

CSS整理與延伸

蔡偉駿 (Jimmy Tsai)

溫馨提醒：

請準備一份HTML檔案內容如下，方便你回答課程Quiz

```
<!DOCTYPE html>
<html>
<head>
  <style>
  </style>
</head>
<body>
</body>
</html>
```

回答Quiz很重要！

回答Quiz很重要！

回答Quiz很重要！

CSS是什麼？

CSS = **C**ascading **S**tyle **S**heets (階層式樣式表)

CSS 誕生於 17 December 1996。

CSS目前出到第3版（CSS3），第4版正在持續規劃中；
本課程使用的CSS語法為CSS3。

CSS是誰發明的？



Håkon Wium Lie (哈肯·維姆·萊)

16 July 1965 - Now

出身於挪威的電腦科學家。

是CSS的發起人。

現為Opera Software的首席技術長。

資料來源：維基百科 https://en.wikipedia.org/wiki/H%C3%A5kon_Wium_Lie

CSS的規格是由哪個組織制定的？

是由World Wide Web Consortium (W3C)所制定的。

官方網站：<https://www.w3.org/>

W3C規定了網頁開發的各種工程規格 (不只CSS，還包含HTML結構和許多層面)，而各家瀏覽器要盡可能地遵循這個規格來開發。

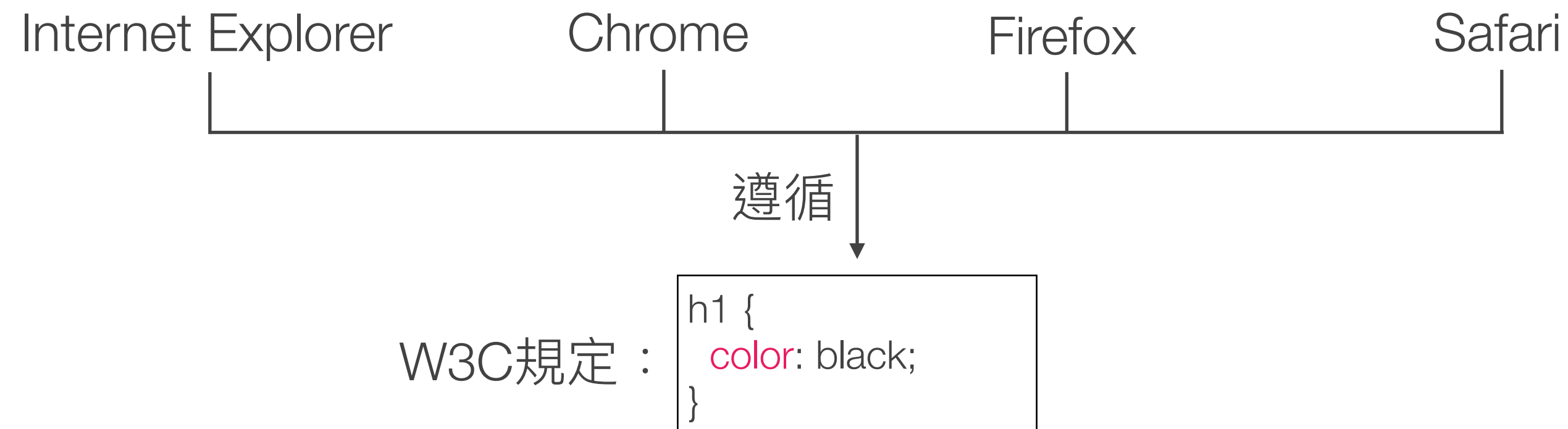
為什麼需要這樣的組織呢？很簡單，做工程如果沒有遵循統一規格，會造成開發上的混亂和不便，就像蓋房子的所有建商都要遵守政府的安全法規來蓋，房子的安全性才有保障。

CSS的規格是由哪個組織制定的？

假設你在沒有W3C的世界，則每個瀏覽器廠商可能會自己出一種字體顏色的寫法，以設成黑色為例：



回到有W3C的世界



在CSS中如何下註解（comment）？

可以使用 `/*` 和 `*/` 來下註解（comment），所有被夾在 `/*` 和 `*/` 中的文字都會被瀏覽器忽略，範例如下：

單行註解

```
h1 {  
  font-size: 60px; /* 這是單行註解 */  
}
```

多行註解

```
h1 {  
  /*  
    這是第1行註解  
    這是第2行註解  
  */  
  font-size: 60px;  
}
```

Quiz 1

請在專案1中隨便新增CSS的註解，並確認註解沒有影響到CSS程式碼的執行。

CSS還可以寫在哪裡？

你也可以在HTML標籤中使用`style`屬性來寫入CSS，透過這種方式寫入的CSS稱為`Inline style` (行內樣式) 範例如下：

使用`style`屬性

```
<p>...</p>  
<p style="font-size: 20px; color: red;">...</p>  
<p>...</p>
```

→ CSS規則只有對這個`<p>`有效，對其他的`<p>`是無效的；通常只有在特別的情境才會單獨對特定標籤設定`style`，大部份時候請盡量用CSS程式碼來寫。

使用CSS程式碼

```
p {  
  font-size: 20px;  
  color: red;  
}
```

→ CSS規則對所有的`<p>`都有效。

CSS還可以寫在哪裡？

style的設定值若和CSS程式碼相衝突，則以**style**的設定覆蓋掉CSS程式碼的設定。

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p {
      font-size: 30px;
      color: white;
    }
  </style>
</head>
<body>
  <p>...</p>
  <p style="color: red;">...</p>
  <p>...</p>
</body>
</html>
```

只有第2個<p>的字體顏色是red，其他<p>的字體顏色是white。

Quiz 2

請使用style屬性將專案1中的<p>的字體大小改成40px。

什麼是襯線體，什麼是無襯線體？

襯線體 = serif

S 襯

1. 每筆畫尾端有襯線 (小裝飾線)。
2. 每筆畫線條粗細較有變化。
3. 中文字體範例：新細明體、宋體。
4. 英文字體範例：Times New Roman。

無襯線體 = sans-serif

S 襯

(sans來自法文，代表“無”或“沒有”的意思。)

1. 每筆畫尾端無襯線。
2. 每筆畫線條粗細較一致。
3. 中文字體範例：微軟正黑體、蘋方。
4. 英文字體範例：Arial、Helvetica。

CSS字體設定細節

```
* {  
  font-family: Arial, "Times New Roman", sans-serif;  
}
```

value	description
family-name	特定字體，例如Arial, Times New Roman。若字體名稱有空白，則使用雙引號 (") 包住。
generic-family	通用字體。共有5種可選，分別是sans-serif (無襯線體)、serif (襯線體)、monospace (等寬體)、cursive (手寫體) 和 fantasy (幻想體)。

CSS字體設定細節

```
* {  
  font-family: Arial, "Times New Roman", sans-serif;  
}
```

每個字體用“，”分隔開來；以上述設定為例，瀏覽器會優先使用Arial字體，若使用者的電腦沒有安裝Arial字體對應的字型檔，則使用Times New Roman字體，若使用者的電腦內沒有安裝Times New Roman字體對應的字型檔，則使用sans-serif字體。

value	description
family-name	特定字體，例如Arial, Times New Roman。若字體名稱有空白，則使用雙引號 (") 包住。
generic-family	通用字體。共有5種可選，分別是sans-serif (無襯線體)、serif (襯線體)、monospace (等寬體)、cursive (手寫體) 和 fantasy (幻想體)。

CSS字體設定細節

```
* {  
  font-family: Arial, "Times New Roman", sans-serif;  
}
```

隸屬於無襯線體這個大家庭的字體這麼多，到底會用哪一種呢？答案是取決於使用者的作業系統。每個作業系統預設的sans-serif不一樣，例如在Mac上，sans-serif預設會使用Helvetica字體，因此瀏覽器會去找Helvetica字體對應的字型檔。

value	description
family-name	特定字體，例如Arial, Times New Roman。若字體名稱有空白，則使用雙引號 (") 包住。
generic-family	通用字體。共有5種可選，分別是sans-serif (無襯線體)、serif (襯線體)、monospace (等寬體)、cursive (手寫體) 和 fantasy (幻想體)。

Quiz 3

請將專案1中CSS的font-family值改成"serif"，看有什麼效果。

字體 vs 字型 vs 字型家族

字體 (Typeface) 是文字外觀設計上的理念和規範，強調的是設計層面的規格。

例如：Arial

字型 (Font) 是實作字體的**產品**，其實就是**字型檔案**，只要是**不同的字型檔就是不同的字型**。下表是以Windows 10上安裝的Arial字體對應的字型檔案為例：

所屬字體	字型名稱	字型檔案		
Arial	Arial	arial.ttf	Arial正體	這4個字型檔都實作了Arial的規範，都屬於Arial字體，差別只在正體、斜體、粗體、粗斜體這種風格(style)上的差異。
	Arial Italic	ariali.ttf	Arial斜體	
	Arial Bold	arialbd.ttf	Arial粗體	
	Arial Bold Italic	arialbi.ttf	Arial粗斜體	

字型家族 (Font-family) 是實作同樣字體的一群字型所構成的集合。

字型家族 (Font-family) 其實就是 **字體 (Typeface)**，這兩者是相等的。

以上表為例：Arial這個字型家族 (集合) 包含了字型Arial，Arial Italic，Arial Bold，和Arial Bold Italic。

Quiz 4

以下表為例，請問Arial Italic和Arial Bold是同一個字型 (Font) 嗎？

所屬字體	字型名稱	字型檔案
Arial	Arial	arial.ttf
	Arial Italic	ariali.ttf
	Arial Bold	arialbd.ttf
	Arial Bold Italic	arialbi.ttf

Quiz 4

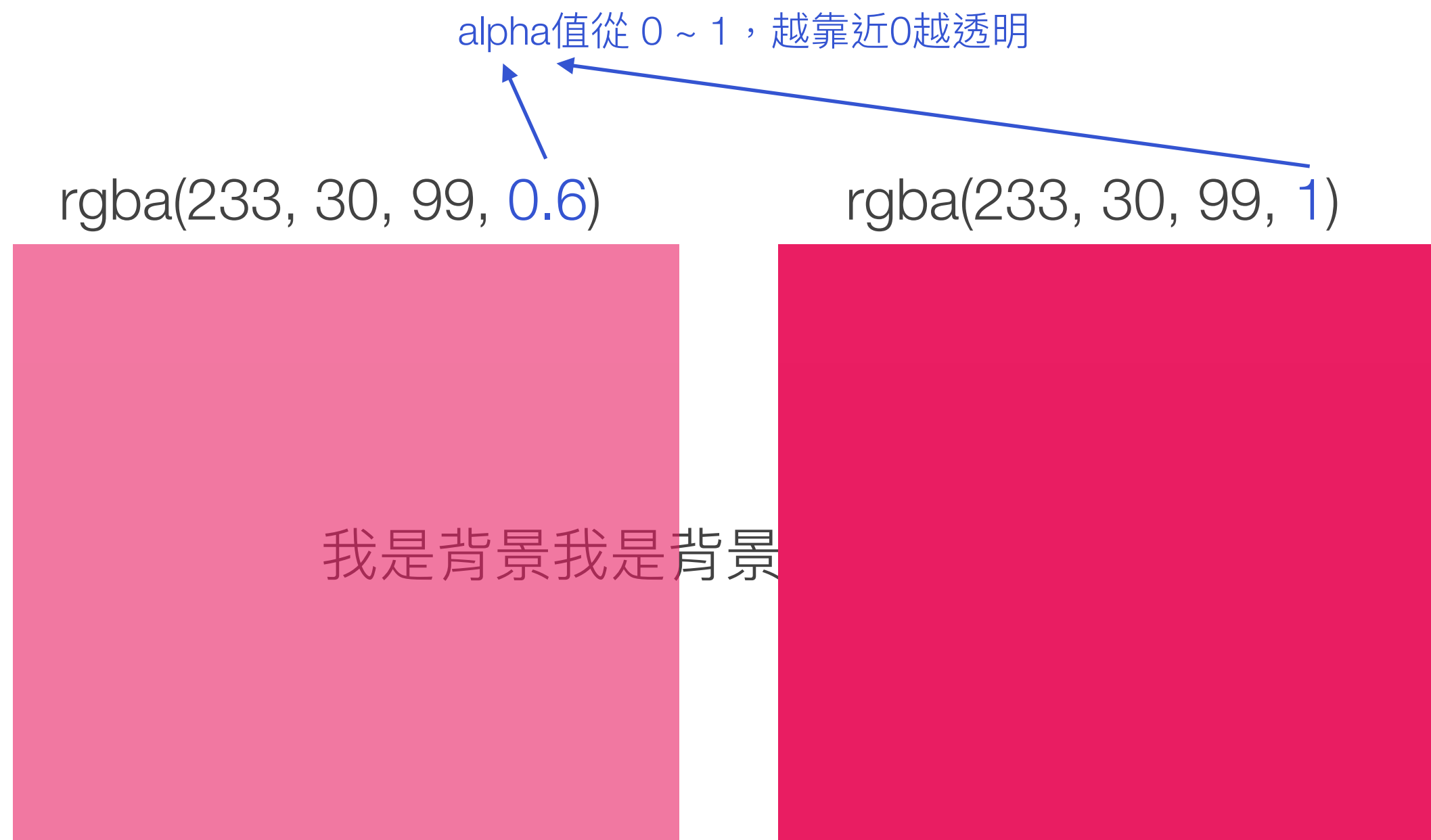
以下表為例，請問Arial Italic和Arial Bold是同一個字型 (Font) 嗎？

所屬字體	字型名稱	字型檔案
Arial	Arial	arial.ttf
	Arial Italic	ariali.ttf
	Arial Bold	arialbd.ttf
	Arial Bold Italic	arialbi.ttf

Ans: No

CSS中的顏色寫法有哪些？

1. 英文名稱，例如 red，green，blue，pink，...等等。名稱和顏色的對照表可參考：https://www.w3schools.com/colors/colors_names.asp
2. rgb (red, **g**reen, **b**lue)，例如 rgb(233, 30, 99)
3. hex (hexadecimal, 16進位)，例如 #e91e63
4. hsl (**h**ue, **s**aturation, **l**ightness)，例如 hsl(340, 82%, 52%)，不常用，本課程不介紹。
5. rgba，例如 rgba(233, 30, 99, 0.6)。rgba中的“a”是alpha值，代表不透明度。



CSS中的顏色寫法有哪些？

rgb 和 hex 的關係：

```
input[type="button"] {  
  background: rgb(233, 30, 99);  
}
```

=

```
input[type="button"] {  
  background: #e91e63;  
}
```

↓
最小值是0，最大是255

請用hex寫法較佳，會讓程式碼短一點，檔案小一點。

寫法	範例
rgb	rgb(233, 30, 99)
hex	#e91e63

10進位	16進位
233	e9
30	1e
99	63

你不用自己算，可以用Chrome的檢查工具幫你換算 (DEMO)

CSS中的顏色寫法有哪些？

顏色以hex表示時會是3個數字（每個數字都是2碼）共6碼寫在一起，如果每個數字本身都重複的話，可將6碼簡寫成3碼即可。例如：

#aabbcc = #abc

#3344ee = #34e

#222222 = #222

#ff0000 = #f00

#ffffff = #fff

Quiz 5

- (1) 猜猜看rgb(255, 255, 255)是什麼顏色
- (2) 猜猜看rgb(0, 0, 0)是什麼顏色
- (3) 請試著不用Chrome檢查工具，自己手算一次
rgb(233, 30, 99) 轉 #e91e63

10進位	16進位
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	a
11	b
12	c
13	d
14	e
15	f

Quiz 5

- (1) 猜猜看rgb(255, 255, 255)是什麼顏色
- (2) 猜猜看rgb(0, 0, 0)是什麼顏色
- (3) 請試著不用Chrome檢查工具，自己手算一次
rgb(233, 30, 99) 轉 #e91e63

Ans: (1) 白色 (2) 黑色

(3) 計算如下：

10進位

16進位

$$233 = 14 \times 16^1 + 9 \times 16^0 = e9$$

$$30 = 1 \times 16^1 + 14 \times 16^0 = 1e$$

$$99 = 6 \times 16^1 + 3 \times 16^0 = 63$$

10進位	16進位
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	a
11	b
12	c
13	d
14	e
15	f

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

我DEMO一下這個問題，幫助大家回憶

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

這個問題應該要換一個問法：

當一個元素的某個CSS屬性(例如: <h1>的color)沒有被我們設定時，瀏覽器該如何決定它的值？

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

瀏覽器在決定<h1>的color要套用什麼值時，會參考2個樣式表 (樣式表 = CSS程式碼) 再加上CSS Initial：

優先權高 1. 開發者 (我們自己) 定義的樣式表

```
<style>
h1 { ... }
</style>
```

2. 瀏覽器定義的樣式表，叫做User Agent Stylesheet，簡稱UA，內建在瀏覽器中。你可以把UA想像成這樣

```
h1 { ... }
p { ... }
...
```

優先權低 3. W3C定義的初始值表格，叫做CSS Initial，也會內建在瀏覽器中。**CSS Initial**會定義所有**CSS屬性的初始值**，而且每個屬性的初始值和元素無關。你可以把CSS Initial想像成一個類似這樣的表格：

屬性	Initial
color	black
margin	0
...	

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

瀏覽器在決定<h1>的color要套用什麼值時，會參考2個樣式表 (樣式表 = CSS程式碼) 再加上CSS Initial：

優先權高

1. 開發者 (我們自己) 定義的樣式表

```
<style>
h1 { ... }
</style>
```

2. 瀏覽器定義的樣式表，叫做User Agent Stylesheet，簡稱UA，內建在瀏覽器中。你可以把UA想像成這樣

```
h1 { ... }
p { ... }
...
```

優先權低

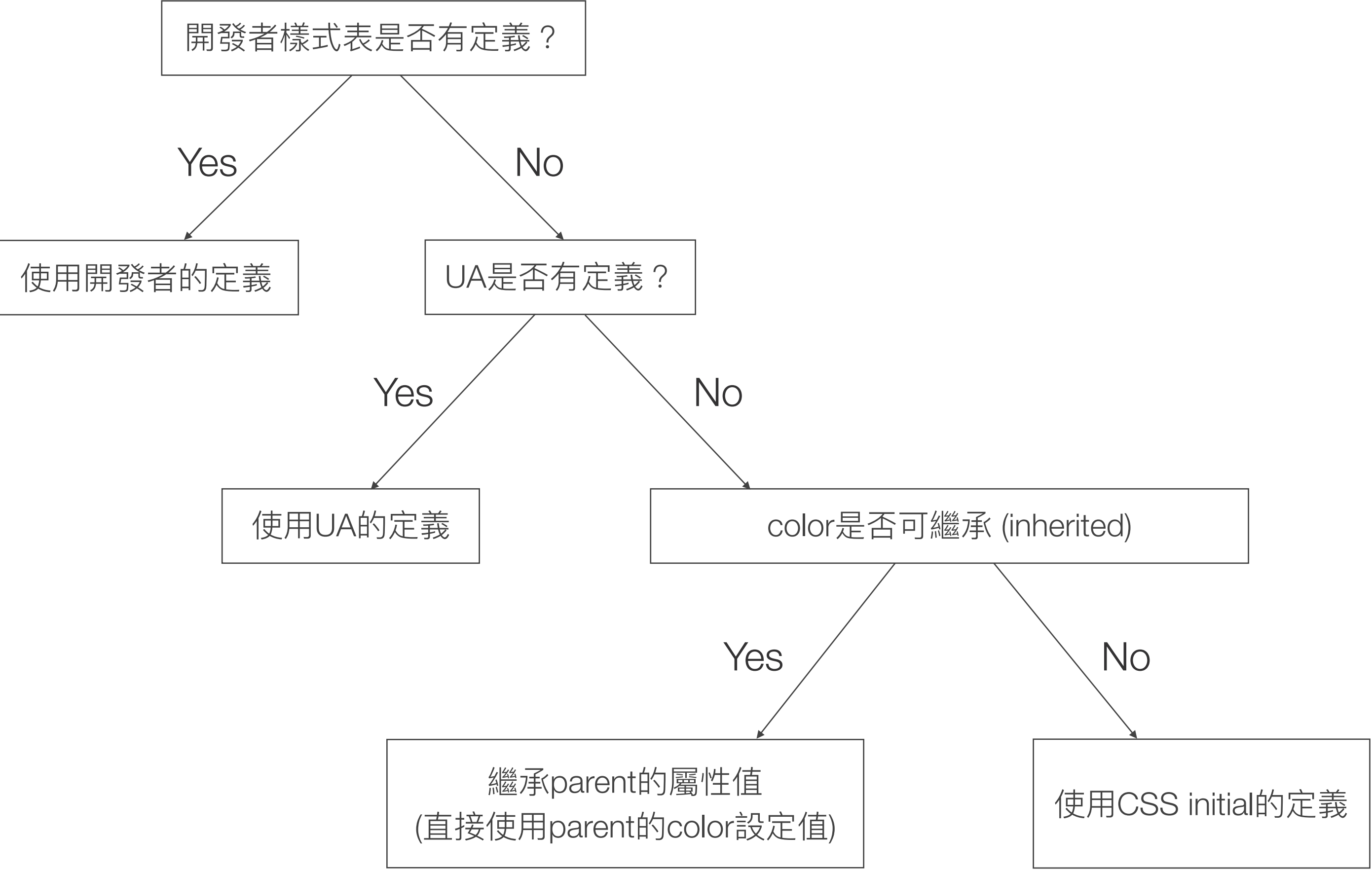
3. W3C定義的初始值表格，叫做CSS Initial，也會內建在瀏覽器中。CSS Initial會定義所有CSS屬性的初始值，而且每個屬性的初始值和元素無關。你可以把CSS Initial想像成一個類似這樣的表格：

屬性	Initial
color	black
margin	0
...	

可以從以下W3C網站查到margin屬性的initial值為0
<https://drafts.csswg.org/css-box-3/>

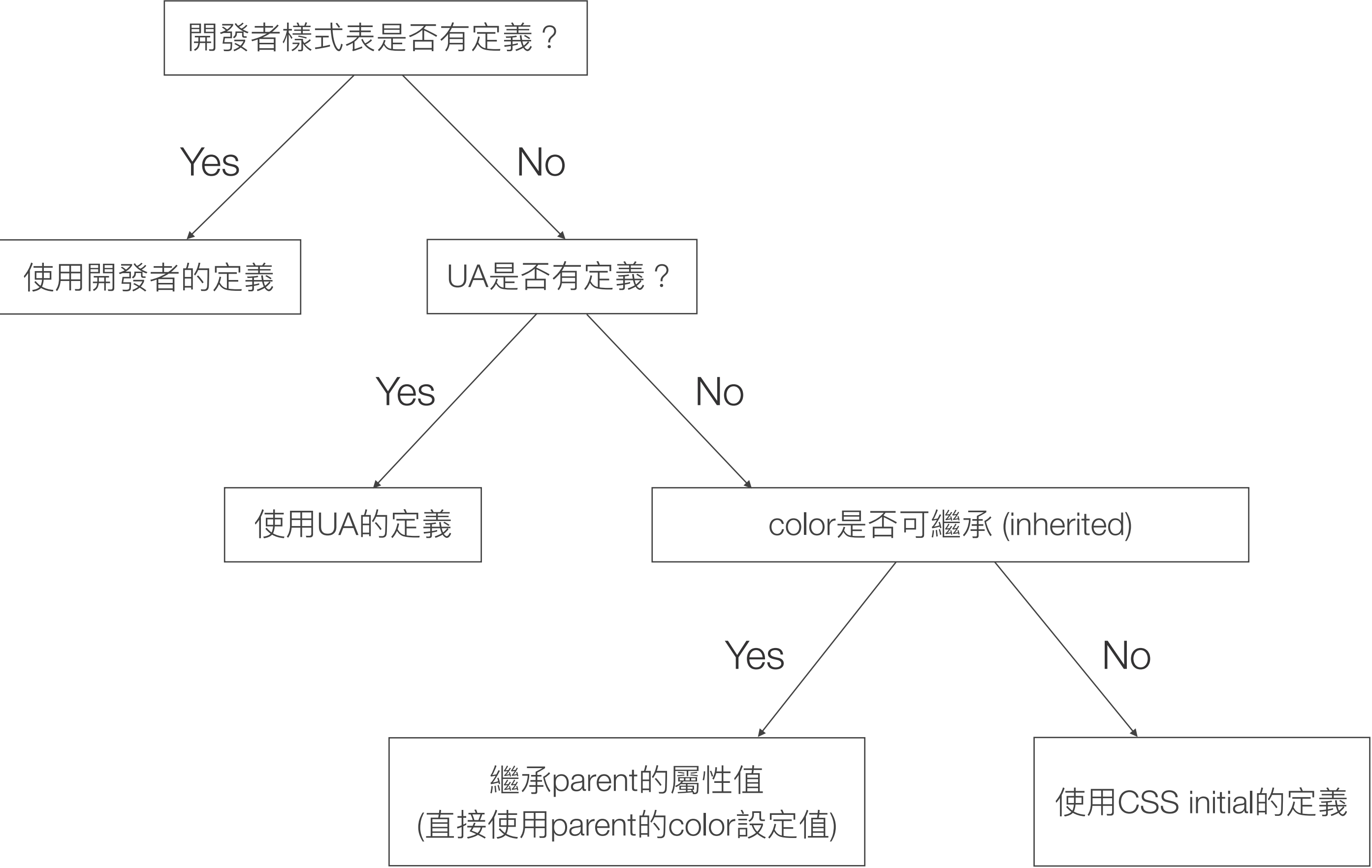
專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

瀏覽器套用CSS樣式的規則 (流程) 如下，我們就以<h1>的color為例 (其他屬性同理)：



專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

瀏覽器套用CSS樣式的規則 (流程) 如下，我們就以<h1>的color為例 (其他屬性同理)：



至於CSS屬性是否為可繼承 (inherited)，你可以參考下面這個網站來查詢 (我順便DEMO要怎麼查)：

<https://www.w3schools.com/cssref/default.asp>

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

專案1中<h1>和<p>的color套用路線：

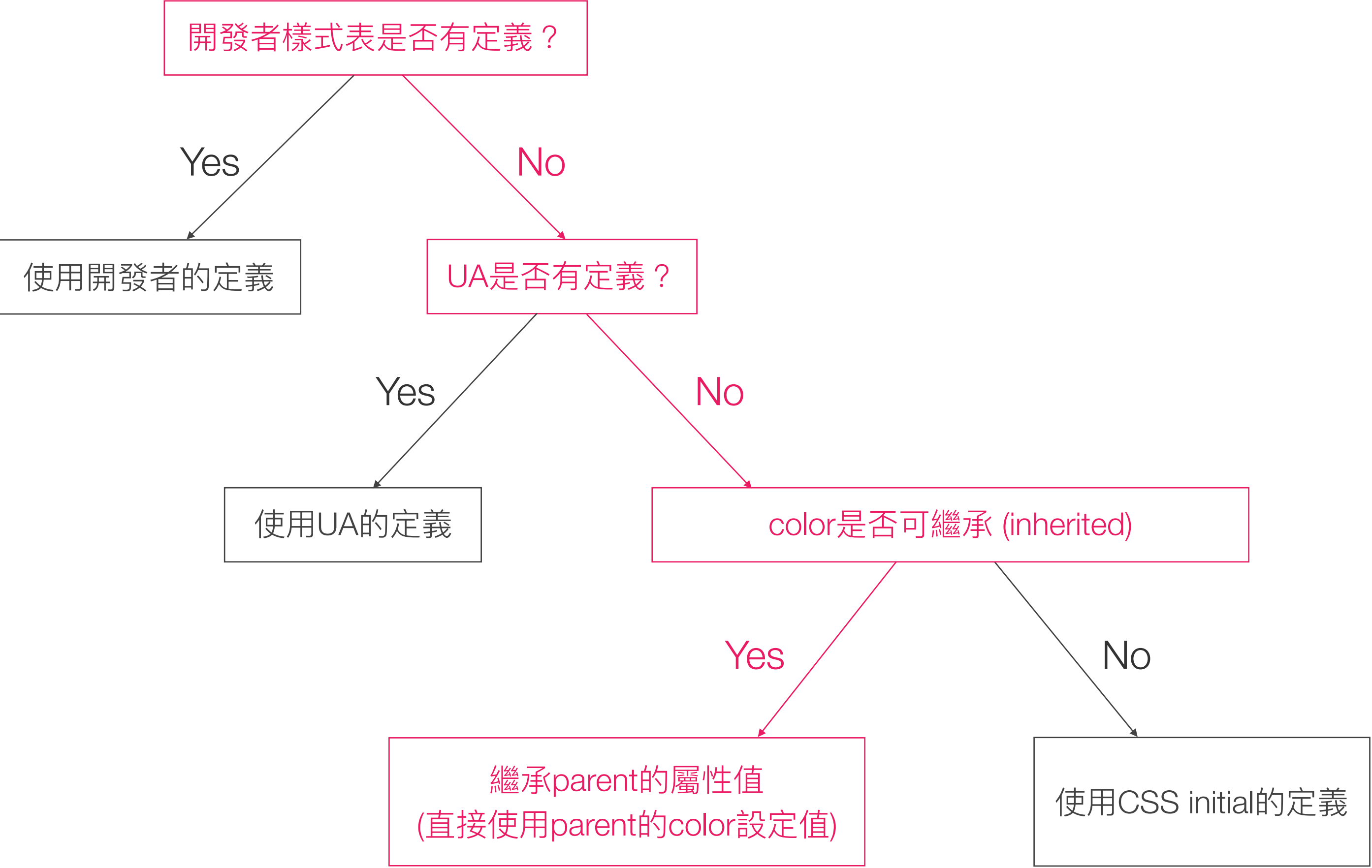
DEMO

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

專案1中<h1>和<p>的color套用路線：

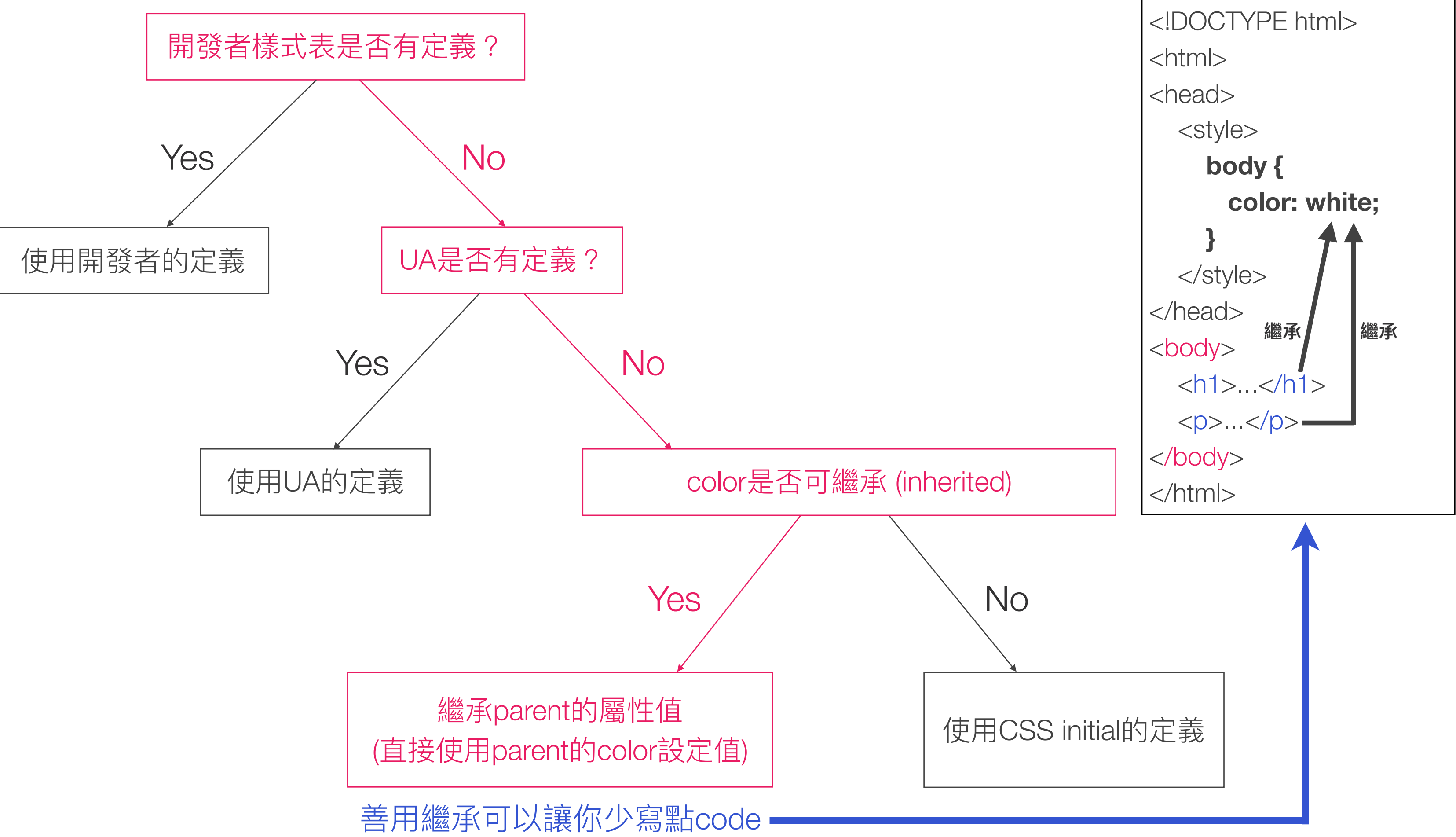
專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

專案1中<h1>和<p>的color套用路線：



專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

專案1中<h1>和<p>的color套用路線：



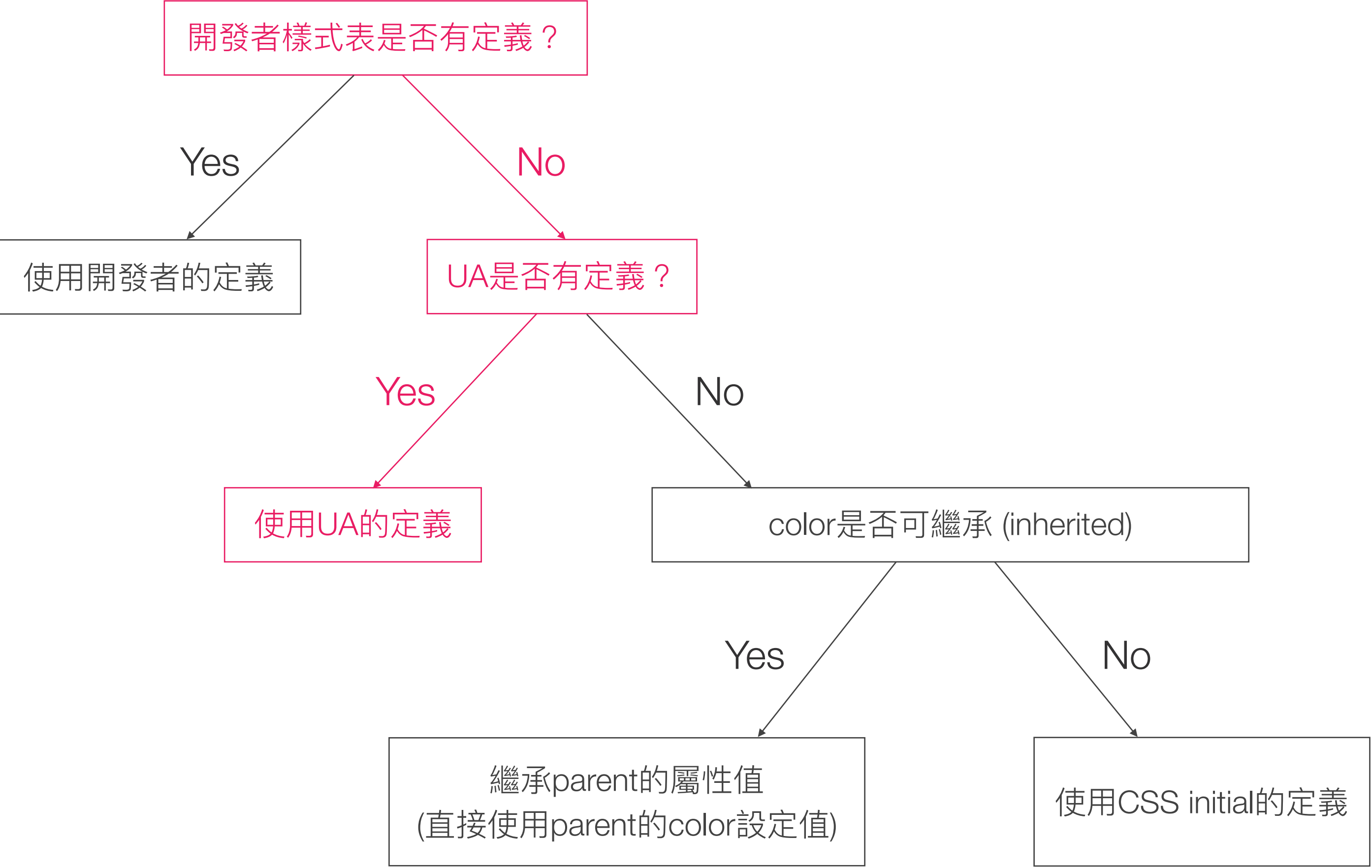
專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

專案1中<input type="button">的color套用路線(在我們未設定它的color之前)：

DEMO

專案1中設定<body>的color屬性，為何對<h1>和<p>有效但對<input type="button">無效？

專案1中<input type="button">的color套用路線(在我們未設定它的color之前)：



Quiz 6

(1) 請直接執行以下程式碼，看<h1>的border是否會被畫出來？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      border: 1px solid red;
    }
  </style>
</head>
<body>
  <h1>光速買</h1>
</body>
</html>
```

(2) 請使用 <https://www.w3schools.com/cssref/default.asp> 查詢border是否為可繼承？

(3) 請在上述程式碼的CSS中添加

```
h1 {
  border: inherit;
}
```

看會有什麼效果。

Quiz 6

(1) 請直接執行以下程式碼，看<h1>的border是否會被畫出來？Ans: No

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      border: 1px solid red;
    }
  </style>
</head>
<body>
  <h1>光速買</h1>
</body>
</html>
```

(2) 請使用 <https://www.w3schools.com/cssref/default.asp> 查詢border是否為可繼承？

Ans: No

(3) 請在上述程式碼的CSS中添加

```
h1 {
  border: inherit;
}
```

看會有什麼效果。

Ans: <h1>的border會被畫出來而且跟<body>的border一模一樣。

Quiz 6

(1) 請直接執行以下程式碼，看<h1>的border是否會被畫出來？Ans: No

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      border: 1px solid red;
    }
  </style>
</head>
<body>
  <h1>光速買</h1>
</body>
</html>
```

(2) 請使用 <https://www.w3schools.com/cssref/default.asp> 查詢border是否為可繼承？

Ans: No

(3) 請在上述程式碼的CSS中添加

```
h1 {
  border: inherit;
}
```

看會有什麼效果。

將border設成**inherit**，這樣就會強迫它去繼承parent的值

Ans: <h1>的border會被畫出來而且跟<body>的border一模一樣。

CSS中的長度單位有哪些？

絕對單位 (物理單位)：如果你關心的是網頁印到紙張上的呈現，那你才會需要使用下表的絕對單位

單位	說明
cm	centimeters 公分
mm	millimeters 公釐
pt	points
pc	picas
in	inches 英吋， $1\text{in} = 2.54\text{cm} = 72\text{pt} = 6\text{pc}$

不論是本課程或大部分的網頁開發，通常沒有印到紙張的需求，因此幾乎不會使用上述的絕對單位。

CSS中的長度單位有哪些？

魔法單位 (magic unit)：如果你關心的是網頁在螢幕上的呈現，那你最好使用魔法單位px (參考：<https://www.w3.org/Style/Examples/007/units.en.html>)

1px的定義為，在不同裝置各自的常用距離下，1px寬的線必須夠小夠清晰。

單位	說明
px	pixel 像素，px不是一個固定的物理長度，px會隨著不同電子裝置的螢幕 (手機、平板電腦、筆電、桌電、...等) 而有不同的物理長度。工程師會想辦法讓1px的線寬在不同裝置各自的常用距離下看起來又小又清晰。例如，你看手機的距離(約25公分)和你看筆電的距離(約45公分)不會一樣，因此1px在這2個裝置上不一定一樣長，但保證在這2個裝置的常用距離下看起來都又小又清晰。 順帶一提，瀏覽器上預設的font-size通常是 16px 。

用px就對了。

不論是開發手機、平板、或桌面版的網頁，螢幕大小或解析度之類的事通通不用擔心，**px**都幫你搞定了！它可以保證看到的大小都很合理。

CSS中的長度單位有哪些？

相對單位：不論網頁在螢幕上或在紙張上呈現，相對單位都適用。

單位	說明
em	以目前元素的font-size為基準做倍數相乘（但有例外，見Quiz）。
%	以parent的尺寸做百分比相乘，例如parent的字體大小、parent的寬度...等，至於是哪種尺寸，取決於你要設定哪種CSS屬性（日後會看到範例）。

