>

此三种方法各有利弊，使用时应择优选择，比较之下第二种方法更为可取

1. 使用过，示例略
2. js继承有5种实现方式：1、继承第一种方式：对象冒充；2、继承第二种方式：call()方法方式；3、继承的第三种方式：apply()方法方式；4、继承的第四种方式：原型链方式，即子类通过prototype将所有在父类中通过prototype追加的属性和方法都追加到Child，从而实现了继承；5、继承的第五种方式：混合方式 混合了call方式、原型链方式（详见：<http://blog.csdn.net/james521314/article/details/8645815）>
3. Header footer nav article section hgroup
4. 1、box-shadow； 2、text-overflow；3、border-colors；4、text-shadow；5、border-sizing；6、border-radius
5. 略
6. 使用Trident内核的浏览器：IE、Maxthon、TT、The World等；

使用Gecko内核的浏览器：Netcape6及以上版本、FireFox、MozillaSuite/SeaMonkey；

使用Presto内核的浏览器：Opera7及以上版本；

使用Webkit内核的浏览器：Safari、Chrome。

1. 1. \* 符号 IE 浏览器能识别 \* 符号，但其他浏览器诸如 Firefox、Opera、Chrome 等不能识别 \* 符号。 例：在 Firefox 和 IE 中呈现不同的文字颜色：color:red;\*color:blue;

//在 Firefox 等非 IE 核心浏览器中，文字呈现红色；而 IE 中呈现蓝色。

2. !important IE7 不但能识别 \* 符号，还能识别 !important，而 IE6 只能识别前者。

例：在 IE6 和 IE7 中呈现不同的文字颜色：color:red !important;color:blue;

//在 IE7 浏览器中，文字呈现红色；而 IE6 中呈现蓝色。

3.综合 1 和 2，利用上述浏览器特性，可在 CSS 中判别 Firefox，IE7，IE6 并加载不同样式。

例：在 Firefox，IE7，IE6 中呈现三种不同文字颜色：color:blue;\*color:red !important;\*color:green;

//在 Firefox 中，文字呈现蓝色，在 IE7 浏览器中，呈现红色；而 IE6 中呈现绿色。

4. \*html 和 \*+html

IE 核心的浏览器能识别\*html 和\*+html，而 Firefox 等非 IE 核心浏览器不能识别。

1. 略
2. 详见：<http://www.cnblogs.com/fighting_cp/archive/2010/09/20/1831844.html>
3. 详见：<http://blog.sina.com.cn/s/blog_c3068a9901019aqx.html>
4. 简单数值类型: 有Undefined, Null, Boolean, Number和String

复杂型：Object。

1. 略
2. ul li{list-style:none}
3. Color:#f83;

**18、一、使用STYLE属性： 将STYLE属性直接加在个别的元件标签里。**

<元件(标签) STYLE="性质(属性)1: 设定值1; 性质(属性)2: 设定值2; ...}例如：

<TD STYLE="COLOR:BLUE; font-size:9pt; font-family:"标楷体";[line](http://mydown.yesky.com/soft/375/34565875.shtml" \t "http://homepage.yesky.com/436/_blank)-height:150%>  
这种用法的优点 是可灵巧[应用](http://search.yesky.com/search.do?wd=%D3%A6%D3%C3" \t "http://homepage.yesky.com/436/_blank)样式於各标签中，但是缺点则是没有整篇文件的『统一性』。  
  
**二、使用STYLE标签： 将样式规则写在<STYLE>...</STYLE>标签之中。**

<STYLE TYPE="text/css">  
<!--  
样式规则表  
-->   
</STYLE>  
例如：   
<STYLE TYPE="text/css">  
<!--  
BODY {  
　　color: BLUE;  
　　background: #FFFFCC;  
　　font-size: 9pt}  
TD, P {  
　　COLOR: GREEN;  
　　font-size: 9pt}  
-->  
</STYLE>

通常是将整个的 <STYLE>...</STYLE>结构写在网页的<HEAD> </HEAD>部份之中。这种用法的优点就是在於整篇文件的统一性，只要是有声明的的元件即会套用该样式规则。缺点就是在个别元件的灵活度不足。  
  
