1、标签选择器：用HTML标签作为选择器

标签名{

属性： 值;

}

2、类选择器：为不同标签作出差异化，可多个标签用，ID选择器只能一次

.类名{

属性：值;

}

3、#ID名{

属性：值;

}

4、font-weight

默认为 normal（400）

number: 100 200 300 400 500 600 700 800 900

5、

复合属性书写方式：

选择器{

font: font-style font-weight font-size/line-height font-family;

}

标红为必须有的属性

e.g.

div{  
 font: italic bolder 20px/50px 'Microsoft Yahei';  
}

6、

text-indent：文本缩进；

1em：当前元素一个汉字的大小

7、

行间距：

行间距=上间距+文本高度+下间距

如果 行间距<=文本大小，则不会有任何行间加高效果

例如：行间距26px，文本大小16px，则上间距为5px，下间距为5px

8、后代选择器，子元素选择器：

后：标签间用空格隔开

只能选择作为某元素的最近一级子元素，

元素1>元素2 {样式}

.nav > a {  
 color: red;  
}

<p class="nav">  
 <a href="#">儿子</a>  
 <p>  
 <a href="#">孙子</a>  
 </p>  
</p>

此时只有儿子变红

9、并集选择器：

选择多组标签，同时为他们定义相同的样式，可用任何形式。

标签1,

标签2，

……{

属性： 值

}

10、伪类选择器：

用于向某些选择器添加特殊的效果，如当鼠标放到文字链接上时变色。

重点记a{} a:hover 的搭配使用

确保顺序： :link - :visited - :hover - :active LVHA 绿哈

因为链接有默认样式，所以要单独指定，一般的body{ }选择器无效

:focus伪类选择器，选取获得焦点（光标）的元素，一般都是<input>用，

input:focus

11、元素的现实模式：

行内元素：一行放多个标签（元素）：span…

1. 相邻行内元素在一行上
2. 宽高设置无效
3. 默认宽度就是内容本身的宽度
4. 行内元素只能容纳文本或其他行内元素

块元素：1）一行一个

2）可以控制 高度、宽度、外边距、内边距

3）宽度默认为容器（父级元素）的100%

4）是一个容器/盒子，里面还可以放块元素、行内元素

注意：文字类的元素内不能使用块元素：<p>标签主要用于存放文字，不能放块元素，特别是<div>

\*行内块元素（官方文档无此说法，是俗称）：同时具有块元素和行内元素的特点

1. 和相邻行内元素在一行上，但是他们之间会有空隙，一行可显示多个
2. 默认宽度就是它本身内容的宽度
3. 高度，行高、外边距、内边距都可以控制
4. <img/><input/><td>(表单)

12、元素显示模式转换：

->块元素：display: block;

->行内元素：display: inline;

->行内块元素：display: inline-block;

ps.当 行高==块元素高度 时，里面的文字可以垂直居中

行高<盒子高度，文字偏上

行高>盒子高度，文字偏下

13、css背景：

background-image: 背景图片  
background-repeat: 重复属性  
background-color: 背景颜色  
background-position: 背景位置  
background-size: 背景大小  
background-attachment:背景是否跟随滚动而滚动，默认为scroll，固定背景fixed

14、

1. 层叠性：

* 样式冲突遵循就近原则
* 样式不冲突不会层叠

1. 继承性：

* 主要继承父元素的文字样式
* 行高的继承写法：12px/1.5 中的1.5是指当前字体大小的1.5倍。如果父标签设置的字体为14px/1.5，子标签的字体设置为12px，则该标签下的行高为12px \* 1.5

1. 优先级：

|  |  |
| --- | --- |
| 选择器 | 权重 |
| 继承 或者 \* | 0，0，0，0 |
| 元素（标签）选择器 | 0，0，0，1 |
| 类选择器，伪类选择器 | 0，0，1，0 |
| ID选择器 | 0，1，0，0 |
| 行内样式style=”” | 1，0，0，0 |
| !important重要的 | 无穷大 |

