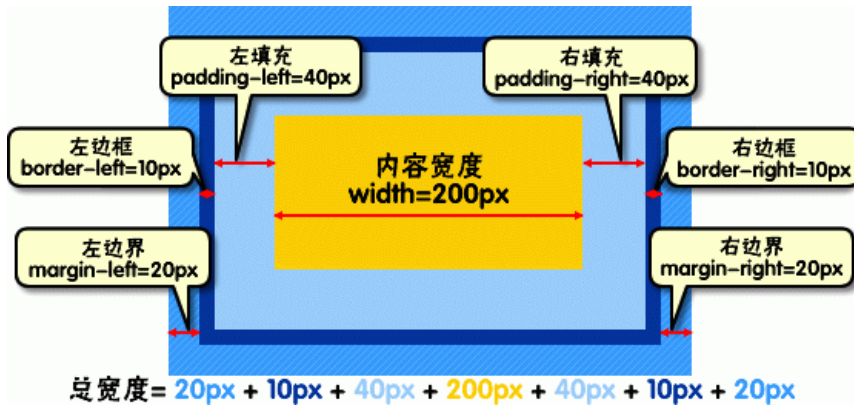


# CSS 盒模型

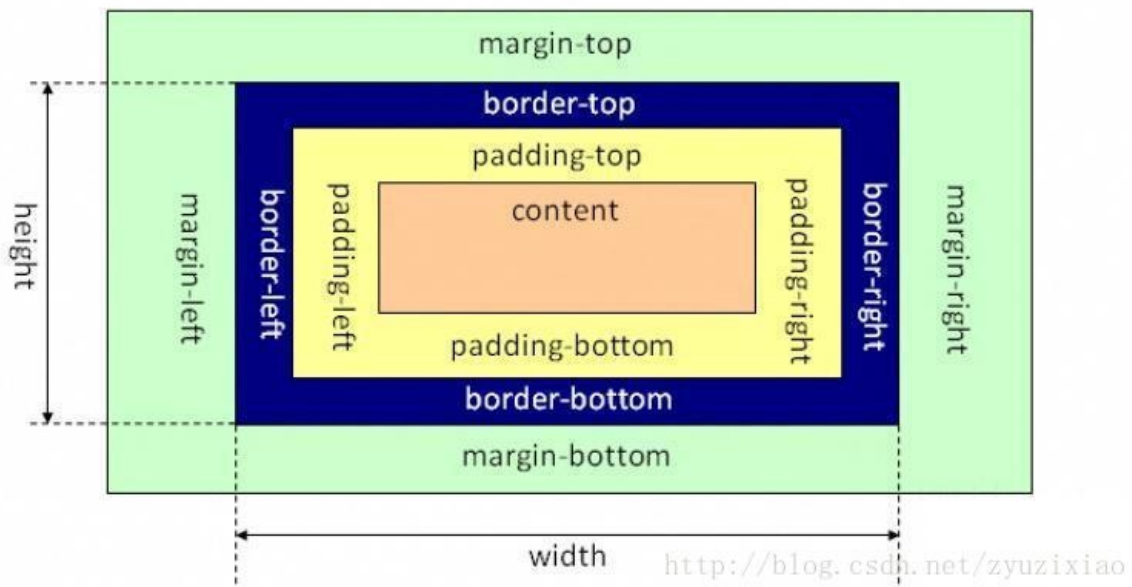
## 盒模型



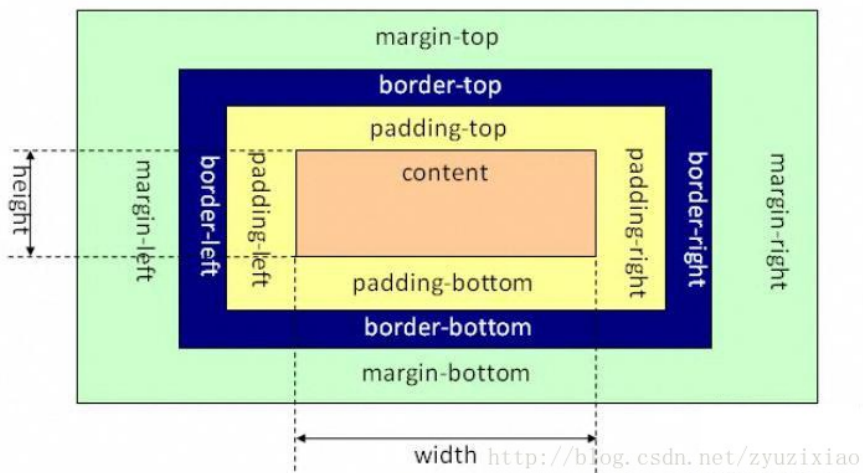
## IE6盒模型与W3C标准盒模型

IE6 盒子模型中，盒子的尺寸包含了内容区，padding，border 这三个部分，而 W3C 的盒子模型中，盒子的尺寸只包含内容区，padding 和 border 被排除在盒子尺寸之外。