**三、使用 LINK标签： 将样式规则写在.css的样式档案中，再以<LINK>标签引入。**

假设我们把样式规则存成一个example.css的档案，我们只要在网页中加入  
<LINK REL=STYLESHEET TYPE="text/css" HREF="example.css">  
即可套用该样式档案中所制定好的样式了。 通常是将LINK标签写在网页的<head></head>部份之中。这种用法的优点就是在於可以把要套用相同样式规则的数篇文件都指定到同一个样式档案即可。缺点也是在个别文件或元件的灵活度不足。

**四、使用@import引入： 跟LINK用法很像，但必　放在<STYLE>...</STYLE> 中。**   
<STYLE TYPE="text/css">  
<!--  
　　@import url(引入的样式表的位址、路径与档名);  
-->  
</STYLE>   
例如：   
<STYLE TYPE="text/css">  
<!--  
　　@import url(http://yourweb/ example.css);  
-->  
</STYLE>  
要注意的是，行末的分号是绝对不可少的！

19、详见：http://www.yes0531.com/oyxw/View\_192.html

20、IE6下这个问题是因为默认行高造成的。解决办法：overflow:hidden;line-height:1px

周考三专业题答案：

1. 答：定义包含文字的容器为：width:xxx;white-space:nowrap;
2. 定义容器为：overflow:hidden;width:xxx;white-space:nowrap;text-overflow:ellipsis;
3. <inputidinputid="test"type="checkbox"value="on"/><labelforlabelfor="test">测试</label>

4、这种情况浏览器会自动进行margin重叠，只显示较大的margin值

解决方案：只设置其中一个div的margin为15px

1. 设置li的vertical-align，值可以为top|text-top|middle|bottom|text-bottom
2. word-wrap:break-word;
3. 略

8、一. 移除超链接的虚线(仅对FF有效)下；二. 给行内元素定义宽度；三. 让固定宽度的页面居中；四. 图片替换技术；.五. 最小宽度

9、**一. 移除超链接的虚线(仅对FF有效)**下,当你点击一个超链接时会在外围出现一个虚线轮廓. 这很容易**解决**, 只需要在标签样式中加入 outline:none .a{outline: none;}**二. 给行内元素定义宽度**如果你给一个行内元素定义宽度,那么它只是在IE6下有效. 所有的元素要么是行内元素要么就好是块元素. 行内元素包括: <span>, <a>, <strong> 和 <em>. 块元素包括<div>, <p>, <h1>, <form>和<li> . 你不能定义行内元素的宽度, 为了**解决**这个问题你可以将行内元素转变为块元素.span { width: 150px; display: block }**三. 让固定宽度的页面居中**为了让页面在浏览器居中显示, 需要相对定位外层div, 然后把margin设置为auto.#wrapper {margin: auto;position: relative;}**四. 图片替换技术**用文字总比用图片做标题好一些. 文字对屏幕阅读机和都是非常友好的.HTML:<h1><span>Main heading one</span></h1>CSS:h1 { background: url(heading-image.gif) no-repeat; }h1 span {position:absolute;text-indent: -5000px;}你可以看到我们对标题使用了标准的<h1>作为标签并且用css来将文本替换为图片. text-indent属性将文字推到了浏览器左边5000px处, 这样对于浏览者来说就看不见了.关掉css,然后看看头部会是什么样子的.**五. 最小宽度**IE6另外一个bug就是它不支持 min-width 属性. min-width又是相当有用的, 特别是对于弹性模板来说, 它们有一个100%的宽度,min-width 可以告诉浏览器何时就不要再压缩宽度了.除IE6以外所有的浏览器你只需要一个 min-width: Xpx; 例如:.container {min-width:300px;}为了让他在IE6下工作, 我们需要一些额外的工作. 开始的时候我们需要创建两个div, 一个包含另一个:<div class="container"><div class="holder">Content</div></div>然后你需要定义外层div的min-width属性.container {min-width:300px;}这时该是IE hack大显身手的时候了. 你需要包含如下的代码:\* html .container {border-right: 300px solid #FFF;}\* html .holder {display: inline-block;position: relative;margin-right: -300px;}As the browser window is resized the outer div width reduces to suit until it shrinks to the border width, at which point it will not shrink any further. The holder div follows suit and also stops shrinking. The outer div border width becomes the minimum width of the inner div.**六. 隐藏水平滚动条**为了避免出现水平滚动条, 在body里加入 overflow-x:hidden .body { overflow-x: hidden; }当你决定使用一个比浏览器窗口大的图片或者flash时, 这个技巧将非常有用.