* ! important用法：

color: white**!important**;

* 注意点：
  + 权重无进位，即类选择器永远大于权重选择器，ID选择器永远大于类选择器；
  + 等级判断从左向右，如果某一位数值相同，则判断下一位数值；
  + **继承的权重是0**，如果该元素没有直接选中，不管父元素权重多高，子元素权重都为0。且a标签有默认样式，父元素样式不影响a标签。

15、盒子模型

* border-style:

dashed 虚线 solid实线 dotted 点线

* border-collapse：collapse；可以让相邻表格（盒子）的边框合并
* 边框border宽度不算在盒子内
* **word-break:break-all：** 例如div宽200px，它的内容就会到200px自动换行，如果该行末端有个英文单词很长（congratulation等），它会把单词截断，变成该行末端为conra(congratulation的前端部分)，下一行为tulation（conguatulation）的后端部分了。

**word-wrap:break-word：** 与上面一样，但区别就是它会把congratulation整个单词看成一个整体，如果该行末端宽度不够显示整个单词，它会自动把整个单词放到下一行，而不会把单词截断掉的。

* padding：
  + padding：5px； 一个值代表四边都有5px内边距
  + padding：5px 10px；上下5，左右10
  + padding：5px 10px 20px；上5，左右10，下20
  + padding：5px 10px 20px 30px；上5右10下20左30
  + 内边距会把盒子撑大，在设计盒子前确认好边距，减去边距后的盒子才是理想的盒子。（巧用padding：因为可以撑开盒子，所以当盒子内字数不同时，直接不给盒子的长宽，只给一个padding，自然把盒子撑开了）
  + 如果(子元素)盒子本身没有指定width/height属性，则此时padding不会撑开盒子大小
* 给margin设置左右都为auto时就可以让盒子居中（非盒子模型（行内元素、行内块元素）：设置text-align：center）
* 盒子嵌套的塌陷问题（父盒子因为子盒子的边距原因跟着一起移动），可以给父盒子添加：overflow: hidden;
* 清除自带的内外边距：\*{margin: 0; padding: 0;}

16、**margin:10px 5px 15px 20px;**

* + 上边距是 10px
  + 右边距是 5px
  + 下边距是 15px
  + 左边距是 20px
* **margin:10px 5px 15px;**
  + 上边距是 10px
  + 右边距和左边距是 5px
  + 下边距是 15px
* **margin:10px 5px;**
  + 上边距和下边距是 10px
  + 右边距和左边距是 5px
* **margin:10px;**
  + 所有四个边距都是 10px

17、li里的原点用：list-style：none去掉

18、圆角边框：border-radius:

应用：

1）圆形盒子：设置宽高一样的正方形，然后给border-radius:设置为50%；

2）：

div{  
 width: 300px;  
 height: 100px;  
 background-color: pink;  
 border-radius: 150px;  
}

宽/高的一半

3）“border-radius:”一个值：全部角

两个值：左上和右下 右上和左下

三个值：左上 右上和左下 右下

四个角：左上 右上 右下 左下

4）border-top-left-radius: ;等等

19、盒子阴影：

box-shadow: 水平阴影的位置（必须）

垂直阴影的位置（必须）

模糊距离

阴影尺寸

阴影颜色

将外部阴影改为内部阴影（inset，默认为outset）;

20、浮动：

1、特性：

1. 脱离标准普通流的控制（浮）移动到指定位置（动）；
2. 浮动的盒子不再保留原先的位置；
3. 任何元素都可以设置浮动的属性，不管原先什么模式的元素，添加浮动后都具有行内块元素相似的特性；
4. 如果块元素没有设置宽度，默认宽度和父级一样宽，但是添加浮动后，大小由内容来决定

2、要点：

1. 浮动和标准流的父盒子搭配：

先用标准流的父元素排列上下位置，之后内部子元素采取浮动，排列左右位置。

1. 一个元素浮动了，理论上其余的兄弟元素也要浮动。

3、清除浮动：

有时候不方便给父盒子添加高度，就要清除浮动，清除浮动之后，父级就会根据浮动的子盒子自动检测高度，父级有了高度，就不影响下面的标准流了。（清除浮动元素脱离标准流造成的影响）

