

2.5.CSS - Box Model與版面定位

一、元素結構

(一) 元素的區別

CSS 把每個 HTML 標籤所建立的元素視為一個獨立的個體，依特性不同又區分為「區塊元素」與「行內元素」。

- **區塊元素**：於 HTML 文件中加入後會自動換行的元素，如<h1>、<p>、<ui>...等。這些些元素預設會自動換行自成一區。
- **行內元素**：於 HTML 文件中加入後仍與前後內容放在同一個文字段落中，如<a>、、...等。這些元素仍會與前後內容連在一起。

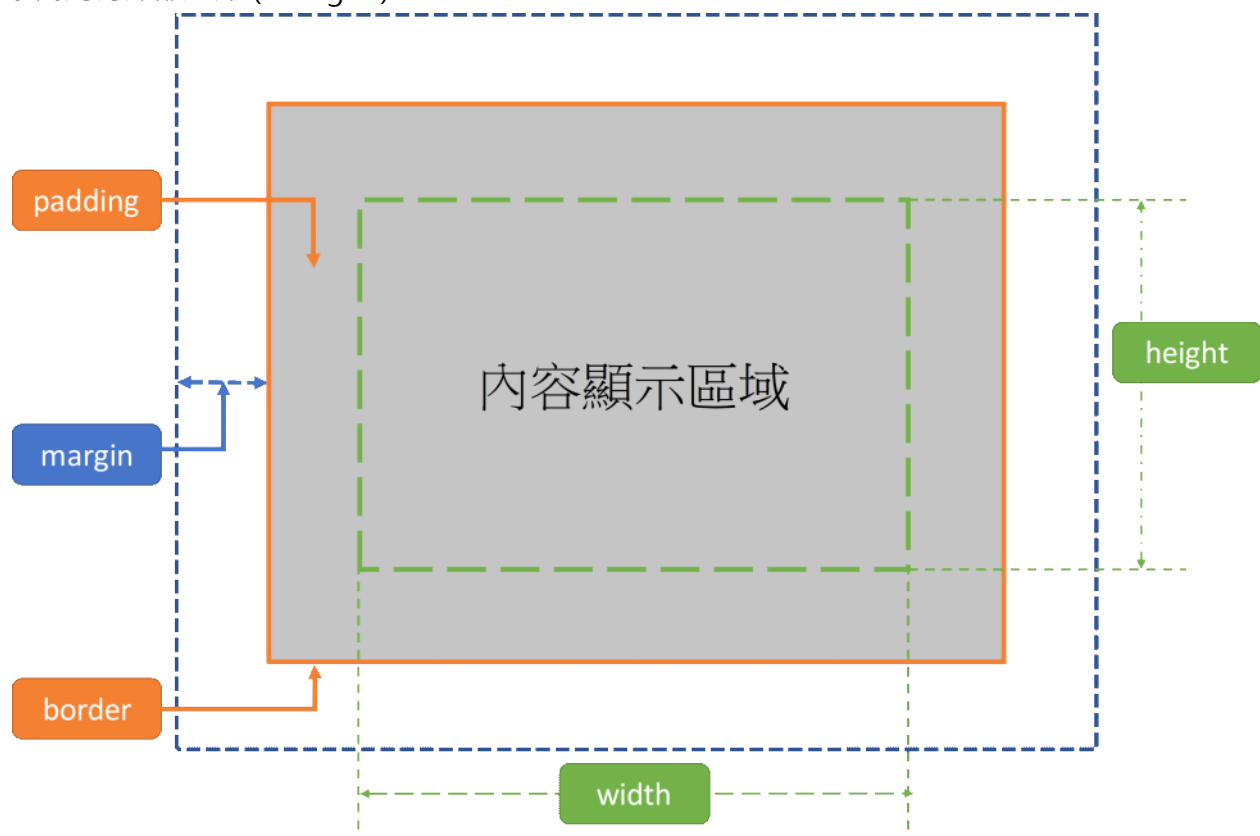
(二) 容器

區塊元素是利用 CSS 進行網頁排版時很重要的結構之一，它可以視為網頁版面上的一個容器，可以使用 CSS 設定這個容器的大小、外觀等。

二、Box Model的結構

區塊元素在加上寬、高、內距、框線及邊界的屬性後，整個外觀就如同一個盒子般，所以稱為盒子模型 (Box Model)。在網頁的版面上可以說是一個個 Box 組成的，只要能熟悉 Box Model 的結構，就能準確的規劃網頁版面。