## ■ IE盒子模型



## ■ 标准盒子模型



IE6 的盒子模型更合理，所以 W3C 在 CSS3 中增加了 box-sizing 属性。该属性有两个可选值，默认的 content-box

和 border-box，选用后者，盒子模型将按 IE6 的方式进行处理。

属性

border边框属性

属性	描述
<a href="#">border</a>	简写属性，设置所有的边框属性。可以按顺序设置如下属性：border-width, border-style, border-color
<a href="#">border-width</a>	简写属性，用于为元素的所有边框设置宽度，或者单独地为各边边框设置宽度。可以指定长度值，比如 2px 或 0.1em；或者使用关键字(thin、medium（默认） 和 thick)。
<a href="#">border-style</a>	用于设置元素所有边框的样式，或者单独地为各边设置边框样式。常见样式有：none（默认），dashed（虚线），dotted（点线），solid（实线），double（双线）。
<a href="#">border-color</a>	简写属性，设置元素的所有边框中可见部分的颜色，或为 4 个边分别设置颜色。颜色值可以是命名颜色，也可以是十六进制和 RGB 值。
<a href="#">border-bottom</a>	简写属性，用于把下边框的所有属性设置到一个声明中。
<a href="#">border-bottom-color</a>	设置元素的下边框的颜色。
<a href="#">border-bottom-style</a>	设置元素的下边框的样式。
<a href="#">border-bottom-width</a>	设置元素的下边框的宽度。
<a href="#">border-left</a>	简写属性，用于把左边框的所有属性设置到一个声明中。
<a href="#">border-left-color</a>	设置元素的左边框的颜色。
<a href="#">border-left-style</a>	设置元素的左边框的样式。
<a href="#">border-left-width</a>	设置元素的左边框的宽度。
<a href="#">border-right</a>	简写属性，用于把右边框的所有属性设置到一个声明中。
<a href="#">border-right-color</a>	设置元素的右边框的颜色。
<a href="#">border-right-style</a>	设置元素的右边框的样式。
<a href="#">border-right-width</a>	设置元素的右边框的宽度。
<a href="#">border-top</a>	简写属性，用于把上边框的所有属性设置到一个声明中。
<a href="#">border-top-color</a>	设置元素的上边框的颜色。
<a href="#">border-top-style</a>	设置元素的上边框的样式。
<a href="#">border-top-width</a>	设置元素的上边框的宽度。

padding内边距属性

属性	描述
<a href="#">padding</a>	简写属性。设置元素的所有内边距属性，默认值为0。接受长度值或百分比值，但不允许使用负值。
<a href="#">padding-bottom</a>	设置元素的下内边距。
<a href="#">padding-left</a>	设置元素的左内边距。
<a href="#">padding-right</a>	设置元素的右内边距。
<a href="#">padding-top</a>	设置元素的上内边距。

margin外边距属性

属性	描述
<a href="#">margin</a>	简写属性。设置所有外边距属性，默认值为0。可以设置为 auto。接受长度值或百分比值（基于父元素的宽度）。外边距可以是负值。
<a href="#">margin-bottom</a>	设置元素的下外边距。
<a href="#">margin-left</a>	设置元素的左外边距。
<a href="#">margin-right</a>	设置元素的右外边距。
<a href="#">margin-top</a>	设置元素的上外边距。

元素内容属性

属性	描述
<a href="#">height</a>	设置元素高度，默认值为auto。
<a href="#">width</a>	设置元素高度，默认值为auto。
<a href="#">max-height</a>	设置元素的最大高度。
<a href="#">max-width</a>	设置元素的最大宽度。
<a href="#">min-height</a>	设置元素的最小高度。
<a href="#">min-width</a>	设置元素的最小宽度。