方法：

1. 额外标签法：

.clear{  
 clear: both;  
}

<div class="clear"></div>

ps.要求这个新的空标签必须是块级元素

1. 给父级添加overflow属性，属性值可以为hidden、auto和scroll

.box{  
 overflow: hidden;

…}

1. :after伪元素法：

给父级添加：

.clearfix:after{  
 content: "";  
 display: block;  
 height: 0;  
 clear: both;  
 visibility: hidden;  
}  
  
.clearfix{  
 \*zoom: 1; /\*IE6/7专有\*/  
}

1. 双伪元素清除浮动：

给父元素添加：

.clearfix:before,  
.clearfix:after{  
 content: "";  
 display: table;  
}  
  
.clearfix:after{  
 clear: both;  
}  
  
.clearfix{  
 \*zoom: 1;  
}

21、善用li+a。行内块元素之间默认有缝隙。

22、浮动的盒子不会有外边距合并的问题（不会出现塌陷）；浮动的元素不会压住下面标准流的文字。

23、定位：

选择器 {

position: static(仅理解，少)/relative/absolute/fixed/sticky（了解）

}

1. relative：
2. 相对于自己原来的位置来移动。
3. 原来在标准流的位置继续占有，后面的盒子仍然以标准流的方式对待它。（不脱标，继续保持原来位置）
4. absolute：
5. 相对于它的祖先元素移动。
6. 如果没有父元素或父元素没有定位，则以浏览器为准定位。
7. 如果有祖、爷、父元素…多级（相对、绝对、固定定位），则以最近一级的有定位祖先元素为参考点移动位置，例如：爷有定位，父无，子则以爷为参考进行定位，若爷、父都有定位，则以父为参考。
8. 不再占有原先的位置（脱标，区别于relative！）

❈相对定位经常用来作为绝对定位的父级。父盒子用于占有位置，子盒子不需要占有位置。子绝父相

1. fixed：

1.可以在浏览器页面滚动时元素的位置不会改变，即根据浏览器的可视窗口来定位。

2.不占有原先的位置（脱标）

技巧：让盒子固定在版心的某个位置：

让固定定位的盒子left:50%，走到浏览器可视区域的一半位置，再margin-left:版心宽度的一半距离

1. sticky：（有点类似于相对定位和固定定位的混合）
2. 以浏览器的可视窗口为参照点移动元素（固定定位特点）
3. 粘性定位占有原先的位置（相对定位特点）
4. 必须添加top，left，right，bottom其中一个才有效
5. 定位的叠放次序：

选择器{z-index: X

}

数值可以是正整数、负整数或0，默认是auto，数值越大越靠上；如果没有值（auto），则书写顺序上，后来的在上面，auto永远比实际的数字小；只有定位的盒子才有z-index属性。

1. 加了绝对/固定定位(相对定位不影响)的盒子不能通过margin:0 auto水平居中（margin会失效），但是可以通过设置盒子的left:50%; margin-left:<盒子宽度的一半>px;来让盒子居中。
2. 定位的特殊性：
3. 行内元素（<span>…）添加绝对或固定定位，可以直接设置高度和宽度；
4. 块级元素（<div>）添加绝对或固定定位，如果不给宽度或者高度，默认大小是内容的大小；
5. 脱标的盒子不会触发外边距塌陷：浮动元素、绝对/固定定位元素都不会触发。
6. 浮动的元素不会压住下面标准流的文字（浮动最初的目的是为了实现文字环绕图片（盒子）的效果），而绝对/固定定位会压住下面标准流所有的内容
7. 如果同时存在left和right属性，默认执行left；