不論是本課程或大部分的網頁開發，通常使用**px**搭配**%**即可（比較講究的網站，會再搭配**em**使用）。

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

母元素
子元素

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

母元素

子元素

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：

<p>

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：

<p>

2em

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

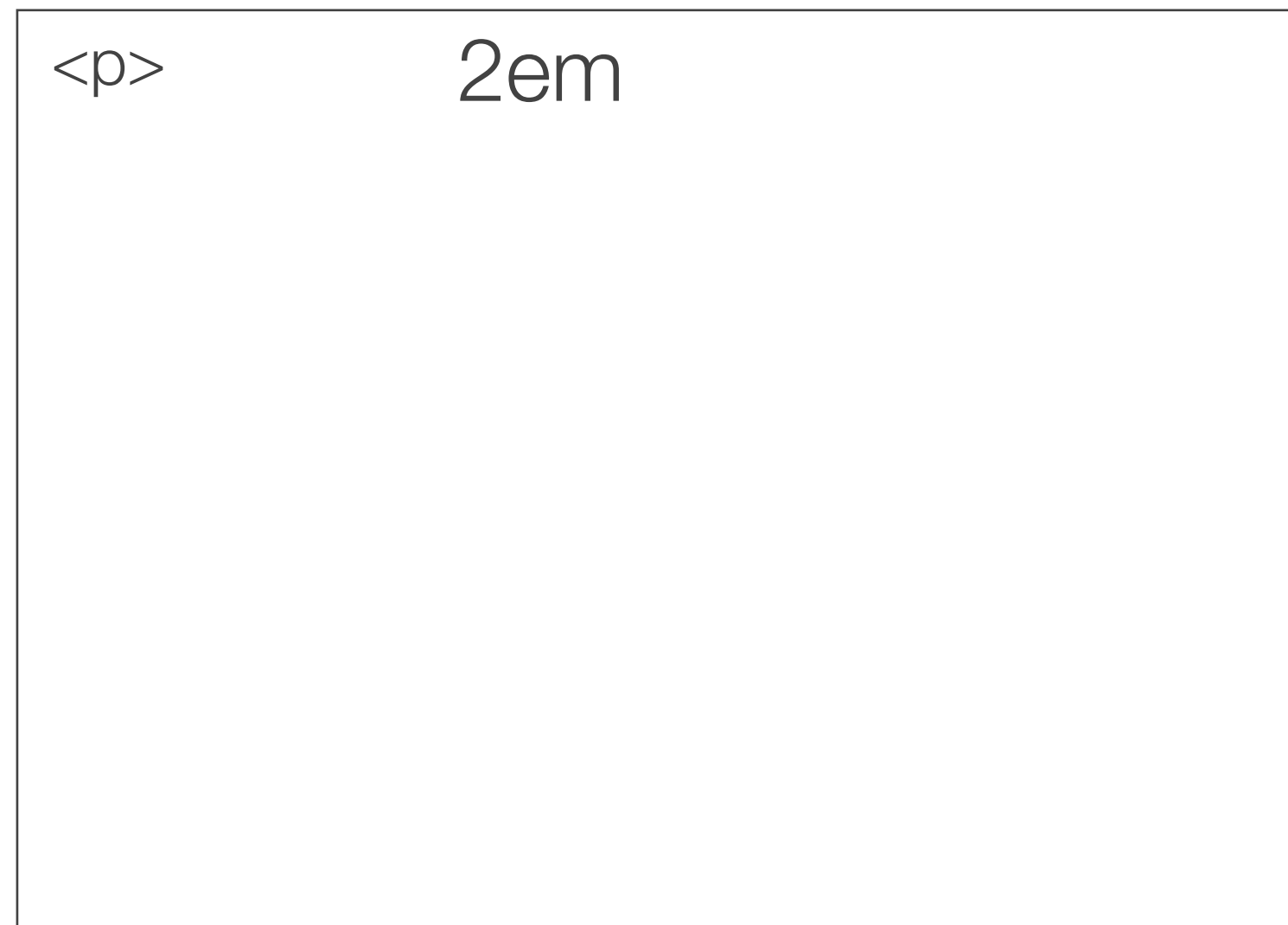
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

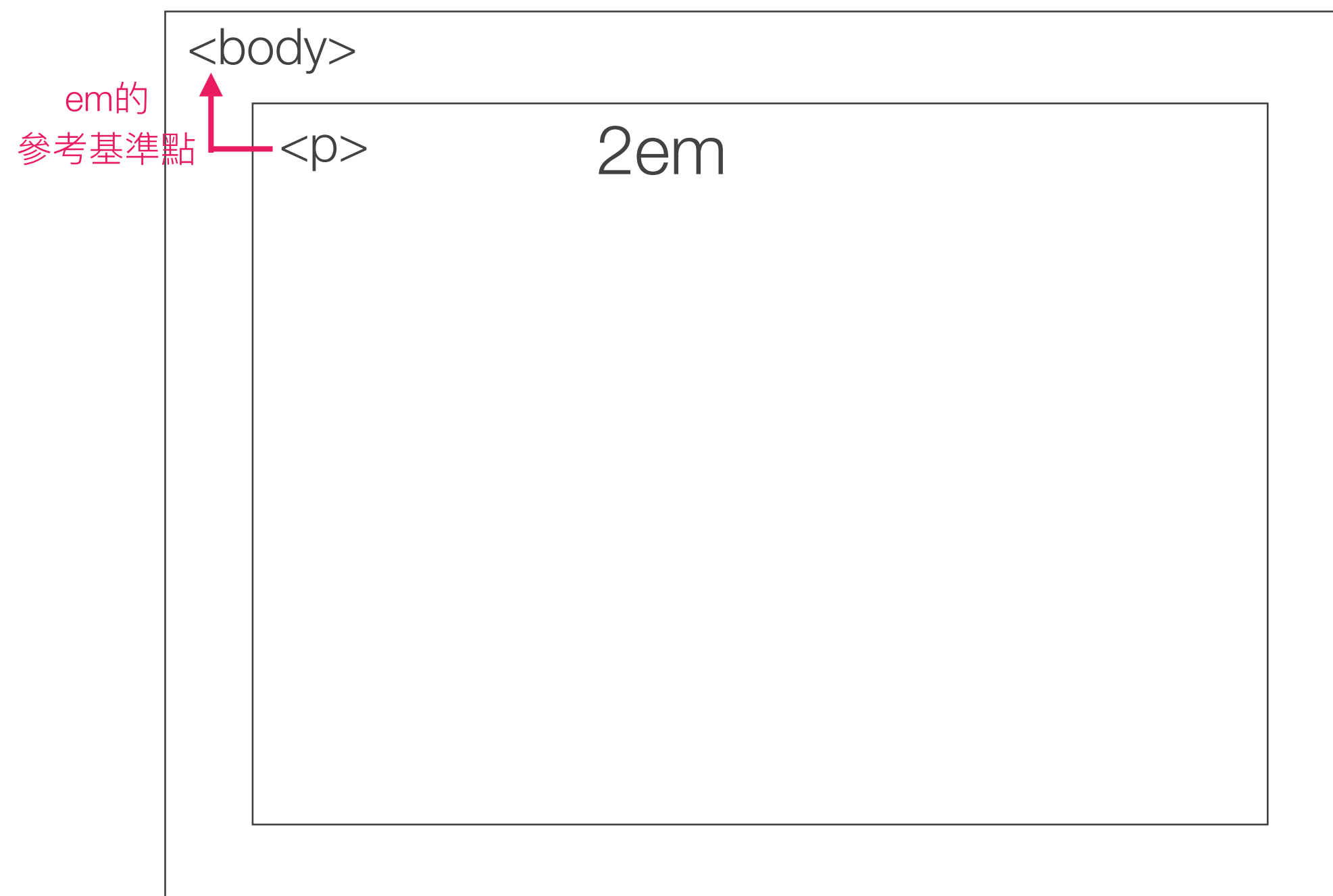
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

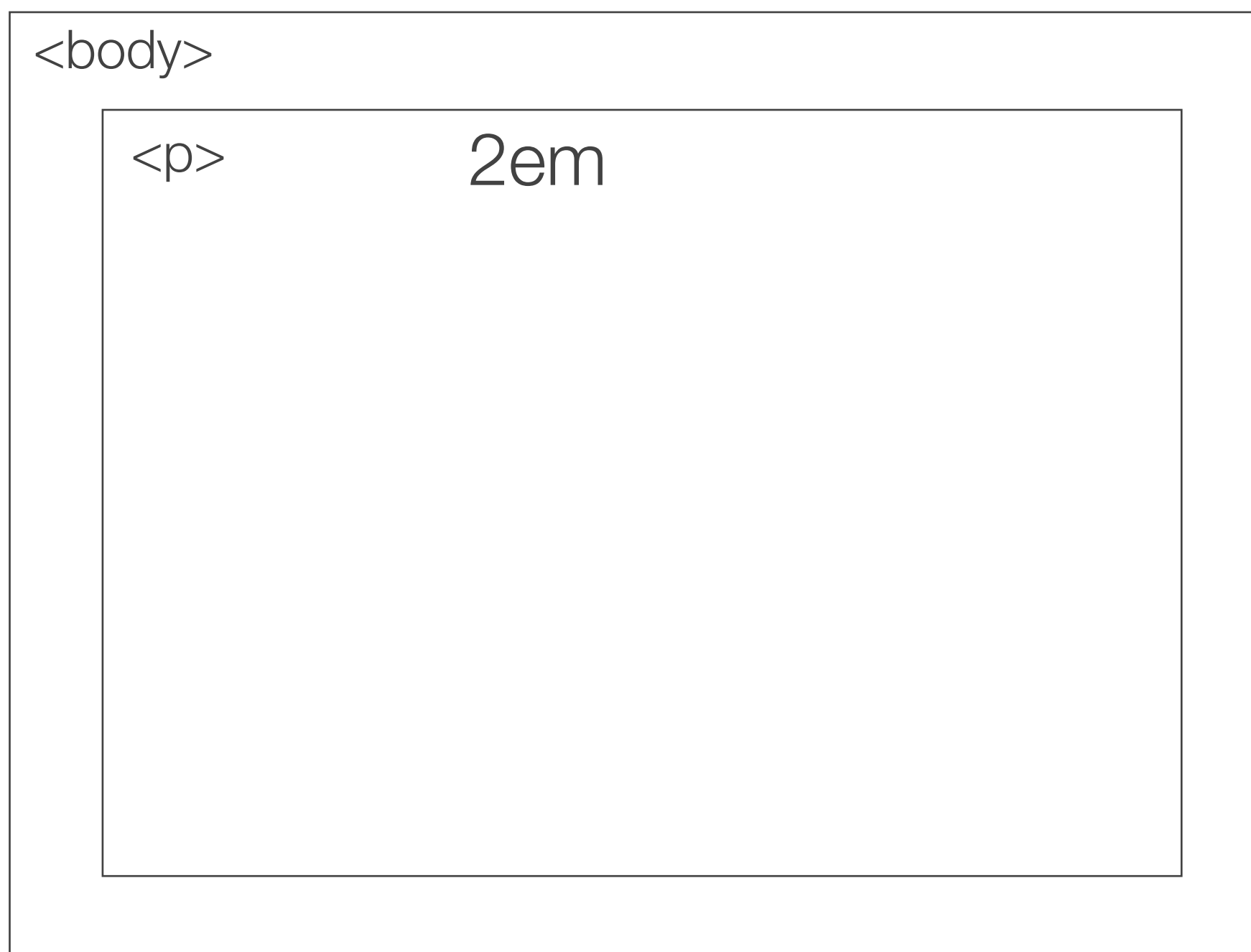
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

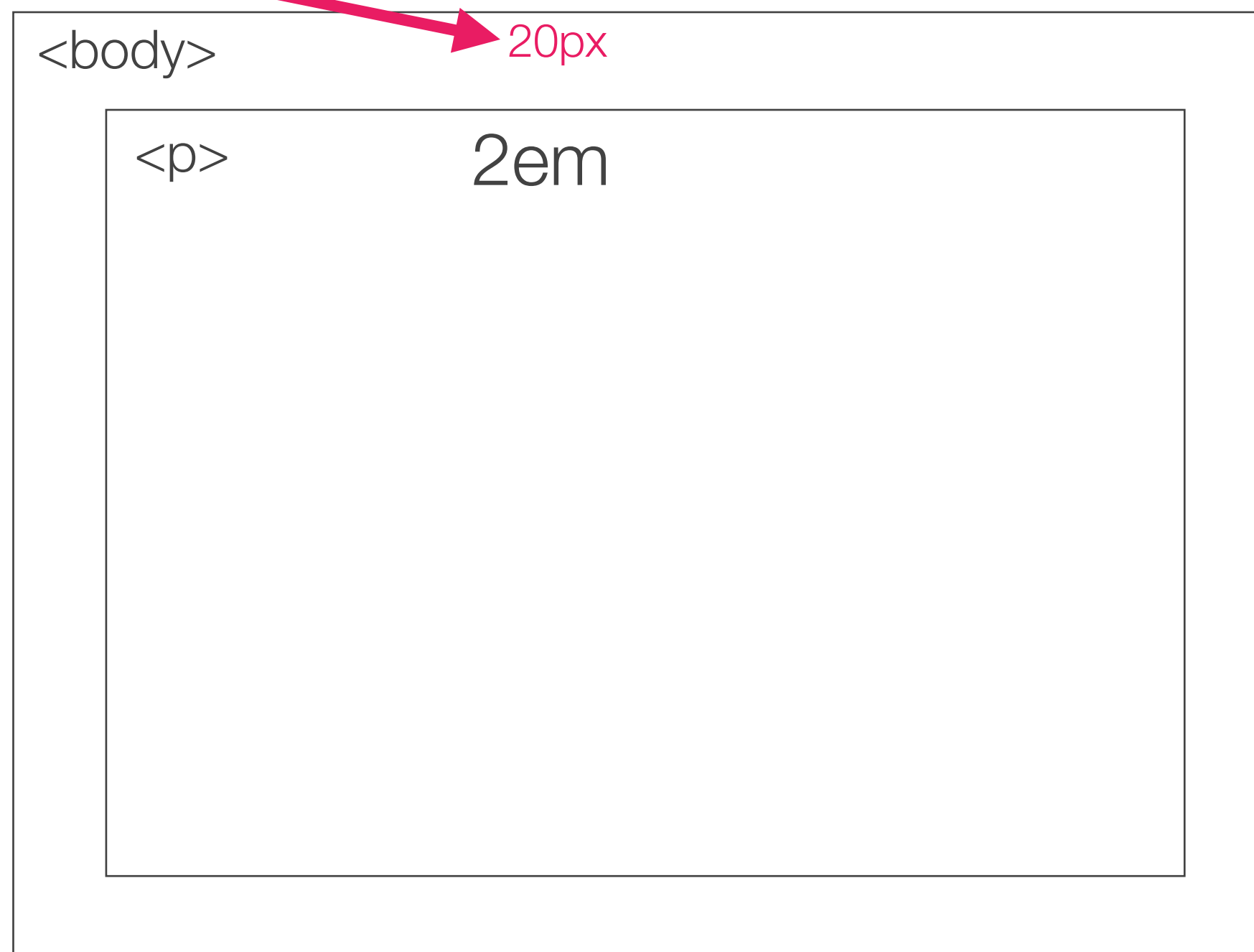
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size :



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

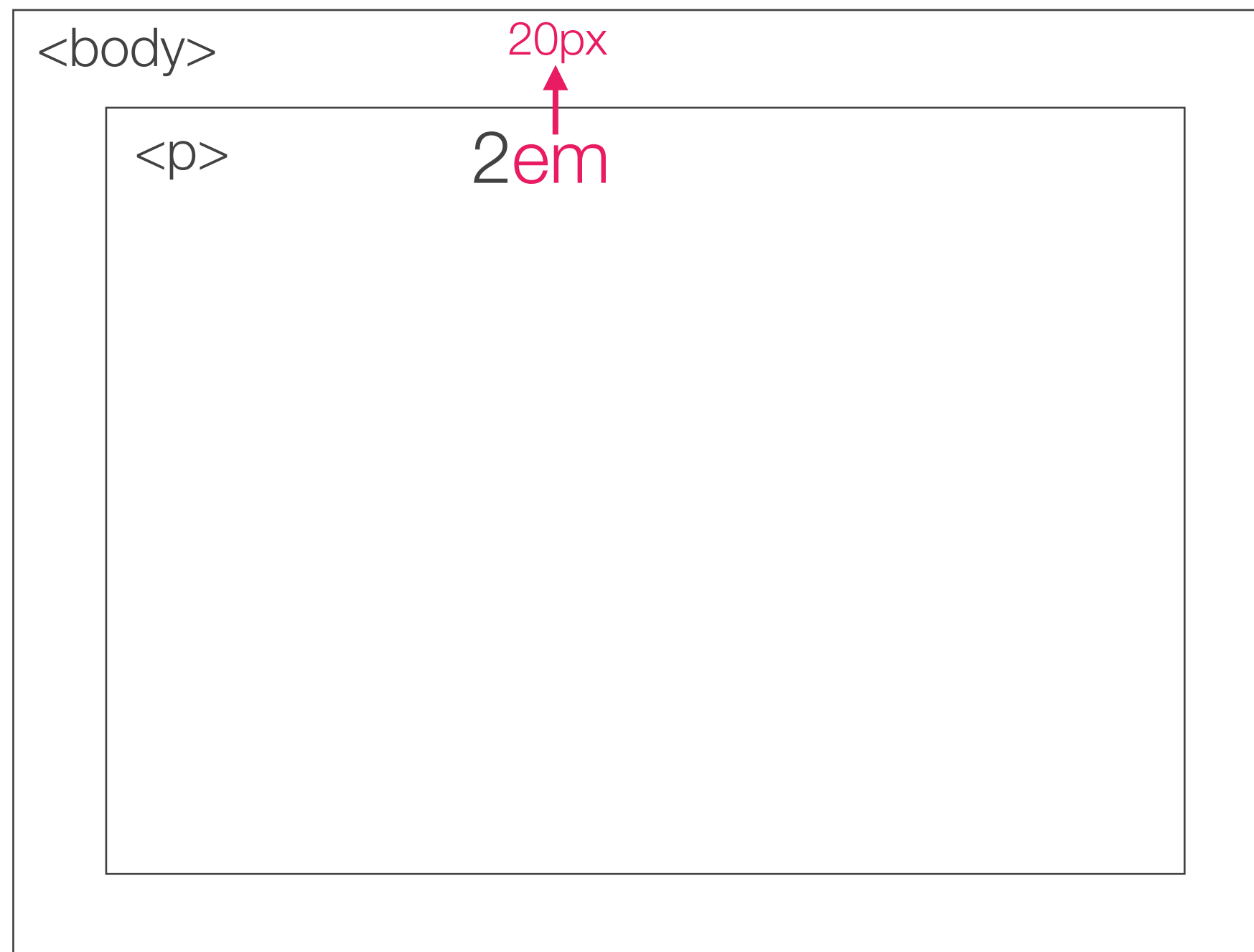
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

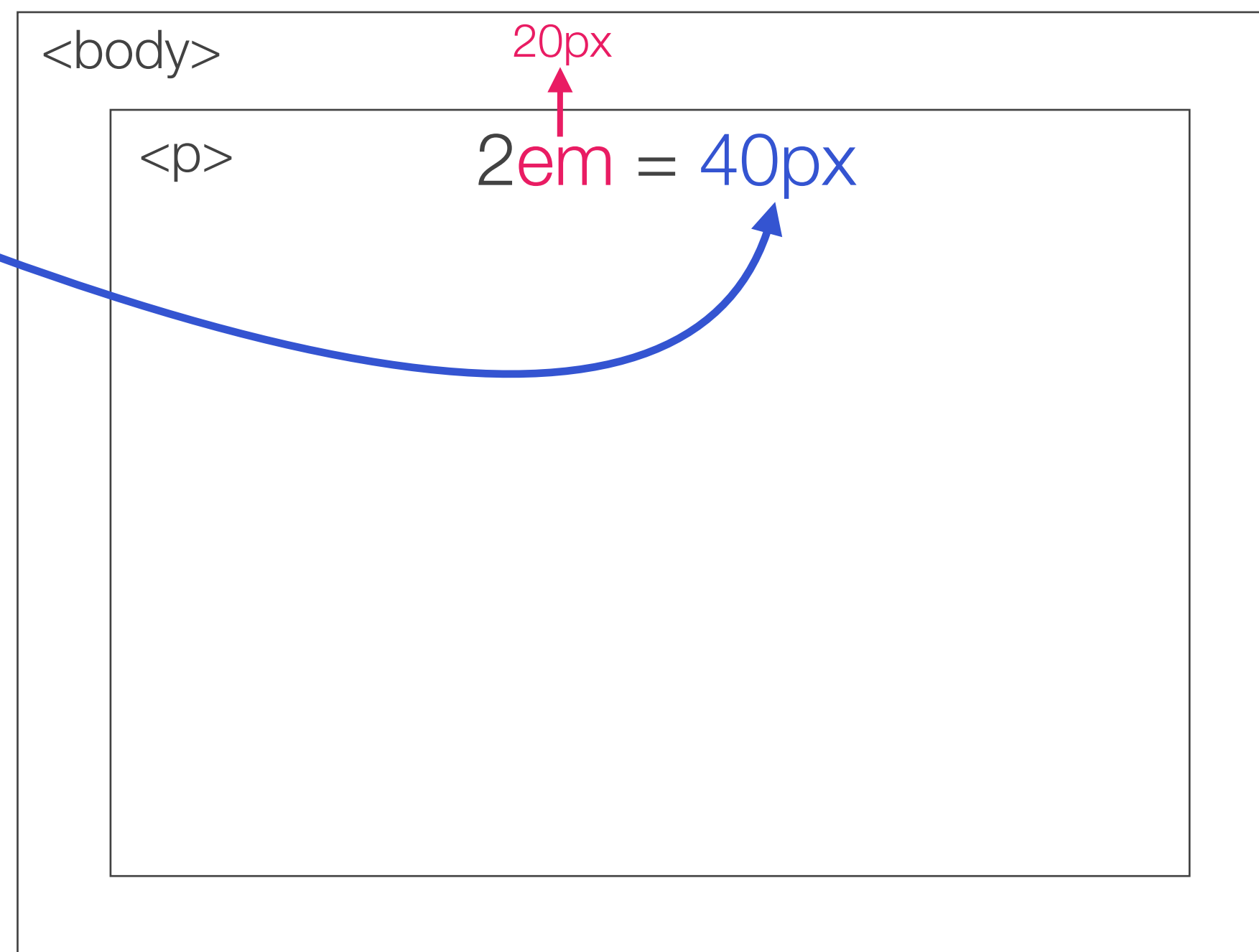
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

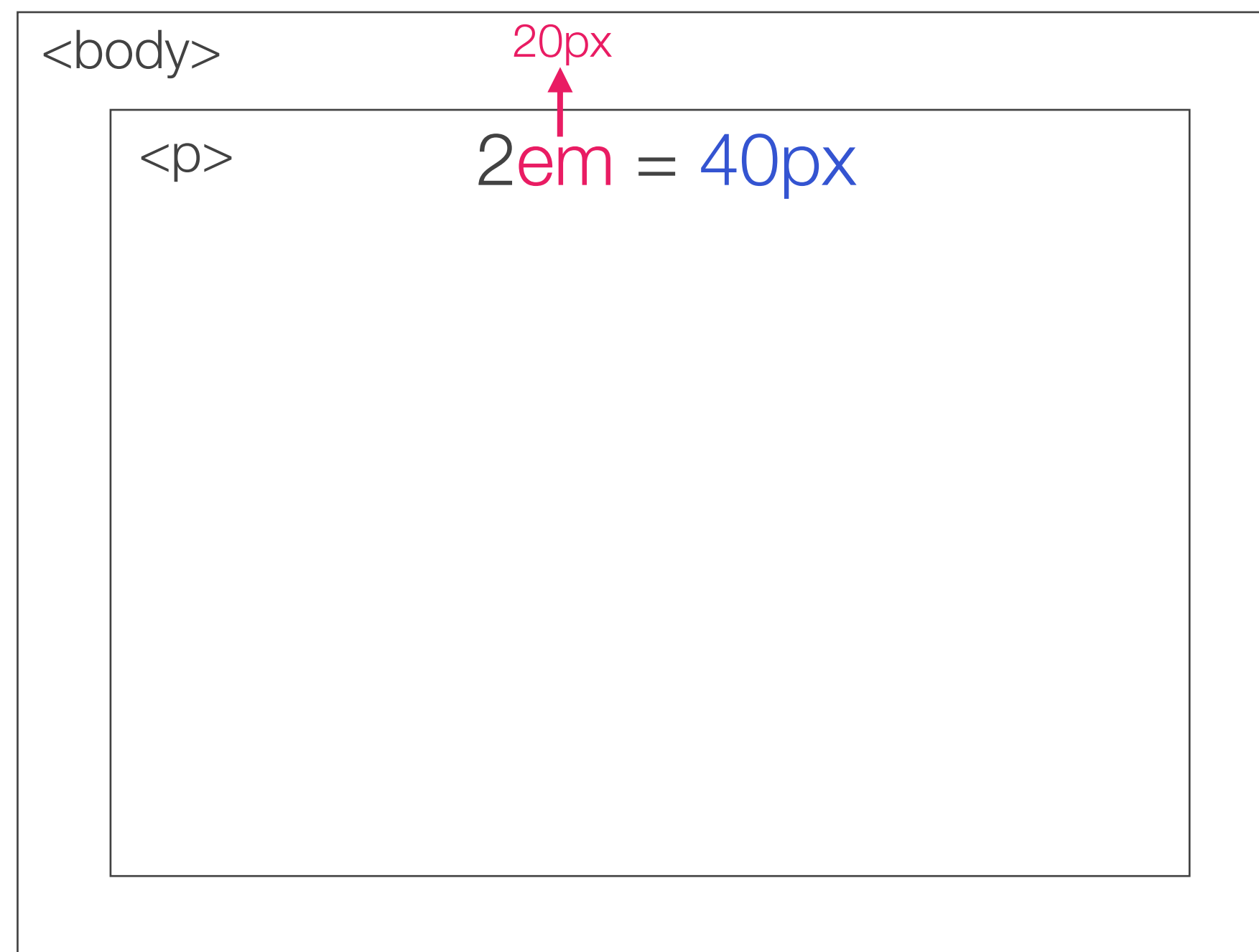
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

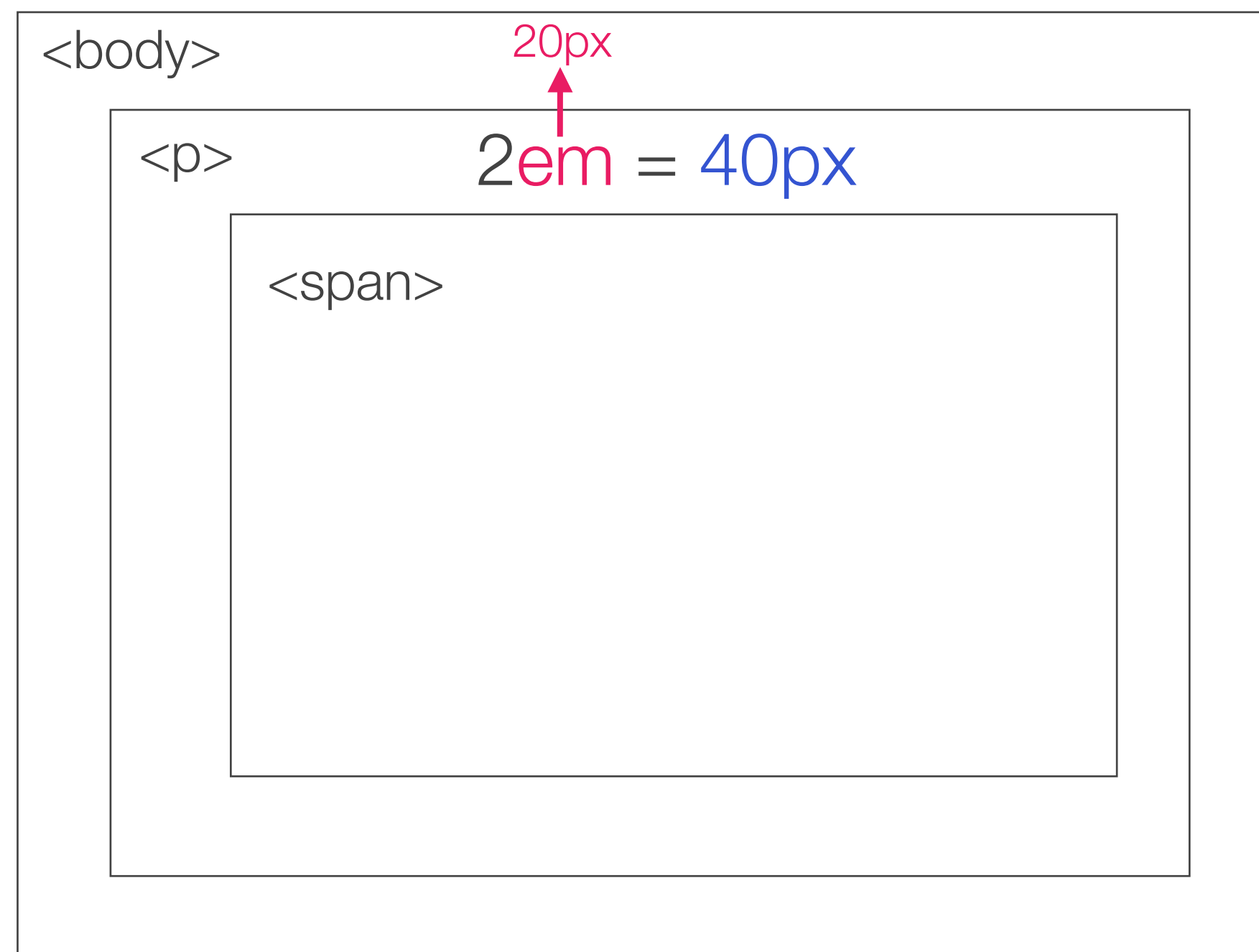
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

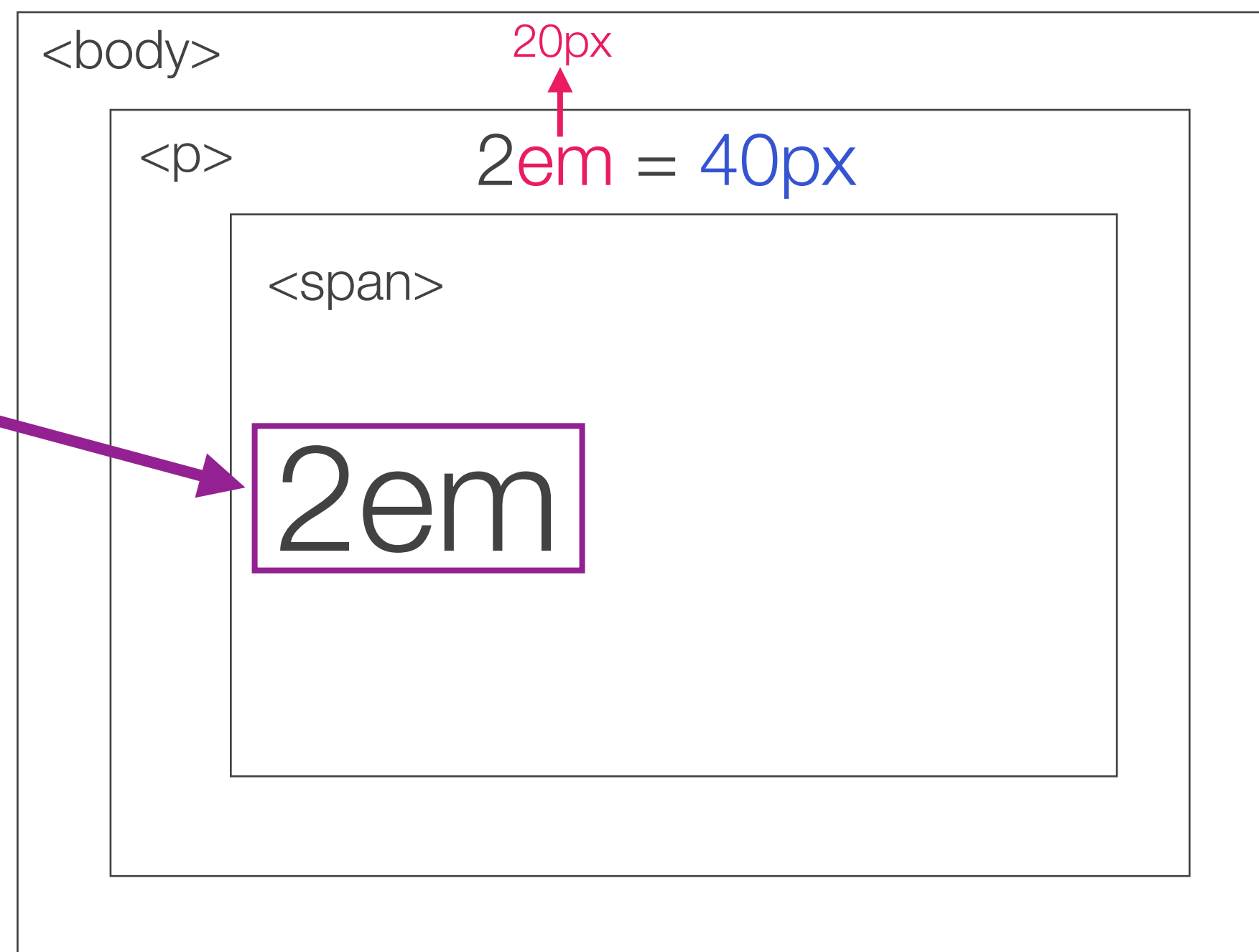
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

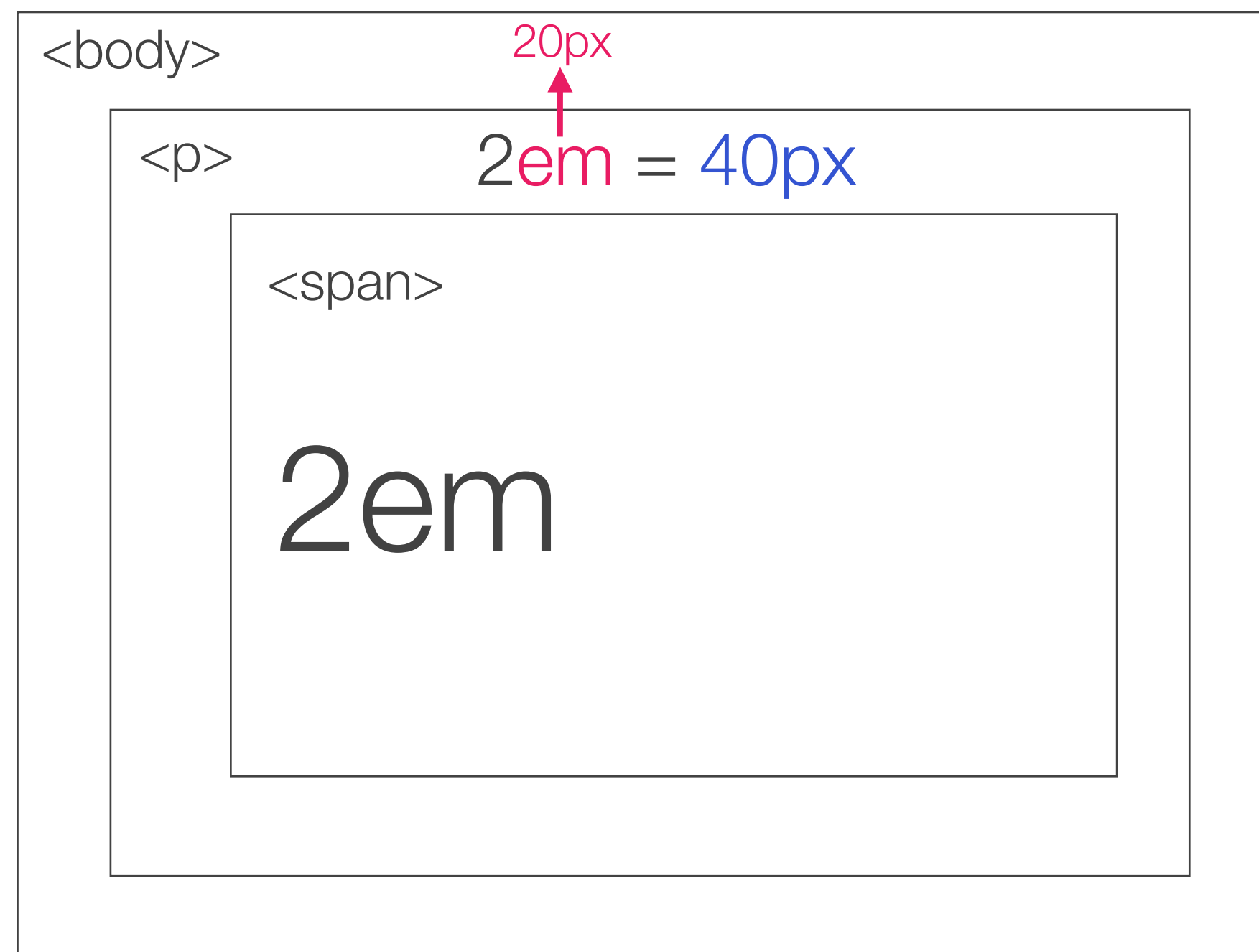
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

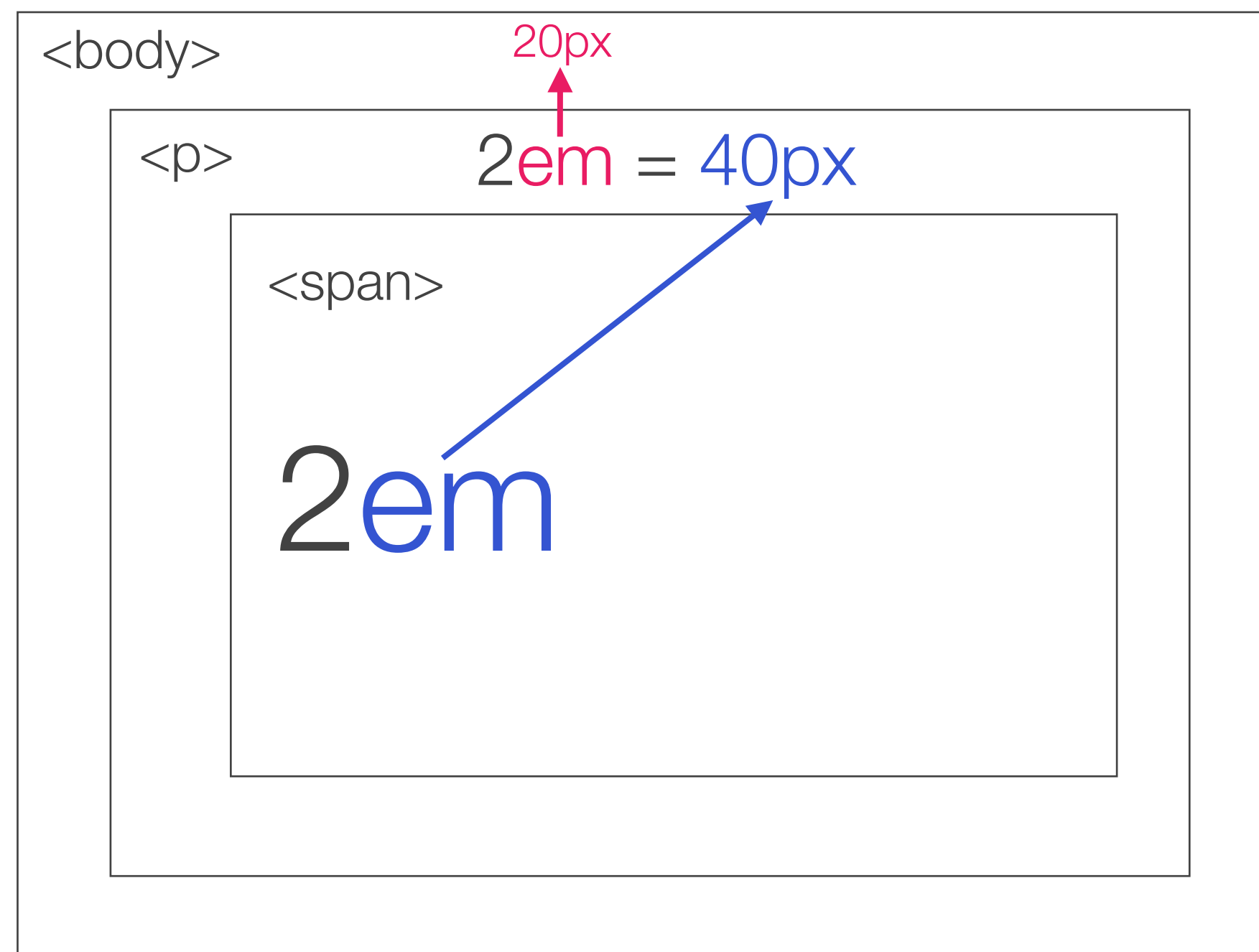
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

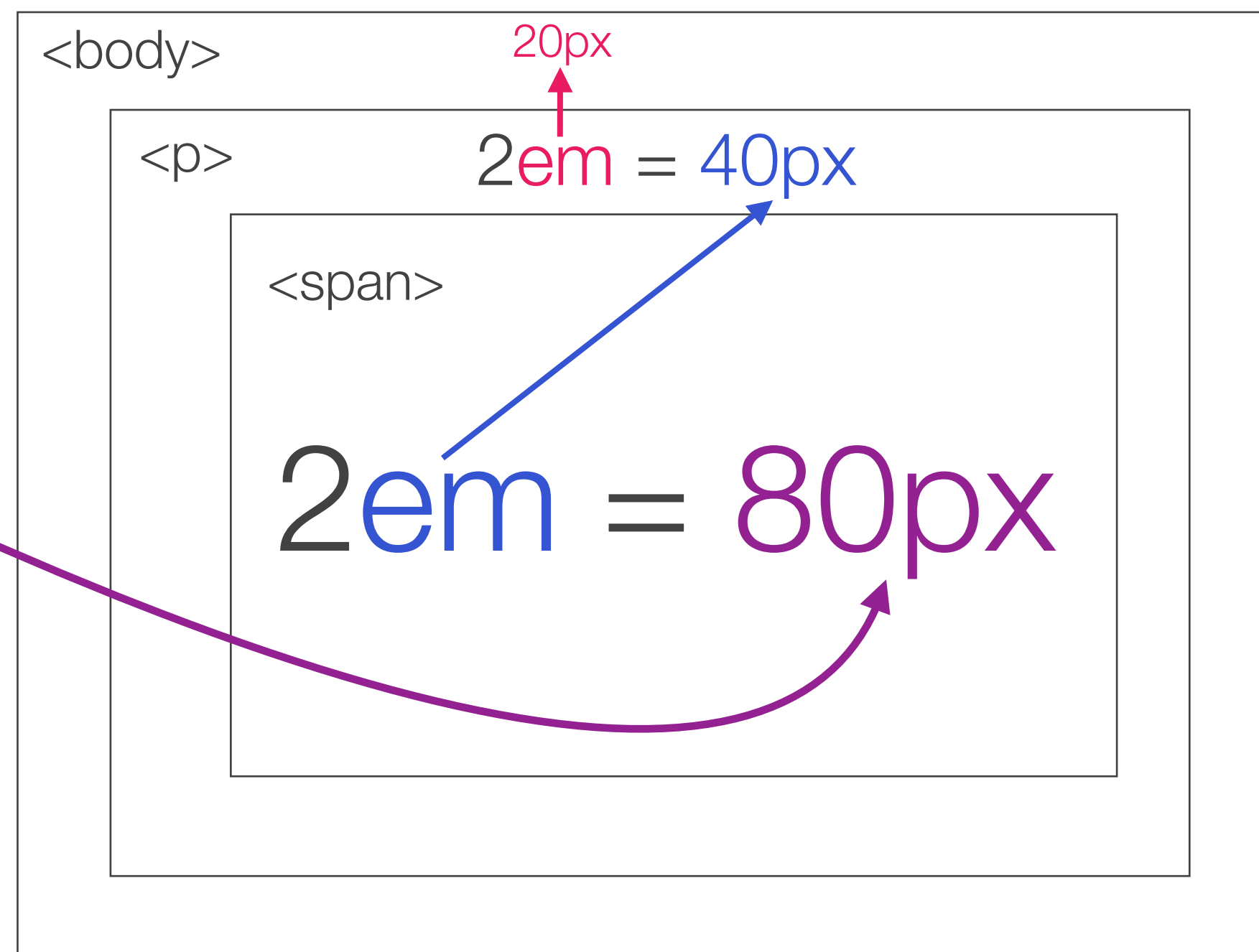
(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

font-size：



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

<p>

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)



Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

<p>

margin: 1.5em

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent的font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

<p>本身的font-size = 40px

margin: 1.5em

Quiz 7

請直接執行以下程式碼，並回答下面三個小題：

(1) <p>的字體大小是幾個px？ (2) 的字體大小是幾個px？ (3) <p>的margin是幾個px？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      font-size: 20px;
    }
    p {
      font-size: 2em;
      margin: 1.5em;
    }
    span {
      font-size: 2em;
    }
  </style>
</head>
<body>
  <p>
    光速買
    <span>最棒</span>
  </p>
</body>
</html>
```

Ans:

- (1) 40px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (2) 80px (用em設定**font-size**屬性則em的基準為**parent**的**font-size**)
- (3) 60px (用em設定**非font-size**的屬性則em的基準為**自己的font-size**)

<p>本身的font-size = 40px

margin: 1.5em = 60px

CSS如何決定版面配置 (Layout) ?

版面配置 (Layout) 指的就是，當HTML元素被render到畫面上時，瀏覽器要去決定：

1. 元素的位置

2. 元素的大小

CSS如何決定版面配置 (Layout) ?

決定元素的位置：

瀏覽器會從HTML檔案的上方往下方一行一行地掃描，先遇到的元素就先render。

上面講的這個Layout原則稱為`flow`（又稱`normal flow`或者`document flow`）。

元素被render後，有些元素會和前後元素斷行（linebreak），有些則不會。

至於會不會斷行這件事取決於元素本身CSS的`display`屬性的設定值，若`display`設成`block`，就會斷行；若`display`設成`inline`或`inline-block`，就不會斷行。

`inline`和`inline-block`的差異，後面會再討論。

就算你都沒設定`display`，瀏覽器早就對每個元素都有預設的`display`值：

1. `display`預設為`block`的元素，有`<html>`，`<body>`，`<h1>`～`<h6>`，`<p>`，`<div>` ...等等。
2. `display`預設為`inline`的元素，有`<strike>`，`<i>`，`<a>`，``，`` ...等等。
3. `display`預設為`inline-block`的元素，有`<input>`，`<button>` ...等等。

CSS如何決定版面配置 (Layout) ?

你可以透過設定`display`，將元素改成block或inline或inline-block，
範例如下：

將h1改成inline

```
h1 {  
  display: inline;  
}
```

將span改成block

```
span {  
  display: block;  
}
```

將h1改成inline-block

```
h1 {  
  display: inline-block;  
}
```

將h1從畫面中隱藏起來

```
h1 {  
  display: none;  
}
```

`display`可以設定的值還有很多，此處只提本課程會用到的部分，更多其他的值請參考

<https://developer.mozilla.org/en-US/docs/Web/CSS/display>

Quiz 8 (1) 請修改專案1的CSS把<p>的display設成inline，看會有什麼結果。
(2) 請修改專案1的CSS把<p>的display設成none，看會有什麼結果。

Quiz 8 (1) 請修改專案1的CSS把<p>的display設成inline，看會有什麼結果。
(2) 請修改專案1的CSS把<p>的display設成none，看會有什麼結果。

DEMO

CSS如何決定版面配置 (Layout) ?

接下來我們再用兩個例子，來讓大家對flow的運作更有感覺。

CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：

HTML

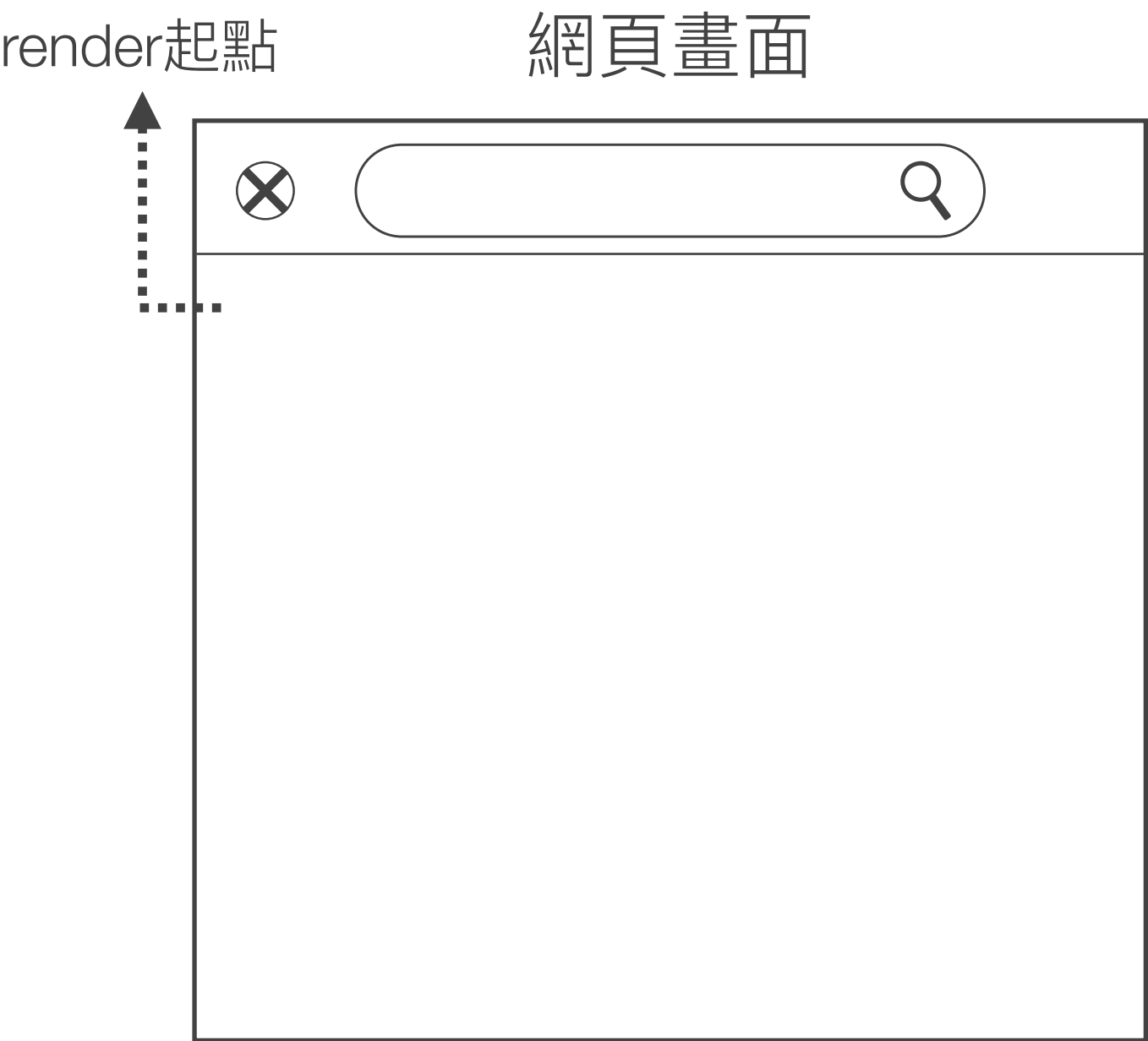
```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <h1>...</h1>
    <h2>...</h2>
    <h3>...</h3>
    <h4>...</h4>
    <h5>...</h5>
    <h6>...</h6>
  </body>
</html>
```

CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：

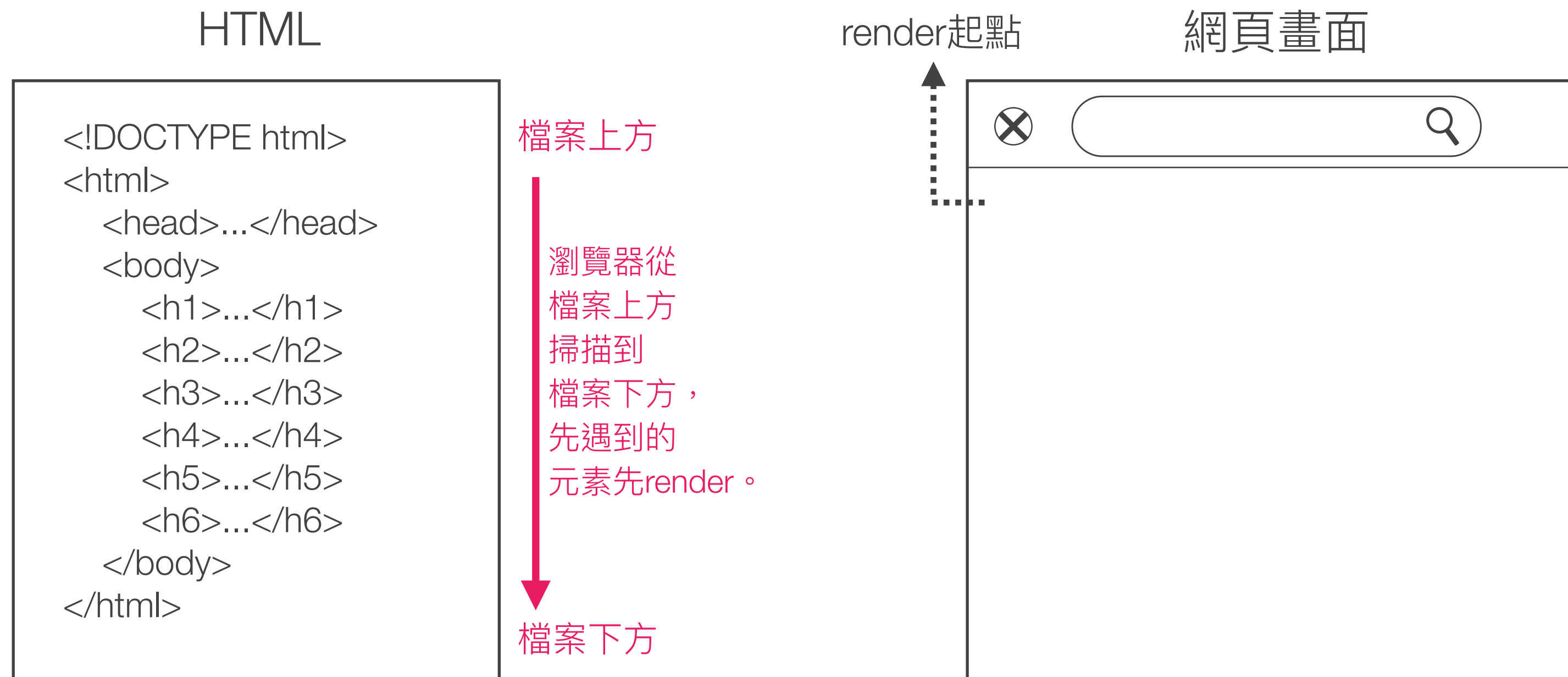
HTML

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <h1>...</h1>
    <h2>...</h2>
    <h3>...</h3>
    <h4>...</h4>
    <h5>...</h5>
    <h6>...</h6>
  </body>
</html>
```



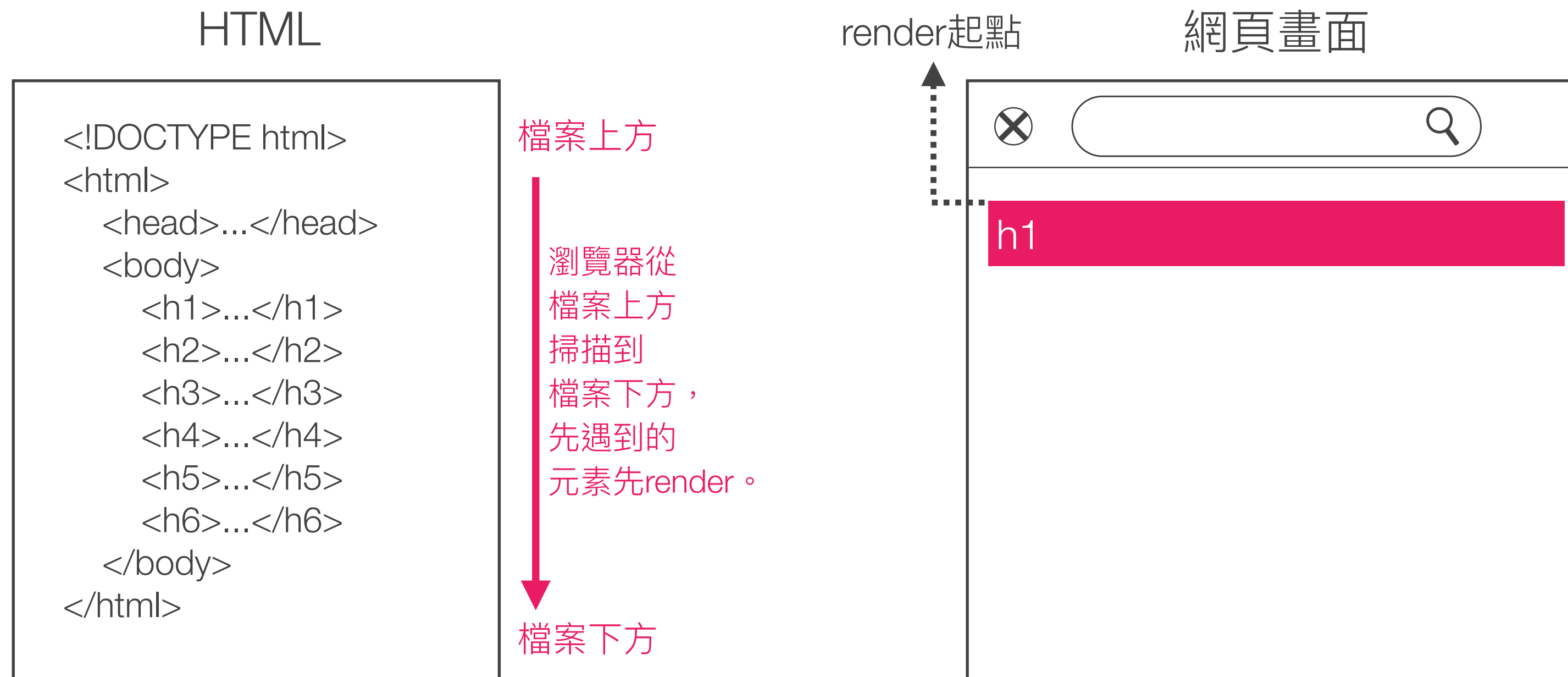
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



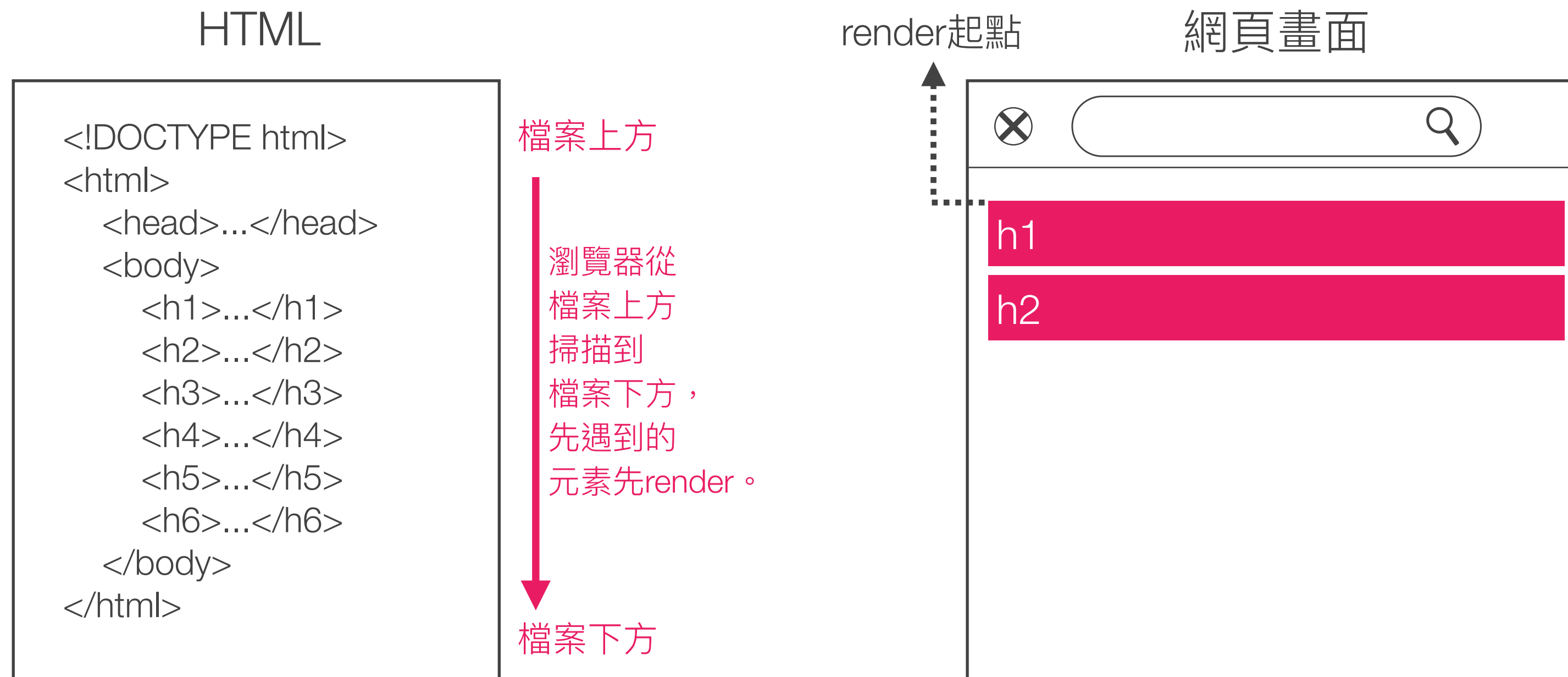
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



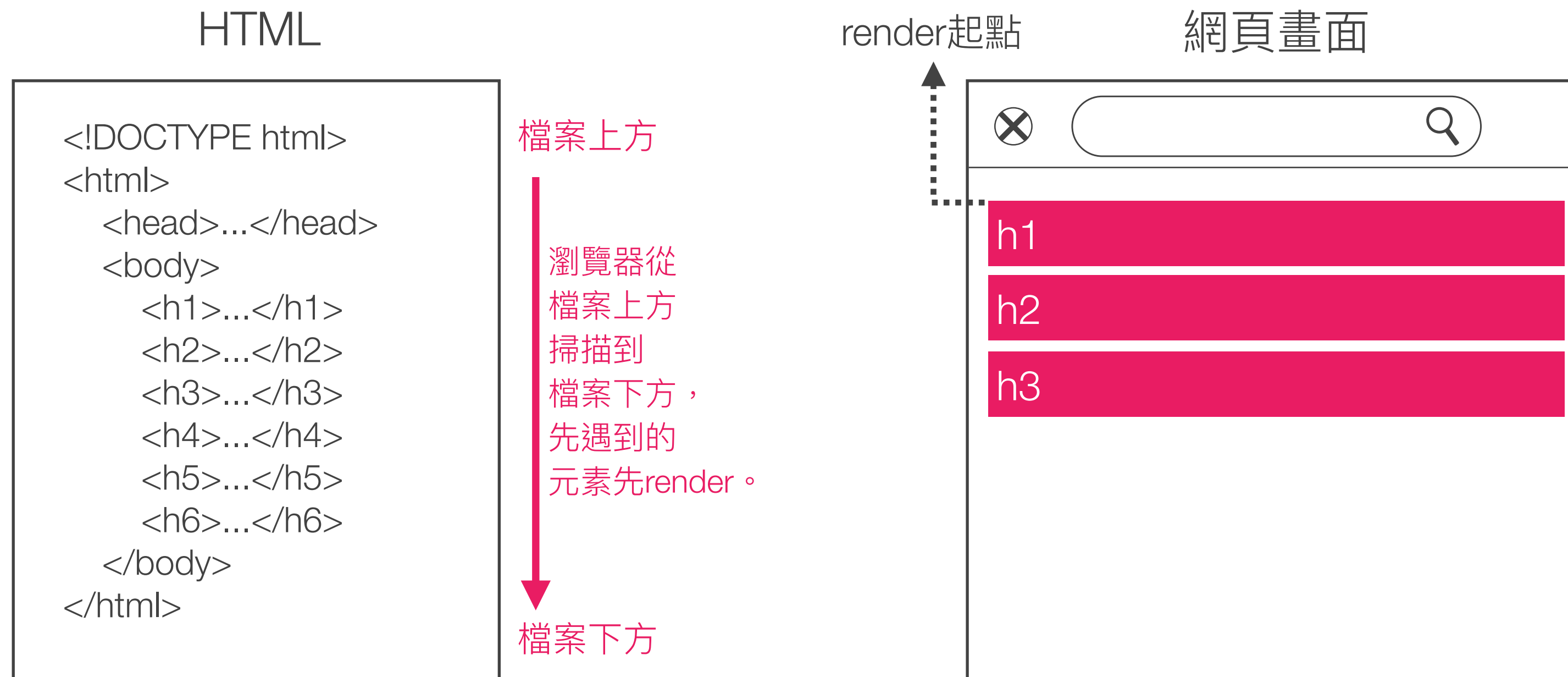
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



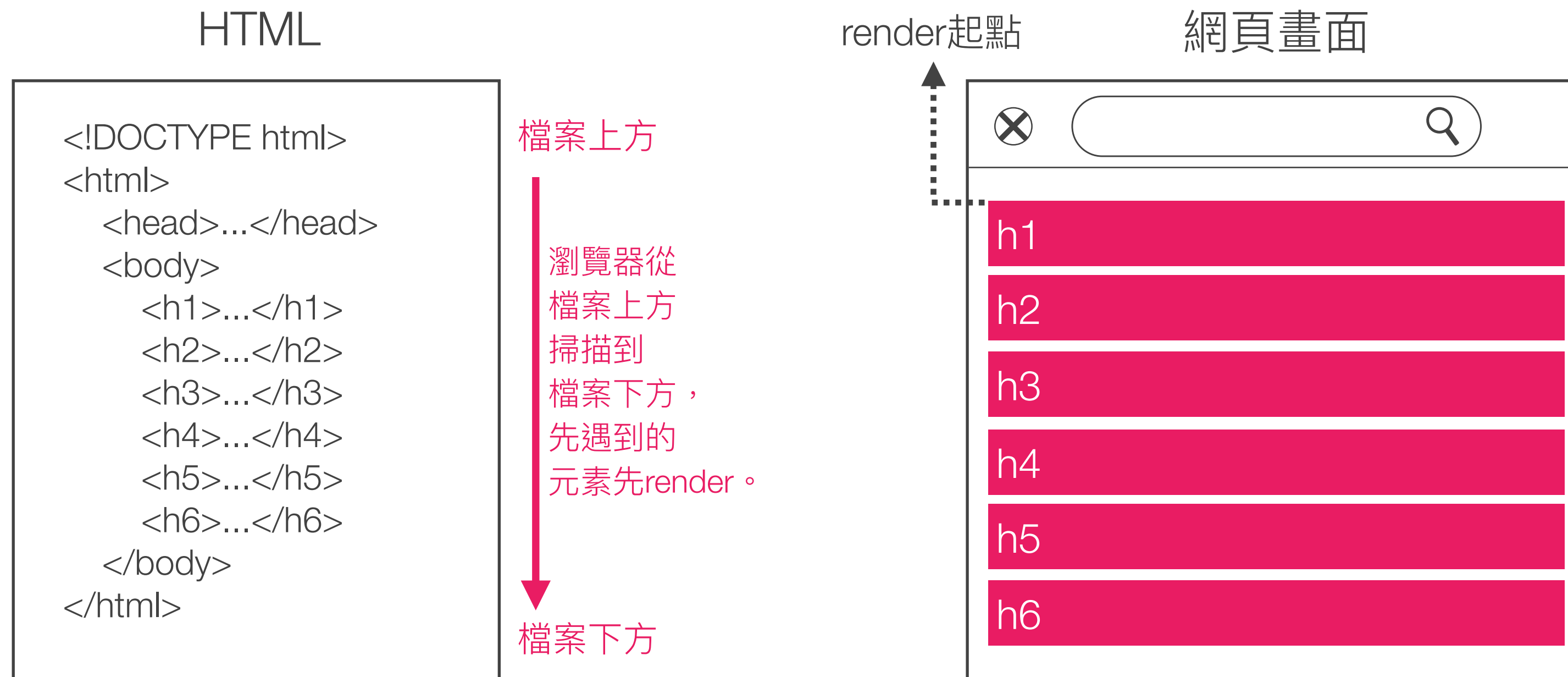
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



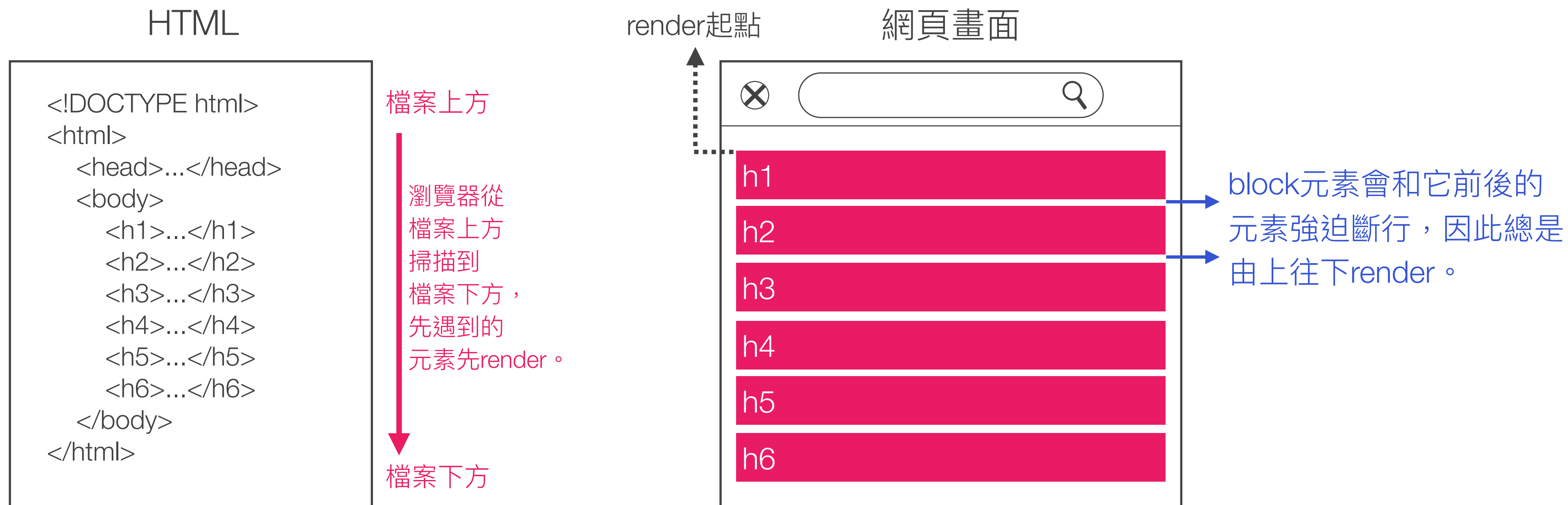
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



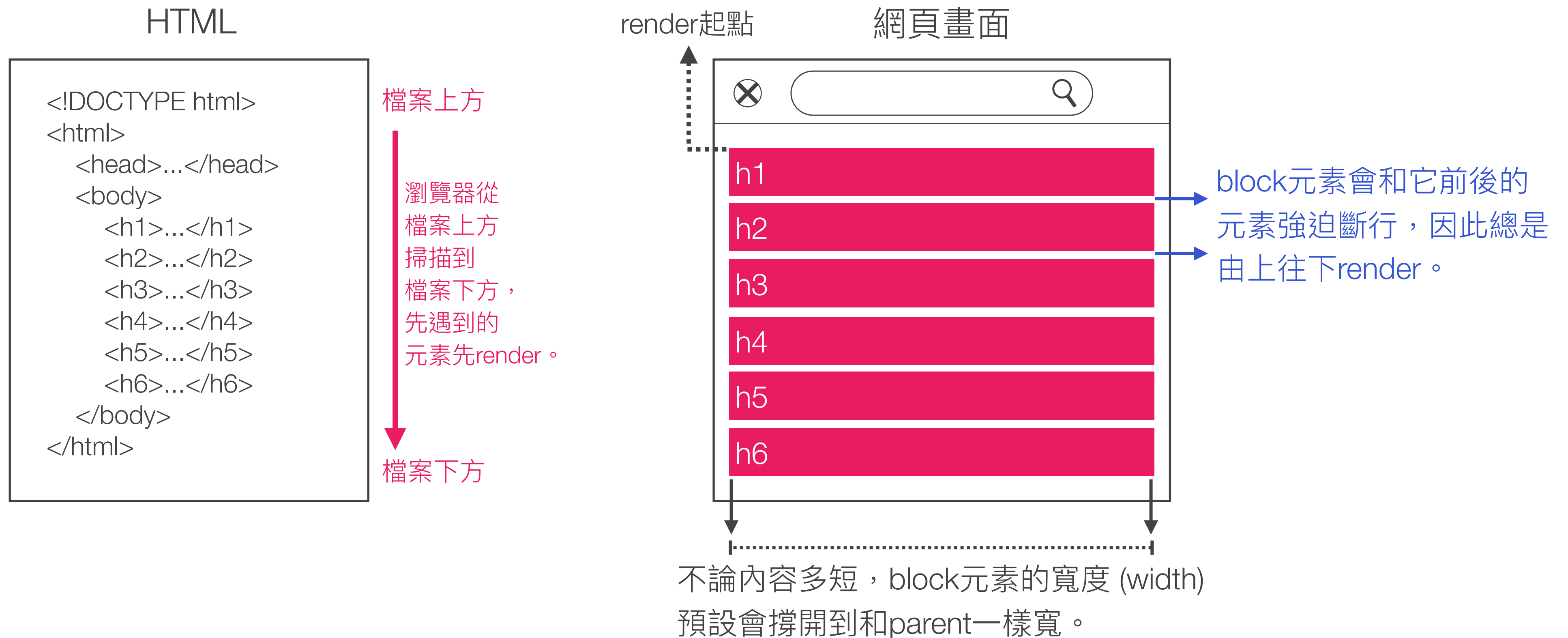
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



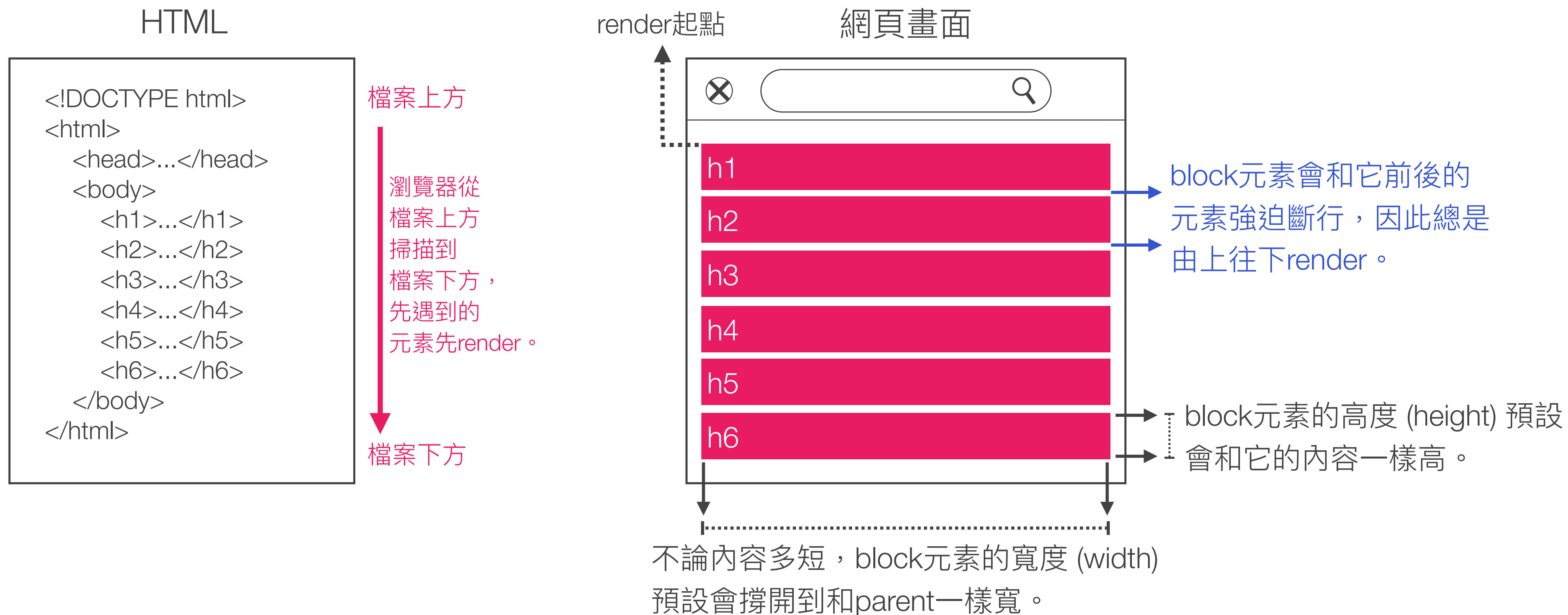
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中都是block元素為例：



CSS如何決定版面配置 (Layout) ?

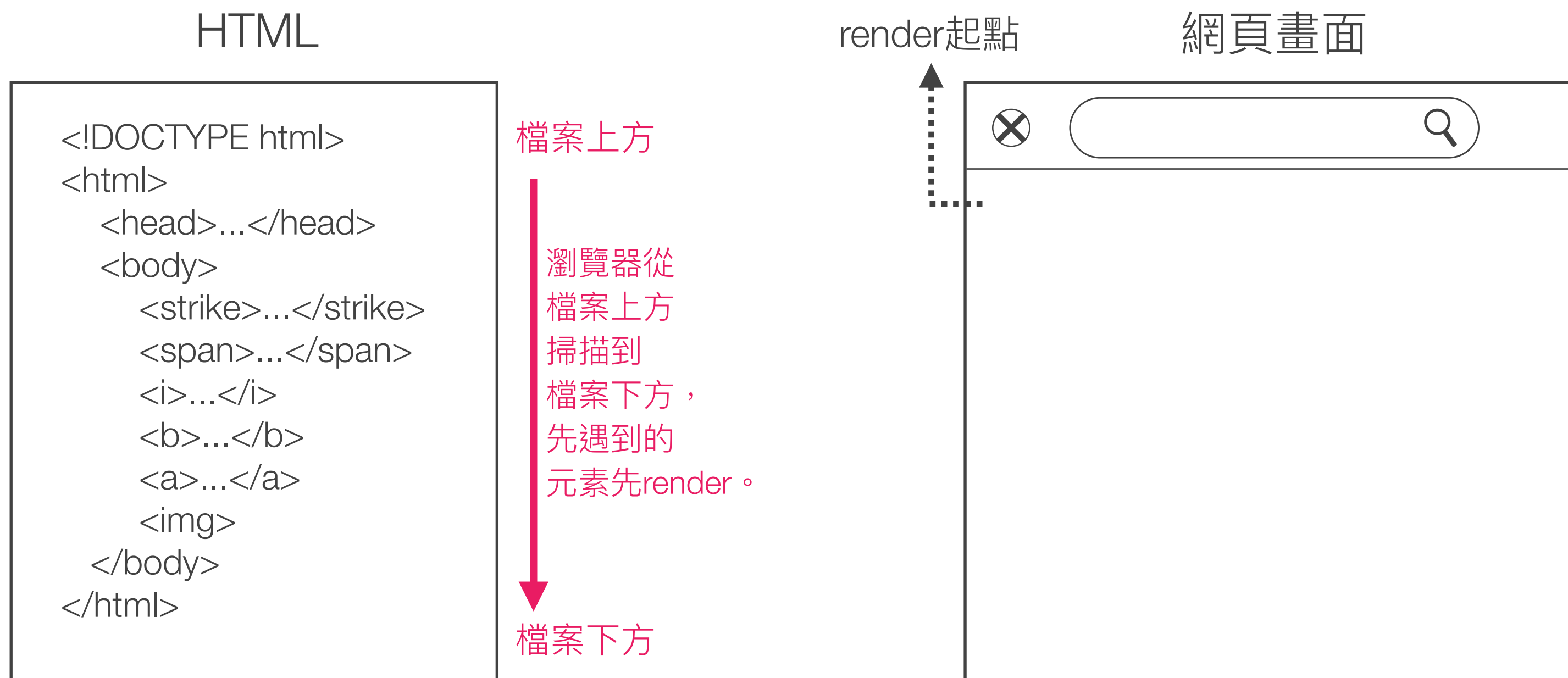
Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：