以下是 Box Model 的示意圖，HTML 標籤包含的區塊元素即是灰色方塊，也就是瀏覽者會實際看到的區域。區塊元素的最外層是外框 (border)，與內容顯示區域有內側距離 (padding)，與其他元素間則有外部邊界 (margin)。



三、內容顯示區域尺寸：width、height

Box Model 可以使用 width、height 兩個屬性來設定顯示的尺寸，尺寸的單位可以使用 px 像素或是百分比，格式如下：

width：數值
height：數值

範例：

Ex:

```
<style>
  .box1 {
    width: 50%;
    height: 200px;
    color: ■ #FFFFFF;
    background-color: ■ #FF0000;
  }
</style>

<div class="box1">
  文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試文
  字測試文字測試文字測試
</div>
```

四、限制內容顯示區域尺寸：max-width、min-width、max-height、min-height


在預設的狀況下，內容顯示區域會隨著瀏覽器大小來改變顯示範圍，max-width 和 max-height 是用來設定最大可以顯示的寬度和高度，而 min-width 和 min-height 則是用來設定最小可以顯示的寬度和高度，格式如下：

max-width：數值；min-width：數值；
max-height：數值；min-height：數值；

範例：

Ex:

```

<style>
  .box1 {
    max-width: 800px;
    min-width: 600px;
    max-height: 200px;
    min-height: 100px;
    background-color:  #FF0000;
  }
</style>

<div class="box1">
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試文字測試文字測試文字測試文字測試文字
  測試文字測試文字測試文字測試文字測試文字測試
  文字測試
</div>

```

五、溢出範圍處理：overflow


當內容超過顯示區域大小時就會有溢出範圍，可用overflow設定顯示方式，「hidden」是隱藏，「scroll」則是顯示捲軸，格式如下：

overflow：hidden / scroll

範例：


Ex:

```

<style>
  .box1 {
    width: 200px;
    height: 200px;
    overflow: hidden;
    background-color:  #FF0000;
  }

```

```

<style>
    .box1 {
        width: 200px;
        height: 200px;
        overflow: hidden;
        background-color:  #FF0000;
    }
</style>

<div class="box1">
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試
</div>

```

六、內距的設定：padding

內距的屬性是指定內容顯示區與外框間的距離，設定值可以為 1~4 個值，單位為 px 像素，格式如下：

padding：上下左右四個方向的距離值

padding：上下的距離值 左右的距離值

padding：上方的距離值 左右的距離值 下方的距離值

padding：上方的距離值 右方的距離值 下方的距離值 左方的距離值



也可以針對單邊設定內距，格式如下

padding- (top / right / bottom / left)：屬性值

範例：

Ex:

```

<style>
    .box1 {
        width: 200px;
        height: 200px;
        padding: 10px;
        color:  #FFFFFF;
        background-color:  #FF0000;
    }

```

```

<style>
    .box1 {
        width: 200px;
        height: 200px;
        padding: 10px;
        color: ■ #FFFFFF;
        background-color: ■ #FF0000;
    }
</style>

<div class="box1">
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試
</div>

```

七、邊界的設定：margin

邊界的屬性是指定外框與其他區域元素間的距離，設定值可以為 1~4 個值，單位為 px 像素，格式如下：

margin：上下左右四個方向的距離值

margin：上下的距離值 左右的距離值

margin：上方的距離值 左右的距離值 下方的距離值

margin：上方的距離值 右方的距離值 下方的距離值 左方的距離值

也可以針對單邊設定內距，格式如下

margin- (top / right / bottom / left)：屬性值

範例：

Ex:

```

<style>
    .box1 {
        padding: 5px;
        margin: 5px;
        color: ■ #FFFFFF;
        background-color: ■ #FF0000;
    }
    .box2 {
        padding: 5px;
        margin: 5px;
        color: □ #000000;
        background-color: ■ #FFFF00;
    }

```



```

        background-color: #FFFF00;
    }
</style>

<div class="box1">
    文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試
    測試
</div>
<div class="box2">
    文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試文字測試
    測試
</div>