值顺序

值的顺序是从top 开始围着元素顺时针旋转的：

top right bottom left

CSS 定义了一些规则，允许为边距指定少于 4 个值。规则如下：

- 如果缺少左边距的值，则使用右边距的值。
- 如果缺少下边距的值，则使用上边距的值。
- 如果缺少右边距的值，则使用上边距的值。



如果为边距指定了 3 个值，则第 4 个值（即左边距）会从第 2 个值（右边距）复制得到。

如果给定了两个值，第 4 个值会从第 2 个值复制得到，第 3 个值（下边距）会从第 1 个值（上边距）复制得到。

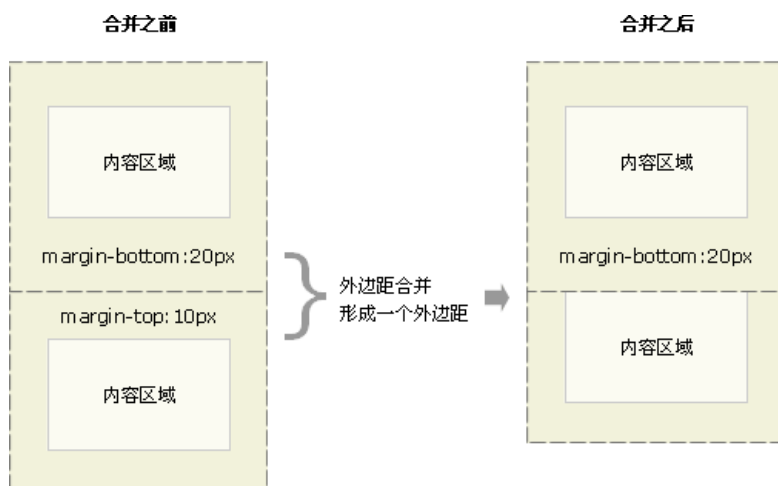
如果只给定一个值，那么其他 3 个边距都由这个值（上边距）复制得到。

外边距合并

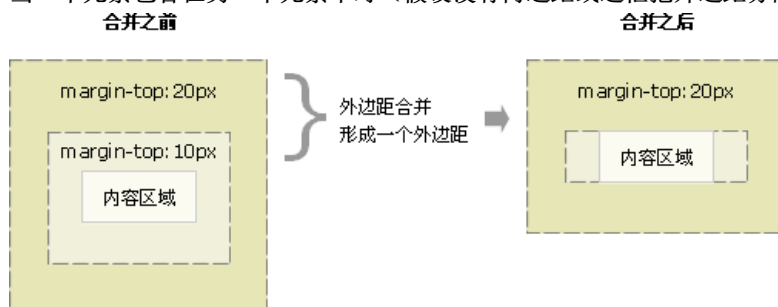
当两个垂直外边距相遇时，它们将形成一个外边距。

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。

当一个元素出现在另一个元素上面时，第一个元素的下外边距与第二个元素的上外边距会发生合并。

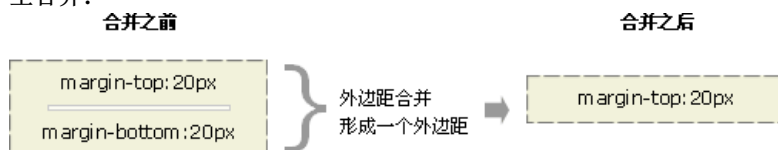


当一个元素包含在另一个元素中时（假设没有内边距或边框把外边距分隔开），它们的上和/或下外边距也会发生合并。



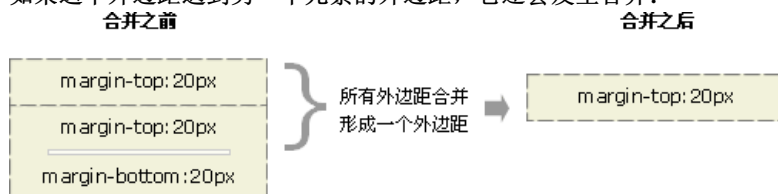
外边距甚至可以与自身发生合并。

假设有一个空元素，它有外边距，但是没有边框或填充。在这种情况下，上外边距与下外边距就碰到了一起，它们会发生合并：



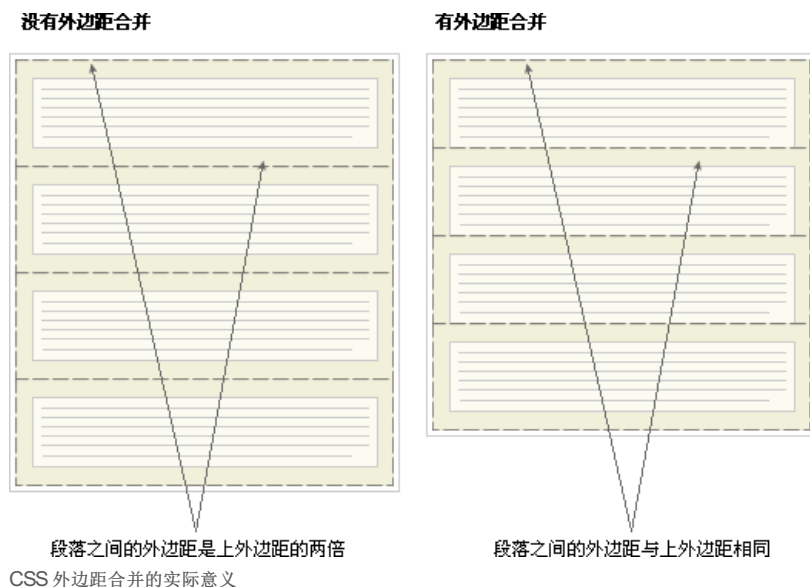
CSS 外边距合并实例 3

如果这个外边距遇到另一个元素的外边距，它还会发生合并：



CSS 外边距合并实例 4

这就是一系列的段落元素占用空间非常小的原因，因为它们的所有外边距都合并到一起，形成了一个小的外边距。



注释：只有普通文档流中块框的垂直外边距才会发生外边距合并。行内框、浮动框或绝对定位之间的外边距不会合并。

### auto值

是很多尺寸的默认值，表示由浏览器渲染时自动计算值。margin、width、height可以设置为auto。

普通文档流中（没有定位或position:relative;）的块状元素：

在水平方向上：

如果不设置width（即width: auto;），则整个元素水平方向占父元素的100%。width的值就等于父元素的宽度减去margin、border、padding的值。所以在父元素宽度变化的时候，该元素的width也会随之变化。