如果同时存在top和bottom属性，默认执行top。

24、元素的显示与隐藏：

1、display属性用于设置一个元素如何显示：

display:none; 隐藏对象，且隐藏后不再占有原来位置。

display:block; 转换为块级元素和显示元素

2、visibility属性用于指定一个元素应可见还是隐藏：

visibility:visible; 元素可见。

visibility:hidden; 元素隐藏，且隐藏后继续占有原来位置。

3、overflow溢出：让一个元素在页面中隐藏或者显示出来：

visible：溢出部分全部显示；

hidden：溢出部分隐藏；

scroll：显示滚动条用来滚动查看溢出部分，不溢出也有滚动条；

auto：溢出时就显示一个滚动条，不溢出的时候就没有滚动条。

注：一般情况下都不会让内容溢出显示，但如果是有定位的盒子，慎用overflow:hidden，因为会隐藏多余的部分。

25、css高级技巧：

1、精灵图的使用：确定精灵图的x、y轴位置，然后类似于 background: url("images/sprites.png") -155px -106px; 使用。

2、字体图标：iconfont

1）将fonts文件夹放到根目录

2）引入代码

@font-face {  
 font-family: 'icomoon';  
 src: url('fonts/icomoon.eot?mbti4g');  
 src: url('fonts/icomoon.eot?mbti4g#iefix') format('embedded-opentype'),  
 url('fonts/icomoon.ttf?mbti4g') format('truetype'),  
 url('fonts/icomoon.woff?mbti4g') format('woff'),  
 url('fonts/icomoon.svg?mbti4g#icomoon') format('svg');  
 font-weight: normal;  
 font-style: normal;  
 font-display: block;  
}

3）使用时指定字体即可

span{  
 font-family: 'icomoon';  
 font-size: 100px;  
 color: pink;  
}

3、css三角：

示例：

div{  
 width: 0;  
 height: 0;  
 line-height: 0;  
 font-size: 0;  
 border: 50px solid transparent;  
 border-left-color: pink;  
}

26、css用户界面样式：

1、鼠标样式cursor：

li{cursor: xxx }

xxx:default/pointer/move/text/not-allowed

2、去掉输入框（text、textarea…）轮廓线：

outline:0; / outline:none;

3、防止拖拽文本域：

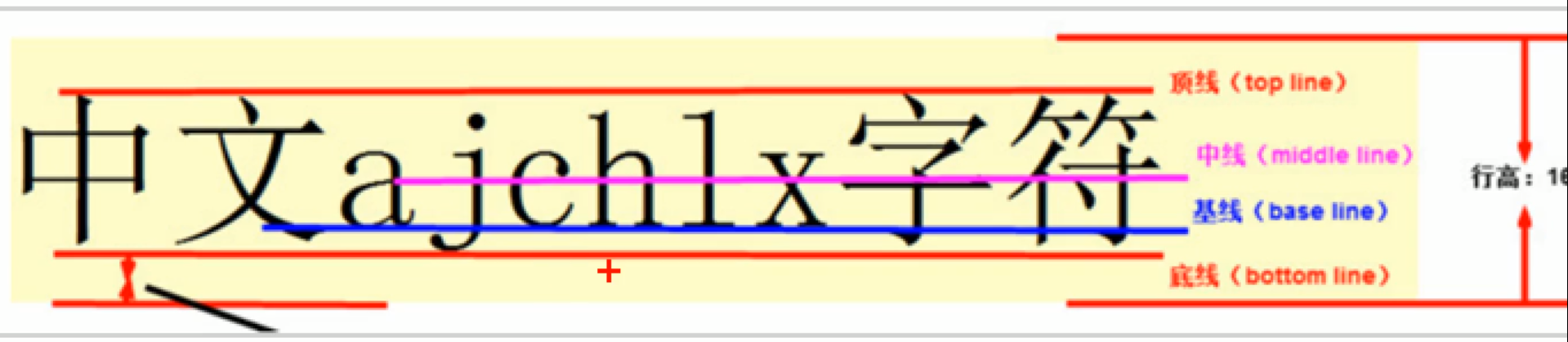
textarea{ resize:none;}

4、vertical-align：用于设置图片或者表单和文字垂直对齐。

官方解释：用于设置一个元素的垂直对齐方式，但是它只针对 行内元素或者行内块元素有效

默认为基线对齐。

vertical-align:baseline | top | middle | bottom



应用：解决图片底部默认空白缝隙问题：

给img添加vertical-align:baseline | top | middle | bottom

或

把img用display:block转换掉

vertical-align: middle;