HTML

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <strike>...</strike>
    <span>...</span>
    <i>...</i>
    <b>...</b>
    <a>...</a>
    <img>
  </body>
</html>
```

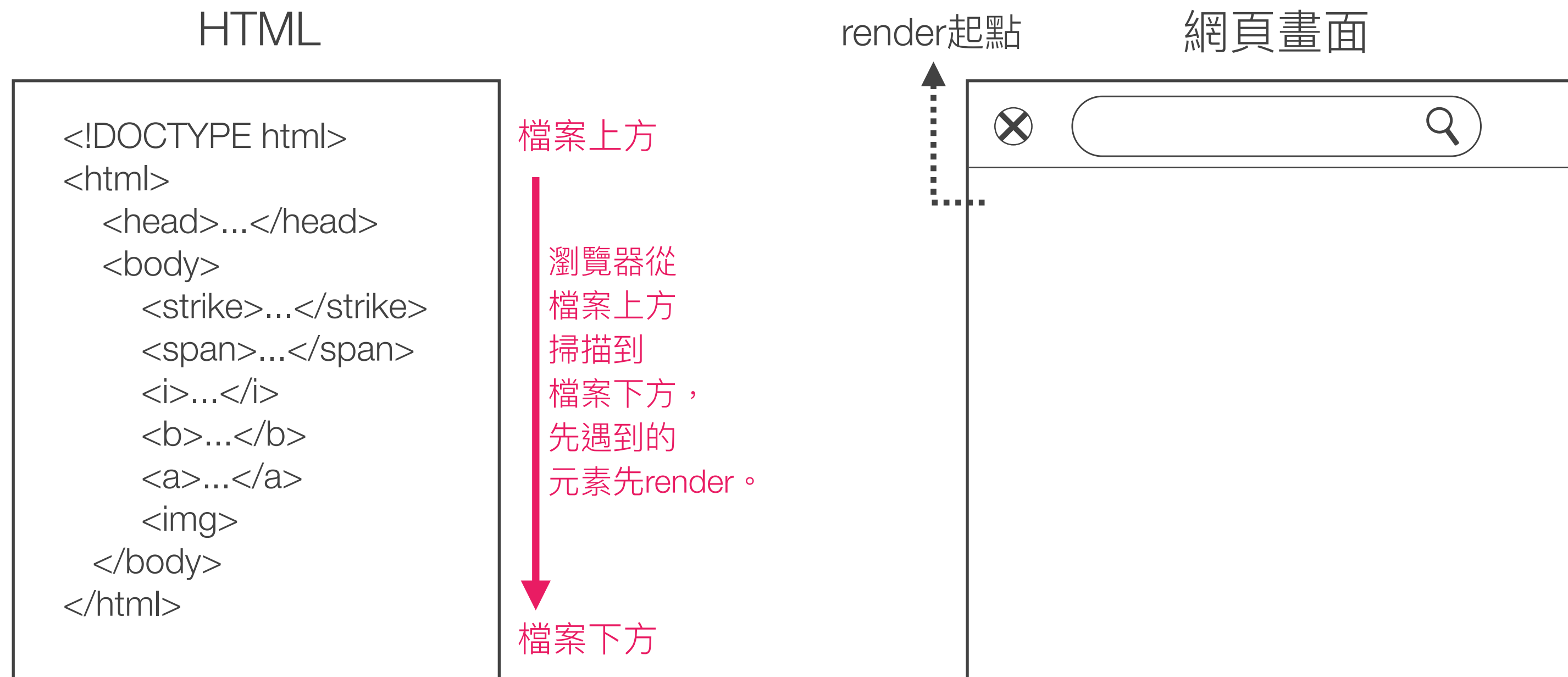
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



inline元素不會和它前後的元素斷行，它總是盡量靠在前一個元素的右邊開始render；一旦parent (母元素，此處為<body>) 的寬度不夠了，才會跳到下一行render。

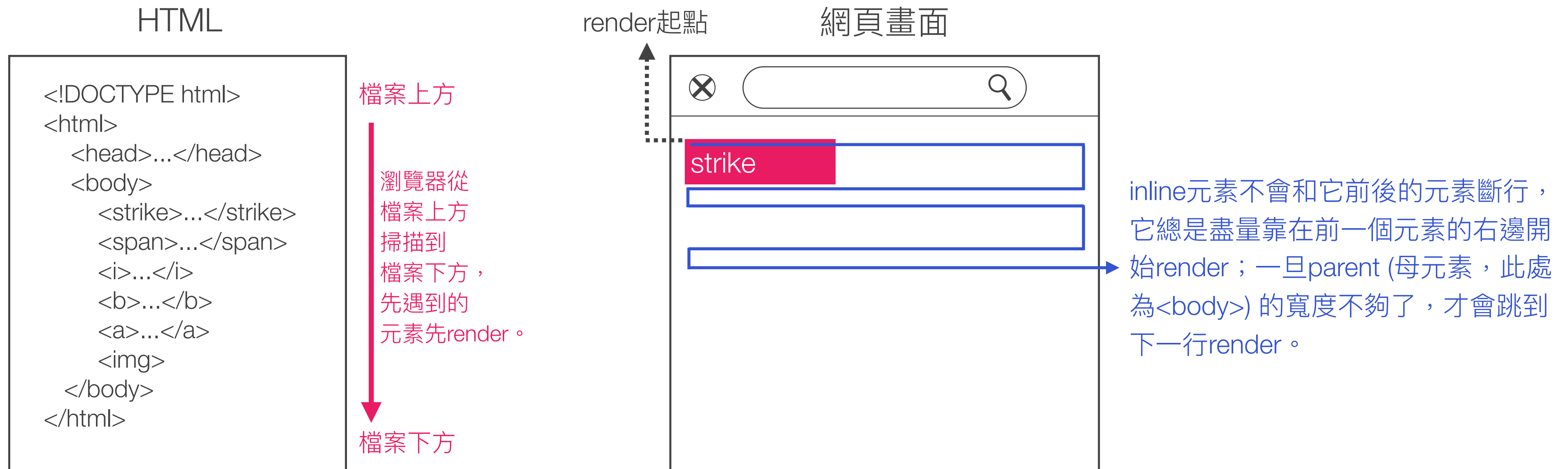
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



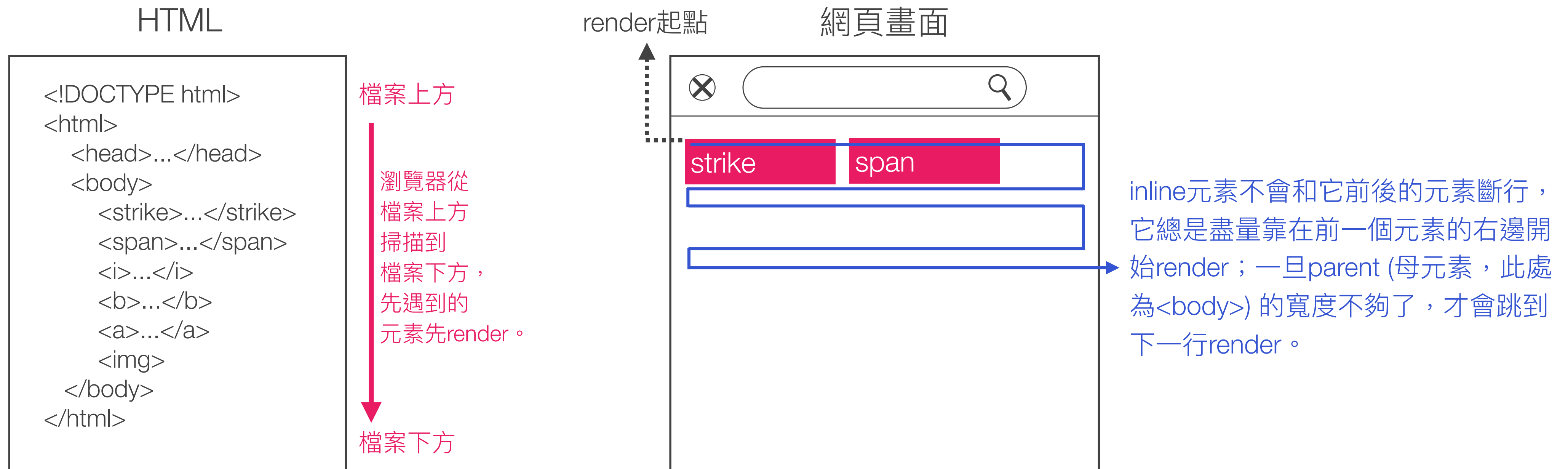
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



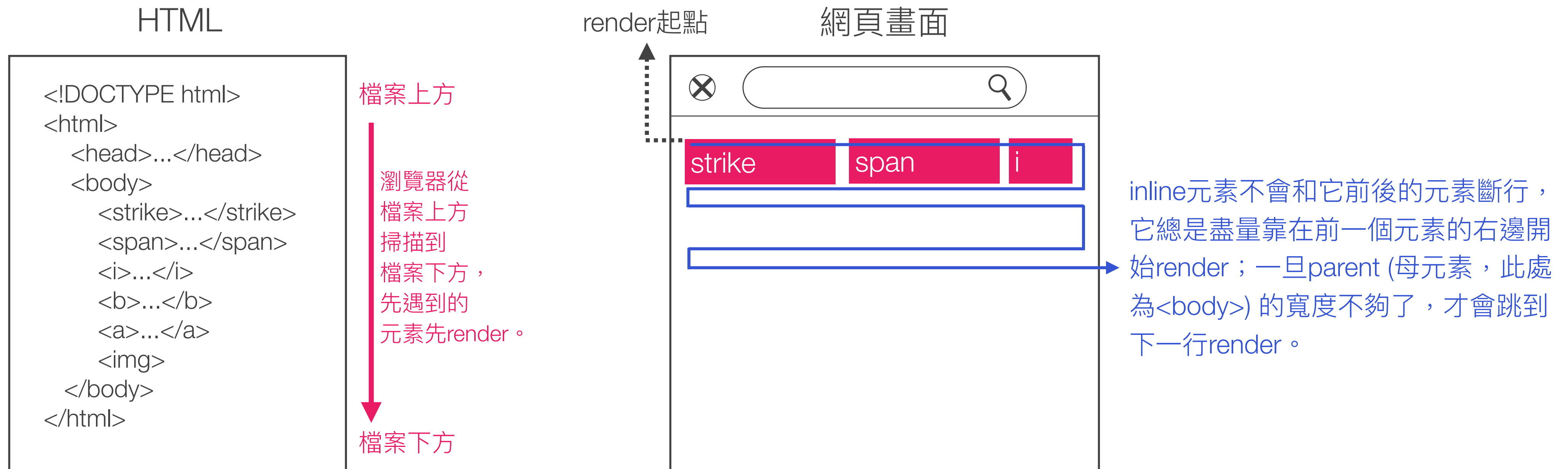
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



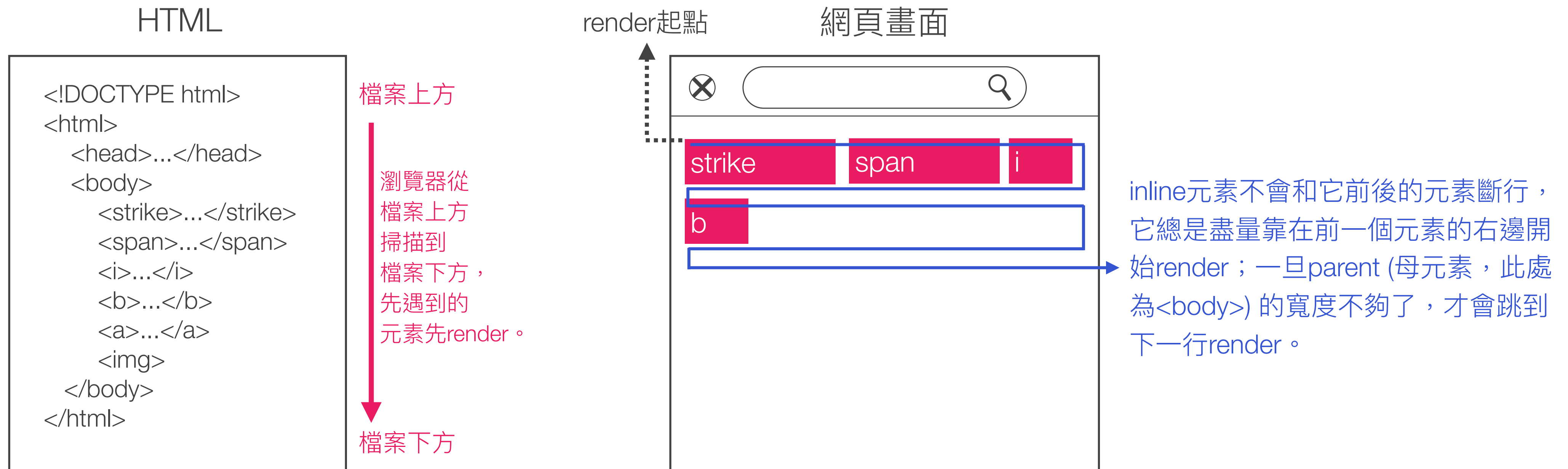
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



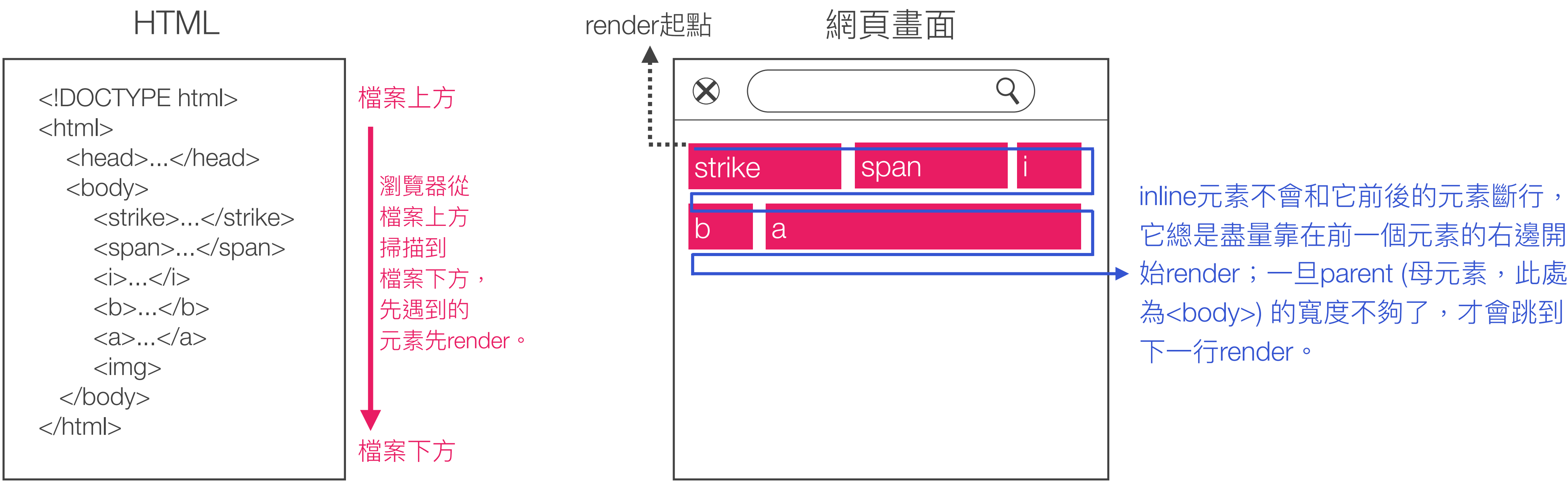
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



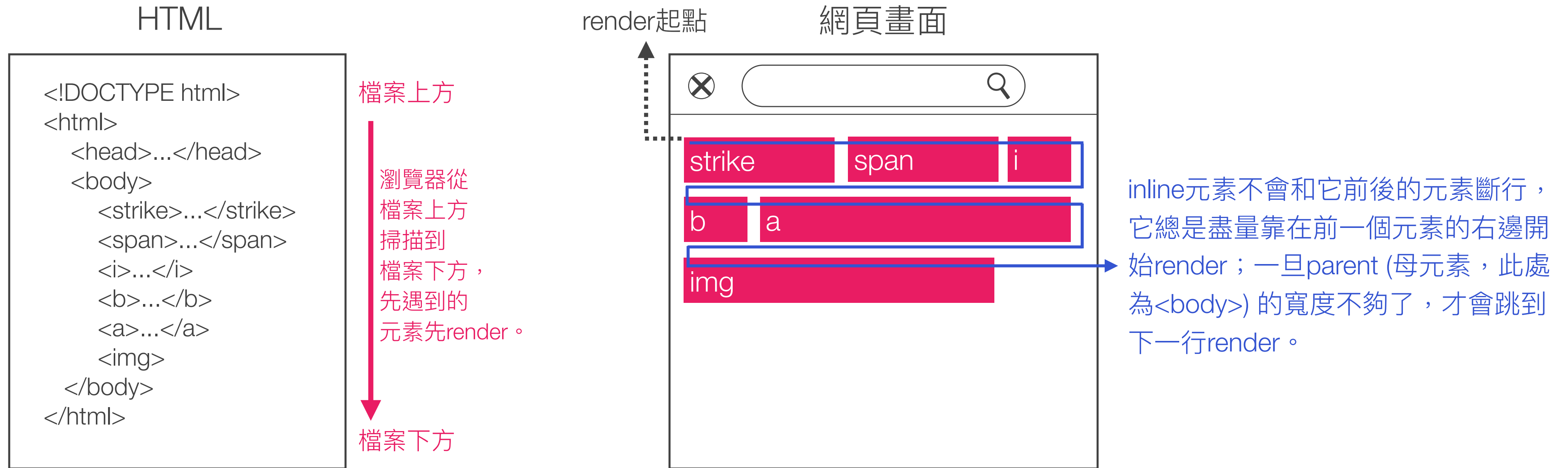
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



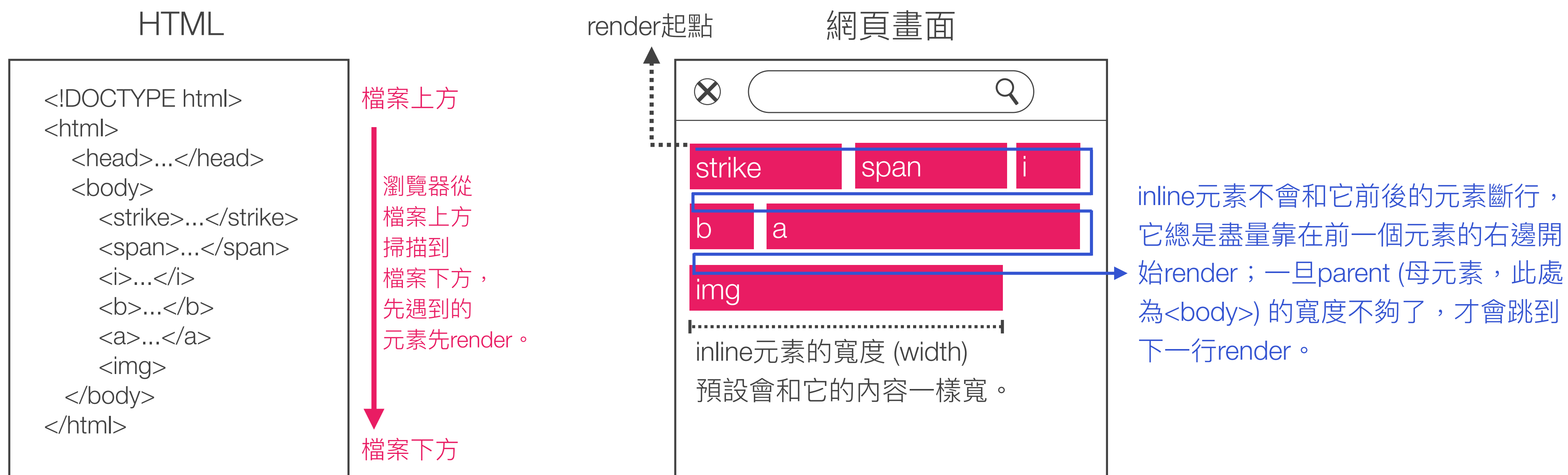
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



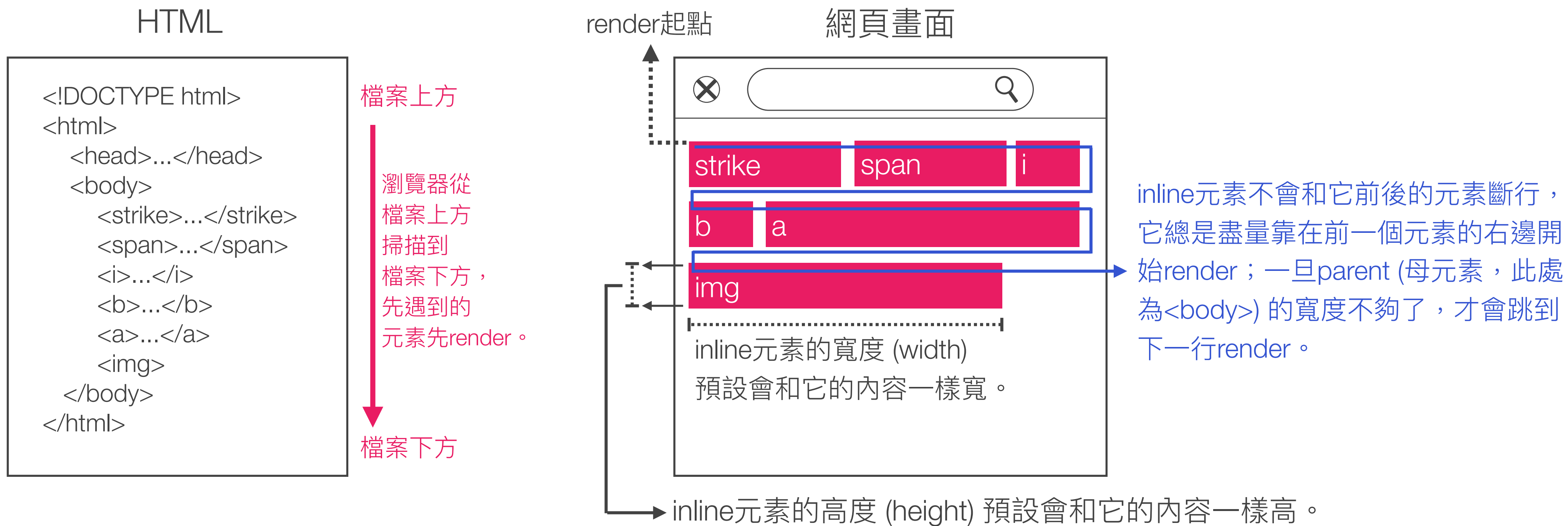
CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



CSS如何決定版面配置 (Layout) ?

Flow的運作 - 假設網頁中的<body>裡面都是inline元素 (假設撰寫HTML程式碼時，每個標籤之間都有斷行)：



Quiz 9

有一份HTML文件如左下方所示，裡面混合了block元素和inline元素，請猜猜看這份HTML文件render後的Layout，大概會是什麼樣子？

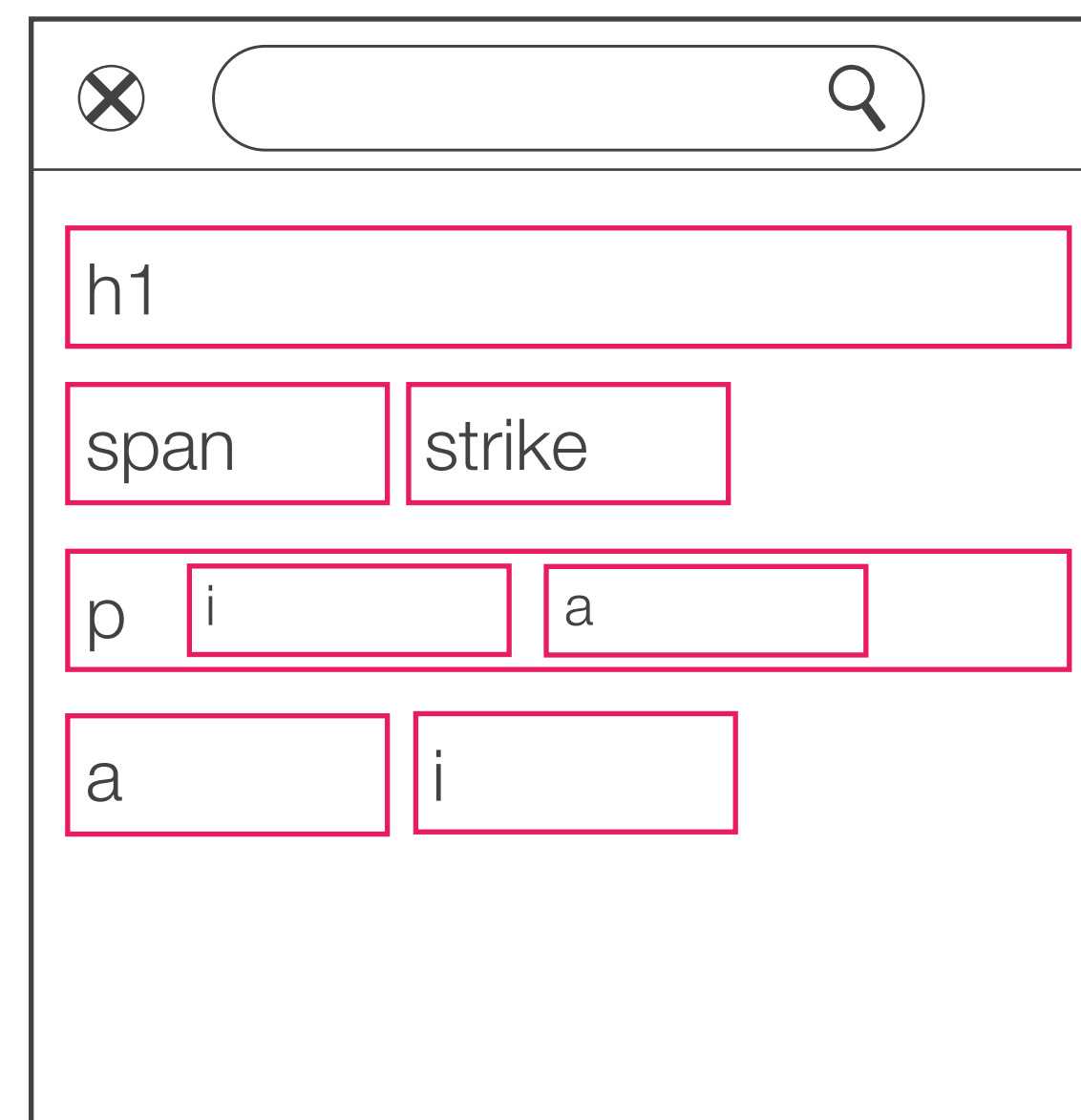
```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <h1>1</h1>
    <span>2</span>
    <strike>3</strike>
    <p>
      <i>4</i>
      <a>5</a>
    </p>
    <a>6</a>
    <i>7</i>
  </body>
</html>
```

Quiz 9

有一份HTML文件如左下方所示，裡面混合了block元素和inline元素，請猜猜看這份HTML文件render後的Layout，大概會是什麼樣子？

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <h1>1</h1>
    <span>2</span>
    <strike>3</strike>
    <p>
      <i>4</i>
      <a>5</a>
    </p>
    <a>6</a>
    <i>7</i>
  </body>
</html>
```

Ans:



CSS如何決定版面配置 (Layout) ?

決定元素的大小：

CSS如何決定版面配置 (Layout) ?

決定元素的大小：

Box Model (盒子模型)

CSS如何決定版面配置 (Layout) ?

決定元素的大小：

每個元素被render後，預設會佔據一個矩形的面積，這個矩形稱為box。

CSS使用了Box Model來描述一個box。

CSS如何決定版面配置 (Layout) ?

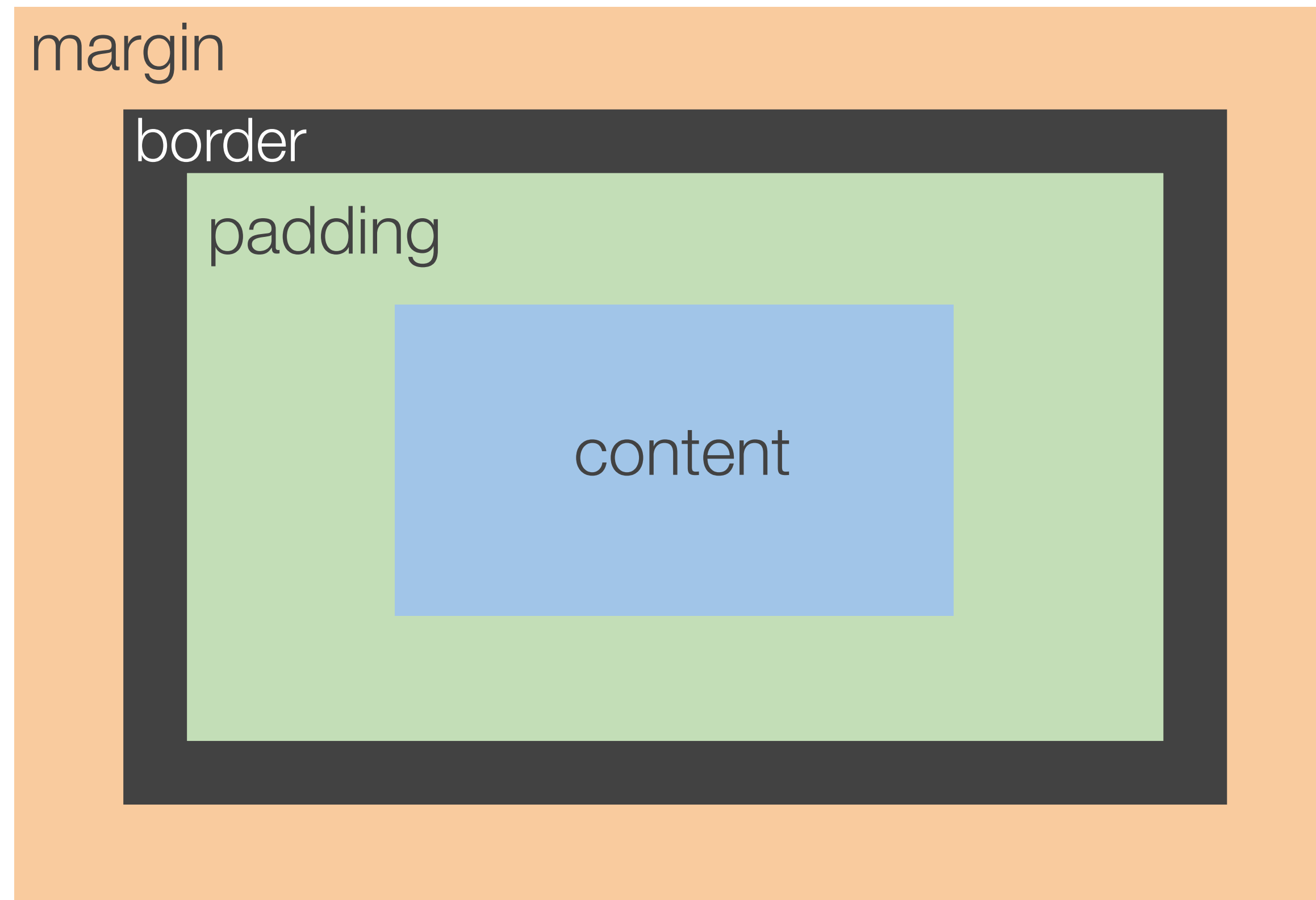
決定元素的大小：

每個元素被render後，預設會佔據一個矩形的面積，這個矩形稱為box。

CSS使用了Box Model來描述一個box。

Box Model將一個box分成4個部分：

content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

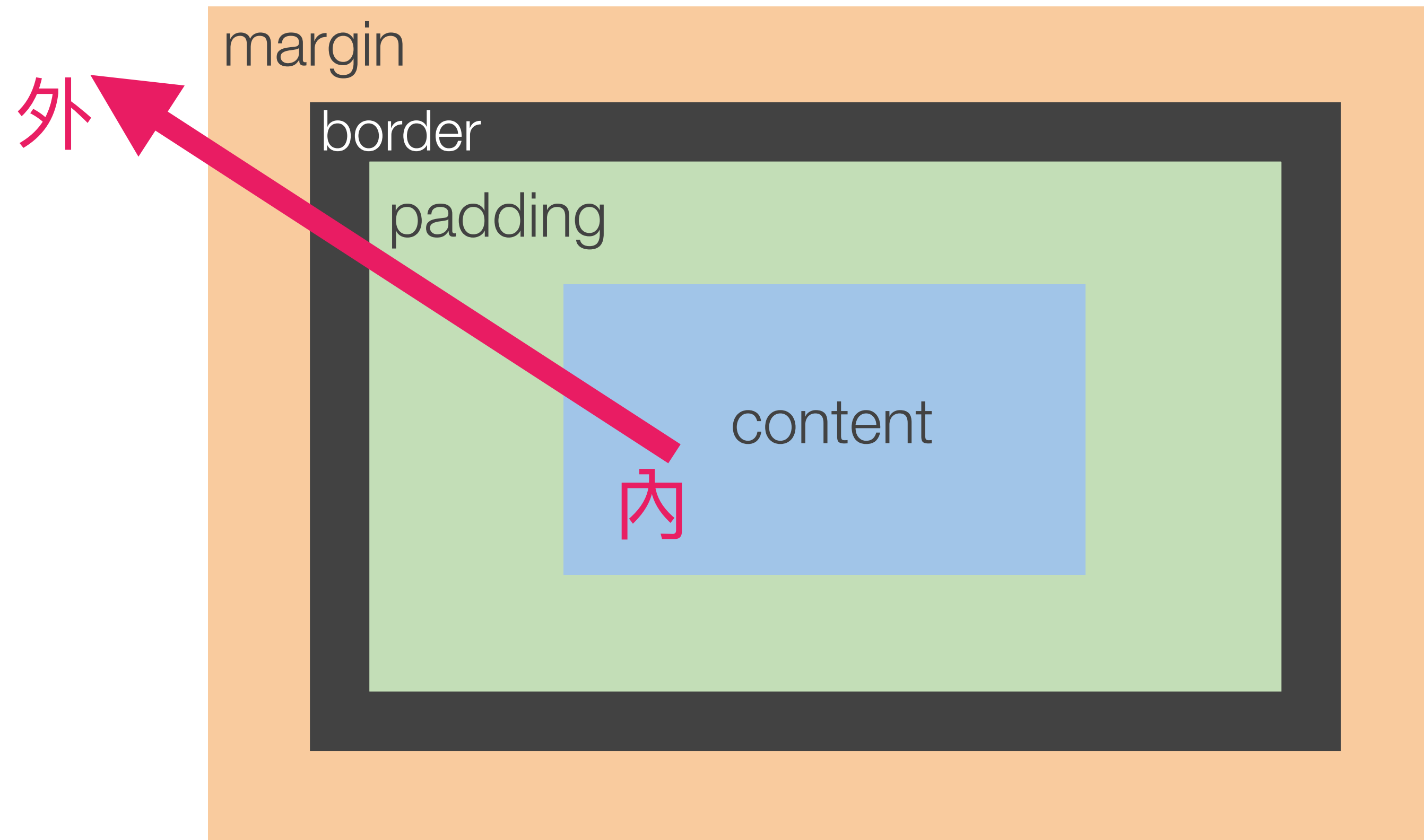
決定元素的大小：

每個元素被render後，預設會佔據一個矩形的面積，這個矩形稱為box。

CSS使用了Box Model來描述一個box。

Box Model將一個box分成4個部分：

content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

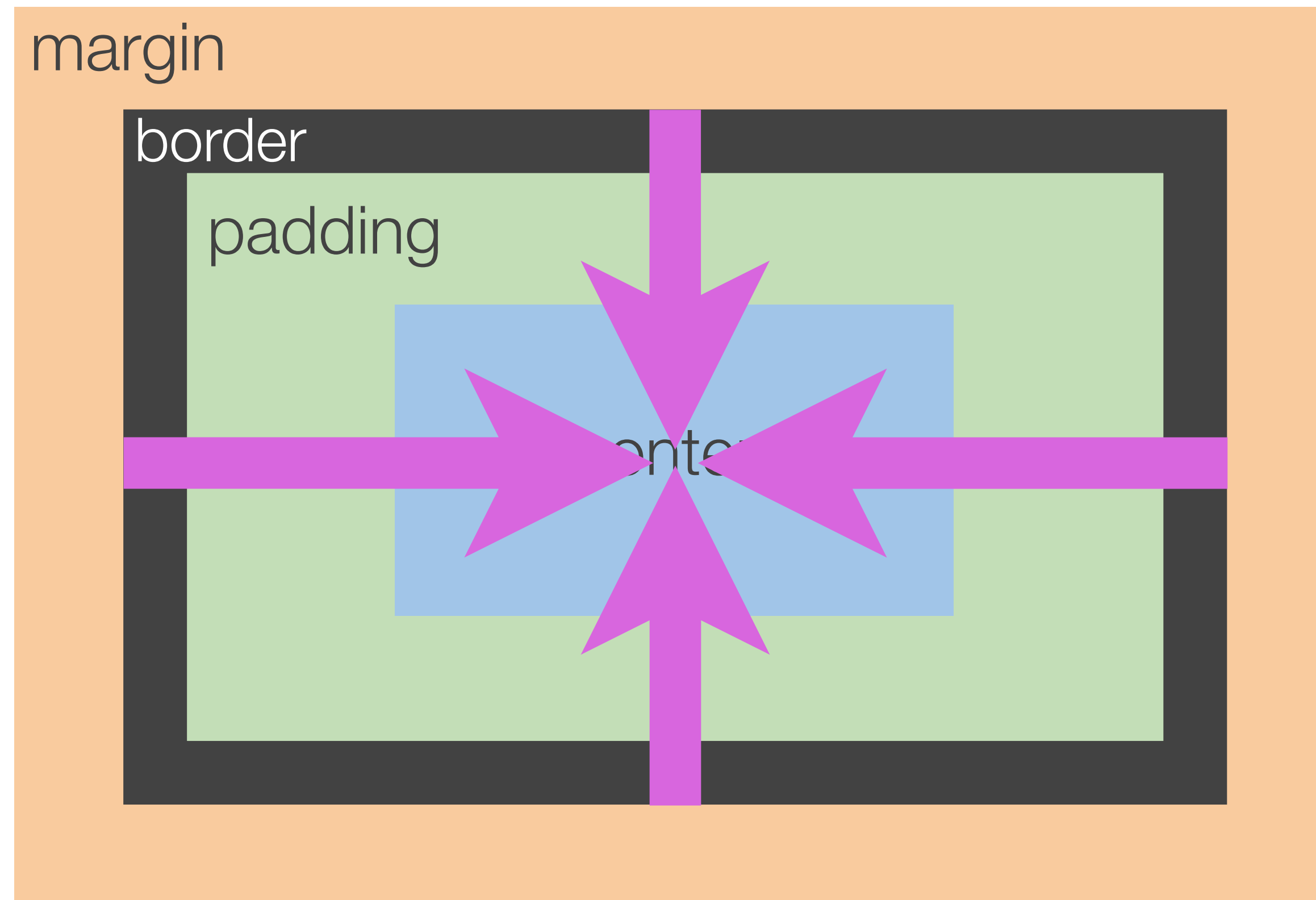
決定元素的大小：

每個元素被render後，預設會佔據一個矩形的面積，這個矩形稱為box。

CSS使用了Box Model來描述一個box。

Box Model將一個box分成4個部分：

content (內容), padding (內距), border (邊界), margin (外距)



元素實際上所佔據的面積，指的是border以內的範圍，也就是：
border + padding + content

CSS如何決定版面配置 (Layout) ?

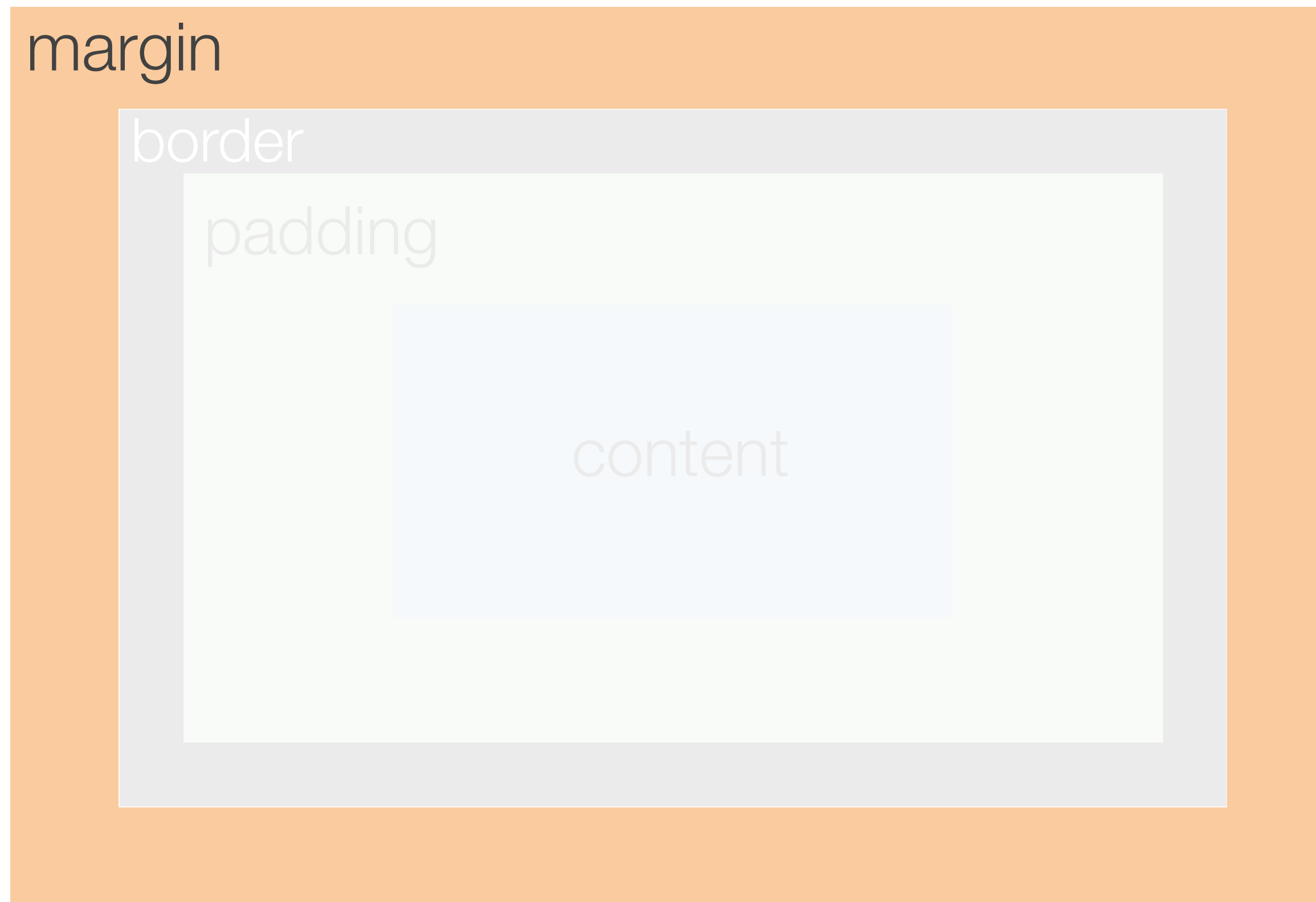
決定元素的大小：

每個元素被render後，預設會佔據一個矩形的面積，這個矩形稱為box。

CSS使用了Box Model來描述一個box。

Box Model將一個box分成4個部分：

content (內容), padding (內距), border (邊界), margin (外距)

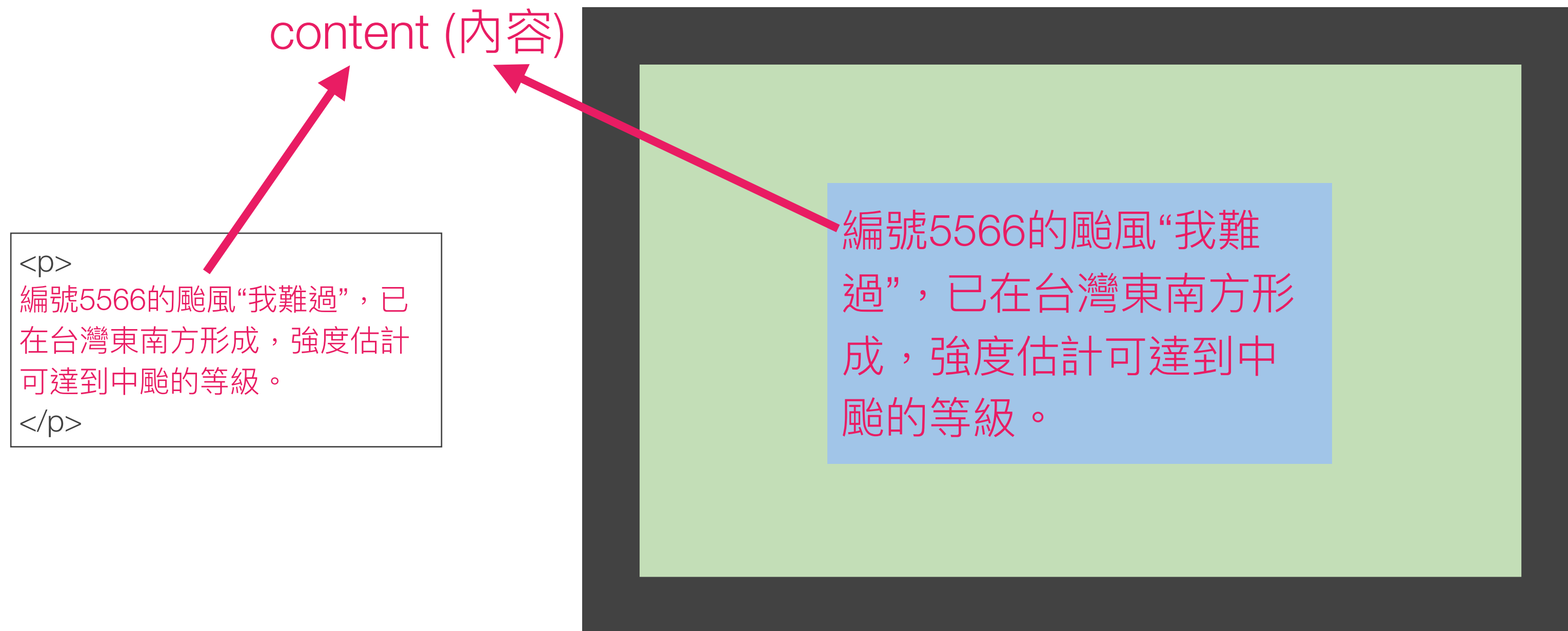


最外面的這個淺橘色區域，稱為margin，它並不屬於元素所佔據的範圍，margin指的是目前元素和其他元素之間的距離。

CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

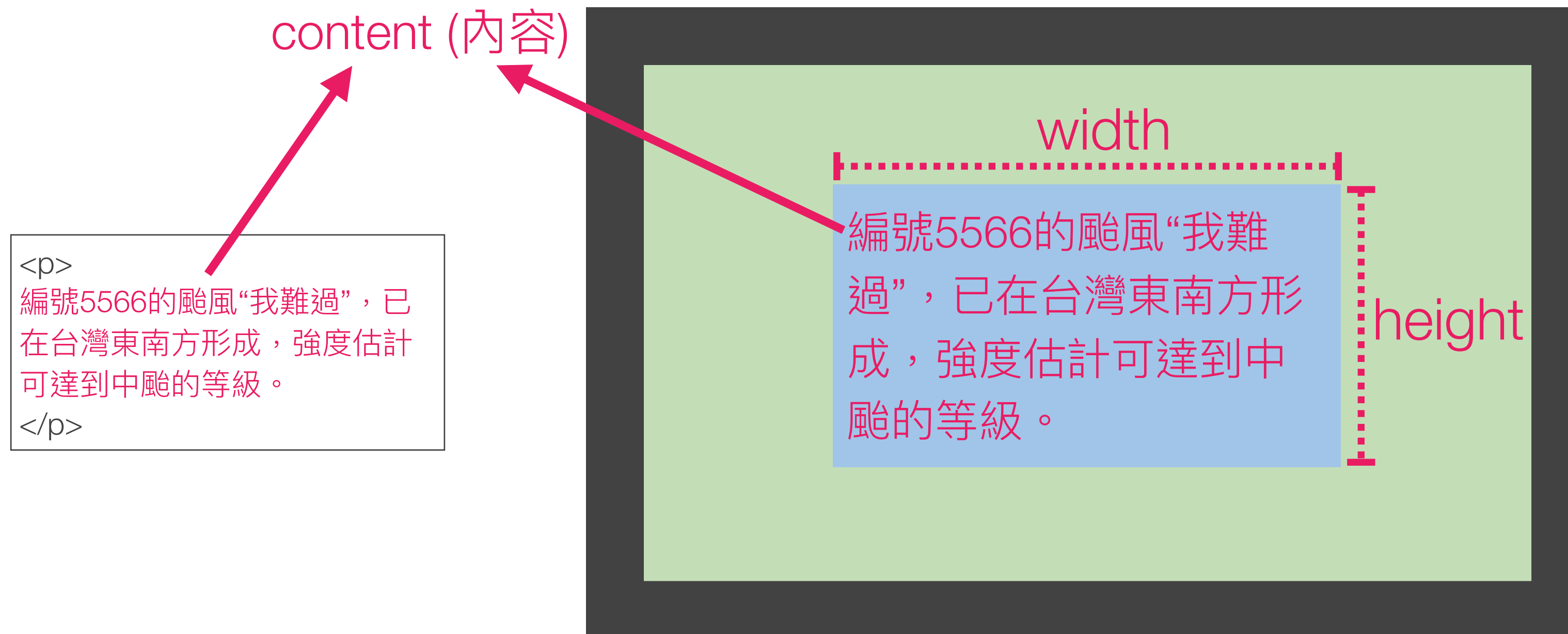
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