如果设置margin: auto, width定宽，那么父元素的宽度减去width、border、padding剩下来的宽度就会分成两等份分别给左右margin（达到居中效果）。

如果width、margin同时设置为auto，则margin为0，width占整个父元素的宽度。

在垂直方向上：

如果不设置height（即height: auto;），则根据块内的内容自动调节高度。

如果设置margin: auto; 则margin为0。因为垂直方向是被设计成可以无限扩展的，内容越多浏览器便产生滚动条来扩展，所以垂直方向找不到一个计算基准，会返回一个false，便成了0。

当块状元素被设为 float:left/right | overflow | position:absolute:

若 width 和 height 没有定值，该元素就会产生包裹性。width 和 height 就变成了内容的宽度和高度。

在具有包裹性的元素上想利用 width : auto; 来让元素宽度自适应浏览器宽度是不行的。

### 负边距

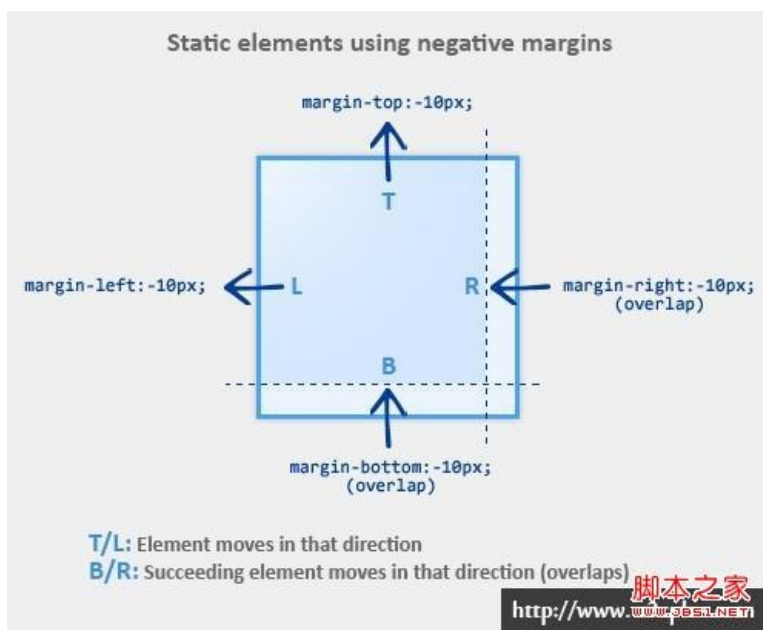
没有脱离文档流的元素（不是浮动元素、绝对定位、固定定位的元素等）：

定宽元素：

设置 margin-top/margin-left 为负数，元素会产生向上/左位移。后面文档流中的其它元素就会流过来填充这部分空间。（不同于相对定位，通过相对定位偏移后，其仍然会坚守着它原来占据的空间，不会让文档流的其它元素乘虚而入。）

设置 margin-bottom/margin-right 为负数，会将后续文档中的元素流拖拉进来，覆盖本来的元素。

即文档流只能向左或向上流动，不能向下或向右移动。



实例：多列等高

不定宽元素：

设置 `margin-left/margin-right` 会增加宽度。

设置 `margin-top/margin-bottom` 为负数，跟定定宽元素效果相同。

实例：去除列表右边框

浮动元素

设置负margin，浮动元素可以向左或者向右流动，多个浮动元素将产生重叠。

三个向左浮动的box，宽度是100px：



给它们都加上`margin-left: -25px;`



只给box3设置`margin-left: -200px;`



绝对定位元素

负margin会基于元素的绝对定位坐标再偏移。

实例：绝对定位实现水平垂直居中