```

八、設定區塊元素水平置中：

將區塊元素的 margin-left、margin-right 左右邊界設為auto，區塊元素即會水平置中。

範例：

Ex:

```

<style>
    .box1 {
        width: 300px;
        padding: 5px;
        margin: 10px auto;
        color: #FFFFFF;
        background-color: #FF0000;
    }
</style>

<div class="box1">
    <p>文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試<br>
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試文字測試文字測試文字
    測試文字測試文字測試文字測試文字測試文字測試
    文字測試文字測試文字測試</p>
</div>

```

九、設定區塊元素與行內元素的狀態：display

區塊元素加入時會自動換行自成一個區域，行內元素則會隨著文字段落流動，使用display屬性，可以讓區塊元素化為行內元素，也能使行內元素顯示為區塊元素，格式如下：

display：顯示狀態

狀態設定	說明
inline	將元素設定為行內元素，元素隨文字段落流動顯示
block	將元素設定為區塊元素，元素前會自動分行成為獨立區域
inline-block	讓區塊元素可以與行內元素一樣隨文字段落流動顯示，但其他屬性仍保有區塊元素的特性
none	隱藏元素，完全不在頁面上顯示，但網頁原始碼仍會呈現

範例：

Ex:

```
<style>
  li {
    display: inline;
    margin-right: 5px;
  }
  .later{
    display: none;
  }
</style>

<ul>
  <li><a href="#home">回到首頁</a></li>
  <li><a href="#about">關於公司</a></li>
  <li class="later"><a href="#product">產品介紹</a></li>
  <li><a href="#contact">聯絡我們</a></li>
</ul>
```

十、顯示或隱藏元素：visibility

使用 display: none 雖然能將該元素隱藏，但它的方式是消失，接下來的元素在顯示時會取代它的位置，若要隱藏元素且保留它的位置則是使用 visibility，格式如下：

visibility: visible (顯示) / hidden (隱藏)

範例：

Ex:

```

<style>
  li {
    display: inline;
    margin-right: 5px;
  }
  .later{
    visibility: hidden;
  }
</style>

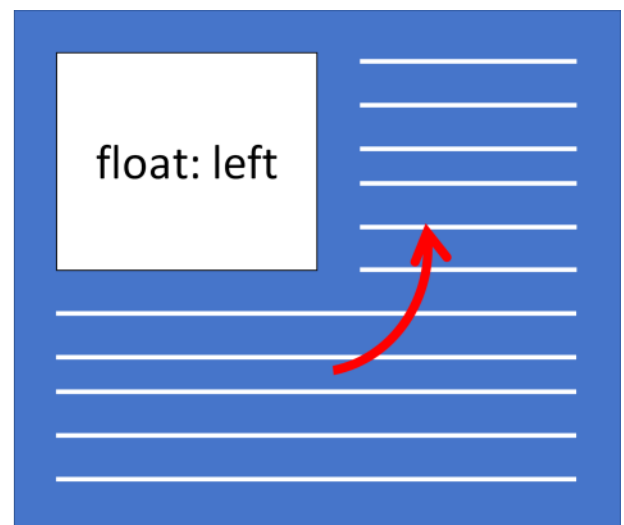
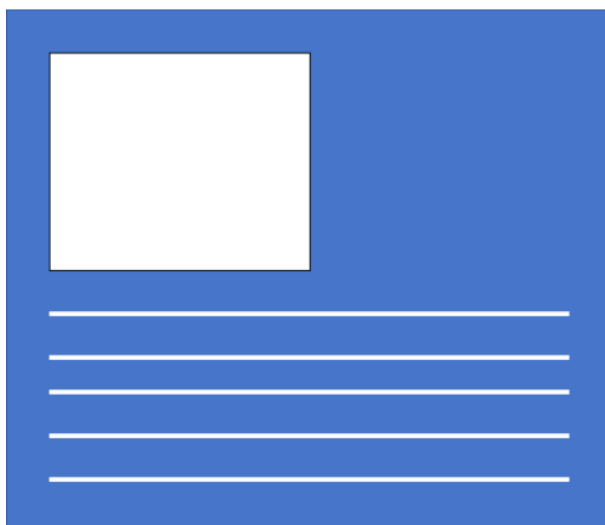
<ul>
  <li><a href="#home">回到首頁</a></li>
  <li><a href="#about">關於公司</a></li>
  <li class="later"><a href="#product">產品介紹</a></li>
  <li><a href="#contact">聯絡我們</a></li>
</ul>