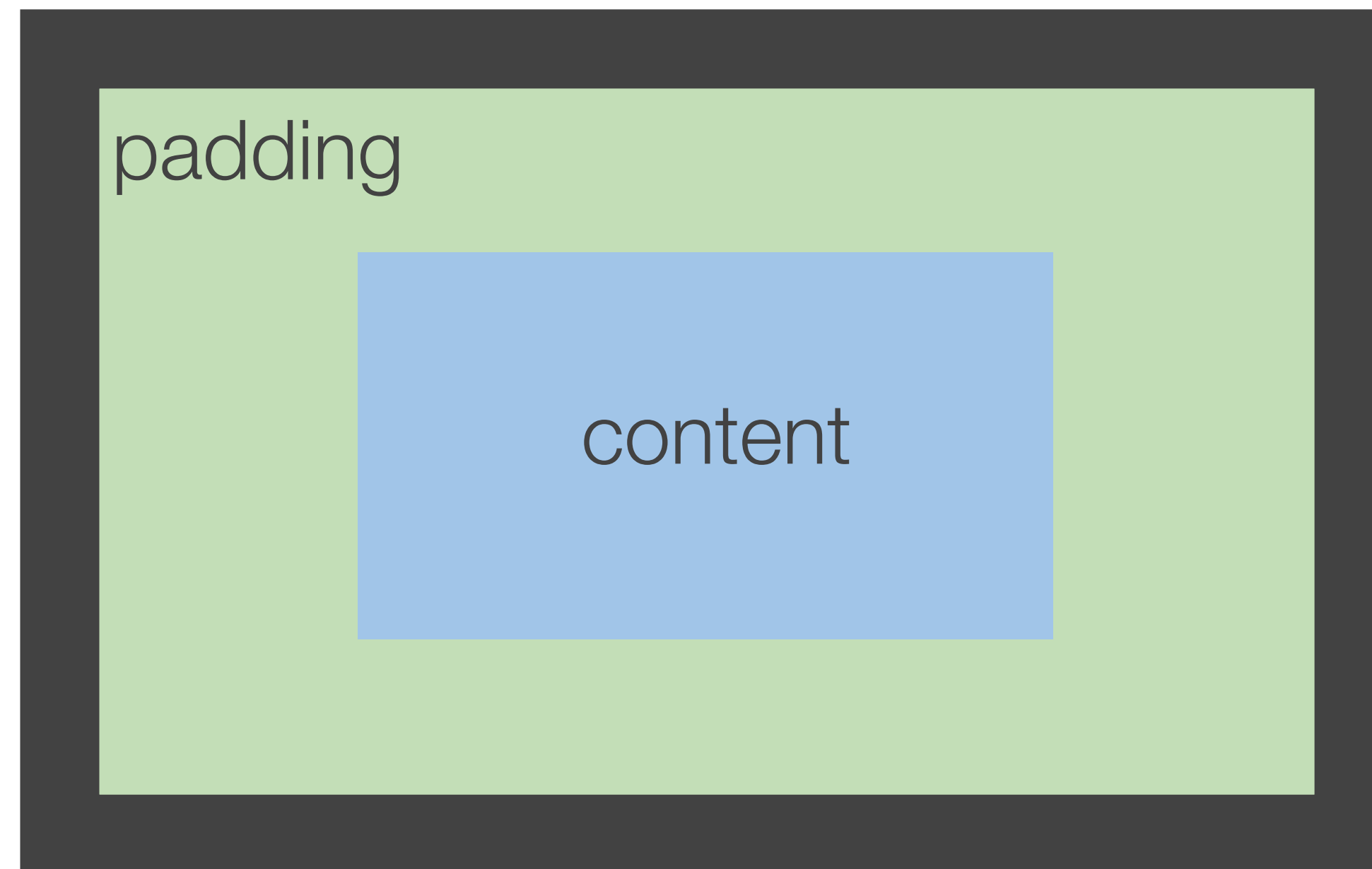
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

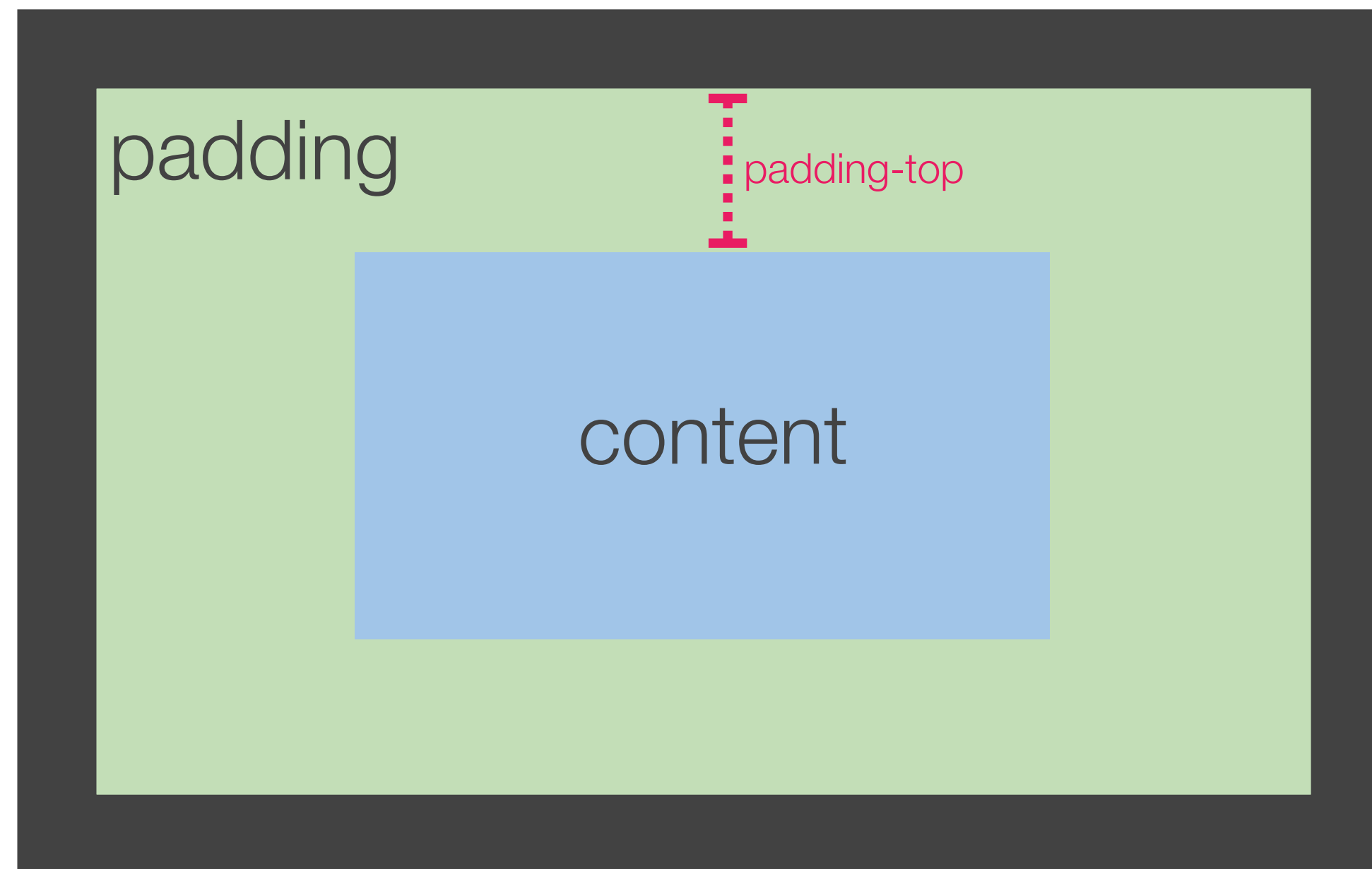
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

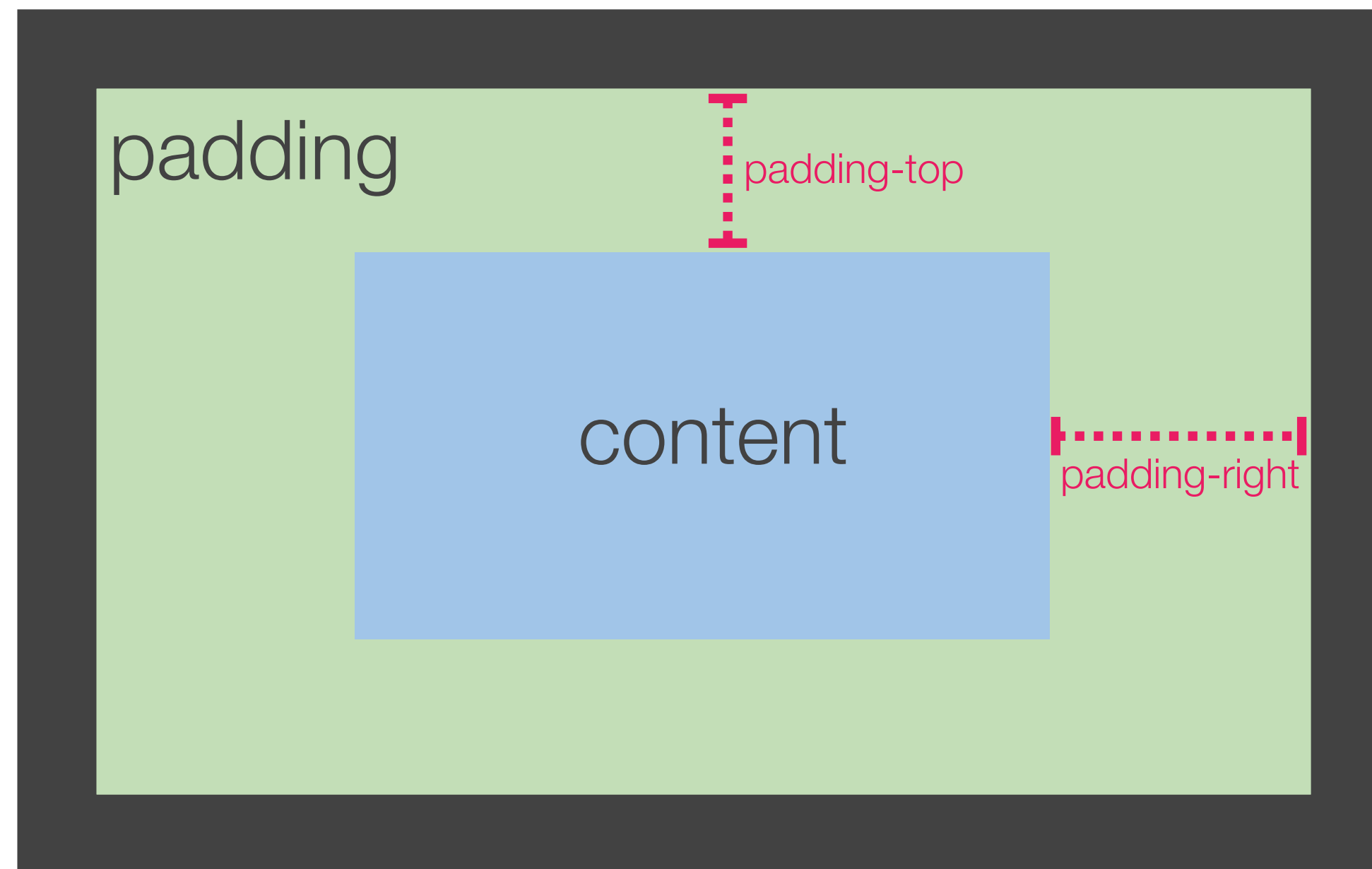
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

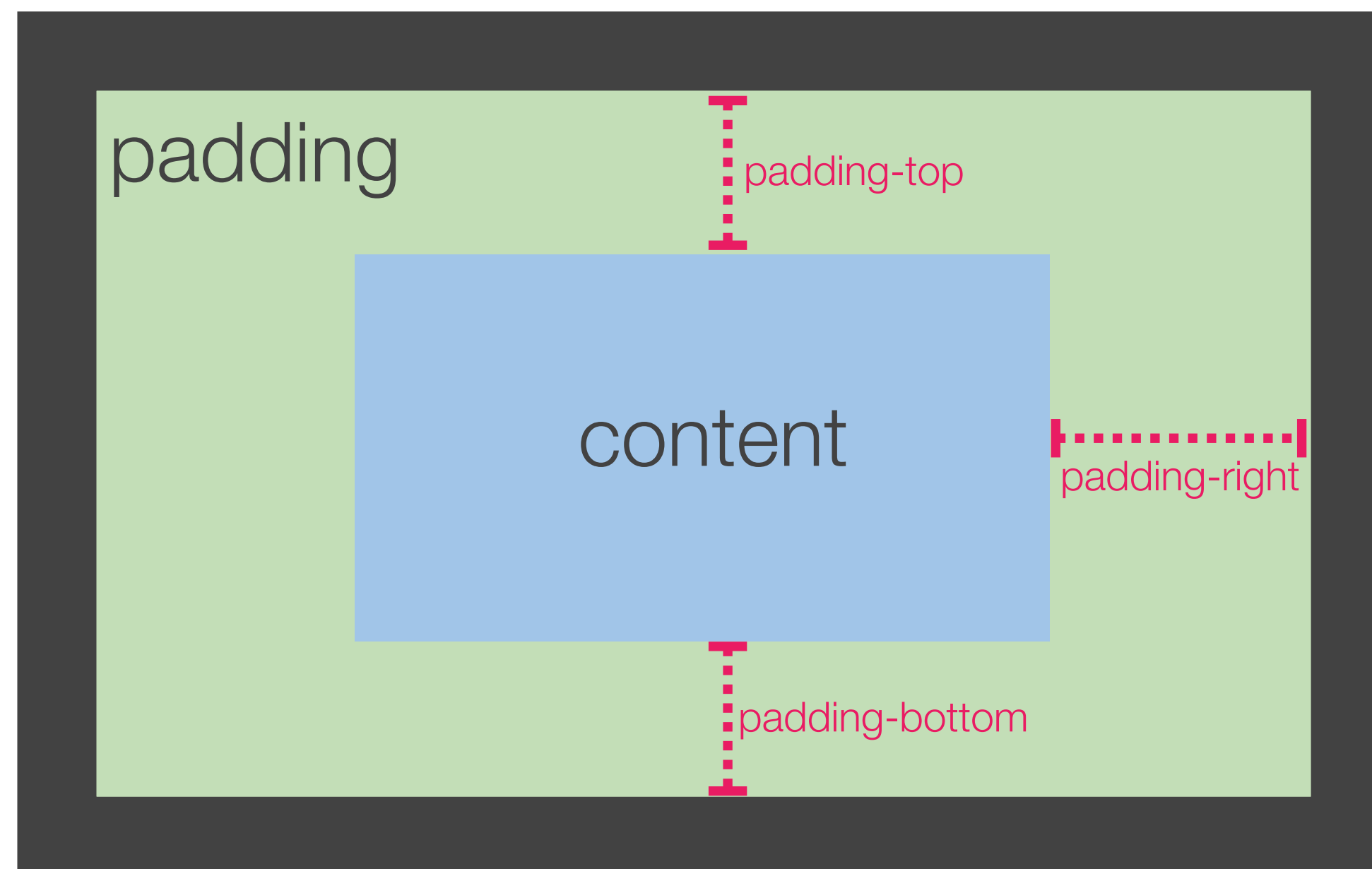
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

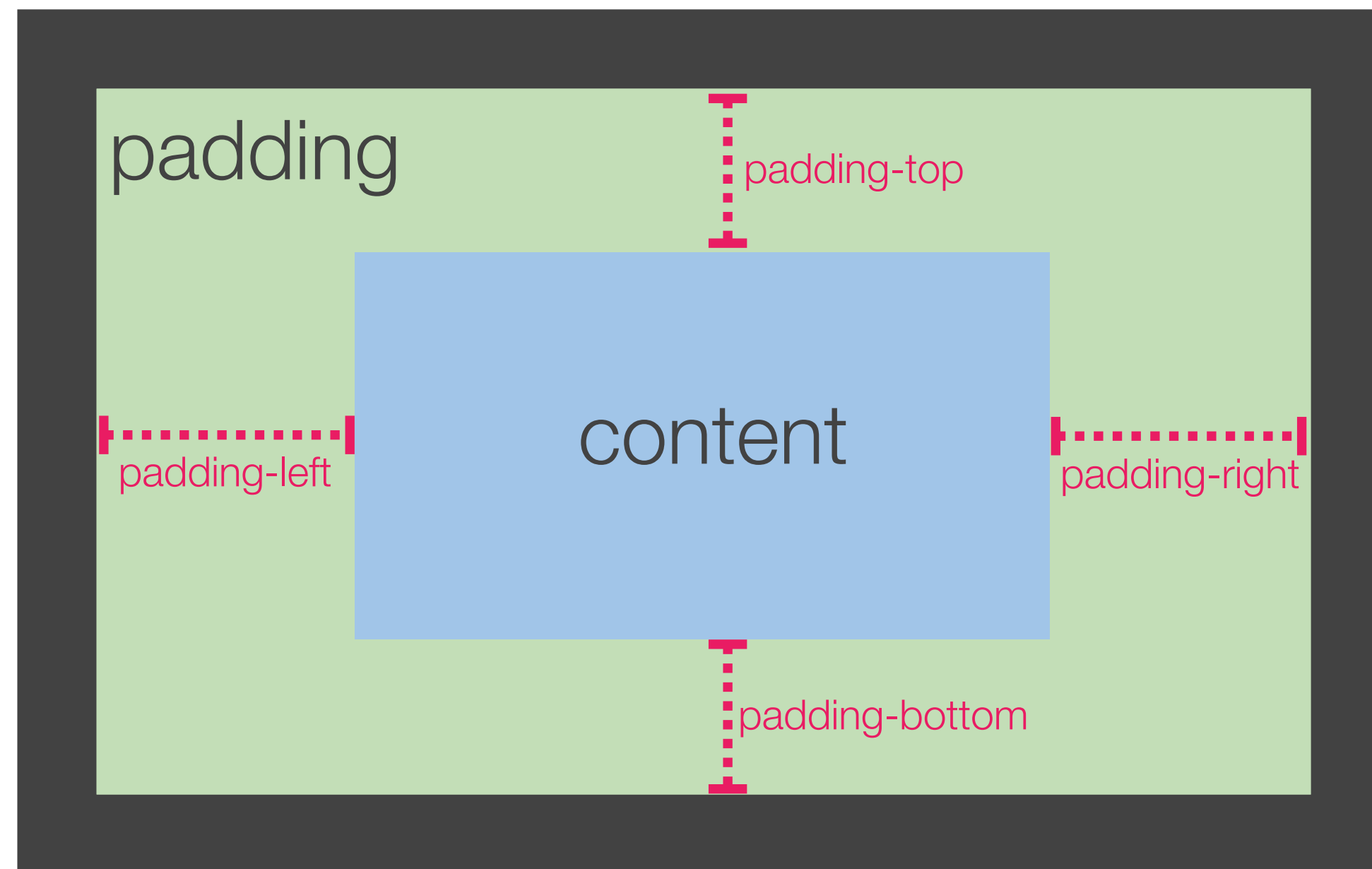
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

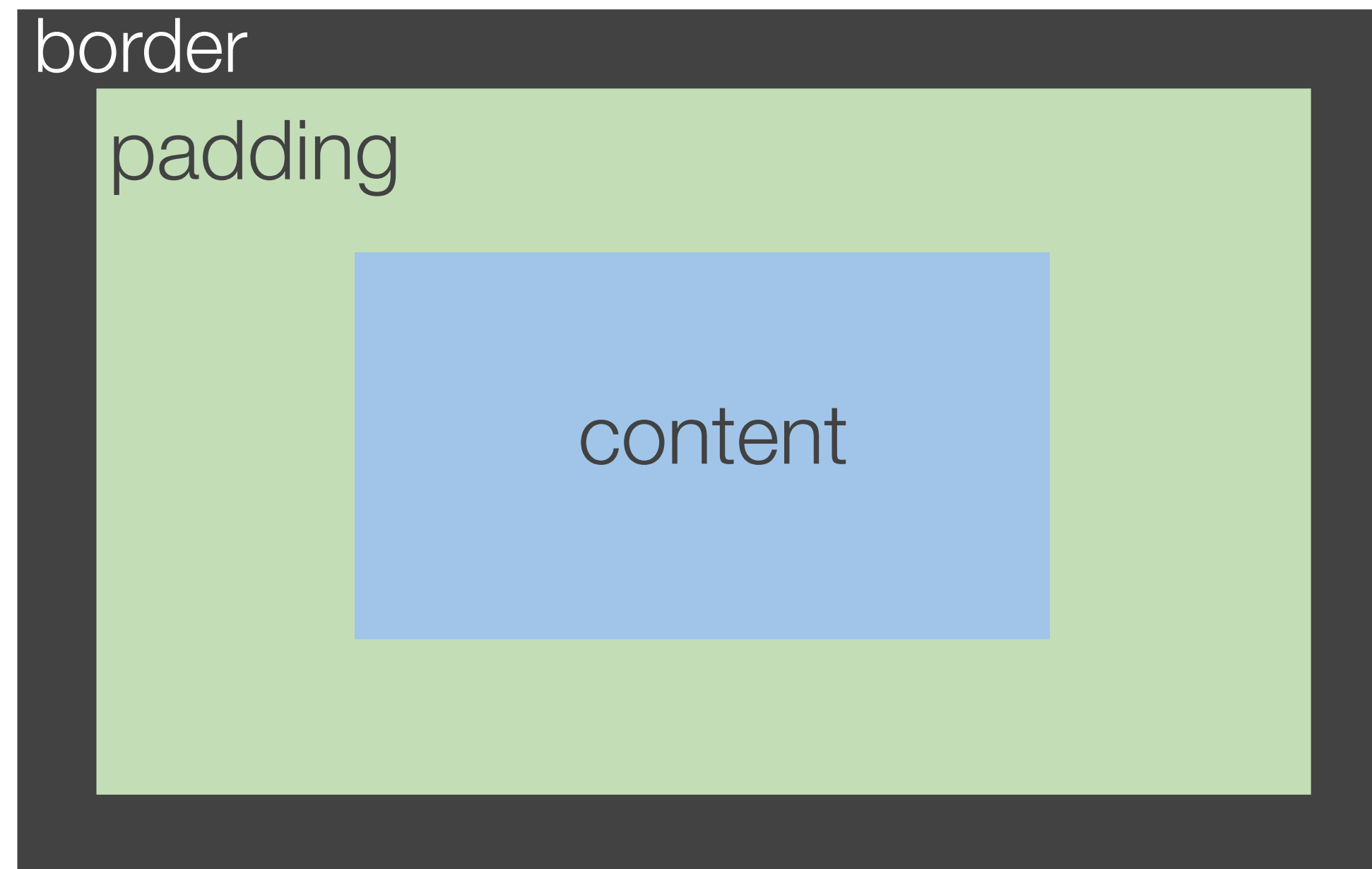
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

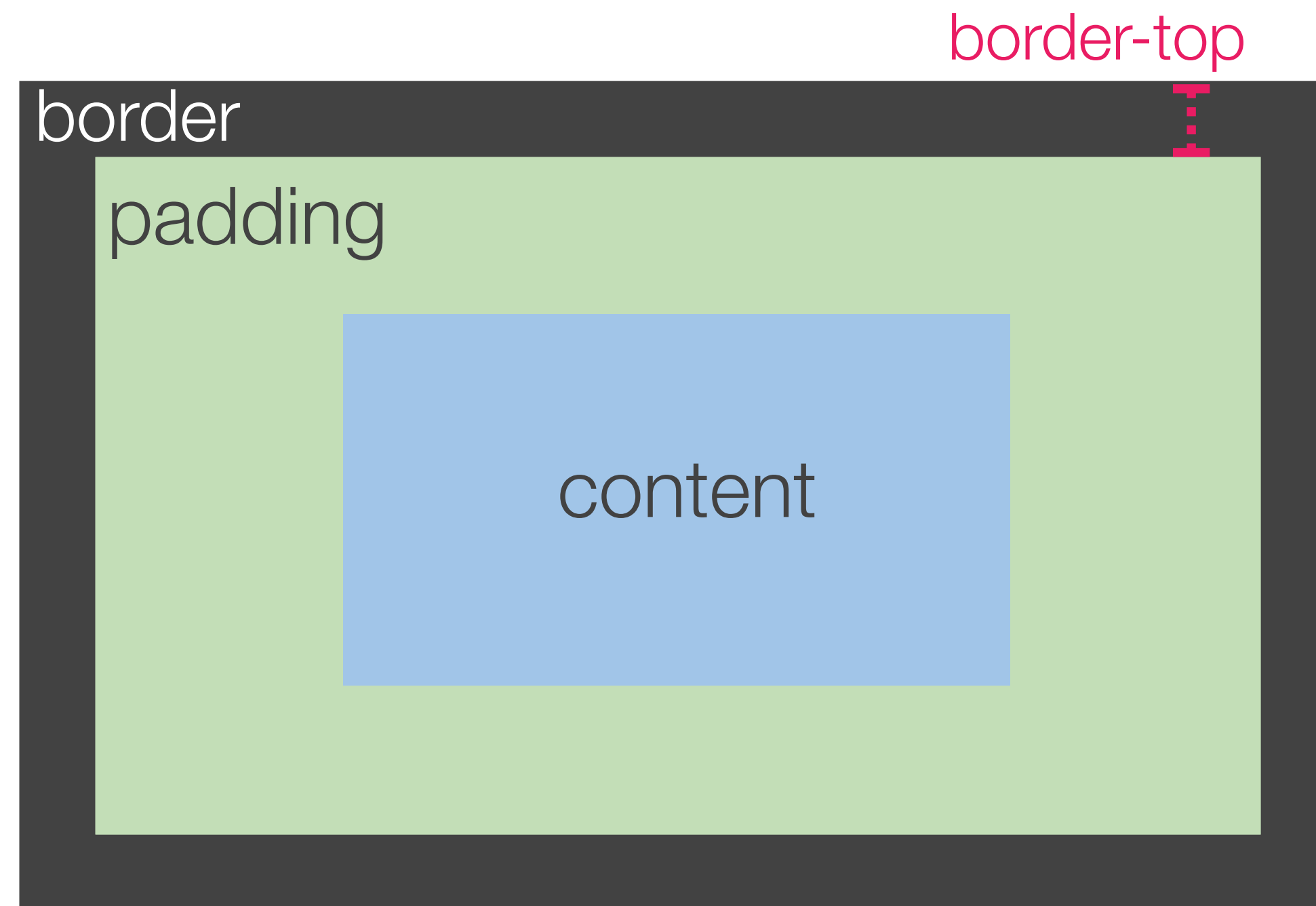
content (內容), padding (內距), **border (邊界)**, margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

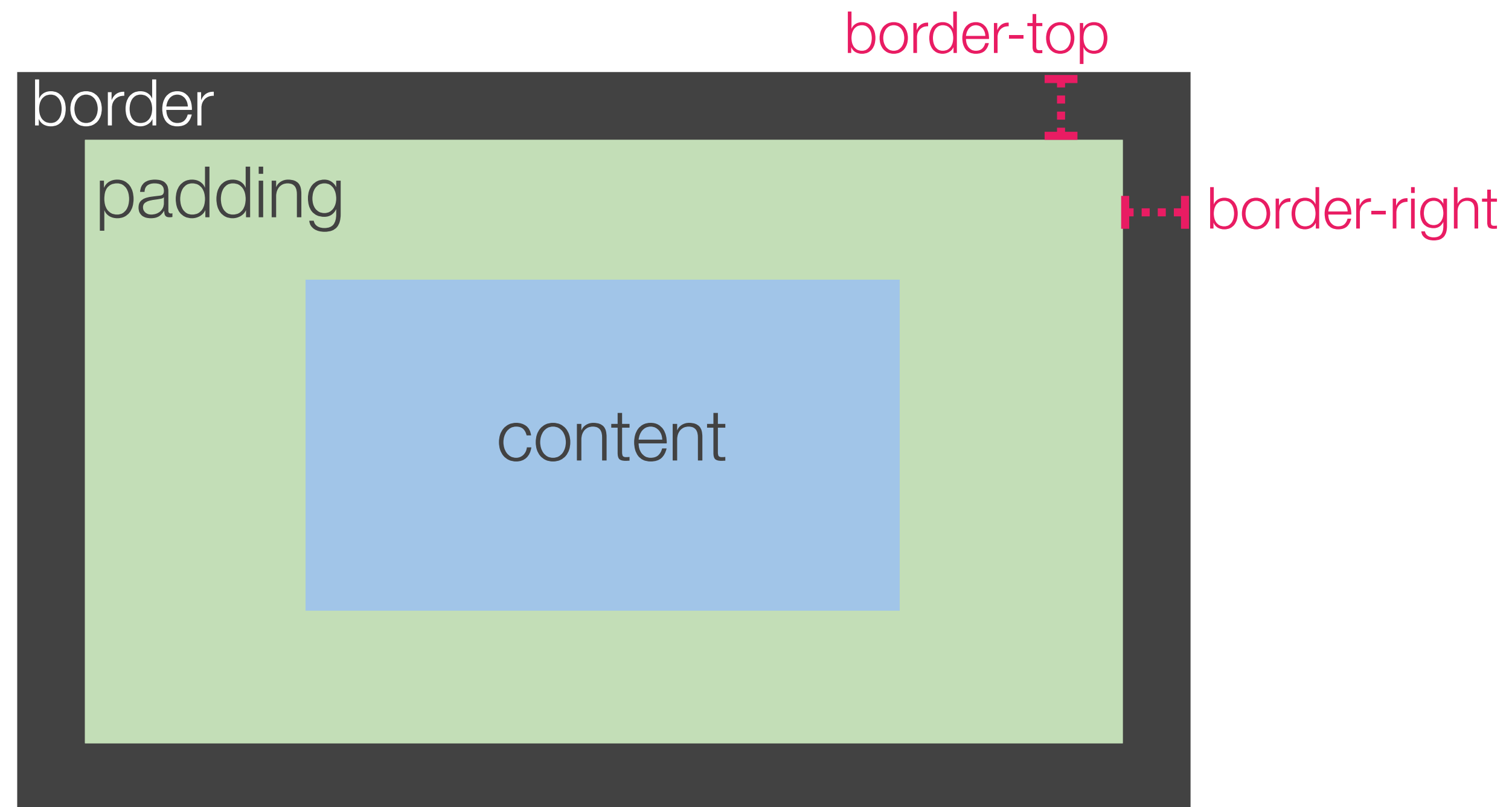
content (內容), padding (內距), **border (邊界)**, margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

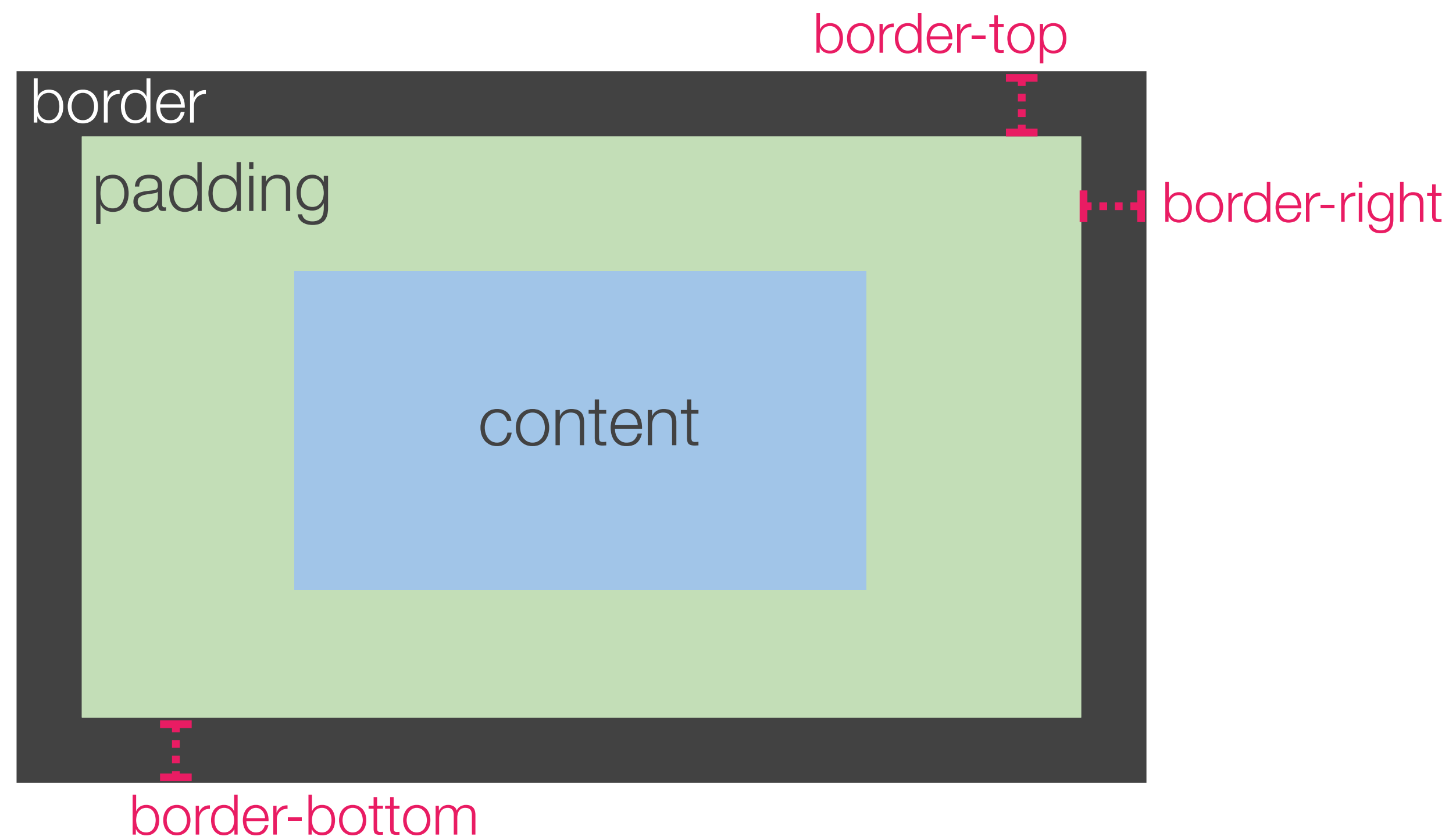
content (內容), padding (內距), **border (邊界)**, margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

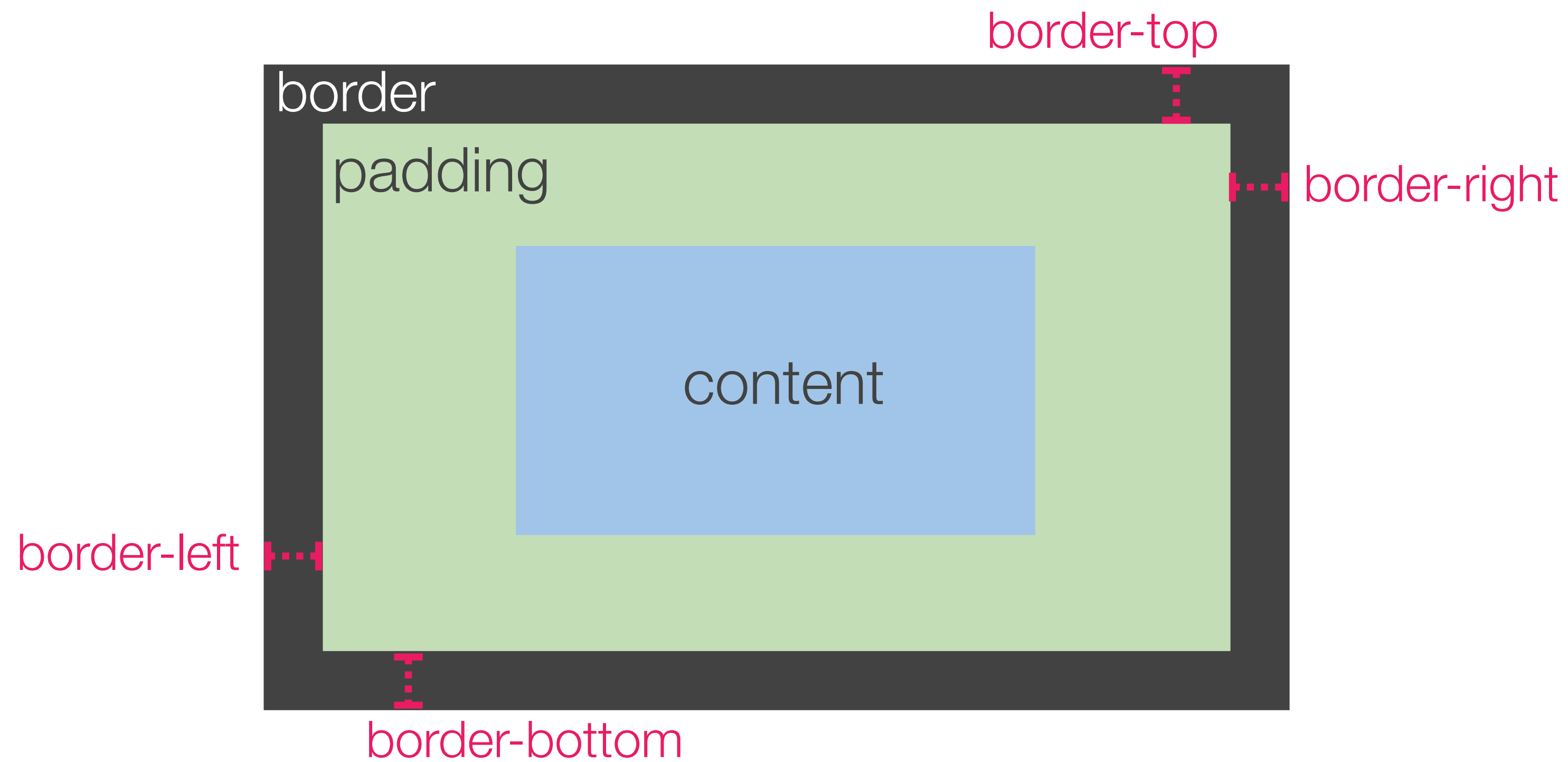
content (內容), padding (內距), **border (邊界)**, margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

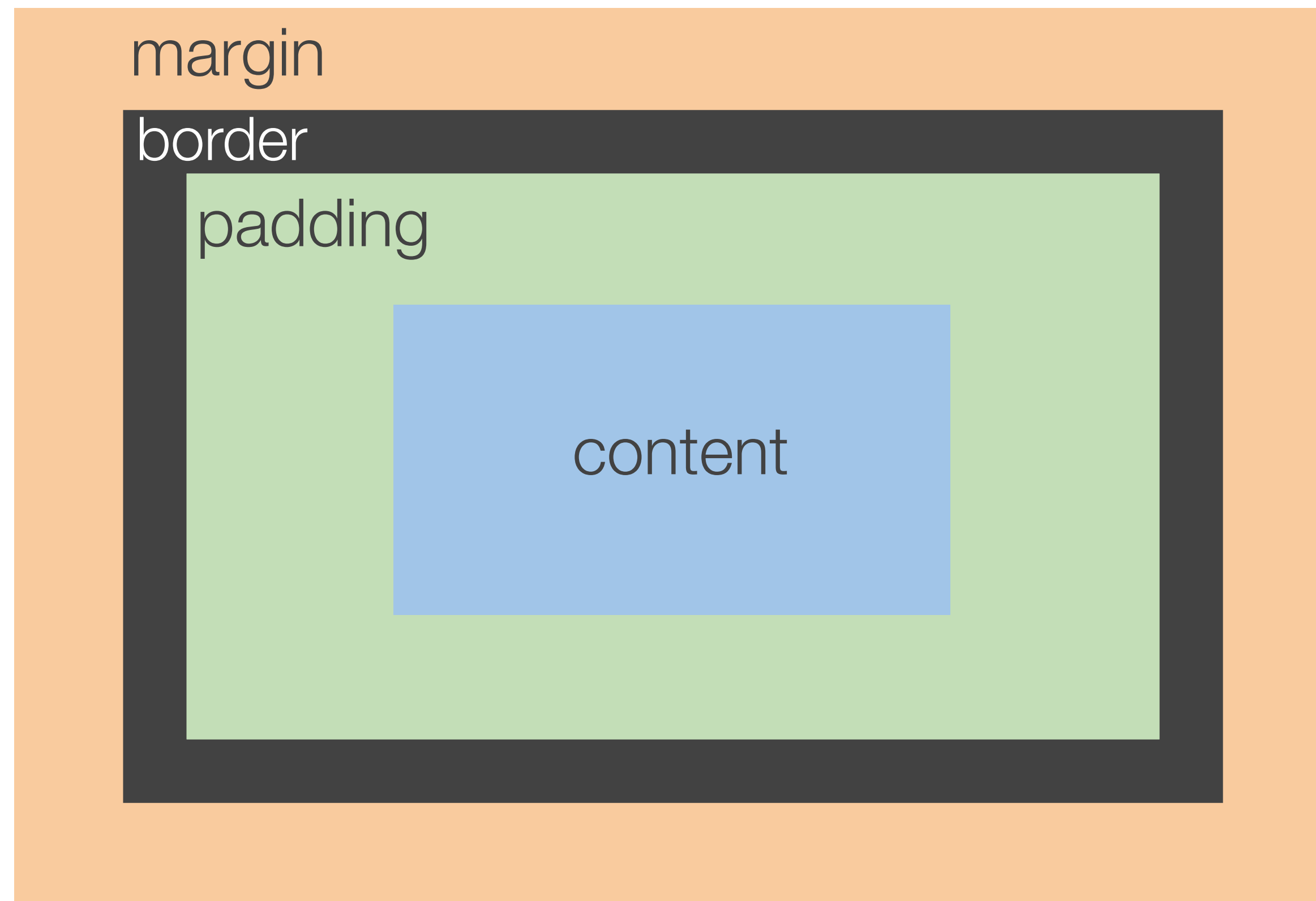
content (內容), padding (內距), **border (邊界)**, margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

