

學習要點:

1. 定位布局
2. box-sizing
3. resize

本章主要探討 HTML5 中 CSS 早期所使用的傳統布局，很多情況下，這些布局方式還是非常有用的。

## 一、定位布局

在使用定位布局前，我們先了解一下定位屬性的用法。CSS2 提供了 **position** 屬性來實現元素的絕對定位和相對定位。

屬性	說明
static	默認值，無定位。
absolute	絕對定位，使用 top、right、bottom、left 進行位移
relative	相對定位，使用 top、right、bottom、left 進行位移
fixed	以窗口為參考定位，使用 top、right、bottom、left 進行位移

//絕對定位，脫離文檔流，以窗口文檔左上角 0,0 為起點

```
header {  
    position: absolute;  
    top: 100px;  
    left: 100px;  
}
```

所謂脫離文檔流的意思，就是本身這個元素在文檔流是佔位的。如果脫離了，就不佔有文檔的位置，好像浮在了空中一般，有了層次感。

由於絕對定位脫離了文檔流，出現層次概念。那麼每個元素到底在哪一層，會不會衝突覆蓋。這時通過 **z-index** 屬性來判定它們的層次關係。

屬性	說明
auto	默認層次
數字	設置層次，數字越大，層次越高

//設置在 100 層上

```
header {  
    z-index: 100;  
}
```

//以窗口參考定位，脫離文檔流，會隨著滾動條滾動而滾動

```
header {  
    position: fixed;  
    top: 100px;  
    left: 100px;  
}
```

//相對定位，不脫離文檔流，佔位偏移

```
header {  
    position: relative;  
    top: 100px;  
    left: 100px;  
}
```

這三種分別都在各自的情況下使用，均比較常用。但還有一種情況，就是：

1. 既要脫離文檔流(這樣元素之間不會相互衝突):
2. 以父元素，比如 **body** 或其他父元素為參考點(這樣可以實現區域性絕對定位):
3. 還必須是絕對定位。

//第一步，將需要設置參考點的父元素設置為相對，且不設置座標

```
body {  
    position: relative;  
}
```

//第二步，如果父元素設置了參考點，子元素的絕對定位將以它為基準

```
header {  
    position: absolute;  
    top: 0px;  
    left: 0px;  
}
```

## 1. 固定布局

//CSS 部分

```
body {  
    width: 960px;  
    margin: 0 auto;  
    position: relative;  
}
```

```
header {
```

```
width: 960px;
height: 120px;
background-color: olive;
position: absolute;
top: 0;
left: 0;
}

aside {
width: 200px;
height: 500px;
background-color: purple;
position: absolute;
top: 120px;
/*left: 200px;*/
right: 0;
}

footer {
width: 960px;
height: 120px;
background-color: gray;
position: absolute;
top: 620px;
}
```

在上面，基本都用了定位來進行固定布局。但細心的可以發現，其實只有右側需要實行絕對定位，其他就按照普通的擺放即可。對於設計成流體布局，只要將長度設置成百分比即可。

## 二、box-sizing

在盒模型那個章節，我們了解到元素盒子如果加入了內邊距 **padding** 和邊框 **border** 後，它的總長度會增加。那麼如果這個元素用於非常精確的佈局時，我們就需要進行計算增減。這其實是比较煩人的操作，尤其是動態設置頁面布局的時候。

CSS3 提供了一個屬性 **box-sizing**，這個屬性可以定義元素盒子的解析方式，從而可以選擇避免掉布局元素盒子增加內邊距和邊框的長度增減問題。

屬性	說明
----	----

content-box	默認值，border 和 padding 設置後用於元素的總長度
border-box	Border 和 padding 設置後不用於元素的總長度

//設置 border-box 讓 border 和 padding 不再額外增加元素大小

```
aside {
    width: 200px;
    height: 500px;
    background-color: purple;
    padding: 10px;
    border: 5px solid red;
    box-sizing: border-box;
    float: left;
}
```

Box-sizing 是 CSS3 推出的，各個廠商在實現時設置了私有前綴。

	Opera	Firefox	Chrome	Safari	IE
支持需帶前綴	無	2 ~ 28	4 ~ 9	3.1 ~ 5	8.0+
支持不帶前綴	10.1+	29+	10+	6+	9.0+

//完整形式

```
-webkit-box-sizing: border-box;
-moz-box-sizing: border-box;
-ms-box-sizing: border-box;
box-sizing: border-box;
```

### 三、resize

CSS3 提供了一個 resize 屬性，來更改元素尺寸大小。

屬性	說明
none	默認值，不允許用戶調整元素大小
both	用戶可以調節元素的寬度和高度
horizontal	用戶可以調節元素的寬度
vertical	用戶可以調節元素的高度

一般普通元素，默認值是不允許的。但如果是表單類的 textarea 元素，默認是允許的。而普通元素需要設置 overflow:auto，配合 resize 才會出現可拖曳的圖形。

//允許修改

```
aside {
```

```
    resize: both;  
    overflow:auto;  
}
```