






# CSS3 框大小

CSS3 `box-sizing` 属性可以设置 `width` 和 `height` 属性中包含了 `padding`(内边距) 和 `border`(边框)。

## 浏览器支持

表格中的数字表示支持该属性的第一个浏览器的版本号。  
紧跟在数字后面的 `-webkit-` 或 `-moz-` 为指定浏览器的前缀。

属性					
box-sizing	10.0 4.0 -webkit-	8.0	29.0 2.0 -moz-	5.1 3.1 -webkit-	9.5

## 不使用 CSS3 box-sizing 属性

默认情况下，元素的宽度与高度计算方式如下：

**width(宽) + padding(内边距) + border(边框) = 元素实际宽度**

**height(高) + padding(内边距) + border(边框) = 元素实际高度**

这就意味着我们在设置元素的 `width/height` 时，元素真实展示的高度与宽度会更大(因为元素的边框与内边距也会计算在 `width/height` 中)。

这个是个较小的框 (width 为 300px , height 为 100px)。

这个是个较大的框 (width 为 300px , height 为 100px)。

以上两个 `<div>` 元素虽然宽度与高度设置一样，但真实展示的大小不一致，因为 `div2` 指定了内边距:

实例

```
.div1 {
  width: 300px;
  height: 100px;
  border: 1px solid blue;
}
```

```
}  
  
.div2 {  
  width: 300px;  
  height: 100px;  
  padding: 50px;  
  border: 1px solid red;  
}
```

[尝试一下 »](#)

使用这种方式如果想要获得较小的那个框且包含内边距，就不得不考虑到边框和内边距的宽度。

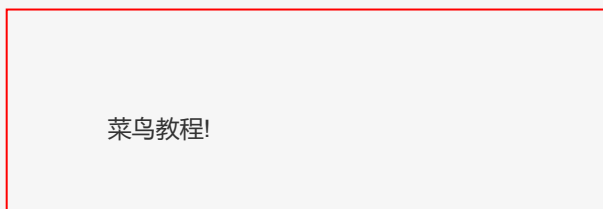
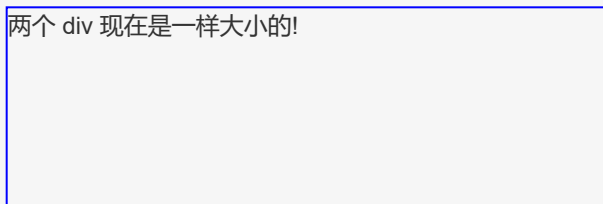
CSS3 的 `box-sizing` 属性很好的解决了这个问题。

## 使用 CSS3 box-sizing 属性

CSS3 `box-sizing` 属性在一个元素的 `width` 和 `height` 中包含 `padding`(内边距) 和 `border`(边框)。

如果在元素上设置了 `box-sizing: border-box;` 则 `padding`(内边距) 和 `border`(边框) 也包含在 `width` 和 `height` 中:

两个 div 现在是一样大小的!



以下是两个 `<div>` 元素添加 `box-sizing: border-box;` 属性的简单实例。

### 实例

```
.div1 {  
  width: 300px;  
  height: 100px;  
  border: 1px solid blue;  
  box-sizing: border-box;  
}  
  
.div2 {  
  width: 300px;  
  height: 100px;  
  padding: 50px;  
  border: 1px solid red;  
  box-sizing: border-box;  
}
```

[尝试一下 »](#)

从结果上看 `box-sizing: border-box;` 效果更好，也正是很多开发人员需要的效果。

以下代码可以让所有元素以更直观的方式展示大小。很多浏览器已经支持 `box-sizing: border-box;` (但是并非所有 - 这就是为什么 `input` 和 `text` 元素设置了 `width: 100%;` 后的宽度却不一样)。

所有元素使用 `box-sizing` 是比较推荐的：

## 实例

```
* {  
  box-sizing: border-box;  
}
```

尝试一下 »

← CSS 分页实例

CSS3 弹性盒子 →

 点我分享笔记