```

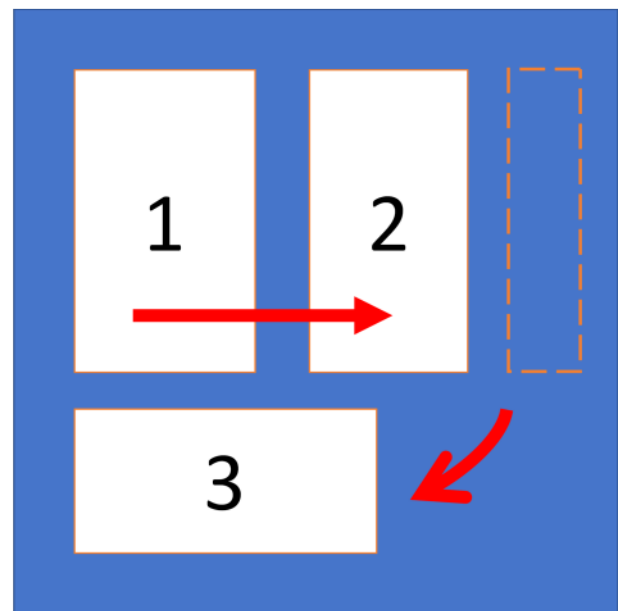
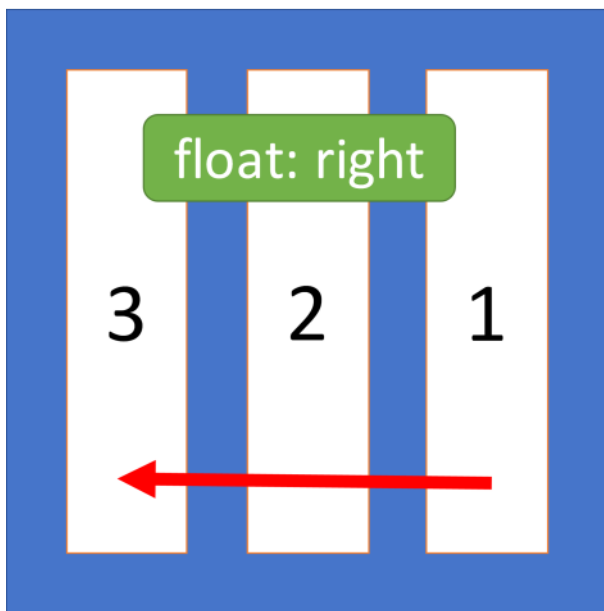
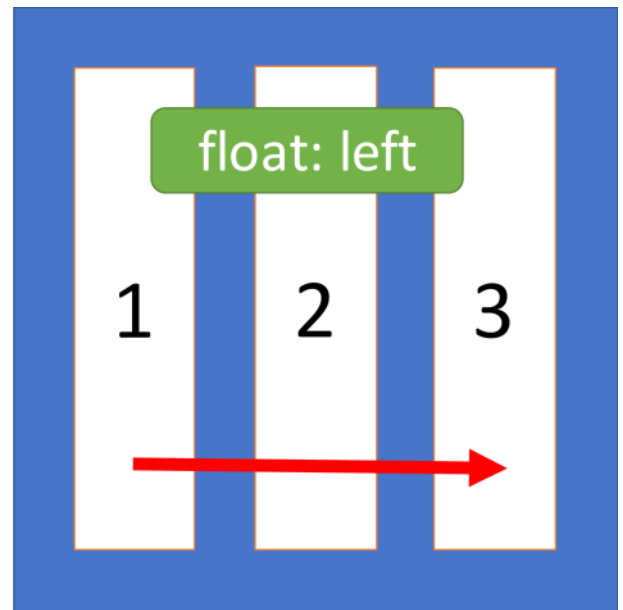
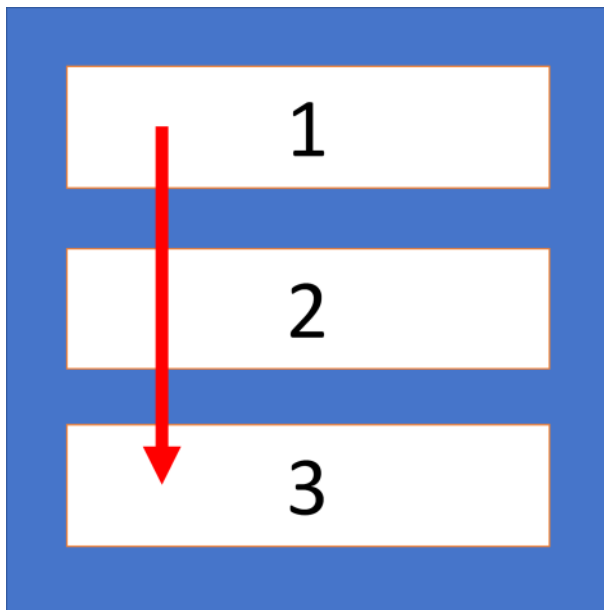
十一、浮動元素：float