可以让图片和文字水平居中

5、单行文本溢出显示省略号

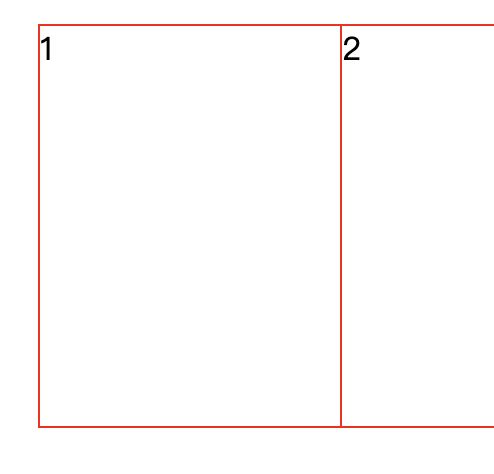
/\*1、先强制一行内显示文本\*/  
 white-space: nowrap;  
 /\*2、超出部分隐藏\*/  
 overflow: hidden;  
 /\*3、文字用省略号替代超出的部分\*/  
 text-overflow: ellipsis;

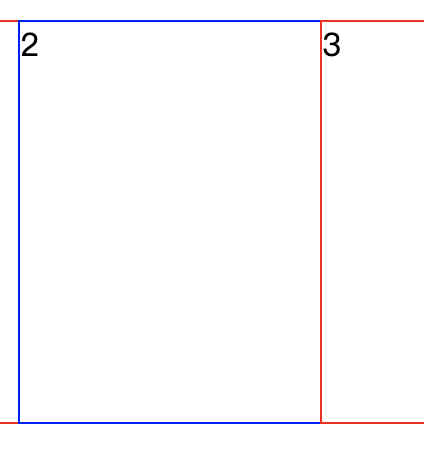
6、多航文本溢出显示省略号：

overflow: hidden;  
text-overflow: ellipsis;  
display: -webkit-box;  
-webkit-line-clamp: 2; （可以更改行数）  
-webkit-box-orient: vertical;

27、布局技巧：

1、margin负值：

让浮动的盒子在边框紧挨着的时候，中间的边框看起来是一条，而不是一粗条；

2、在做经过盒子效果时，一边边框不变色，解决方法可以提高盒子的层级，例如，如果没有定位，则加相对定位，如果有盒子已定位则加z-index（在子绝父相的情况下，父肯定是相对定位，此时用z-index）；

3、三角增强：



.box1{  
 width: 0;  
 height: 0;  
 border-top: 100px solid transparent;  
 border-right: 50px solid skyblue;  
 /\*border-bottom: 0 solid blue;\*/  
 /\*border-left: 0 solid green;\*/  
}

要点为下边框为0，上边框调大（高）。左/右边框控制底。

简写：

.box{  
 width: 0;  
 height: 0;  
 border-color: transparent red transparent transparent;  
 border-style: solid;  
 border-width: 22px 8px 0 0;  
}

例如：

.miaosha{  
 position: relative;  
 float: left;  
 width: 90px;  
 height: 100%;  
 background-color: red;  
 text-align: center;  
 color: white;  
 font-weight: 700;  
 margin-right: 8px;  
}  
.miaosha i{  
 position: absolute;  
 right: 0;  
 top: 0;  
 width: 0;  
 height: 0;  
 border-color: transparent white transparent transparent;  
 border-style: solid;  
 border-width: 24px 10px 0 0;  
}  
.origin{  
 font-size: 12px;  
 color: gray;  
 text-decoration: line-through;  
}