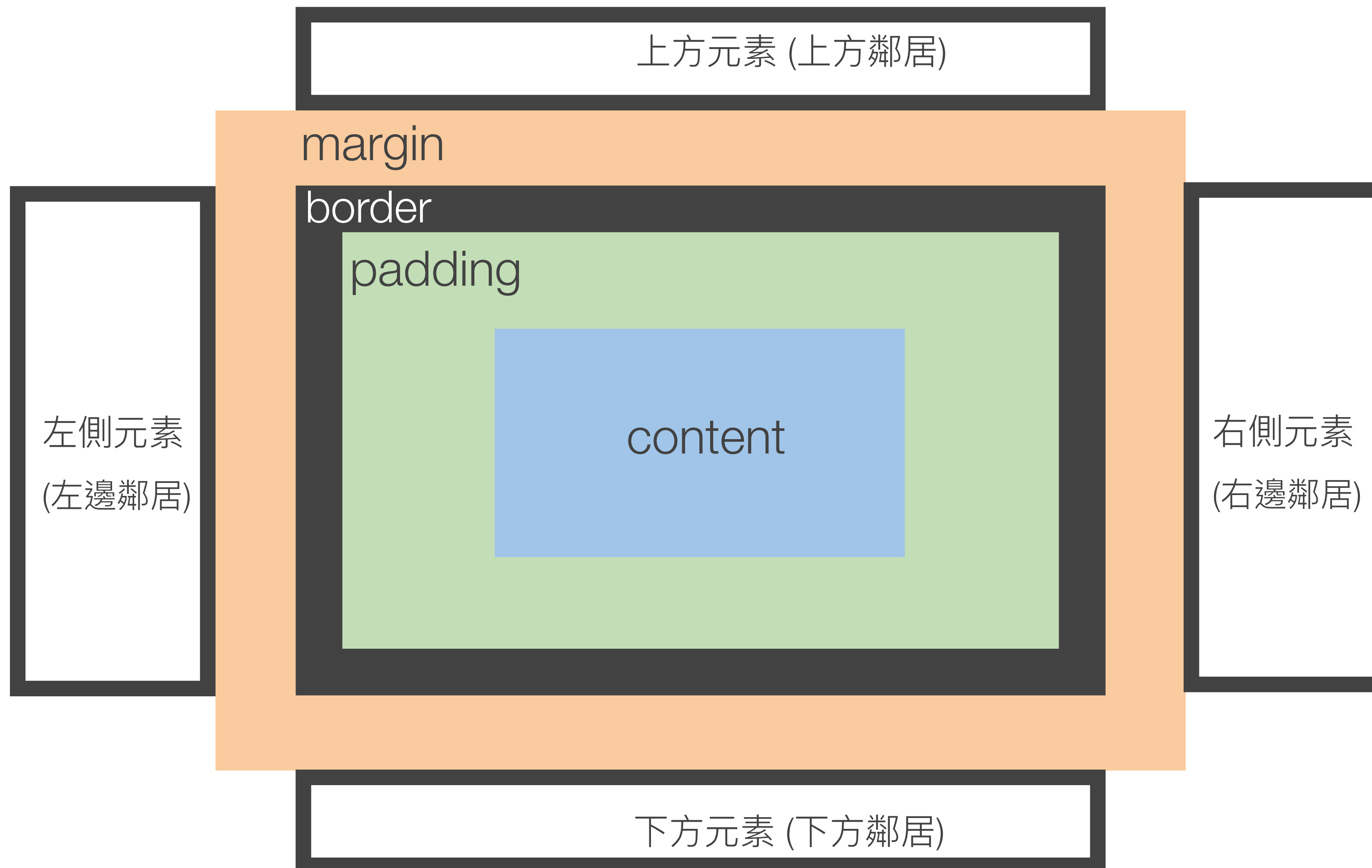
content (內容), padding (內距), border (邊界), **margin (外距)**



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

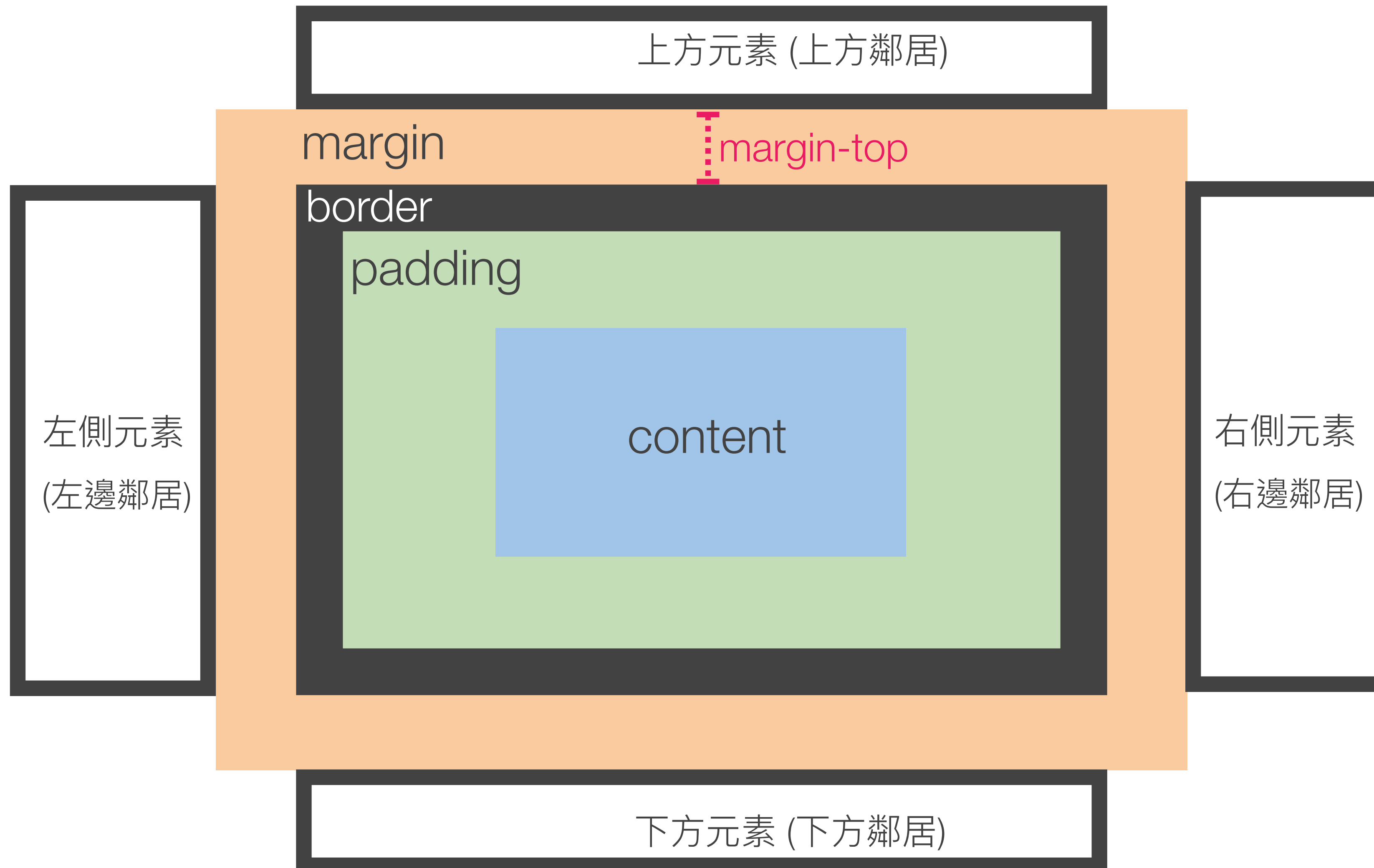
content (內容), padding (內距), border (邊界), **margin (外距)**



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

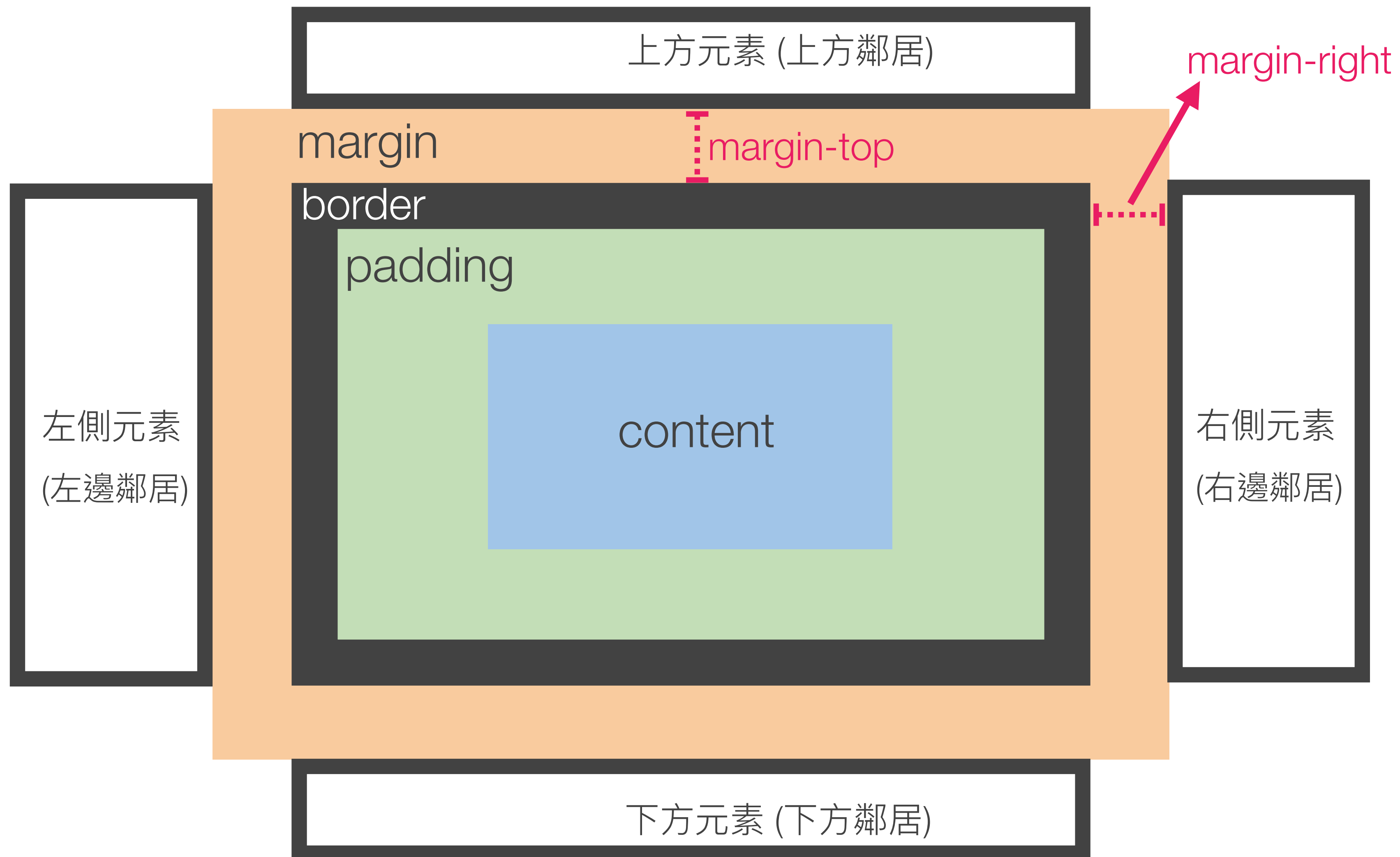
content (內容), padding (內距), border (邊界), **margin (外距)**



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

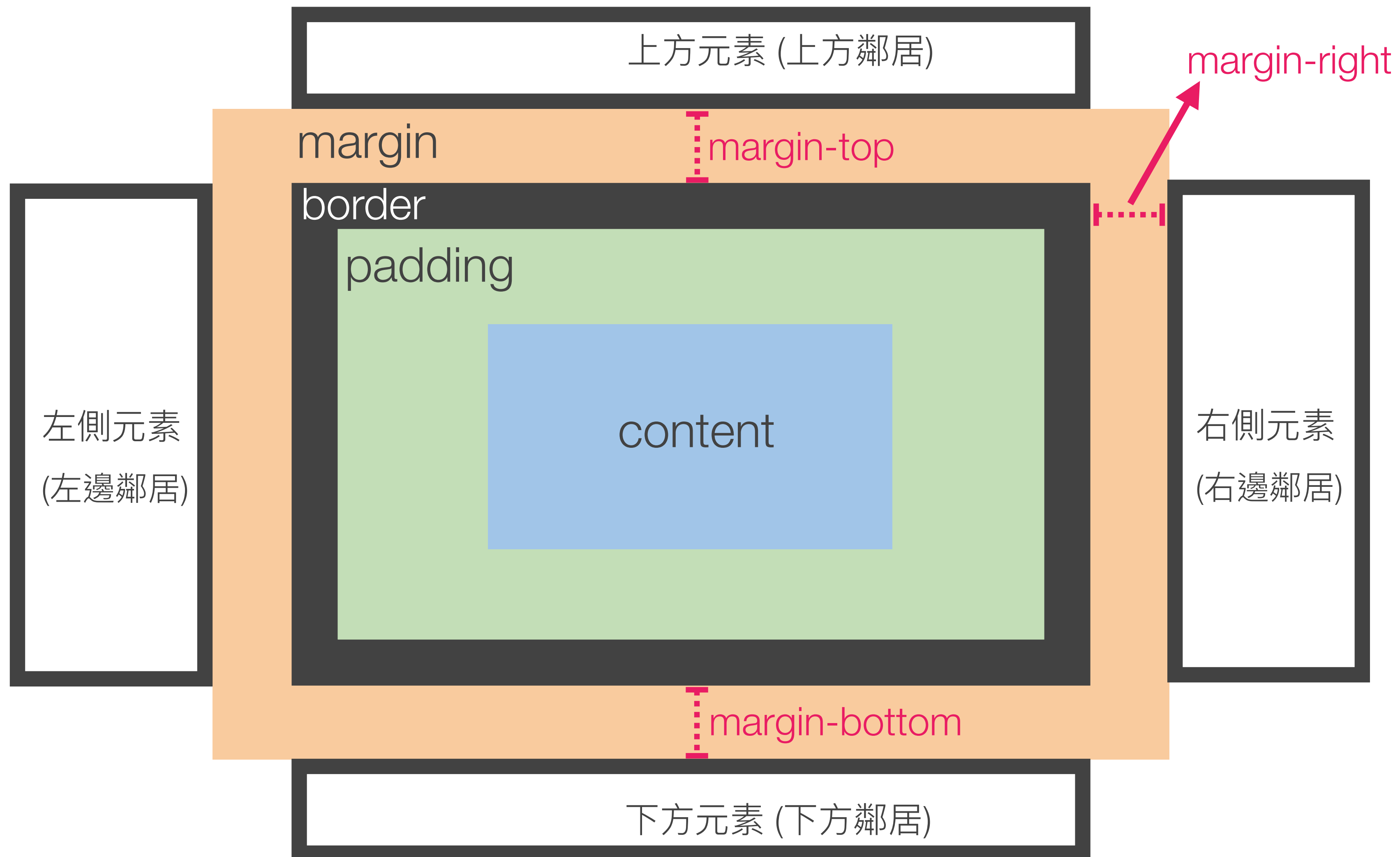
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

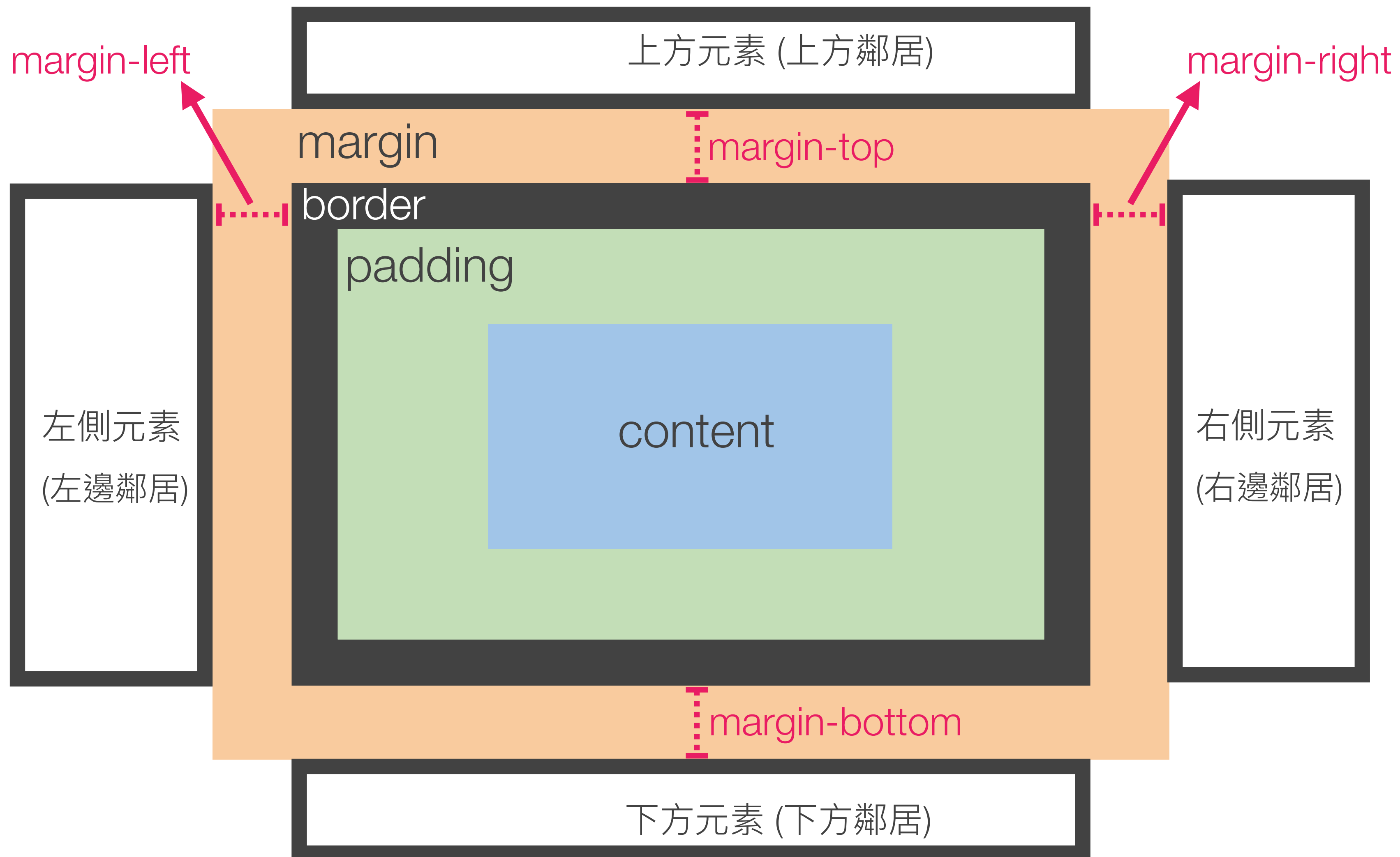
content (內容), padding (內距), border (邊界), margin (外距)



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

content (內容), padding (內距), border (邊界), margin (外距)



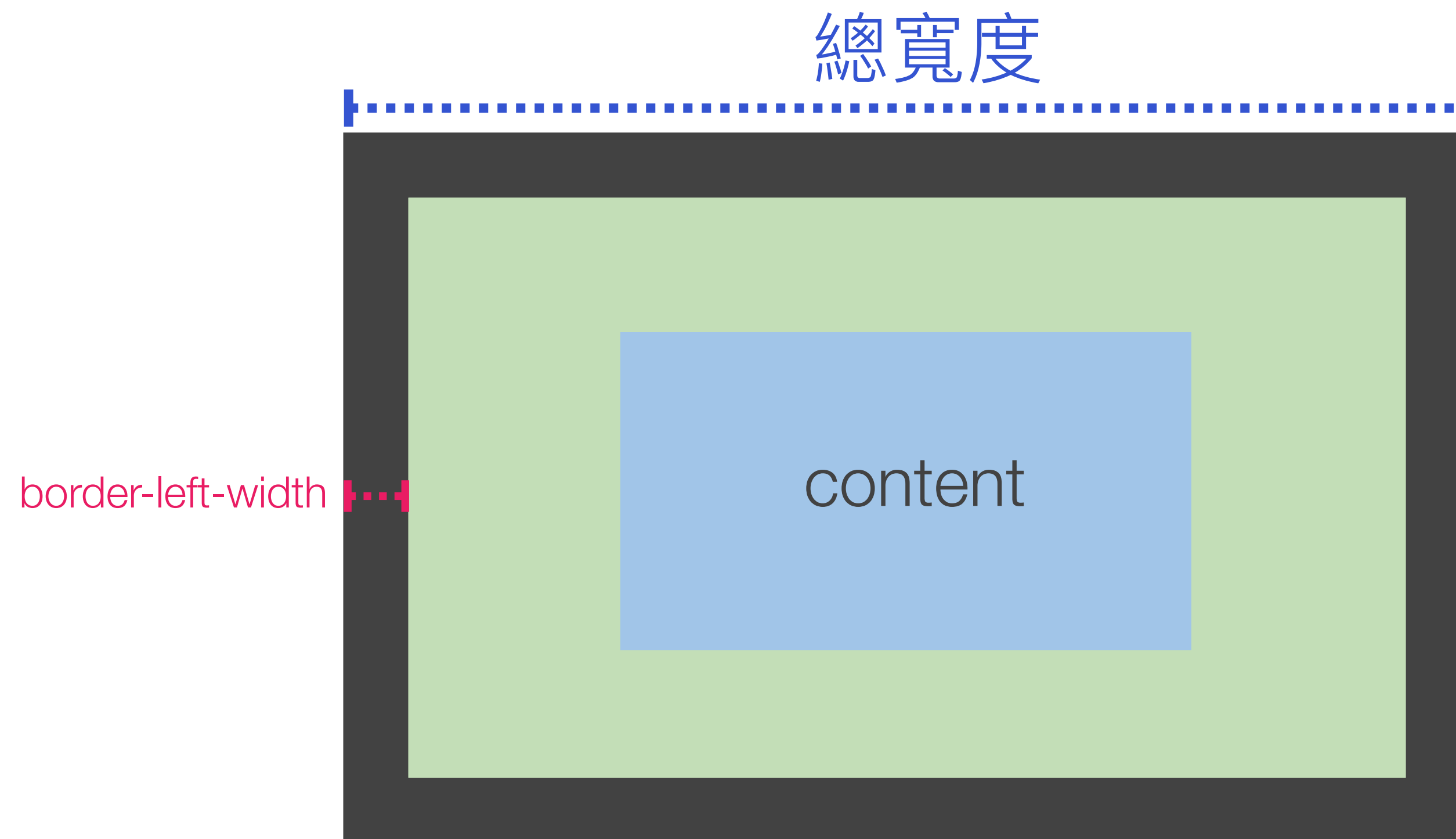
CSS如何決定版面配置 (Layout) ?

元素的總寬度 =



CSS如何決定版面配置 (Layout) ?

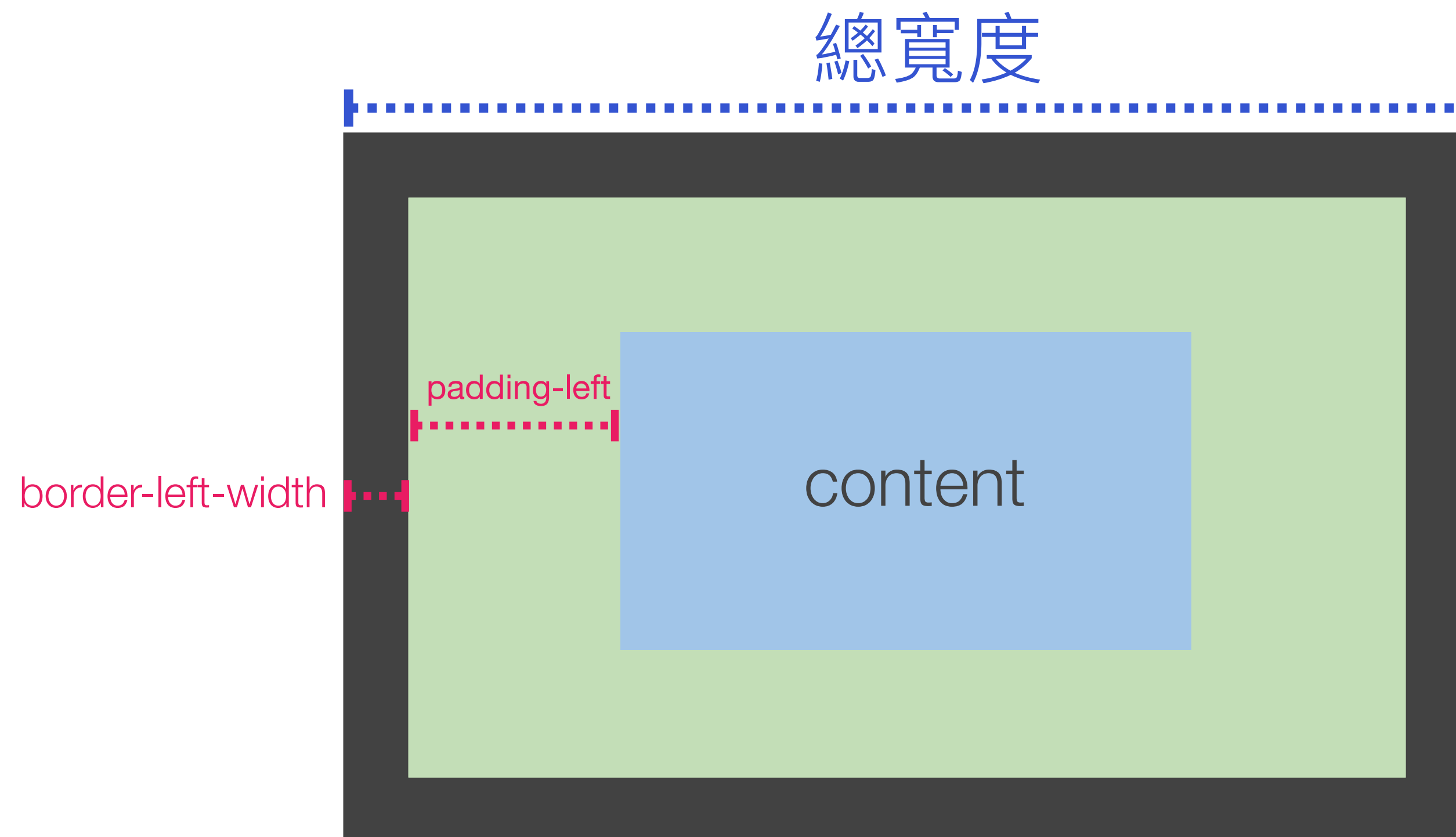
元素的總寬度 =
border-left-width



CSS如何決定版面配置 (Layout) ?

元素的總寬度 =

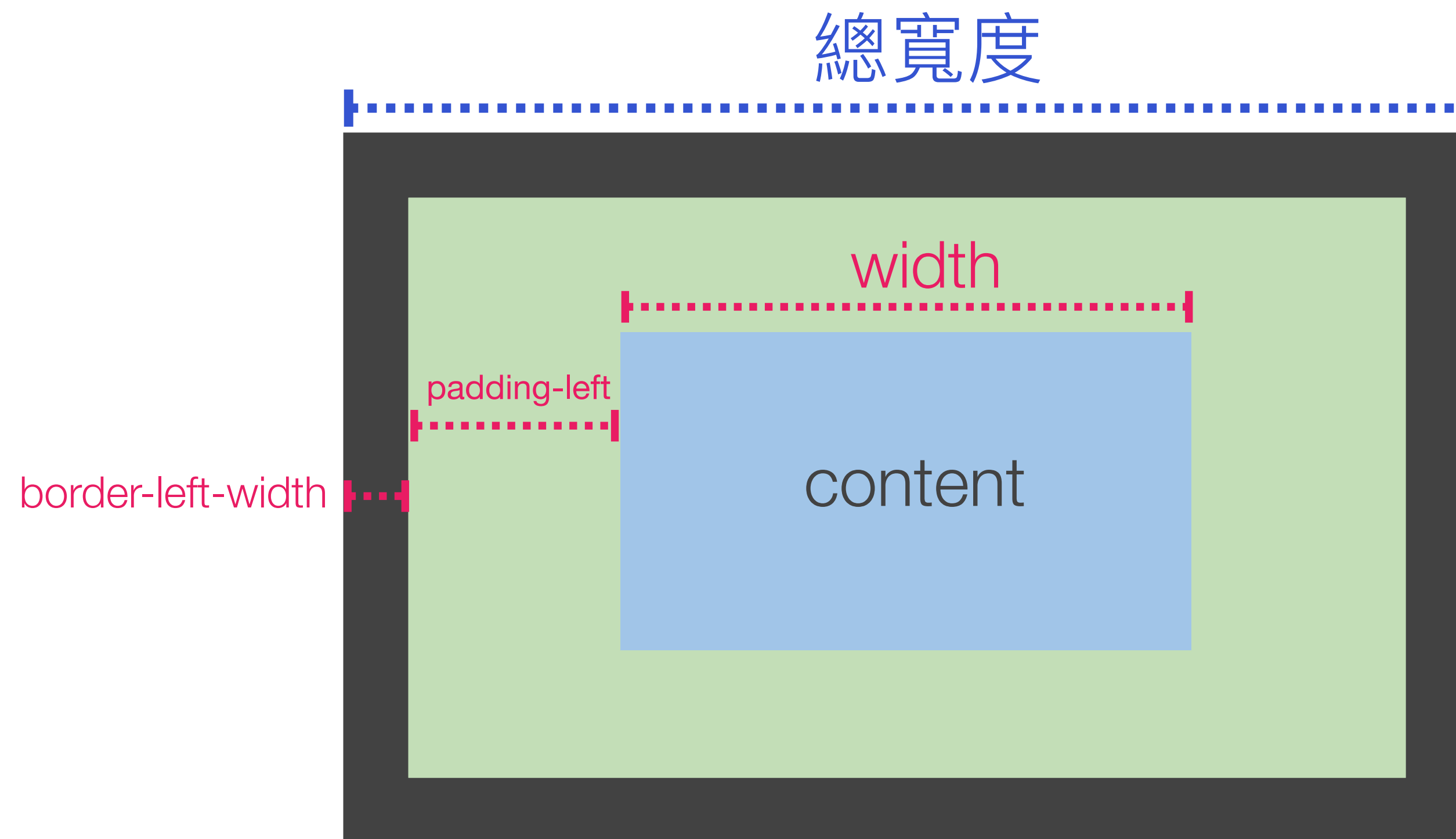
`border-left-width + padding-left`



CSS如何決定版面配置 (Layout) ?

元素的總寬度 =

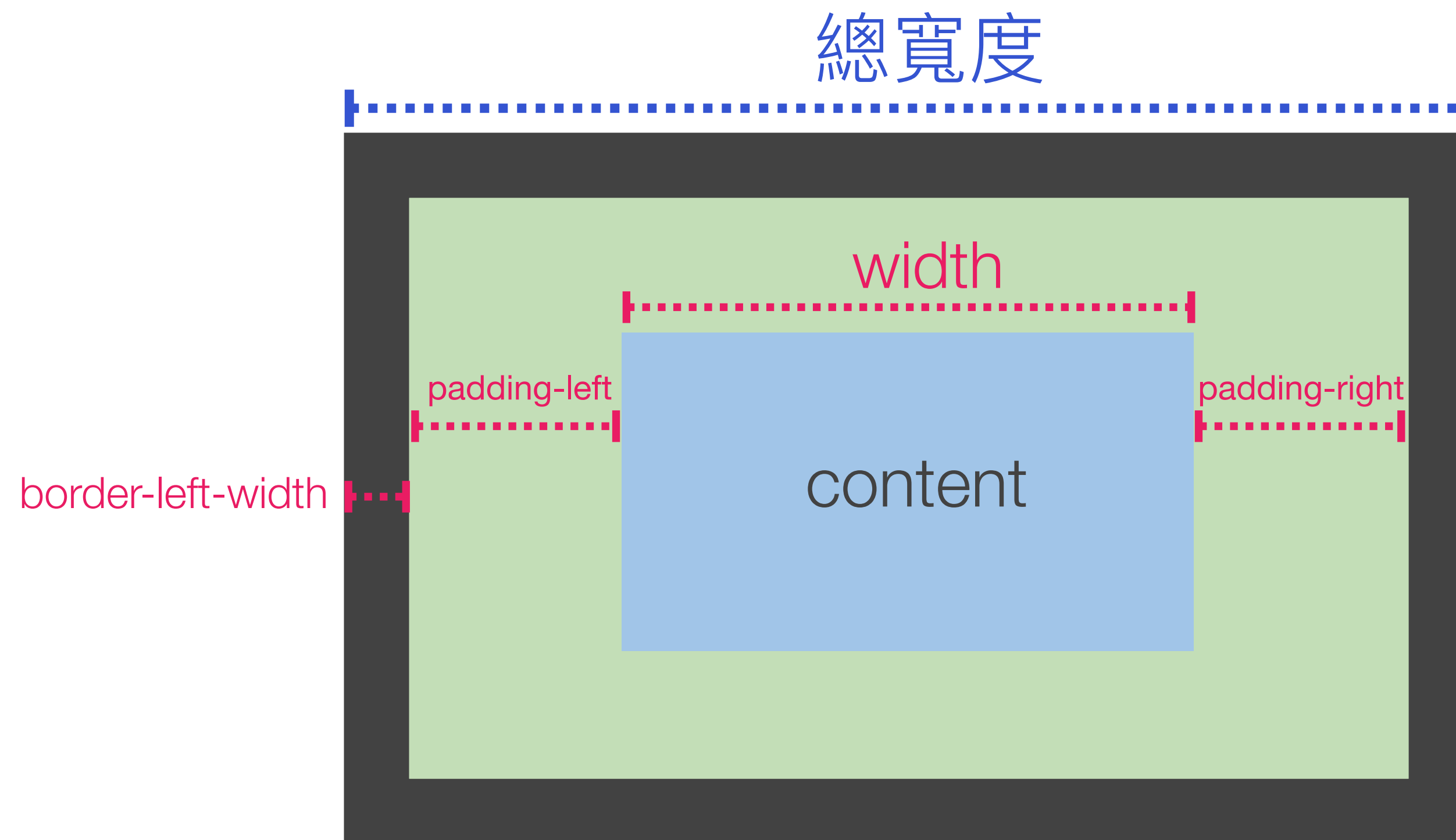
`border-left-width + padding-left + width`



CSS如何決定版面配置 (Layout) ?

元素的總寬度 =

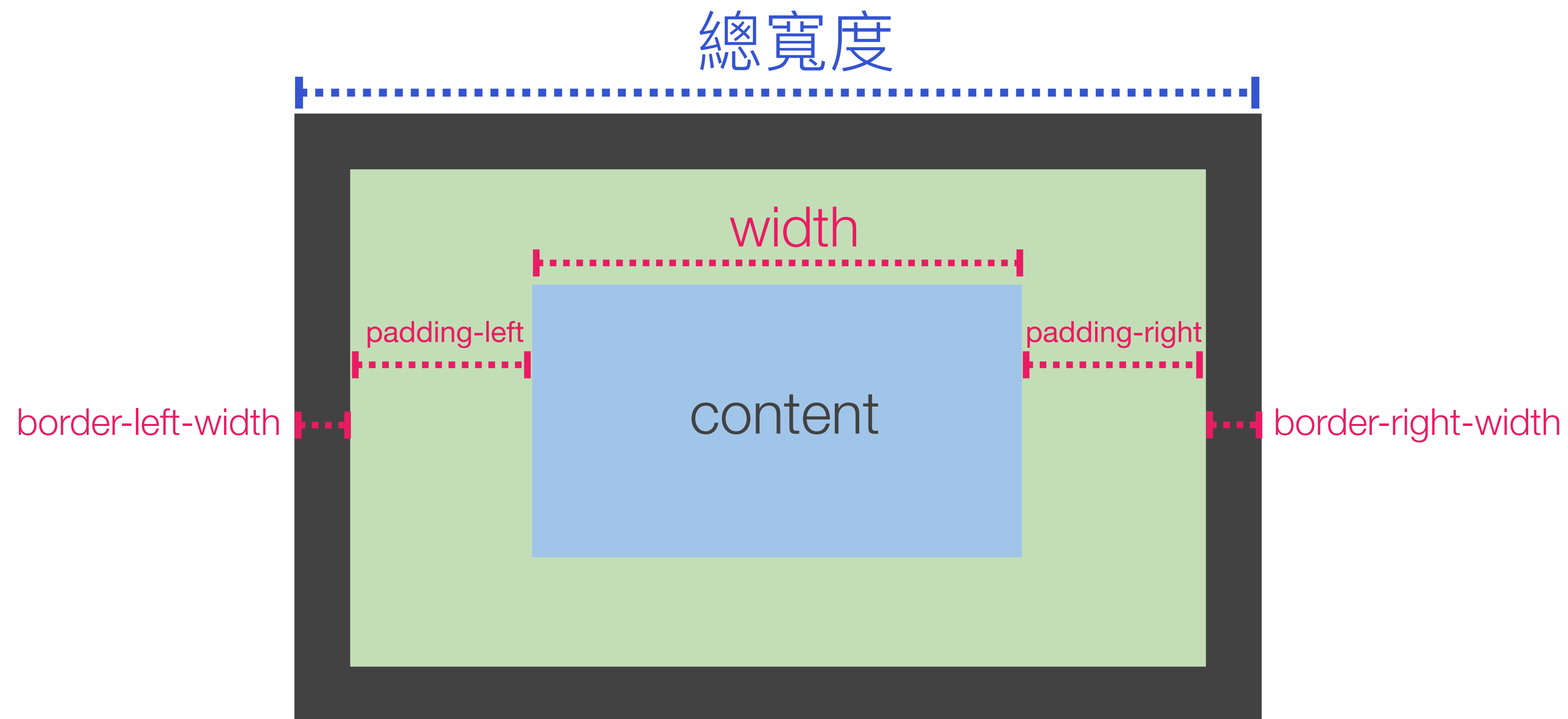
`border-left-width + padding-left + width + padding-right`



CSS如何決定版面配置 (Layout) ?

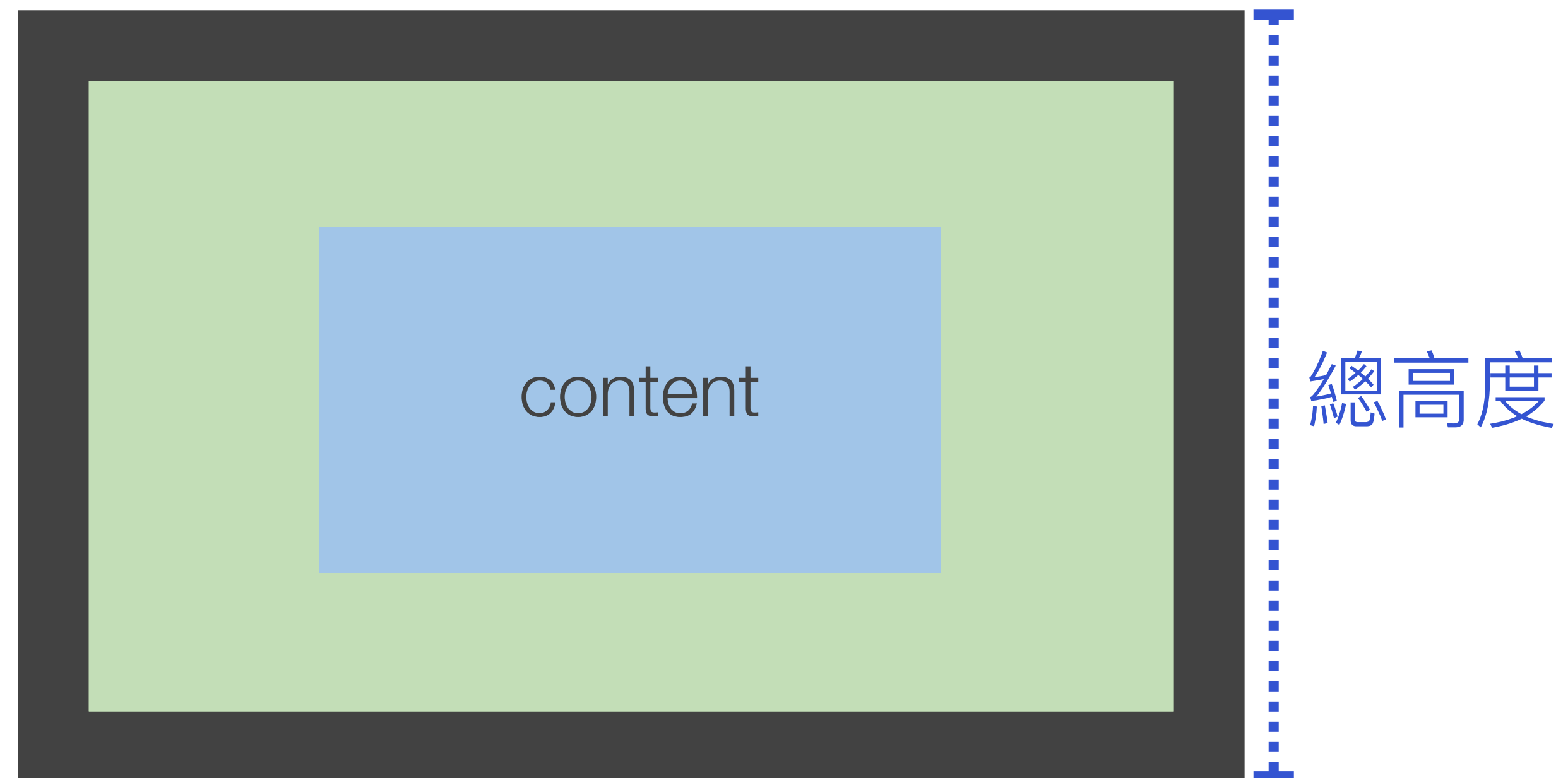
元素的總寬度 =

$\text{border-left-width} + \text{padding-left} + \text{width} + \text{padding-right} + \text{border-right-width}$



CSS如何決定版面配置 (Layout) ?