在 CSS 中 float 屬性可以將元素設置到所在容器的最左方或最右方成為浮動元素，跟在該元素後方的內容就接著流動到剩下的位置，格式如下：

float: left (左) / right (右) / none (取消)







設定多個區塊元素 float 屬性的顯示方式



範例：

Ex:

```

<style>
  .box1{
    width: 25%;
    height: 100px;
    float: left;
    background-color:  #FC0;
  }
  .box2{
    width: 25%;
    height: 100px;
    float: left;
    background-color:  #FF0;
  }
  .box3{
    width: 25%;
    height: 100px;
    float: left;
    background-color:  #F00;
  }
  .box4{
    width: 40%;
    height: 100px;
    float: left;
    background-color:  #CDD;
  }
</style>

```

```

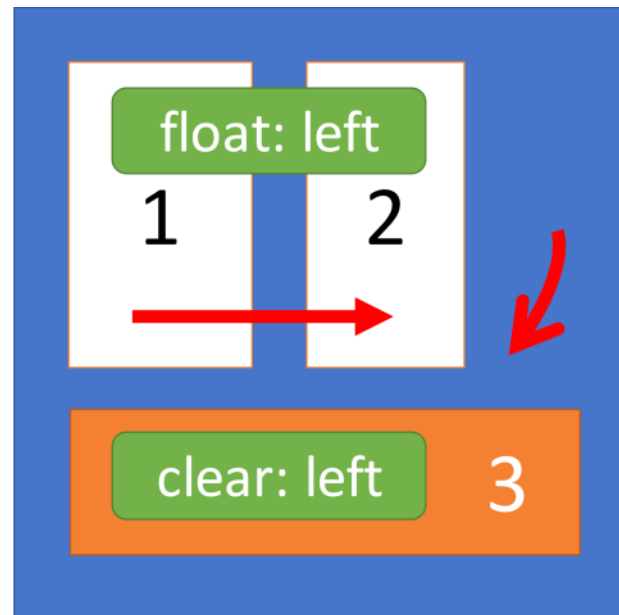
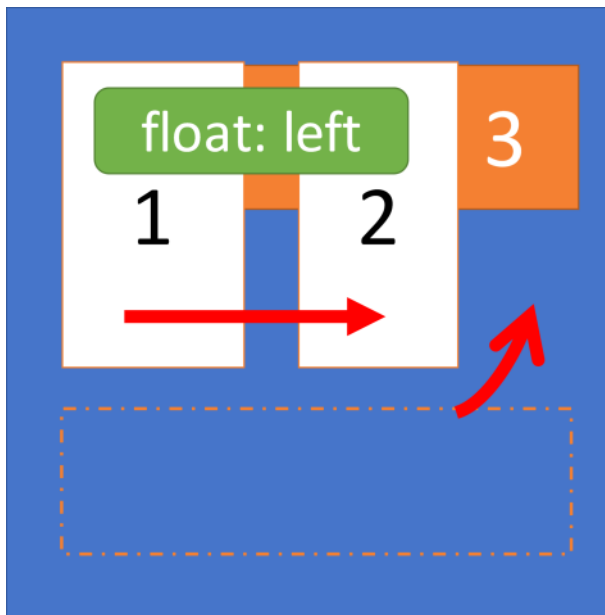
<div class="box1">box1</div>
<div class="box2">box2</div>
<div class="box3">box3</div>
<div class="box4">box4</div>