<div class="price">  
 <span class="miaosha">  
 $1650  
 <i></i>  
 </span>  
 <span class="origin">$5660</span>  
</div>



28、css的初始化：京东为例：

29、HTML5新增语义化标签：

\* {  
 margin: 0;  
 padding: 0  
}  
  
em, i {  
 font-style: normal  
}  
  
li {  
 list-style: none  
}  
  
img {  
 border: 0;  
 vertical-align: middle  
}  
  
button {  
 cursor: pointer  
}  
  
a {  
 color: #666;  
 text-decoration: none  
}  
  
a:hover {  
 color: #c81623  
}  
  
button, input {  
 font-family: Microsoft YaHei, Heiti SC, tahoma, arial, Hiragino Sans GB, "\5B8B\4F53", sans-serif  
}  
  
body {  
 -webkit-font-smoothing: antialiased;  
 background-color: #fff;  
 font: 12px/1.5 Microsoft YaHei, Heiti SC, tahoma, arial, Hiragino Sans GB, "\5B8B\4F53", sans-serif;  
 color: #666  
}  
  
.hide, .none {  
 display: none  
}  
  
.clearfix:after {  
 visibility: hidden;  
 clear: both;  
 display: block;  
 content: ".";  
 height: 0  
}  
  
.clearfix {  
 \*zoom: 1  
}  
  
.mod\_price {  
 font-size: 14px;  
 color: #e33333  
}

<header><nav><article><section><aside><footer>

在IE9中需要把这些元素转换为块级元素。

<audio> 尽量用mp3



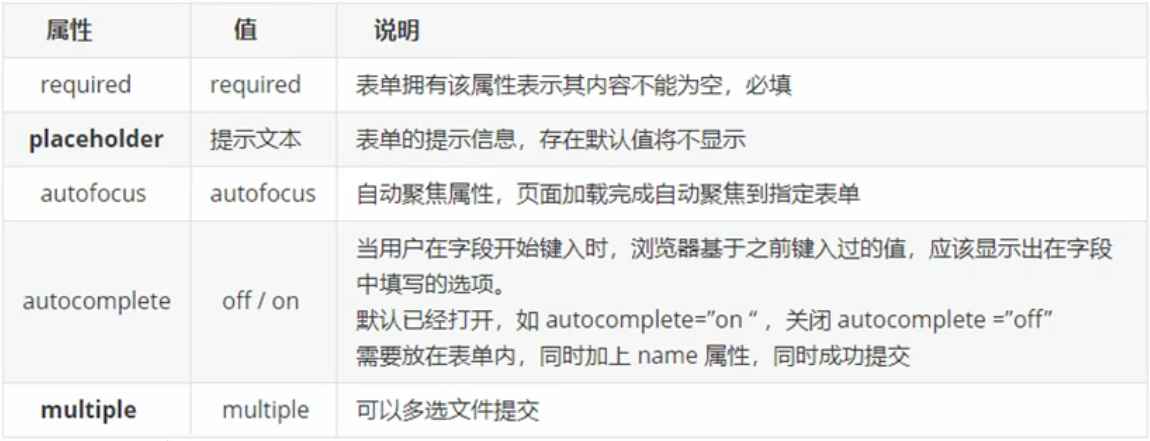
<video>尽量用mp4



<input> html5新增：主要记 number tel search



表单属性：



30、css3属性选择器：当该标签有中括号内的属性时才选中，即同时满足选择器和属性：

1、选择器[属性]{}。

2、选择器[属性=值]{} (例如input标签里多种type（type=“text”和type=“password”），然后可以选出仅text或password)。

3、选择器[属性^=值]{} 属性选择器可以选择属性 值 开头相同的某些元素。

选择器[属性$=值]{} 属性选择器可以选择属性 值 结尾相同的某些元素。

选择器[属性\*=值]{} 属性选择器可以选择属性 值 有相同字段的某些元素。



注：类选择器、属性选择器、伪类选择器，权重为10。

31、结构伪类选择器：



关于E:nth-child(n)/E:nth-of-type(n)中n的用法：

* n可以是数字，关键字和公式；
* n如果是数字，就是选择第n个子元素，里面数字从1开始…；
* n可以是关键字：even偶数，odd奇数；
* 当n作公式的时候，n从0开始递增即2\*n=0 2 4 6…；
* 例：n+5：从第五个开始/-n+5：只有前五个，包括第五个；
* nth-child会把所有盒子都排序号，即使是不同的标签。