1. 第一个不占位，另一个是占位
2. var reg = /^[a-zA-Z][a-zA-Z\_0-9]{5,29}$/;

12、callee

返回正被执行的 Function 对象，也就是所指定的 Function 对象的正文。

[function.]arguments.callee

可选项 function 参数是当前正在执行的 Function 对象的名称。

说明

callee 属性的初始值就是正被执行的 Function 对象。

callee 属性是 arguments 对象的一个成员，它表示对函数对象本身的引用，这有利于匿名函数的递归或者保证函数的封装性，例如下边示例的递归计算1到n的自然数之和。而该属性仅当相关函数正在执行时才可用。还有需要注意的是callee拥有length属性，这个属性有时候用于验证还是比较好的。arguments.length是实参长度，arguments.callee.length是形参长度，由此可以判断调用时形参长度是否和实参长度一致。

示例

//callee可以打印其本身

function calleeDemo() {

alert(arguments.callee);

}

//用于验证参数

function calleeLengthDemo(arg1, arg2) {

if (arguments.length==arguments.callee.length) {

window.alert("验证形参和实参长度正确！");

return;

} else {

alert("实参长度：" +arguments.length);

alert("形参长度： " +arguments.callee.length);

}

}

//递归计算

var sum = function(n){

if (n < = 0)

return 1;

else

return n ＋arguments.callee(n - 1)

}

比较一般的递归函数：

var sum = function(n){

if (1==n) return 1;

else return n + sum (n-1);

调用时：alert(sum(100));

其中函数内部包含了对sum自身的引用，函数名仅仅是一个变量名，在函数内部调用sum即相当于调用

一个全局变量，不能很好的体现出是调用自身，这时使用callee会是一个比较好的方法

13、　同步：脚本会停留并等待服务器发送回复然后再继续

　 　异步：脚本允许页面继续其进程并处理可能的回复

　　跨域问题简单的理解就是因为JS同源策略的限制，a.com域名下的JS无法操作b.com或c.a.com下的对象，

　　PS：(1)如果是端口或者协议造成的跨域问题前端是无能为力的

　　(2) 在跨域问题上，域仅仅通过URL的首部来识别而不会尝试判断相同的IP地址对应的域或者两个域是否对应一个IP

　　前端对于跨域的解决办法：

　　(1) document.domain+iframe

　　(2) 动态创建script标签

14、　call()与apply()区别

　　call方法：

　　语法：call(thisObj, Object)

　　定义：调用一个对象的一个方法，以另一个对象替换当前对象。

　　说明：

　　call 方法可以用来代替另一个对象调用一个方法。call 方法可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。

　　如果没有提供 thisObj 参数，那么 Global 对象被用作 thisObj。

　　apply方法：

　　语法：apply(thisObj, [argArray])

　　定义：应用某一对象的一个方法，用另一个对象替换当前对象。

　　说明：

　　如果 argArray 不是一个有效的数组或者不是 arguments 对象，那么将导致一个 TypeError。

　　如果没有提供 argArray 和 thisObj 任何一个参数，那么 Global 对象将被用作 thisObj， 并且无法被传递任何参数。

1. <script type="text/javascript">

Array.prototype.delrepeat = function() {

var arr = this;

var \_arr = new Array();

for (var i in arr) {

if (i == 'delrepeat') continue;

if (\_arr.length == 0) \_arr.push(arr[i]);

for (var j = 0; j < \_arr.length; j++) {

if (arr[i] == \_arr[j]) {

break;

}

if (j > \_arr.length - 2) \_arr.push(arr[i]);

}

}

return \_arr;

}

alert([1, 1, 7, ,9,1, 2, 2, 3, 4, 5,9, 5].delrepeat());

</script>

16：略

17、（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。

18、使用 prototype 来实现。

19、document.all.cb1[0].disabled = true;

20、readonly 不可编辑，但可以选择和复制；值可以传递到后台

disabled 不能编辑，不能复制，不能选择；值不可以传递到后台

　周考四专业题答案：

此套专业题的相关答案请参考前面三周的相关对应题的答案