```

十二、去除浮動元素：clear

在版面中若沒有全部的區塊元素都設定 float 屬性時，有設定的區塊元素會浮動在版面上，沒有設定的區塊就會流動到這些區塊的下層，造成版面錯亂，可以在沒有設定 float 屬性的區塊加上 clear 屬性，即可解除 float 的影響，讓版面正常排列，格式如下：

clear: left (左) / right (右) / both (左右) / none (取消)



範例：

Ex:

```
<style>
  .box1{
    width: 25%;
    height: 100px;
    float: left;
    background-color: #FC0;
  }
  .box2{
    width: 25%;
    height: 100px;
    float: left;
    background-color: #FF0;
  }
  .box3{
    width: 25%;
    height: 100px;
    float: left;
    background-color: #F00;
  }
  .box4{
    clear: both;
    background-color: #CDD;
  }
</style>
```

十三、設定元素位置：position

在 CSS 中 position 屬性可以將元素精確的依指定方式設定由基準點開始的位置，指定位置的屬

性可以使用 top、right、bottom、left，指定的值可以是數值、百分比或是 auto，格式如下：

position: 指定位置方式

指定方式	說明
static	預設方式，不指定顯示位置
relative	以元素現在位置為基準點指定位置
absolute	以元素的父層元素 position 基準為基準點指定位置
fixed	以瀏覽器畫面的左上角為基準點指定位置

範例：

position: relative定位

Ex:

```
<style>
  div{
    width: 400px;
    border: solid orange 1px;
    padding: 5px;
    margin: 10px auto;
  }
  h2{
    color: white;
    background-color: tomato;
    padding: 5px;
    margin: 0px;
    position: relative;
    top: 10px;
    left: 10px;
  }
</style>

<div>
  <h2>關於我們</h2>
  <p>我們跟每個對電腦資訊充滿學習興趣，對未知領域充滿探索熱情的你一樣，努力的提供每個讀者及時的資訊與學習方法，讓學習不只是提昇個人視野的途徑，更是開拓個人生涯的武器。</p>
</div>
```

position: absolute定位

Ex:

```

<style>
  div{
    width: 400px;
    border: solid orange 1px;
    padding: 5px;
    margin: 10px auto;
    position: relative;
  }
  h2{
    color: white;
    background-color: tomato;
    padding: 5px;
    margin: 0px;
    position: absolute;
    top: 0px;
    left: 0px;
  }
</style>

<div>
  <h2>關於我們</h2>
  <p>我們跟每個對電腦資訊充滿學習興趣，對未知
  領域充滿探索熱情的你一樣，努力的提供每個讀者
  及時的資訊與學習方法，讓學習不只是提昇個人視
  野的途徑，更是開拓個人生涯的武器。</p>
</div>

```

position: fixed定位

Ex:

```
<style>
  div{
    width: 400px;
    border: solid ■orange 1px;
    padding: 5px;
    margin: 10px auto;
  }
  h2{
    color: ■white;
    background-color: ■tomato;
    padding: 5px;
    margin: 0px;
    position: fixed;
    bottom: 0px;
    left: 0px;
  }
</style>

<div>
  <h2>關於我們</h2>
  <p>我們跟每個對電腦資訊充滿學習興趣，對未知領域充滿探索熱情的你一樣，努力的提供每個讀者及時的資訊與學習方法，讓學習不只是提昇個人視野的途徑，更是開拓個人生涯的武器。</p>
  <p>我們跟每個對電腦資訊充滿學習興趣，對未知領域充滿探索熱情的你一樣，努力的提供每個讀者及時的資訊與學習方法，讓學習不只是提昇個人視野的途徑，更是開拓個人生涯的武器。</p>
  <p>我們跟每個對電腦資訊充滿學習興趣，對未知領域充滿探索熱情的你一樣，努力的提供每個讀者及時的資訊與學習方法，讓學習不只是提昇個人視野的途徑，更是開拓個人生涯的武器。</p>
  <p>我們跟每個對電腦資訊充滿學習興趣，對未知領域充滿探索熱情的你一樣，努力的提供每個讀者及時的資訊與學習方法，讓學習不只是提昇個人視野的途徑，更是開拓個人生涯的武器。</p>
</div>
```