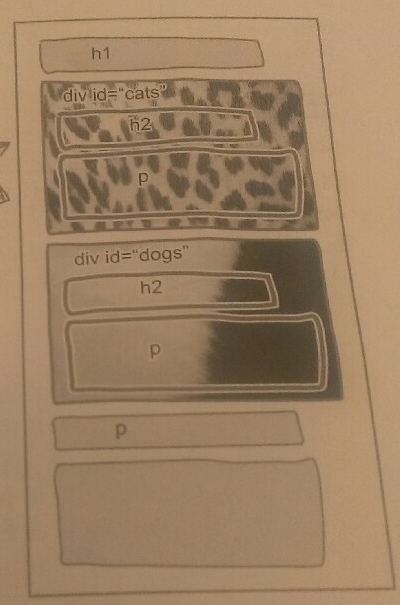
Div與Span

**Div**

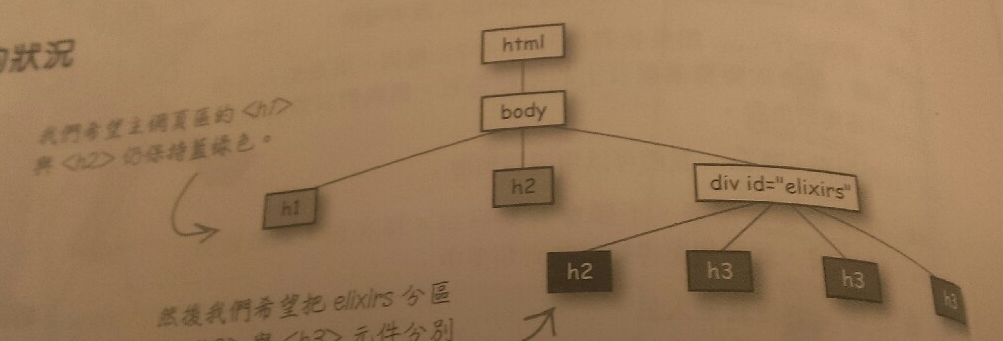
可以把div想像成一個容器，將一群有關係的元件放在一起，將網頁分割成邏輯分區或是群組。例如:在製作寵物清單網頁時，就能用div分成貓類一個區塊、狗類一個區塊。如下圖



使用div能更凸顯網頁架構，有助於讓其他人了解，也較易維護網頁。再者我們也需要架構，以利套用樣式到分區上。

**後代選擇器**

當只想更改某個區塊裡的元件樣式，而不影響區域外的相同元件樣式，例如:下圖中想更改div裡的h2而不影響到body的h2，這種情況下則使用後代選擇器，



寫法如下:

1. 如果div沒設id

div h2{

Color:black;

}

1. div有設id

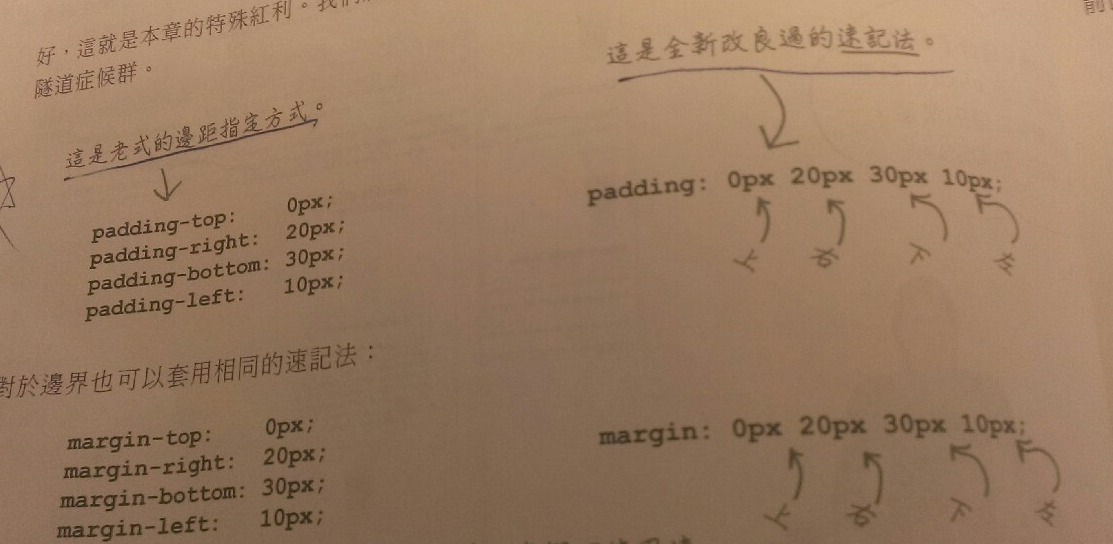
#id h2{

Color:black;

}

**速記法**

很多特性名稱設定值時能使用比較簡潔的速記法如下圖:

