复习：

盒子模型

X轴总空间=MarginLeft + BorderLeft + PaddingLeft + Width+ PaddingRight + BorderRight + MarginRight

Y轴总空间=MarginTop + BorderTop + PaddingTop + Height+ PaddingBottom + BorderBottom + MarginBottom

**区块元素**：

·无浮动：竖直方向上排布，相邻元素的Margin会合并，Padding各自独立；

·浮动之后：Padding各自独立；竖直/水平方向Margin上都各自独立

**内联元素**：水平方向上排布，一行不够自动排布到下一行，

Margin：水平方向上各自独立，不合并；竖直方向上无效。

Padding：水平方向上各自独立；竖直方向上有效但不占用页面空间(即上下两行的padding可能会重叠)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Margin X | Margin Y | Padding X | Padding Y |
| BLOCK不浮动 | 占用空间 | 占空间、会合并 | 占用空间 | 占用空间 |
| BLOCK 浮动 | 占空间、不合并 | 占空间、不合并 | 占空间 | 占空间 |
| INLINE | 占空间、不合并 | 无效 | 占空间 | 有效、不占空间(重叠) |

元素的定位

1. 流/静态定位： 默认/position:static; 不能指定位置
2. 浮动定位: float:left/right; 不能严格指定位置
3. 相对定位: position:relative; 使用left/top/right/bottom进行定位

仍占用页面空间；以“其自己的静态定位点”为定位原点

1. 绝对定位: position:absolute; 使用left/top/right/bottom进行定位。

不占用页面空间；以“最近的已定位的祖先元素的padding起点”为定位原点，且随着页面内容的滚动而滚动。

1. 固定定位: position:fixed; 使用left/top/right/bottom进行定位

不占用页面空间；以“body”为定位原点，且不随着页面内容的滚动而滚动。

\*{}

div{}

#main{}

.product{}

div, span, #d1, .str{}

span.title{}

div span{}

div > span{}

1.CSS复杂选择器

提示：复杂选择器的使用可以减少页面中id和class的出现频率；使用jQuery可以兼容所有下述复杂选择器。

|  |  |  |  |
| --- | --- | --- | --- |
| 复杂选择器 | 示例 | 含义 | 版本/兼容性 |
| 选1 选2 | div ul{...} | 子代选择器 | 1 |
| 选1 > 选2 | div > ul{...} | 直接子代选择器 | 2 |
| 选1 + 选2 | input + span{...} | 相邻兄弟选择器；选择紧挨的下一个兄弟元素(选中0/1个) | 2/ IE6-不支持 |
| 选1 ~ 选2 | div ~ span{...} | 通用兄弟选择器；  匹配选择器1的兄弟元素中匹配选择器2的元素 | 3  注意：div后的span兄弟会选定；之前的不会 |
| [属性名] | [href] | 选择具有指定HTML属性的元素 | 2 |
| [属性名=属性值] | [type=”text”] | 匹配具有指定属性且属性值为指定值 | 2 |
| [属性名~=属性值] | [class~=strong] | 匹配具有指定的属性且属性值中包含指定的**完整的单词**(不是单词不行) | 3 |
| [属性名\*=属性值] | [class\*=str] | 匹配具有指定的属性且属性值中包含指定的字母组合(不必是完整的单词) | 3 |
| [属性名^=属性值] | [class^=str] | 匹配具有指定的属性且属性值以指定的字母开头 | 3 |
| [属性名$=属性值] | [class$=str] | 匹配具有指定的属性且属性值以指定的字母结尾 | 3 |
| [属性名|=属性值] |  | 匹配具有指定的属性且属性值以指定的完整的单词开头(要么只有它，要么其后有-) | 3 |
| :target | :target{}  div:target{} | 目标伪类选择器；选定【当前锚点的】**目标元素** | IE8-不支持 |
| :enabled | :enabled{...} | 匹配当前启用的表单元素 |  |
| :disabled | :disabled{...} | 匹配当前禁用的表单元素 |  |
| :checked | :checked{...} | 匹配当前选定的表单元素 |  |
| :first-child | span:first-child{}  div :first-child{} | 匹配父元素中的第一个子元素(必须是标签，纯文本不算) |  |
| :last-child | p:last-child{ }  div :last-child{} | 匹配作为父元素中最后一个子元素出现的元素 |  |
| :only-child | p:only-child{ } | 匹配作为父元素中唯一子元素出现的元素 |  |
| :empty | div:empty{display:none;} | 匹配没有子元素且没有任何文本内容的元素 |  |
| :not(选1) | div:not(.strong){}  span:not([class=content]) | 否定选择器；匹配不被选择器1选定的元素 |  |
| ::selection | ::selection{ } | 匹配一段文字中被选择部分 |  |

说明： :xxx 伪类选择器 ::xxx 伪元素选择器

2.内容生成选择器

XHTML规定：页面内容交给标签来处理；页面表现交给CSS来处理。

但CSS3有些“内容生成”选择器不单单可以呈现样式，还可以向页面中添加内容。

选择器1:before{

.....

content: 纯文本/图像/计数器;

}

选择器1:after{

.....

content: 纯文本/图像/计数器;

}

content属性必须配合:before/:after选择器使用。

计数器的使用：

body{

counter-reset: 计数器名 初始值 [计算器2名 初始值]...;

}

p:before{

content: ‘《’ counter(计数器名) ’》’;

counter-increment: 计数器名 [步增值];

}

|  |
| --- |
| 课外资料：程序是如何执行循环的——循环三要素  mc = 1; /\*1.循环开始前定义一个循环变量，赋初始值\*/  while(true){  mc = mc - 2; /\*2.循环变量要发生改变\*/  print(mc); /\*3.使用循环变量\*/  } |

3.与内容多列显示相关的CSS样式

column-count: 竖直方向上列的数量

column-gap: 内容列与列间距

column-style: 集合属性 width style color

注意浏览器的兼容性：

·IE11直接使用上述属性

·FF添加-moz-前缀

·Chrome/Safari/Opera添加-webkit-前缀