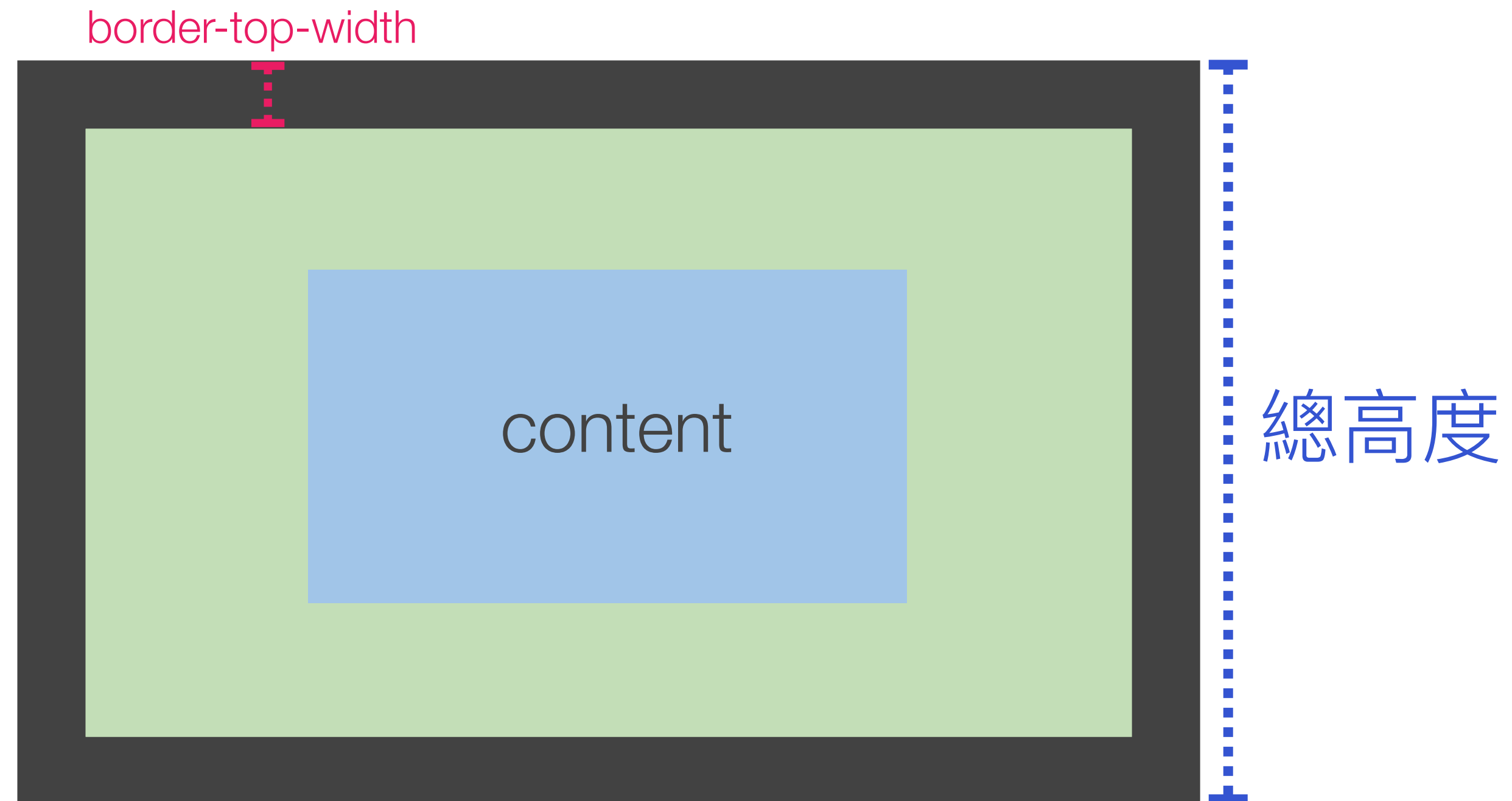
元素的總高度 =



CSS如何決定版面配置 (Layout) ?

元素的總高度 =

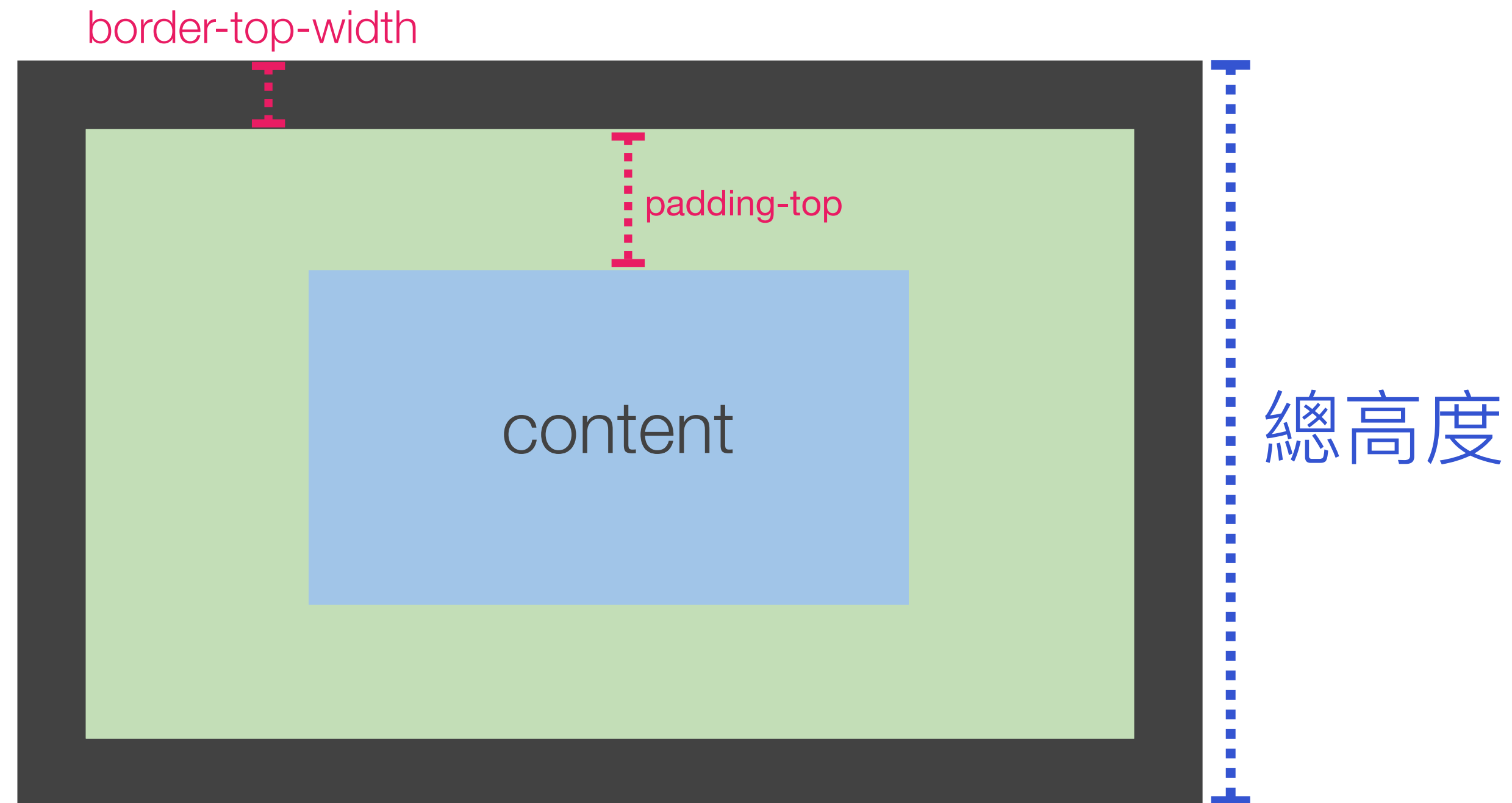
border-top-width



CSS如何決定版面配置 (Layout) ?

元素的總高度 =

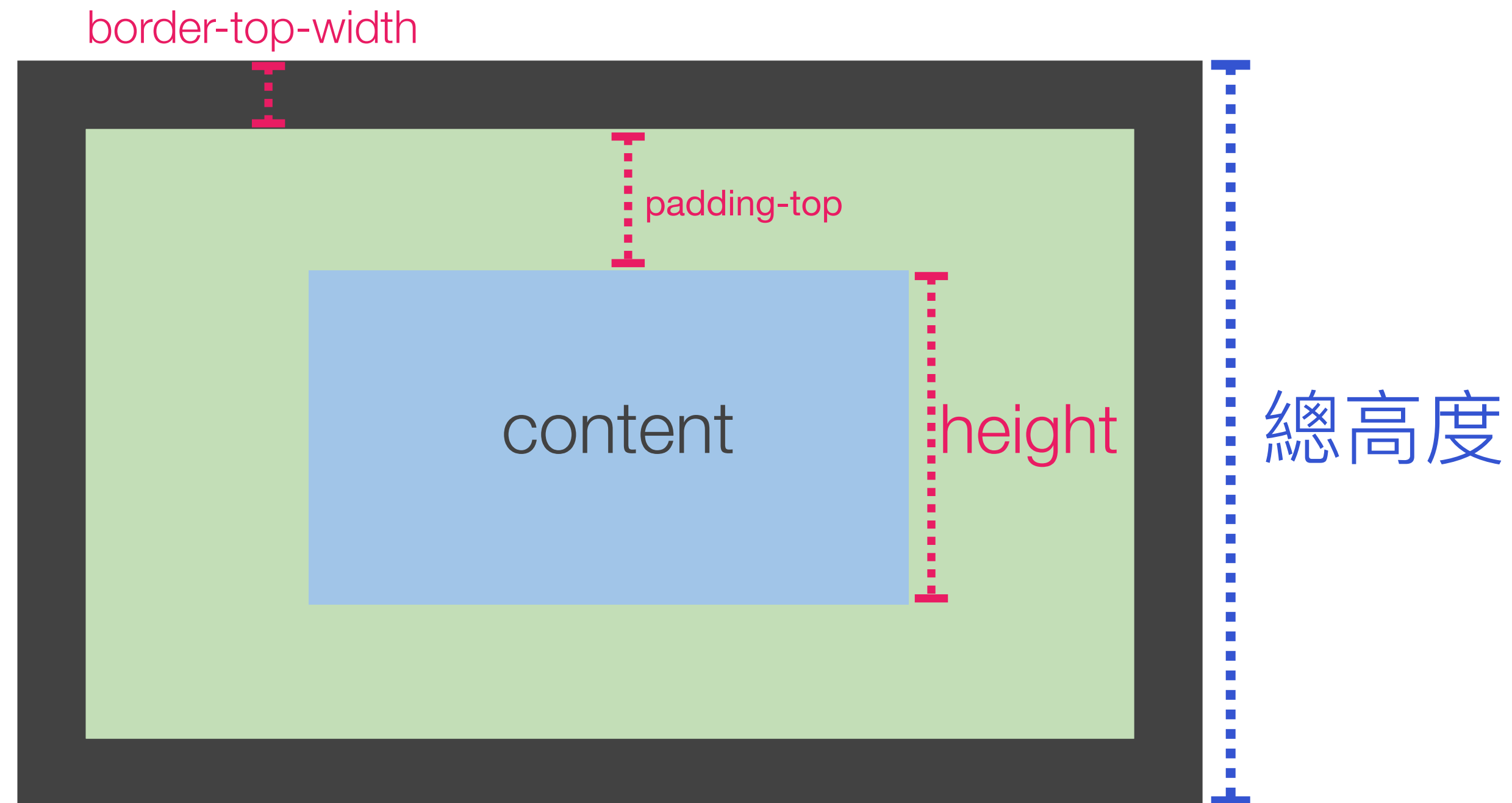
`border-top-width + padding-top`



CSS如何決定版面配置 (Layout) ?

元素的總高度 =

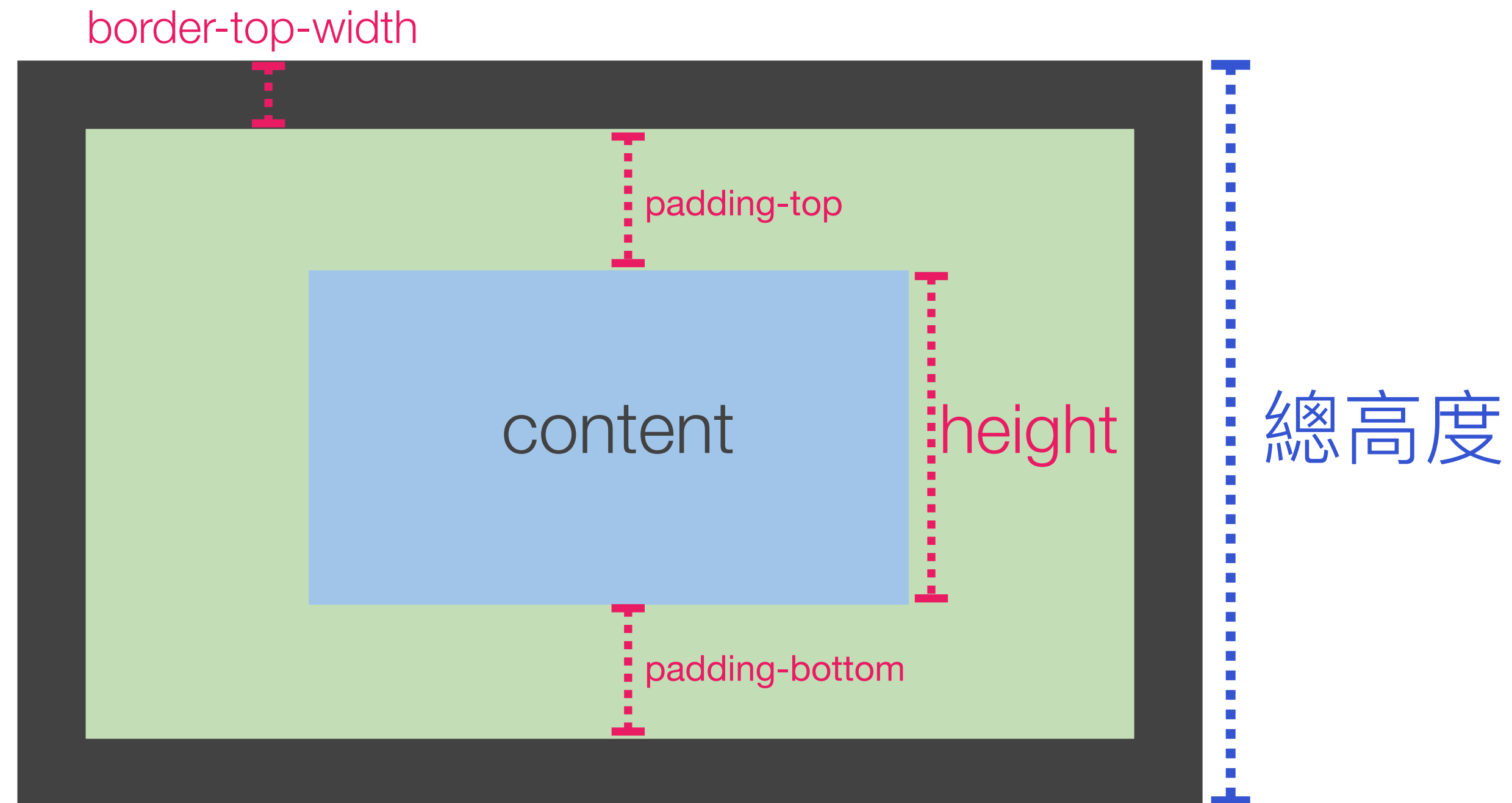
$\text{border-top-width} + \text{padding-top} + \text{height}$



CSS如何決定版面配置 (Layout) ?

元素的總高度 =

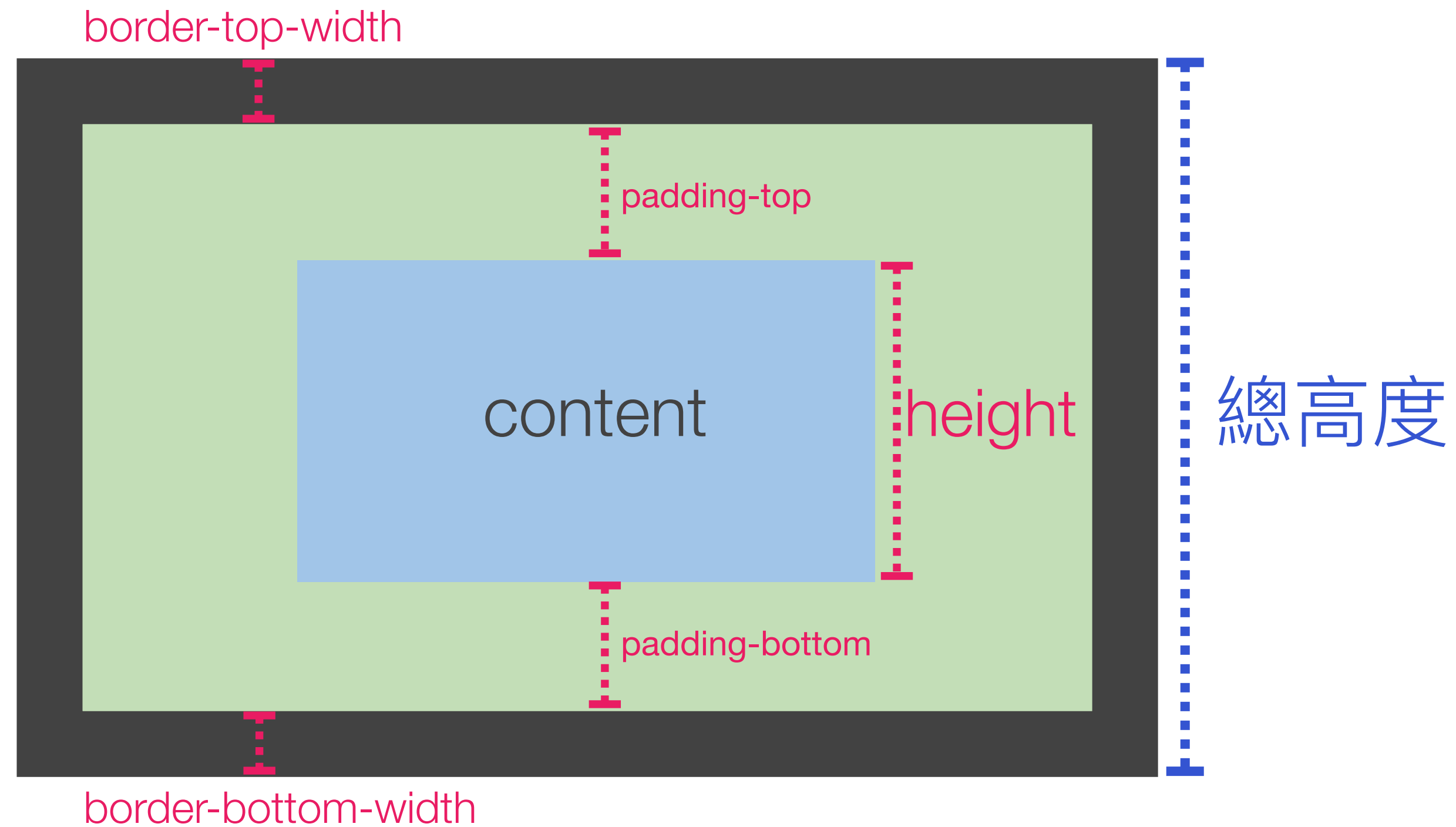
$\text{border-top-width} + \text{padding-top} + \text{height} + \text{padding-bottom}$



CSS如何決定版面配置 (Layout) ?

元素的總高度 =

$\text{border-top-width} + \text{padding-top} + \text{height} + \text{padding-bottom} + \text{border-bottom-width}$



CSS如何決定版面配置 (Layout) ?

Box Model將一個box分成4個部分:

content (內容), padding (內距), border (邊界), margin (外距)

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body { margin: 0; }
    p {
      width: 200px;
      height: 150px;
      padding: 40px;
      margin: 30px;
      border: 20px solid black;
      background: lightgray;
    }
    div {
      background: pink;
      border: 1px solid black;
    }
  </style>
</head>
<body>
  <p>
    編號5566的颱風“我難過”，已在台灣東南方形成，強度估計可達到中颱的等級。
  </p>
  <div>我是下一個元素</div>
</body>
</html>
```

DEMO

Quiz 10

(1) 請使用<div>在空白網頁上畫出一個矩形，這個矩形的設定如下：

文字內容為“光速買光速買”

字體大小為60px，背景是粉紅色(pink)，width為200px，height為100px，

所有方向的padding皆為50px，所有方向的margin皆為30px，

border為40px寬的紅色(red)實心線。

(2) 請問這個矩形的總寬度和總高度各為多少？

Quiz 10

(1) 請使用<div>在空白網頁上畫出一個矩形，這個矩形的設定如下：

文字內容為“光速買光速買”

字體大小為60px，背景是粉紅色(pink)，width為200px，height為100px，

所有方向的padding皆為50px，所有方向的margin皆為30px，

border為40px寬的紅色(red)實心線。

(2) 請問這個矩形的總寬度和總高度各為多少？

Ans:

(1) 見DEMO

(2) 總寬度 = 380px，總高度 = 280px

CSS如何決定版面配置 (Layout) ?

width屬性細節：

```
div {  
  width: 10px;  
}
```

value	description
auto	預設值，由瀏覽器自行決定。
數字+單位	設成特定長度，例如10px，10em，10pt，10cm。使用em指的是以目前元素的font-size為基準去做倍數相乘。
數字+%	將寬度設為parent寬度的特定百分比，例如50%，指的是將目前元素的寬度，設成它parent寬度的50%
initial	使用CSS initial值。
inherit	設成和parent (母元素) 一樣寬。

CSS如何決定版面配置 (Layout) ?

width屬性注意事項1：

當你沒有對元素設定width的時候，瀏覽器名義上會將元素的width，預設為auto，但實際上，
這個auto，對block，inline，或inline-block元素，都有不同的效果。

auto對於block元素而言，代表寬度要和母元素 (parent) 一樣寬；

auto對於inline元素而言，代表寬度要和內容一樣寬；

auto對於inline-block元素而言，針對不同的標籤，會有不同的設定，例如，在Chrome中，
<input>和<button>，都是inline-block元素，但Chrome對這兩個元素的寬度，卻各自有不同的
預設長度。

CSS如何決定版面配置 (Layout) ?

width屬性注意事項2：

只有block和inline-block元素可以設定width，inline元素則無法設定width (設定了也沒用)。

display	相同之處	不同之處
inline	不會和前後的元素斷行 (除非遇到block元素)	無法設定width
inline-block	不會和前後的元素斷行 (除非遇到block元素)	可以設定width

Quiz 11

根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Quiz 11

根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

Quiz 11

根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

$\text{<h1>寬度} = \text{<div>寬度} \times 0.4$
 $= ? \times 0.4$

Quiz 11

根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

$$\begin{aligned}\text{<h1>寬度} &= \text{<div>寬度} \times 0.4 \\ &= ? \times 0.4\end{aligned}$$

Quiz 11

根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

$\text{<div>寬度} = \text{<body>寬度} \times 0.8$
 $= ? \times 0.8$

$\text{<h1>寬度} = \text{<div>寬度} \times 0.4$
 $= ? \times 0.4$

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

<body>的寬度沒有任何設定

<div>寬度 = <body>寬度 x 0.8
= ? x 0.8

<h1>寬度 = <div>寬度 x 0.4
= ? x 0.4

Quiz 11

根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

因為<body>是block元素，因此<body>預設會和<html>一樣寬。

<body>寬度 = <html>寬度 = ?

<div>寬度 = <body>寬度 x 0.8
= ? x 0.8

<h1>寬度 = <div>寬度 x 0.4
= ? x 0.4

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

<html>寬度 = 網頁視窗寬度 = 800

<body>寬度 = <html>寬度 = ?

<div>寬度 = <body>寬度 x 0.8
= ? x 0.8

<h1>寬度 = <div>寬度 x 0.4
= ? x 0.4

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

<html>寬度 = 網頁視窗寬度 = 800

<body>寬度 = <html>寬度 = 800

<div>寬度 = <body>寬度 x 0.8
= ? x 0.8

<h1>寬度 = <div>寬度 x 0.4
= ? x 0.4

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

<html>寬度 = 網頁視窗寬度 = 800

<body>寬度 = <html>寬度 = 800

<div>寬度 = <body>寬度 x 0.8

= 800 x 0.8

= 640

<h1>寬度 = <div>寬度 x 0.4

= ? x 0.4

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

<html>寬度 = 網頁視窗寬度 = 800

<body>寬度 = <html>寬度 = 800

<div>寬度 = <body>寬度 x 0.8

= 800 x 0.8

= 640

<h1>寬度 = <div>寬度 x 0.4

= 640 x 0.4

= **256**

Quiz 11 根據以下程式碼，請問<h1>在網頁中的width為多少？假設你的網頁視窗寬度為800px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 80%;
      }
      h1 {
        width: 40%;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 256px

<html>寬度 = 網頁視窗寬度 = 800

<body>寬度 = <html>寬度 = 800

<div>寬度 = <body>寬度 x 0.8

= 800 x 0.8

= 640

<h1>寬度 = <div>寬度 x 0.4

= 640 x 0.4

= **256**

我順便DEMO一下，將網頁視窗動態改變寬度，看會發生什麼事。

Quiz 12 根據以下程式碼，的總寬度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        margin: 0;
      }
      span {
        width: 200px;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Quiz 12 根據以下程式碼，的總寬度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        margin: 0;
      }
      span {
        width: 200px;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Ans: 是“光速買”這3個字render後的寬度。

因為預設是inline元素，幫inline元素設width不會有作用，因此答案不是200px。

Quiz 13 根據以下程式碼，的總寬度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        margin: 0;
      }
      span {
        display: inline-block;
        width: 200px;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Quiz 13 根據以下程式碼，的總寬度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        margin: 0;
      }
      span {
        display: inline-block;
        width: 200px;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Ans: 200px。

因為我們已經用display將變成inline-block的元素，因此這個時候設定width就會有效。

我就不在瀏覽器上驗證，請同學自行驗證看看。

CSS如何決定版面配置 (Layout) ?

和寬度有關的疑難雜症小補充：

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      h1 {
        width: 200px;
        background: yellow;
      }
    </style>
  </head>
  <body>
    <h1>HiHiHiHiHiHiHiHi</h1>
  </body>
</html>
```

DEMO

CSS如何決定版面配置 (Layout) ?

和寬度有關的疑難雜症小補充：

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      h1 {
        width: 200px;
        background: yellow;
      }
    </style>
  </head>
  <body>
    <h1>HiHiHiHiHiHiHiHiHi</h1>
  </body>
</html>
```

原因是，當block元素的文字內容很長，然後又都沒有空格或斷行的時候，文字內容的寬度就不會被block元素的寬度限制住，反而還會溢出去。

CSS如何決定版面配置 (Layout) ?

和寬度有關的疑難雜症小補充：

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      h1 {
        width: 200px;
        background: yellow;
      }
    </style>
  </head>
  <body>
    <h1>HiHiHiHiHiHiHiHi</h1>
  </body>
</html>
```

原因是，當block元素的文字內容很長，然後又都沒有空格或斷行的時候，文字內容的寬度就不會被block元素的寬度限制住，反而還會溢出去。

英文和印度文，都會有這個問題，至於其他語言，請同學自行嘗試。

CSS如何決定版面配置 (Layout) ?

和寬度有關的疑難雜症小補充：

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      h1 {
        width: 200px;
        background: yellow;
      }
    </style>
  </head>
  <body>
    <h1>HiHiHiHiHiHiHiHiHi</h1>
  </body>
</html>
```

原因是，當block元素的文字內容很長，然後又都沒有空格或斷行的時候，文字內容的寬度就不會被block元素的寬度限制住，反而還會溢出去。

英文和印度文，都會有這個問題，至於其他語言，請同學自行嘗試。

解法是，可以將word-wrap或overflow-wrap，設為break-word，來解決這個問題：

CSS如何決定版面配置 (Layout) ?

和寬度有關的疑難雜症小補充：

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      h1 {
        width: 200px;
        background: yellow;
      }
    </style>
  </head>
  <body>
    <h1>HiHiHiHiHiHiHiHi</h1>
  </body>
</html>
```

原因是，當block元素的文字內容很長，然後又都沒有空格或斷行的時候，文字內容的寬度就不會被block元素的寬度限制住，反而還會溢出去。

英文和印度文，都會有這個問題，至於其他語言，請同學自行嘗試。

解法是，可以將word-wrap或overflow-wrap，設為break-word，來解決這個問題：

```
h1 {
  width: 200px;
  background: yellow;
  overflow-wrap: break-word;
}
```

DEMO

CSS如何決定版面配置 (Layout) ?

height屬性細節：

```
div {  
  height: 10px;  
}
```

value	description
auto	預設值，由瀏覽器自行決定。
數字+單位	設成特定長度，例如10px，10em，10pt，10cm。使用em指的是以目前元素的font-size為基準去做倍數相乘。
數字+%	將高度設為parent高度的特定百分比，例如50%。
initial	使用CSS initial值。

inherit 設成和parent (母元素) 一樣高。

CSS如何決定版面配置 (Layout) ?

height屬性注意事項1：

當你沒有對元素設定height的時候，瀏覽器名義上會將元素的height，預設為auto，但實際上，

auto對於block或inline元素而言，代表高度要和內容一樣高；

auto對於inline-block元素而言，針對不同的標籤，會有不同的設定，例如，在Chrome中，<input>和<button>，都是inline-block元素，但Chrome對這兩個元素的高度，卻各自有不同的預設長度。

CSS如何決定版面配置 (Layout) ?

height屬性注意事項2：

只有block和inline-block元素可以設定height，inline元素則無法設定height (設定了也沒用)。

display	相同之處	不同之處
inline	不會和前後的元素斷行 (除非遇到block元素)	無法設定width和height
inline-block	不會和前後的元素斷行 (除非遇到block元素)	可以設定width和height

Quiz 14

根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

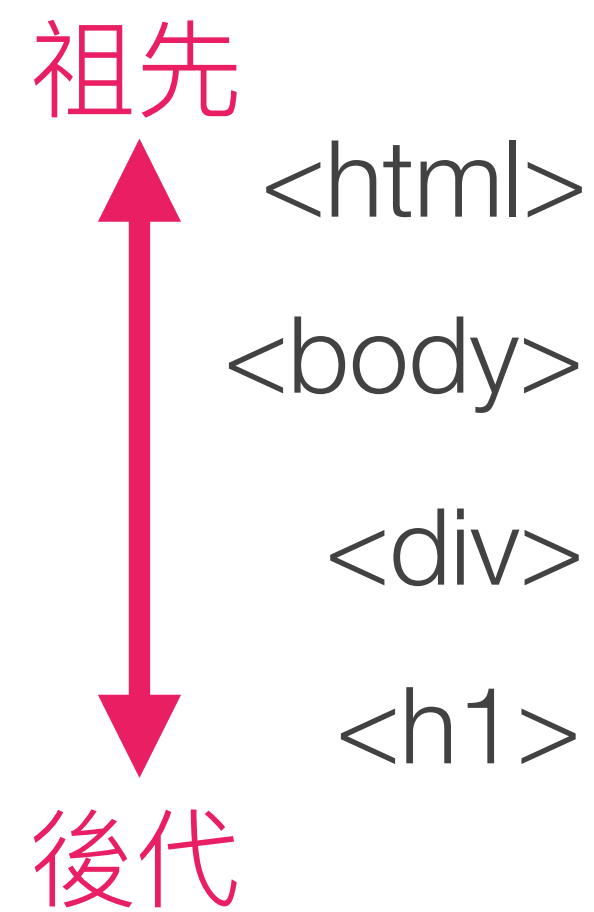
因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

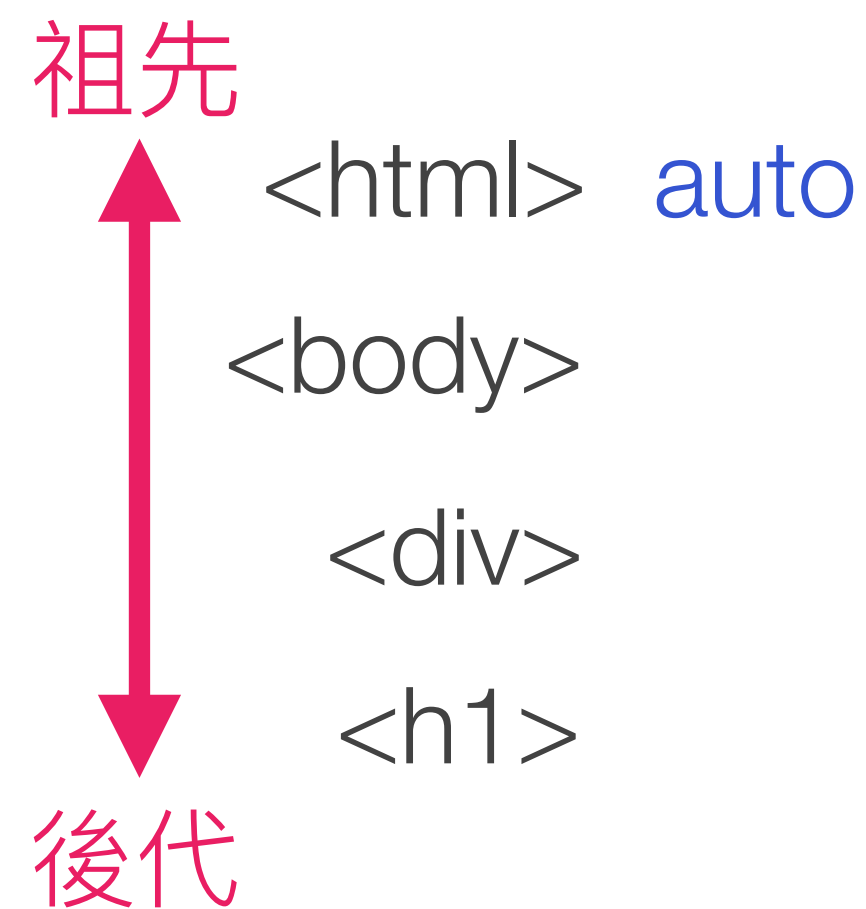


Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。



Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



<html> auto

<body> auto

<div>

<h1>

後代

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



<html> auto

<body> auto

<div> 80% --> <body>的高度 x 0.8 = auto x 0.8

<h1>

後代

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



<html> auto

<body> auto

雞生蛋，蛋生雞

<div> 80% --> <body>的高度 x 0.8 = auto x 0.8

<h1>

後代



Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



後代

<html> auto

<body> auto

<div> ~~80%~~ auto

<h1>

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



<html> auto

<body> auto

<div> ~~80%~~ auto

<h1> 40%

後代

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



<html> auto

<body> auto

<div> ~~80%~~ auto

<h1> ~~40%~~ auto

後代

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



後代

<html> auto

<body> auto

<div> ~~80%~~ auto

<h1> ~~40%~~ auto **0px**

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

祖先



後代

<html>	auto	0px
<body>	auto	0px
<div>	80% auto	0px
<h1>	40% auto	0px

Quiz 14 根據以下程式碼，請問<h1>在網頁中的height為多少？假設你的瀏覽器視窗高度為300px。

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      div {
        width: 300px;
        height: 80%;
        background: aqua;
      }
      h1 {
        width: 200px;
        height: 40%;
        background: pink;
      }
    </style>
  </head>
  <body>
    <div>
      <h1></h1>
    </div>
  </body>
</html>
```

Ans: 0px

因為<html>和<body>的height預設值為auto，當母元素的height為auto，而子元素的height為%，這個時候子元素的height設定，會被瀏覽器視為auto。

我順便DEMO

1. 將<html>和<body>都設成高度100%，看會發生什麼事情？
2. 在第1件事成立的前提下，將視窗動態的改變高度，看會發生什麼事情？

Quiz 15 根據以下程式碼，的總高度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      span {
        height: 200px;
        background: aqua;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Quiz 15 根據以下程式碼，的總高度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      span {
        height: 200px;
        background: aqua;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Ans: 是“光速買”這3個字render後的高度。

因為預設是inline元素，幫inline元素設定height是不會有作用的，因此答案不會是200px。

同學請自行在瀏覽器上驗證結果。

Quiz 16 根據以下程式碼，的總高度為多少？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      span {
        display: inline-block;
        height: 200px;
        background: aqua;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Quiz 16 根據以下程式碼，的總高度為多少？

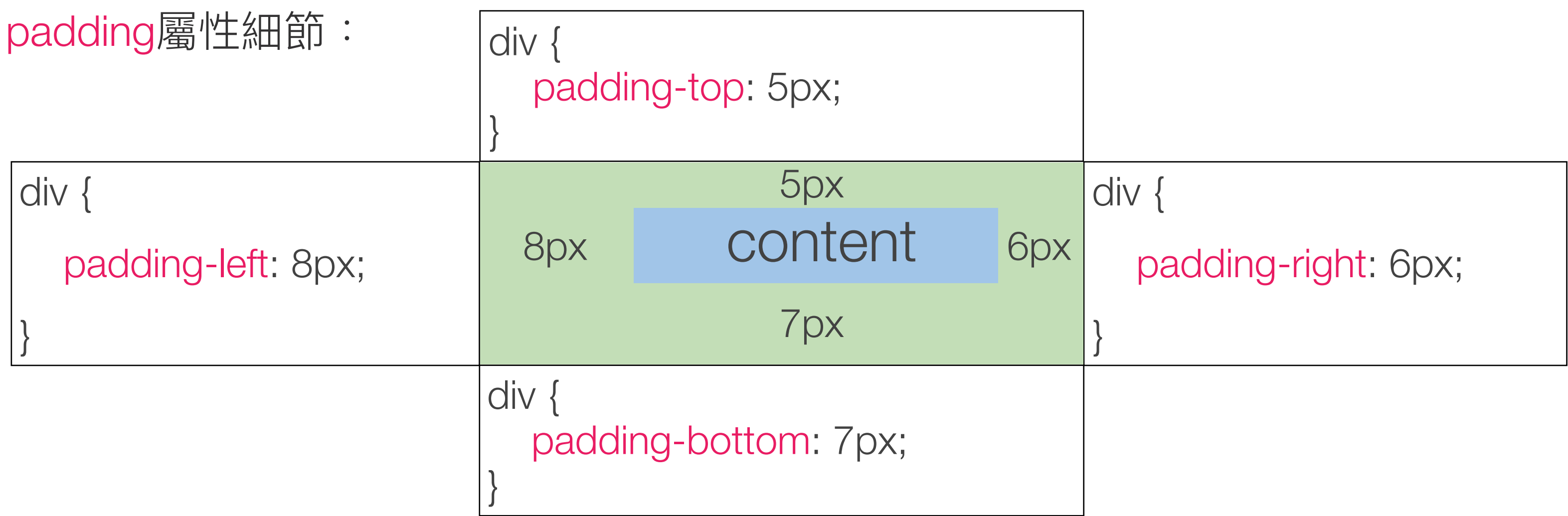
```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        margin: 0;
        padding: 0;
      }
      span {
        display: inline-block;
        height: 200px;
        background: aqua;
      }
    </style>
  </head>
  <body>
    <span>光速買</span>
  </body>
</html>
```

Ans: 200px。

因為我們已經用display，將變成inline-block的元素，因此設定height，就會有效。

同學請自行在瀏覽器上驗證結果。

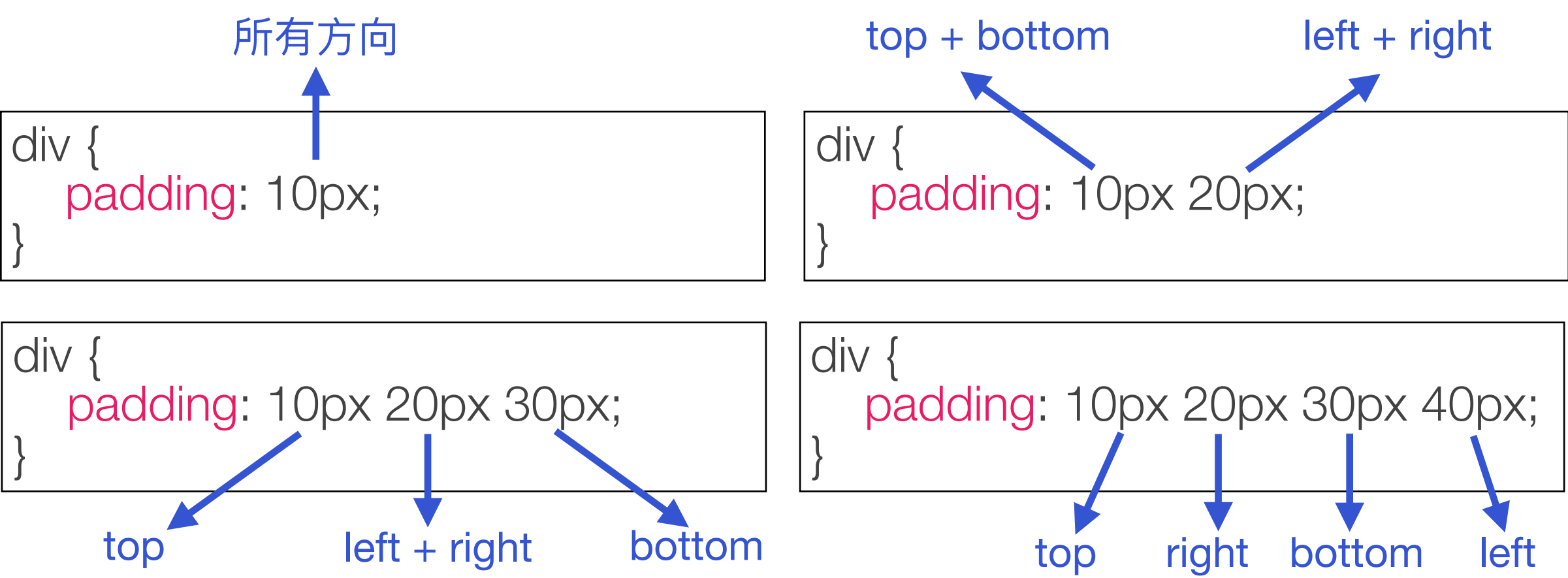
CSS如何決定版面配置 (Layout) ?



value	description
數字+單位	設成特定長度，例如10px，10em，10pt，10cm。使用em指的是以目前元素的font-size為基準去做倍數相乘。
數字+%	將padding設為parent寬度的特定百分比，例如50%，指的是parent寬度的50%，那就算你使用%去設定垂直方向 (top或bottom) 的padding，它們參考的基準，仍然是parent的寬度，而不是高度。
initial	使用CSS initial值。
inherit	設定成和parent的padding一樣。

CSS如何決定版面配置 (Layout) ?

padding屬性細節：



value	description
數字+單位	設成特定長度，例如10px，10em，10pt，10cm。使用em指的是以目前元素的font-size為基準去做倍數相乘。
數字+%	將padding設為parent寬度的特定百分比，例如50%，指的是parent寬度的50%，那就算你使用%去設定垂直方向 (top或bottom) 的padding，它們參考的基準，仍然是parent的寬度，而不是高度。
initial	使用CSS initial值。
inherit	設定成和parent的padding一樣。

CSS如何決定版面配置 (Layout) ?

margin屬性細節：

1. margin的設定邏輯和padding完全一樣，你只要把前兩頁中的"padding"關鍵字都改成margin即可。
但margin可以是負的，而padding只能是大於等於0的值。
2. 對於有設定width的block元素來說，margin值可設成auto讓整個元素自動水平置中：

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      background: aqua;
      width: 400px;
      margin: 0 auto;
    }
  </style>
</head>
<body>
  <div>光速買</div>
</body>
</html>
```

DEMO

Quiz 17 請執行以下程式碼看有什麼結果；如果將<div>的margin改成"10px 0 0 0"，又會是什麼結果？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        padding: 0;
      }
      p {
        background: aqua;
        margin: 0;
      }
      div {
        background: pink;
        margin: -10px 0 0 0;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

Quiz 17 請執行以下程式碼看有什麼結果；如果將<div>的margin改成"10px 0 0 0"，又會是什麼結果？

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      * {
        padding: 0;
      }
      p {
        background: aqua;
        margin: 0;
      }
      div {
        background: pink;
        margin: -10px 0 0 0;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

Ans: **見DEMO**

CSS如何決定版面配置 (Layout) ?