例：

section div:nth-child(1){  
 background-color: #ff8500;  
}

<section>  
 <p>1</p>  
 <div>2</div>  
 <div>3</div>  
</section>

这里没有文字会被选取，因为是先识别nth-child(1)，但第一个（p标签）不是div，所以不会被选中

如果是section div:nth-of-type(1){  
 background-color: #00a4ff;  
 }

，则 2 的背景会变色，因为这个指的是div的第一个，即先识别标签，再识别序号n，跟nth-child正好相反。

小结：

* 结构伪类选择器一般用于选择父级里面的第几个孩子。
* nth-child对父元素里面所有孩子排序选择（序号是固定的），先找到第n个孩子，然后看看是否和E匹配。
* nth-of-type对父元素里面指定子元素进行排序选择，先去匹配E，然后再根据E找第n个孩子。
* 类选择器、属性选择器、伪类选择器，权重为10。

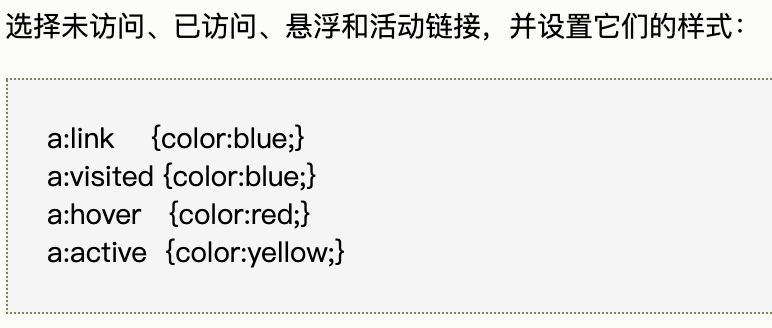
32、伪元素选择器：

伪元素选择器可以帮助我们利用css创建新标签元素，而不需要html标签，从而简化html结构。

|  |  |
| --- | --- |
| 选择符 | 间接 |
| ::before | 在元素内部的前面插入内容 |
| ::after | 在元素内部的后面插入内容 |

注意：

* Before和after创建一个元素，但是属于行内元素。
* 语法：element::before/after。
* Before和after必须有content属性。
* Before在父元素内容的前面创建元素，after在父元素内容的后面插入元素。
* 伪元素选择器和标签选择器一样权重为1。
* Content还能在使用字体图标时，直接填入编号展示，如：content: '\e928';
* 当要对伪元素做:hover这类的操作时，



代码为 属性:伪类选择器::before{}（前面已写div::before{}）

（注意区分伪类选择器和伪元素选择器！ ）

33、css3盒子模型：

box-sizing:content-box/border-box

box-sizing:content-box的盒子大小为width+padding+border

box-sizing: border -box的盒子大小为width

如果设定为border-box，padding和border就不会撑大盒子了，前提是padding和border不会超过width宽度

34、css其他：

* css3滤镜filter：将模糊、颜色、偏移等图形效果应用于元素。

filter:函数();(函数为js里的函数)blur([…px])模糊函数;

* 属性:calc() 函数，可以在一些属性里做简单的+-\*/运算

width:calc(100%-20px);

35、css3的过渡效果：（重点，\*为必填项）

常和:hover配合使用。

transition: \*要过渡的属性 \*时间（s） 运动曲线 何时开始;

属性：要变化的css属性，宽度高度背景颜色内外边距，所有属性写all；

何时开始：当触发变化条件后（如鼠标放上去之后），几秒后才开始变化，默认为0立刻开始变化；

注：多个属性的改变要用逗号分割！