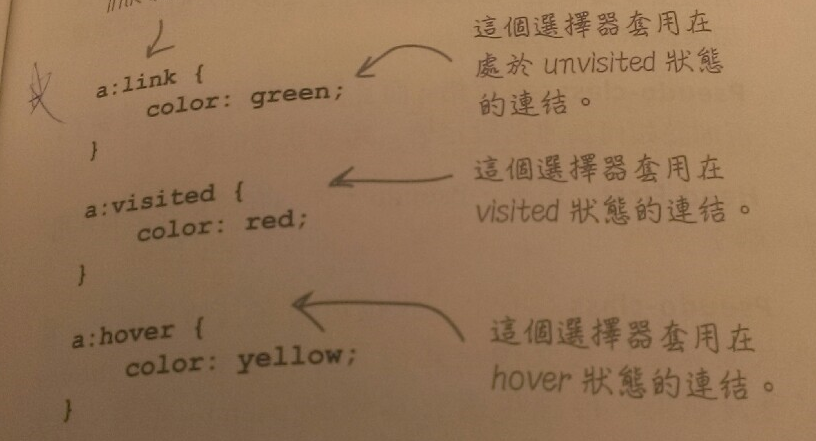


**Span**

類似div，但是是用在行內元件，而div是用於區塊元件，實際的例子就是如果你想讓某段文字有樣式，則可使用對這段文字標籤span元件，利用span設定樣式。

**擬類型(pseudo-classes)**

<a>元件要根據狀態改變樣式，在css中使用擬類型來設定樣式，下圖中為擬類型的用法，分別為當連結沒被造訪、造訪過與當滑鼠移過時的樣式設定。



擬類型其行為像類型，但不是真正的類型，當你開啟一份html檔案時是找不到:visited :link或hover類型。

**階層**

除了網頁設計者編寫的所有樣式表外，還環繞著使用者的樣式表與瀏覽器的預設樣式。當瀏覽器要呈現一個h1元件時，想知道它的font-size特性有以下幾個步驟。

第一步

**收集所有的樣式表:**

當瀏覽器要套用元件上的樣式時，最優先的為作者樣式，然後是讀者樣式，最後才是瀏覽器的預設樣式。

第二步

**尋找所有符合的宣告:**

逐一檢查全部的樣式表，將任何對應<h1> font-size特性的規則找出來。

第三步

**為所有相符宣告作排列:**

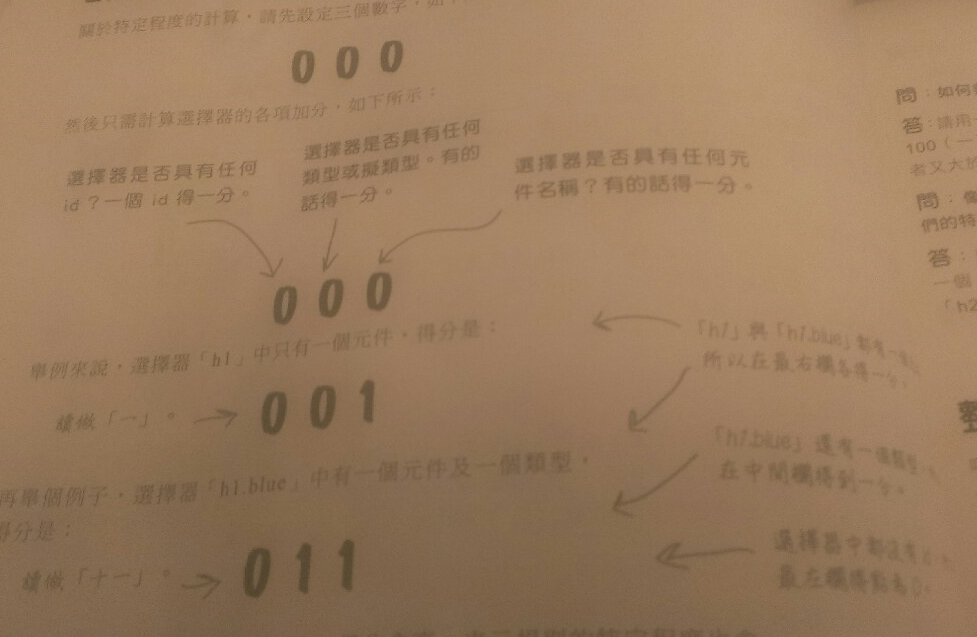
將相符的規則，依作者、讀者、瀏覽器,，依序作排列。其中讀者可在css特性中加上!important覆寫作者的樣式，這樣特性在排列時就會跑到第一位。

第四步

**依宣告特定程度作排列:**

如果規則能夠精準地塞選出某個元件，直覺上就覺得這項規則的特定程度比

較高，也可使用數字計算，如下圖



第五步

**有衝突的規則再依出現在樣式表的順序排列:**

樣式表較晚出現的可覆寫先前的任何規則。