對inline元素設定垂直方向 (top和bottom) 的padding和margin會有不正常的效果；
水平方向 (left和right) 則一切正常。

CSS如何決定版面配置 (Layout) ?

對inline元素設定垂直方向 (top和bottom) 的padding和margin會有不正常的效果；
水平方向 (left和right) 則一切正常。

DEMO

inline-with-vertical-padding-margin.html

(這份檔案也放在"materials"資料夾)

Quiz 18 請修改前一個DEMO的code，將的display改成inline-block，看會有什麼結果。

Quiz 18 請修改前一個DEMO的code，將的display改成inline-block，看會有什麼結果。

Ans: **DEMO**

CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

width (線寬)



CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

width (線寬)

style (實線/虛線)

CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

width (線寬)

style (實線/虛線)

color (顏色)

CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

width (線寬)

style (實線/虛線)

color (顏色)

style (實線/虛線)常見值：

1. solid => 實線
2. dashed => 虛線
3. dotted => 點線

CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

=

```
div {  
  border-width: 1px;  
  border-style: solid;  
  border-color: black;  
}
```

width (線寬)

style (實線/虛線)

color (顏色)

style (實線/虛線)常見值：

1. solid => 實線
2. dashed => 虛線
3. dotted => 點線

CSS如何決定版面配置 (Layout) ?

border屬性細節：

一次設定所有方向的border

```
div {  
  border: 1px solid black;  
}
```

width (線寬) style (實線/虛線) color (顏色)

=

```
div {  
  border-width: 1px;  
  border-style: solid;  
  border-color: black;  
}
```

style (實線/虛線)常見值：

1. solid => 實線
2. dashed => 虛線
3. dotted => 點線

將border的線寬設為0 (不畫border)

```
div {  
  border: 0;  
}
```

```
div {  
  border: none;  
}
```

CSS如何決定版面配置 (Layout) ?

border屬性細節：

只設定上方border

```
div {  
  border-top: 1px solid black;  
}
```

==

```
div {  
  border-top-width: 1px;  
  border-top-style: solid;  
  border-top-color: black;  
}
```

只設定右方border

```
div {  
  border-right: 1px solid black;  
}
```

==

```
div {  
  border-right-width: 1px;  
  border-right-style: solid;  
  border-right-color: black;  
}
```

只設定下方border

```
div {  
  border-bottom: 1px solid black;  
}
```

==

```
div {  
  border-bottom-width: 1px;  
  border-bottom-style: solid;  
  border-bottom-color: black;  
}
```

只設定左方border

```
div {  
  border-left: 1px solid black;  
}
```

==

```
div {  
  border-left-width: 1px;  
  border-left-style: solid;  
  border-left-color: black;  
}
```

Quiz 19

- (1) 請用<div>畫一個矩形，width為200px，height為100px，margin為0，padding為0，左邊的border為黃色虛線(dashed)，且線寬為10px，下面的border為綠色點線(dotted)且線寬為20px。
- (2) 請問這個矩形的總寬度為多少？總高度為多少？

Quiz 19 (1) 請用<div>畫一個矩形，width為200px，height為100px，margin為0，padding為0，
左邊的border為黃色虛線(dashed)，且線寬為10px，下面的border為綠色點線(dotted)且線寬為20px。

(2) 請問這個矩形的總寬度為多少？總高度為多少？

Ans:

(1) 見DEMO

(2) 總寬度 = 210px，總高度 = 120px

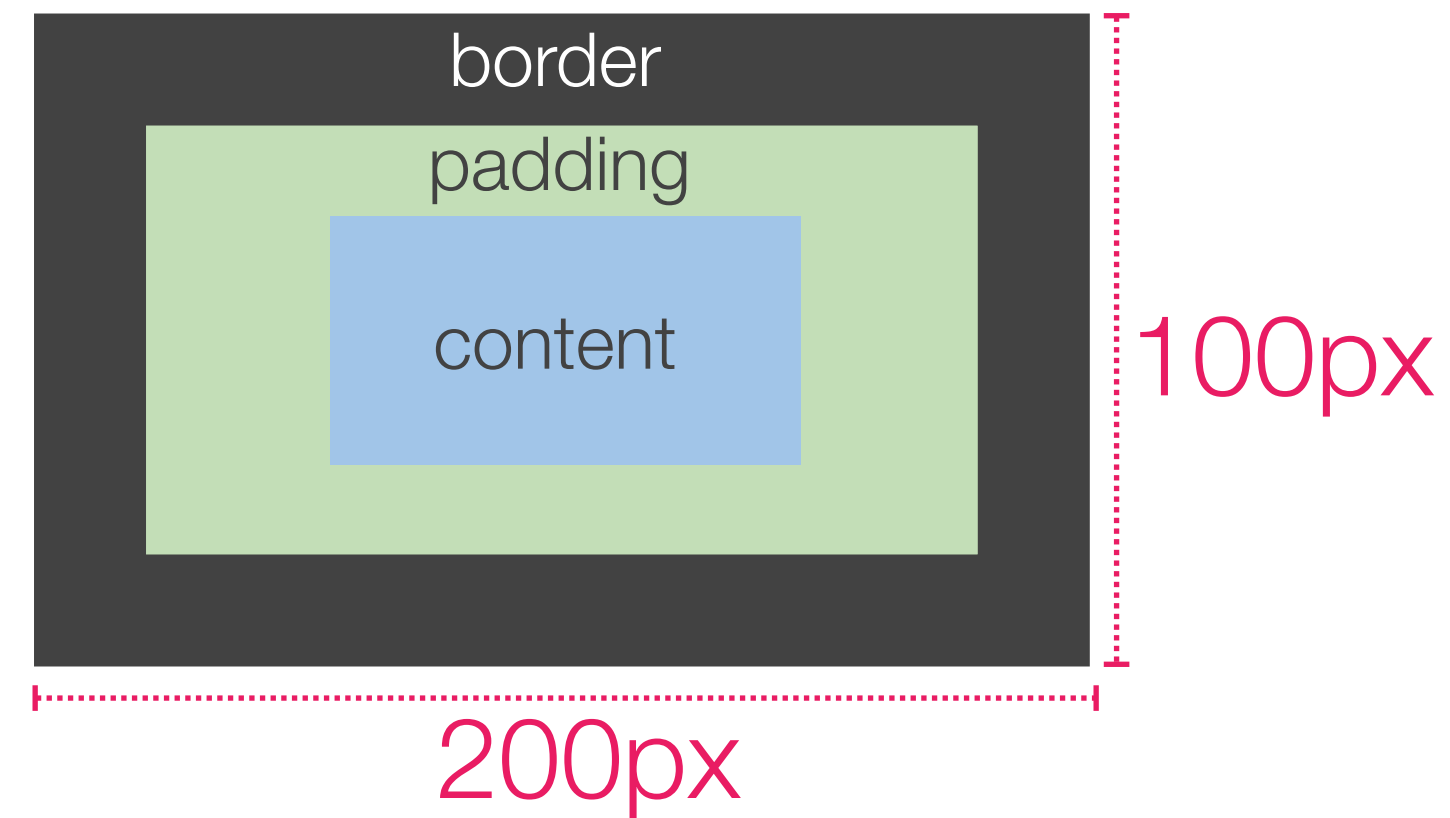
Box Sizing

自己算總寬度和總高度真麻煩，可不可以讓我直接設定總寬度和總高度？

可以！請使用`box-sizing`屬性，讓`width`變成總寬度，讓`height`變成總高度。

```
div {  
  box-sizing: border-box;  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px solid black;  
}
```

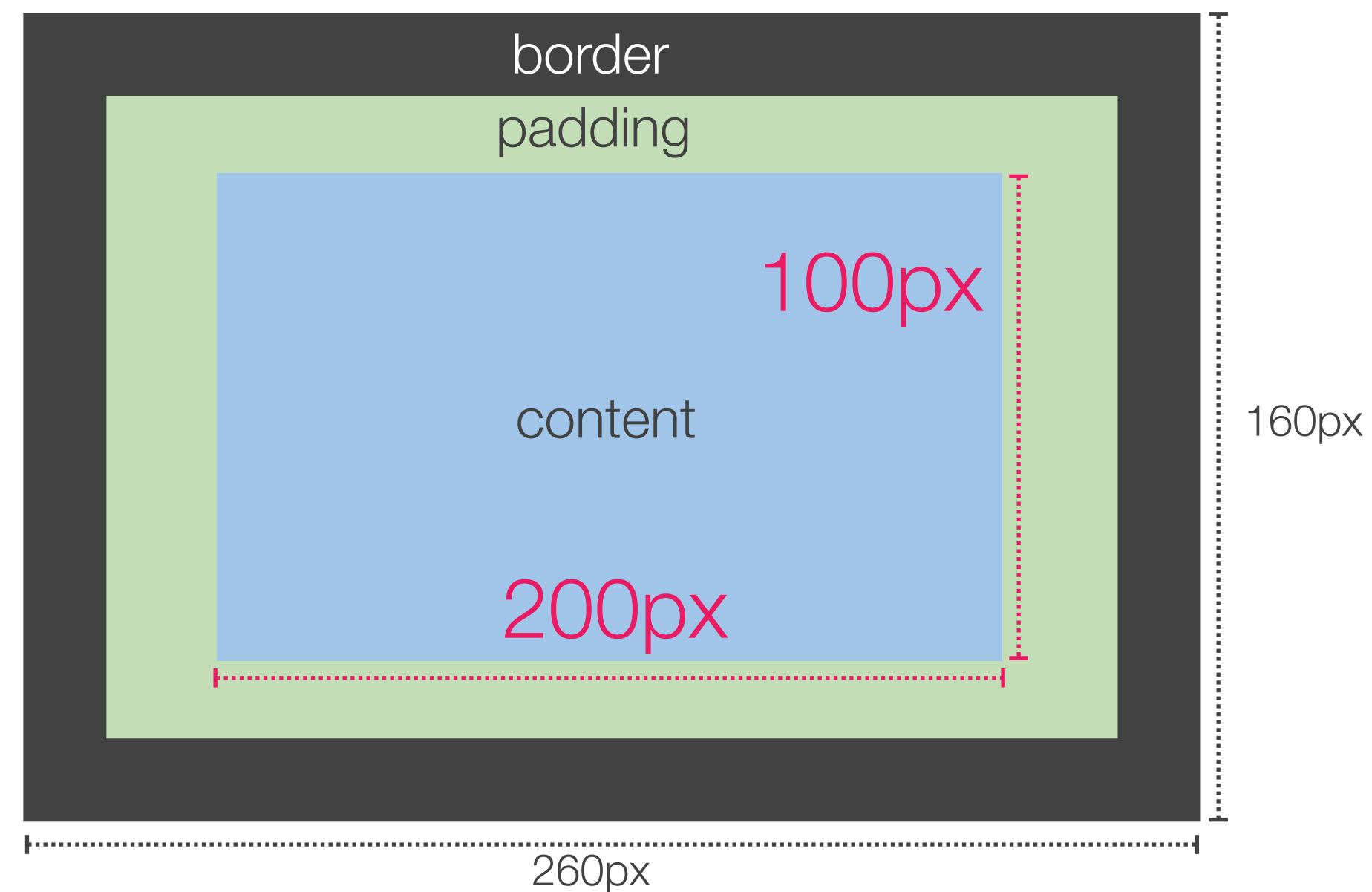
=>



V.S.

```
div {  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px solid black;  
}
```

=>



Quiz 20

請在有<div>的網頁套用以下CSS，並比較有無設定box-sizing的差別。

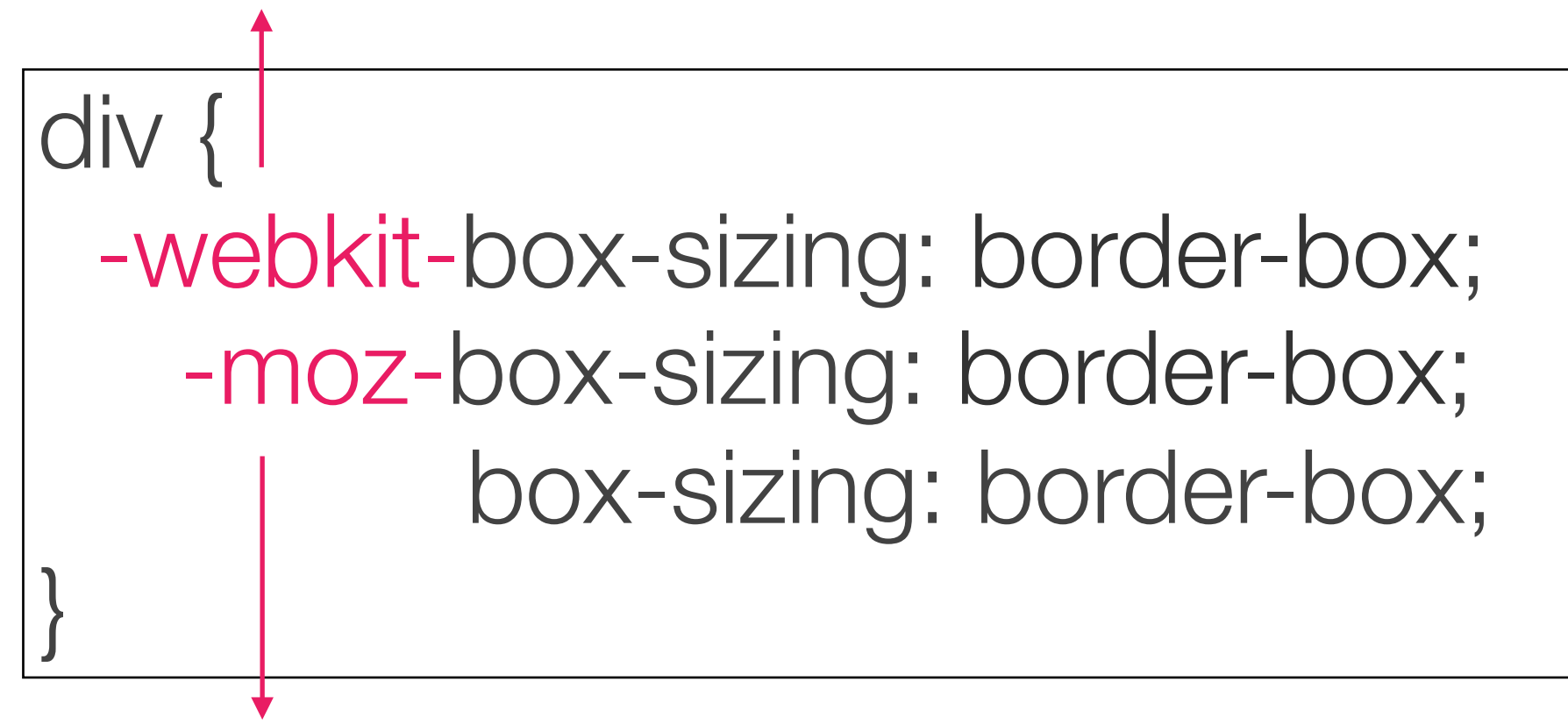
```
div {  
  box-sizing: border-box;  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 10px solid black;  
}
```

前綴字 (prefix)

較舊的瀏覽器不支援box-sizing這個CSS屬性，必須要加前綴字 (prefix) 才能在較舊的瀏覽器上使用。

舊版的Chrome和Safari

```
div {  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}
```



舊版的Firefox

那到底支援到多舊的瀏覽器？教大家使用以下網站查詢

<https://caniuse.com>

我需要自己去查CSS屬性對於舊瀏覽器的支援度嗎？

如果要支援較舊的瀏覽器，難道我要幫每個CSS屬性去查對應的**前綴字 (prefix)** 然後再一個一個自己加進去嗎？

答案是不用，我們可以使用**Prepros**這個工具來自動幫CSS3中比較新的語法加前綴字，這樣就可以輕鬆地支援舊的瀏覽器，你不用自己一個一個加。

Prepros網址：
<https://prepros.io>



DEMO

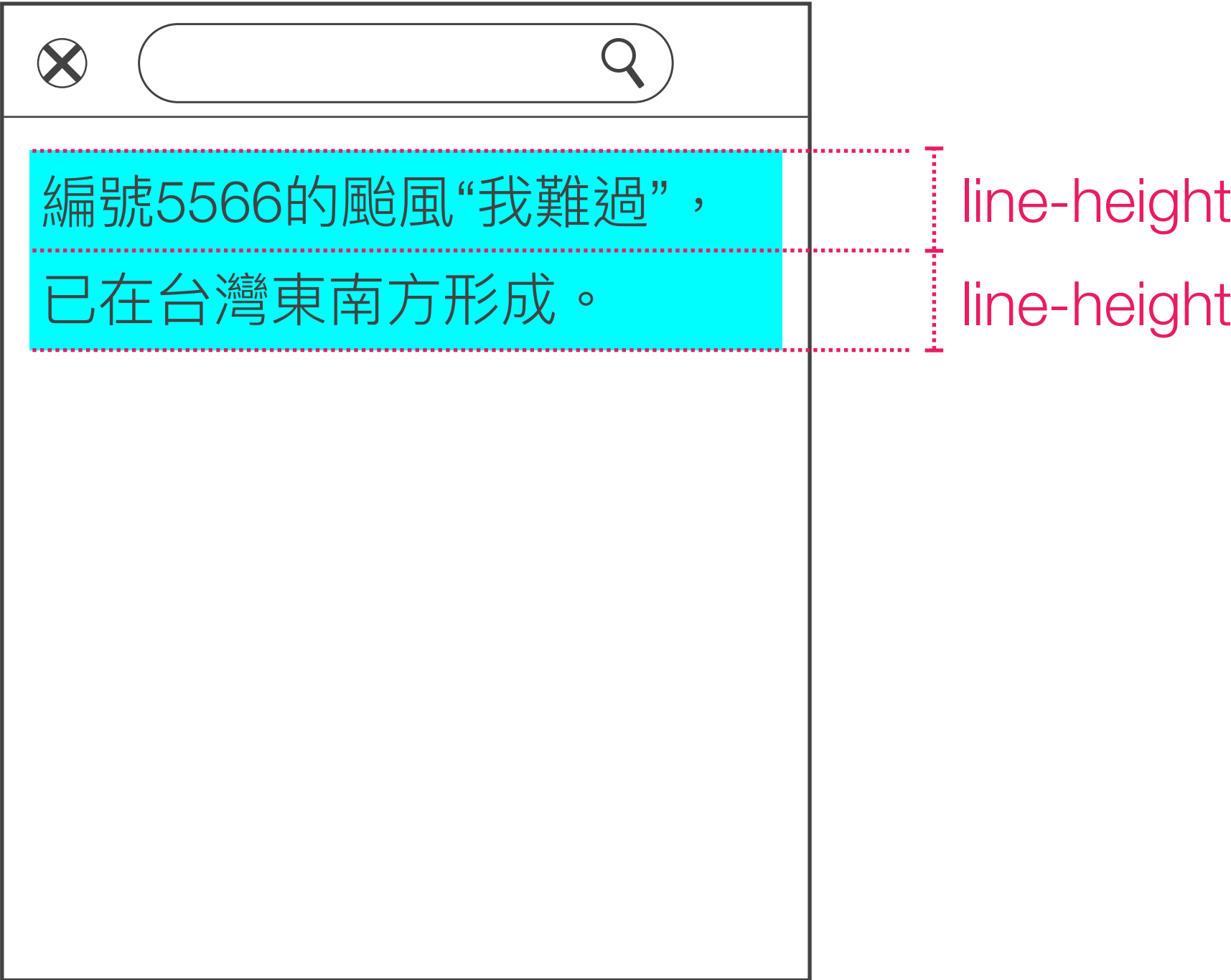
其他會影響元素大小的常用CSS屬性

line-height屬性決定行高，也就是每一行文字的高度。大部分的元素預設值是normal。

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        background: aqua;
      }
    </style>
  </head>
  <body>
    <p>
      編號5566的颱風“我難過”，已在
      台灣東南方形形成。
    </p>
  </body>
</html>
```

網頁畫面



其他會影響元素大小的常用CSS屬性

line-height屬性細節：

```
p {  
  line-height: 25px;  
}
```

value	description
normal	行高標準值，以Chrome實測結果，大約是文字目前font-size的1.4倍左右（僅供參考）。
數字	將行高設成元素目前font-size的特定倍數，例如1.6。
數字+單位	設成特定長度，例如25px，25em，25pt，25cm。使用em指的是以目前元素的font-size為基準去做倍數相乘。
數字+%	將行高設成元素目前font-size的特定百分比，例如160%。
initial	使用CSS initial值。
inherit	設定成和parent一樣。

Quiz 21

請為下面程式碼的<p>添增CSS屬性line-height，並且試著設成以下值看會有什麼結果：

- | | |
|------------|-----------|
| (1) normal | (5) 160% |
| (2) 1.4 | (6) 1.6em |
| (3) 1 | (7) 60px |
| (4) 1.6 | (8) 15px |

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        background: aqua;
        font-size: 30px;
      }
    </style>
  </head>
  <body>
    <p>
      編號5566的颱風“我難過”，
      已在台灣東南方形成。
    </p>
  </body>
</html>
```

其他會影響元素大小的常用CSS屬性

letter-spacing屬性決定字元間距。大部分的元素預設值是normal。letter-spacing的值可以是負的。

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      h1 {
        font-size: 20px;
        letter-spacing: 10px;
      }
      h2 {
        font-size: 20px;
        letter-spacing: 5px;
      }
      h3 {
        font-size: 20px;
        letter-spacing: -5px;
      }
    </style>
  </head>
  <body>
    <h1>光速買</h1>
    <h2>光速買</h2>
    <h3>光速買</h3>
  </body>
</html>
```

網頁畫面



其他會影響元素大小的常用CSS屬性

letter-spacing屬性細節：

```
p {  
  letter-spacing: 5px;  
}
```

value	description
normal	標準值 = 0px = 0。
數字+單位	設成特定長度，例如5px，5em，5pt，5cm；也可以是負的，例如-5px。使用em指的是以目前元素的font-size為基準去做倍數相乘。
initial	使用CSS initial值。
inherit	設定成和parent一樣。

Block元素的排版小陷阱 - Margin Collapse

什麼是Margin Collapse？

垂直方向(top或bottom)相鄰的margin，在沒有border或padding的分隔下，這些margin中只有最大的會起作用，其他都會collapse (塌陷)成0，這個現象稱為margin collapse。Margin collapse只會發生在block元素身上。

Block元素的排版小陷阱 - Margin Collapse

什麼是Margin Collapse？

垂直方向(top或bottom)相鄰的margin，在沒有border或padding的分隔下，這些margin中只有最大的會起作用，其他都會collapse (塌陷)成0，這個現象稱為margin collapse。Margin collapse只會發生在block元素身上。

Margin Collapse的解決方法

法1. 在目標元素的parent設定border。

法2. 在目標元素的parent設定padding。

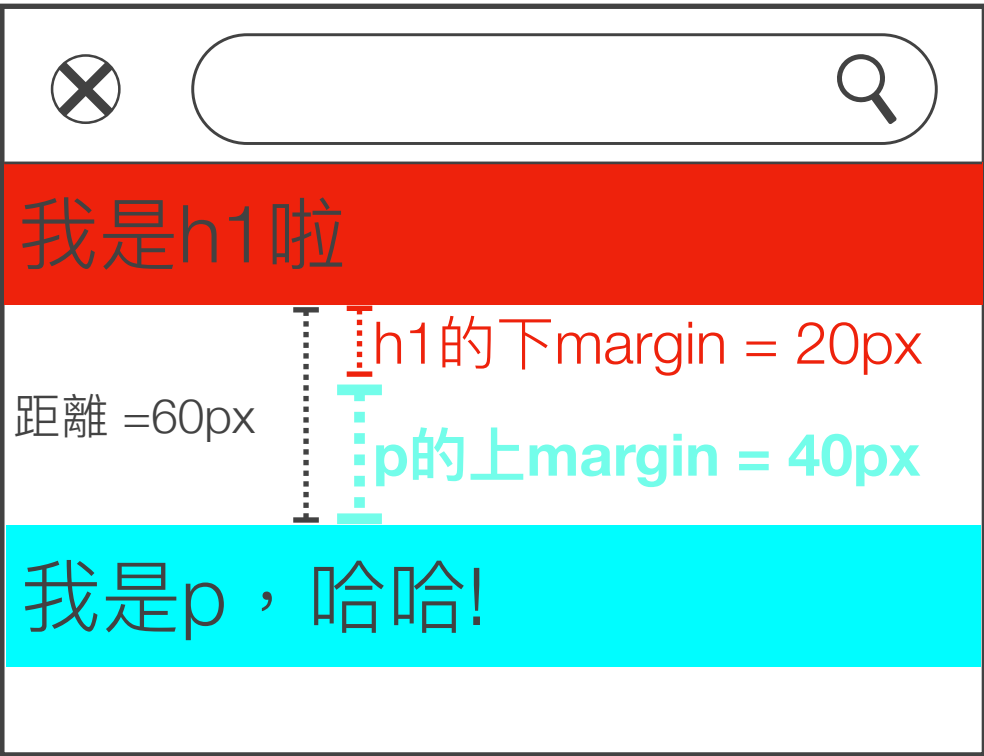
法3. 你真的需要那麼多個margin嗎？你可不可以只用一個margin來解決？
或者搭配其他的padding來解決？

Block元素的排版小陷阱 - Margin Collapse

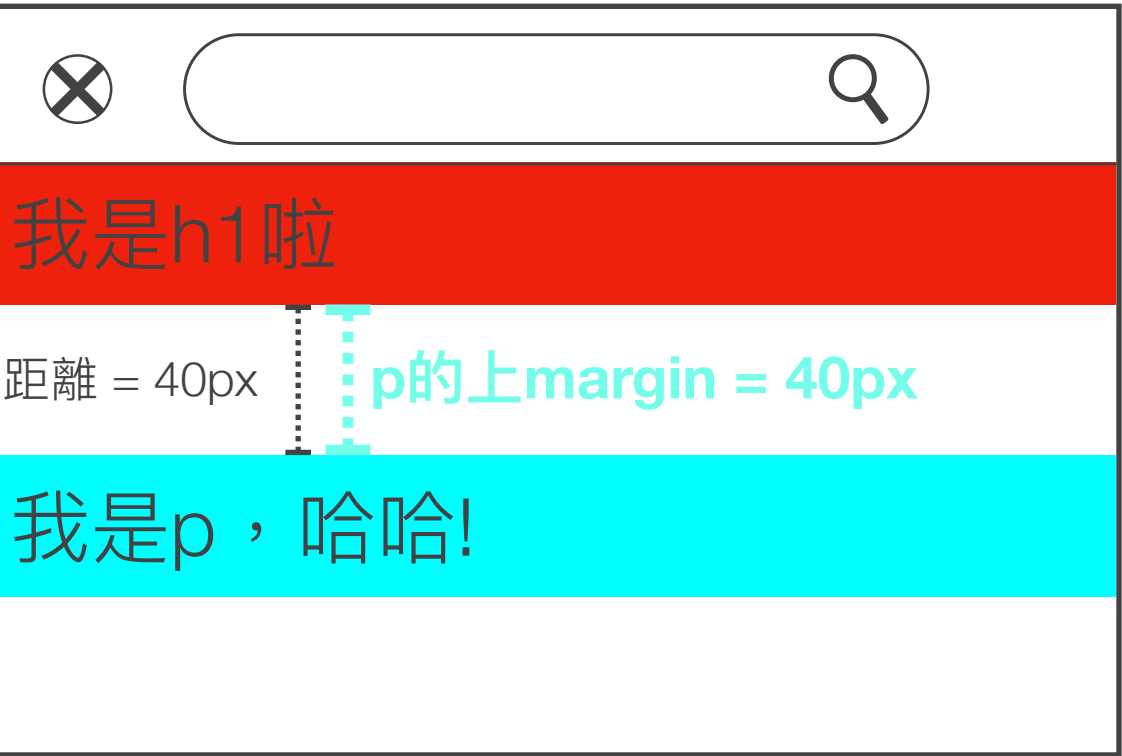
範例1: 相鄰sibling

```
<!DOCTYPE html>
<html>
<head>
  <style>
    * { padding: 0; margin: 0; }
    h1 {
      margin: 0 0 20px 0;
      background: red;
    }
    p {
      margin: 40px 0 0 0;
      background: aqua;
    }
  </style>
</head>
<body>
  <h1>我是h1啦</h1>
  <p>我是p，哈哈!</p>
</body>
</html>
```

你想要的結果 (網頁畫面)



實際上的結果 (網頁畫面)



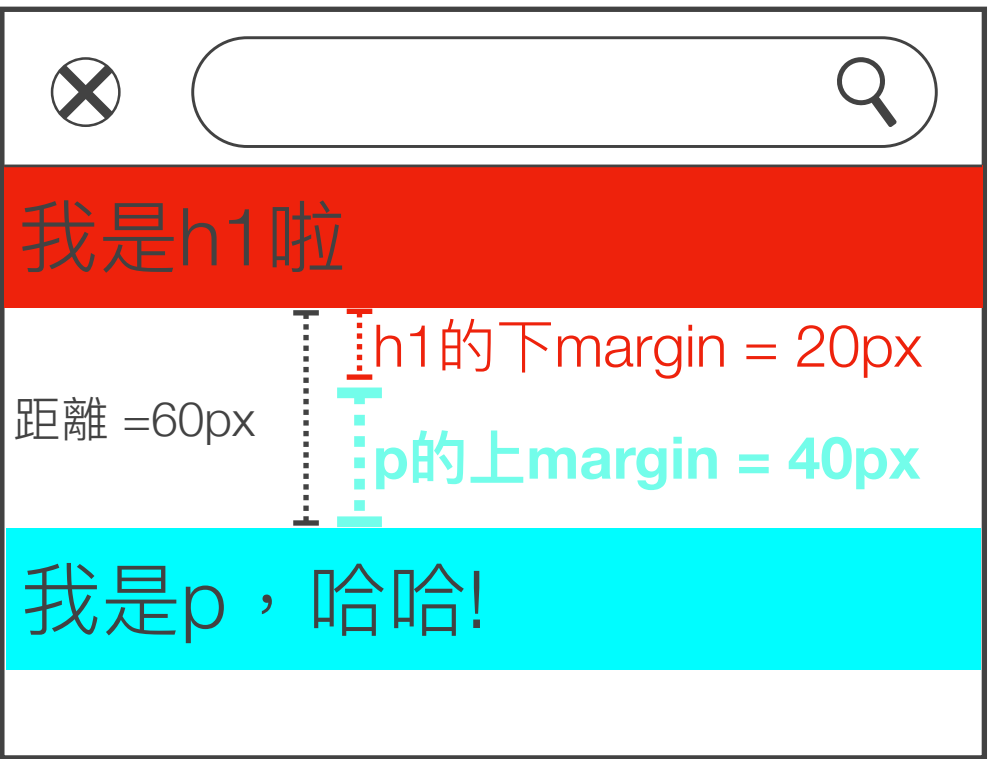
DEMO

Block元素的排版小陷阱 - Margin Collapse

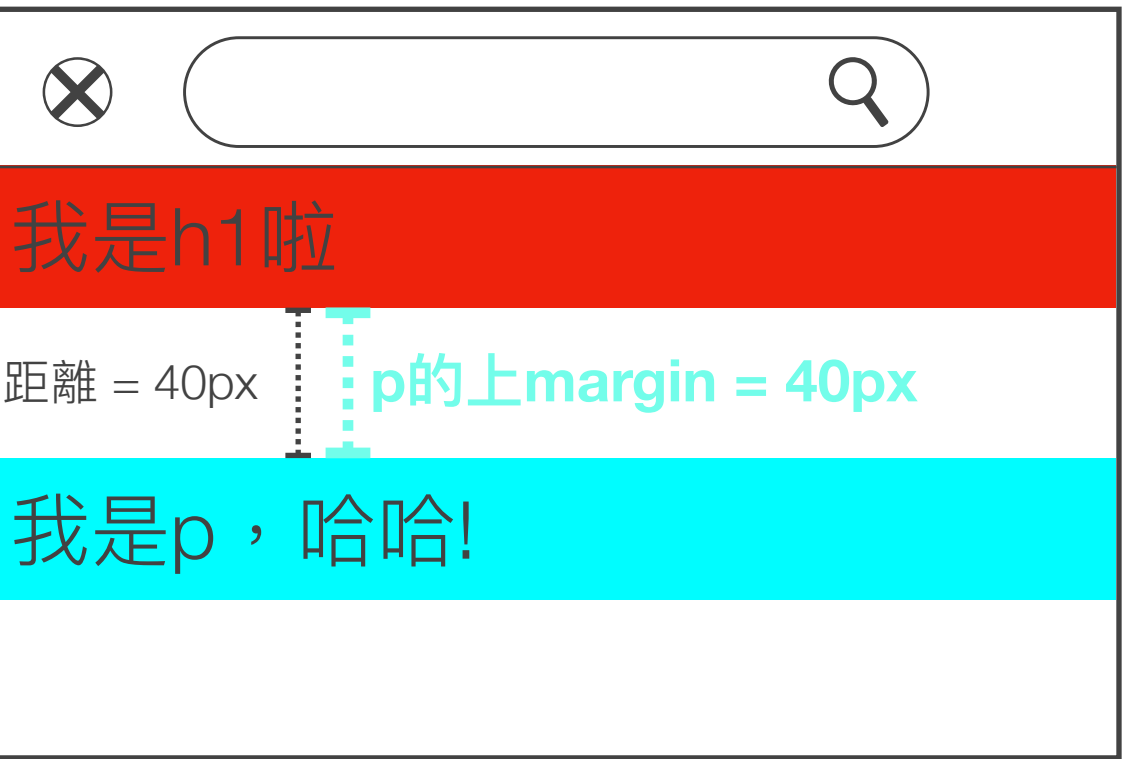
範例1: 相鄰sibling

```
<!DOCTYPE html>
<html>
<head>
  <style>
    * { padding: 0; margin: 0; }
    h1 {
      margin: 0 0 20px 0;
      background: red;
    }
    p {
      margin: 40px 0 0 0;
      background: aqua;
    }
  </style>
</head>
<body>
  <h1>我是h1啦</h1>
  <p>我是p，哈哈!</p>
</body>
</html>
```

你想要的結果 (網頁畫面)



實際上的結果 (網頁畫面)



DEMO

參考解法：

Step 1. 把h1的下margin改成0

Step 2. 把p的上margin改成60px

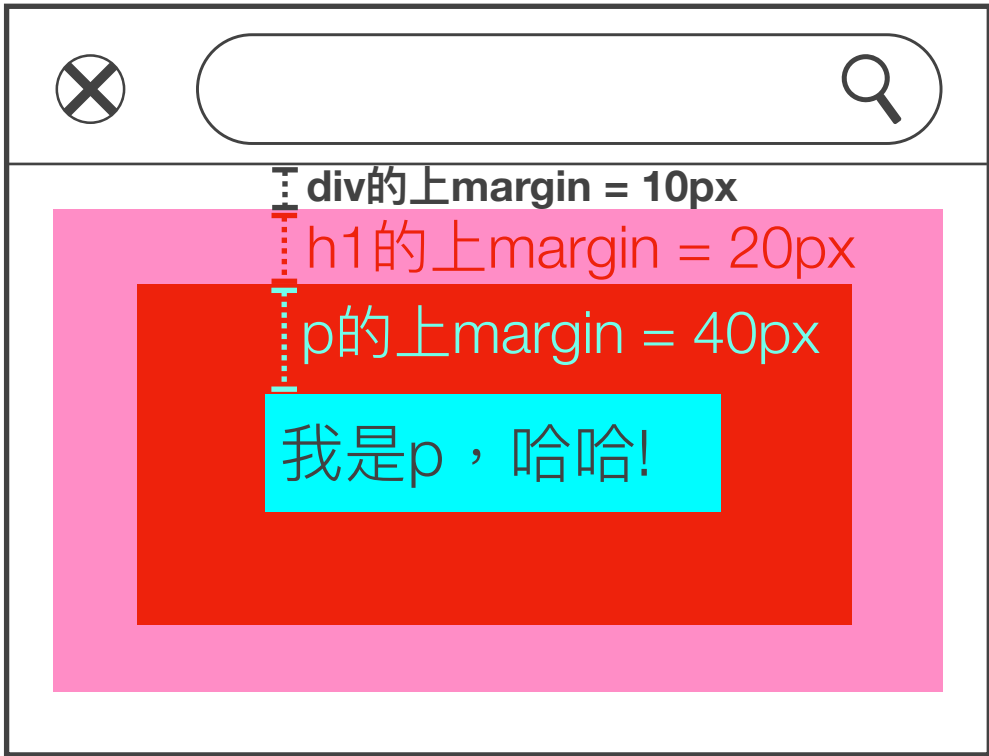
這樣就完成了。

Block元素的排版小陷阱 - Margin Collapse

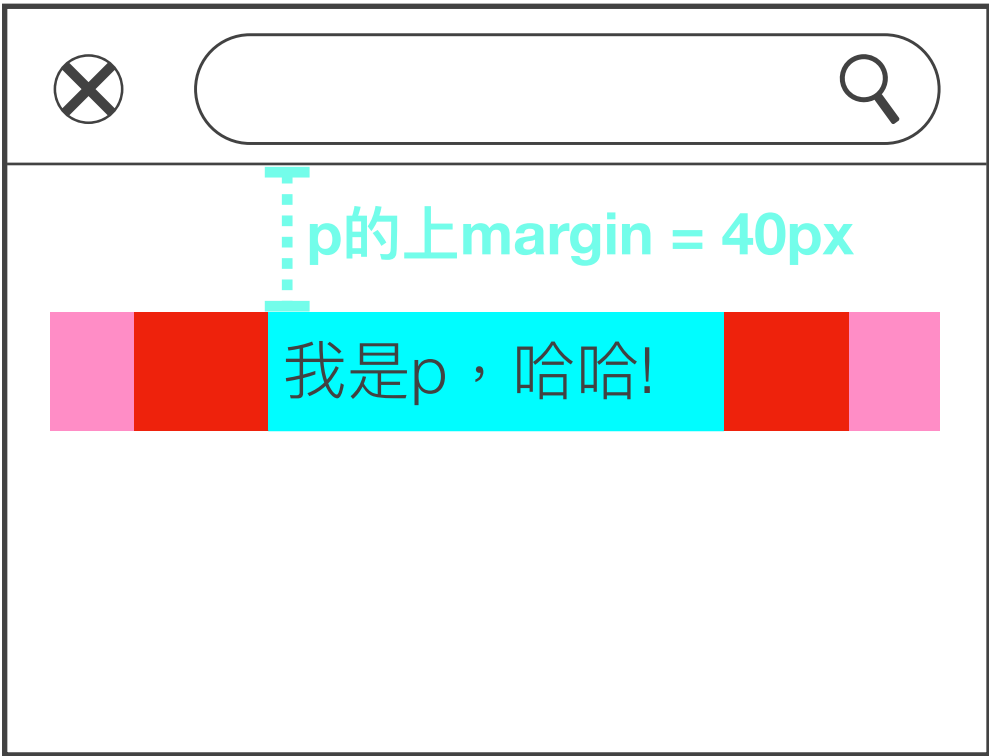
範例2: parent & child

```
<!DOCTYPE html>
<html>
<head>
<style>
  * { padding: 0; margin: 0; }
  div {
    margin: 10px;
    background: pink;
  }
  h1 {
    margin: 20px;
    background: red;
  }
  p {
    margin: 40px;
    background: aqua;
  }
</style>
</head>
<body>
<div>
  <h1>
    <p>我是p，哈哈!</p>
  </h1>
</div>
</body>
</html>
```

你想要的結果 (網頁畫面)



實際上的結果 (網頁畫面)



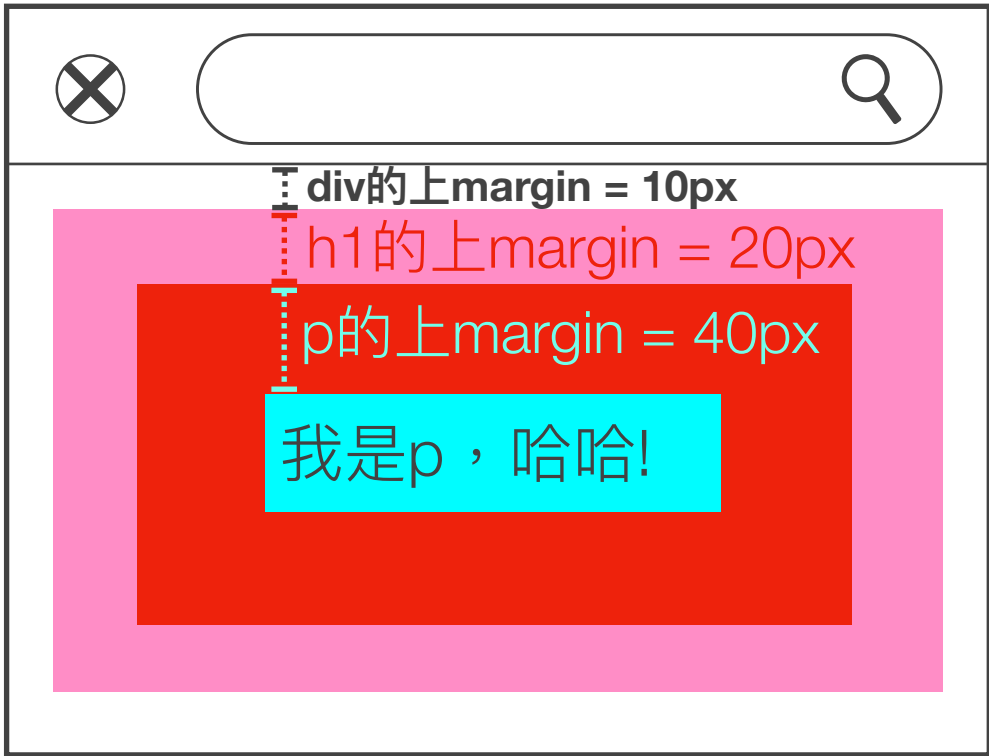
DEMO

Block元素的排版小陷阱 - Margin Collapse

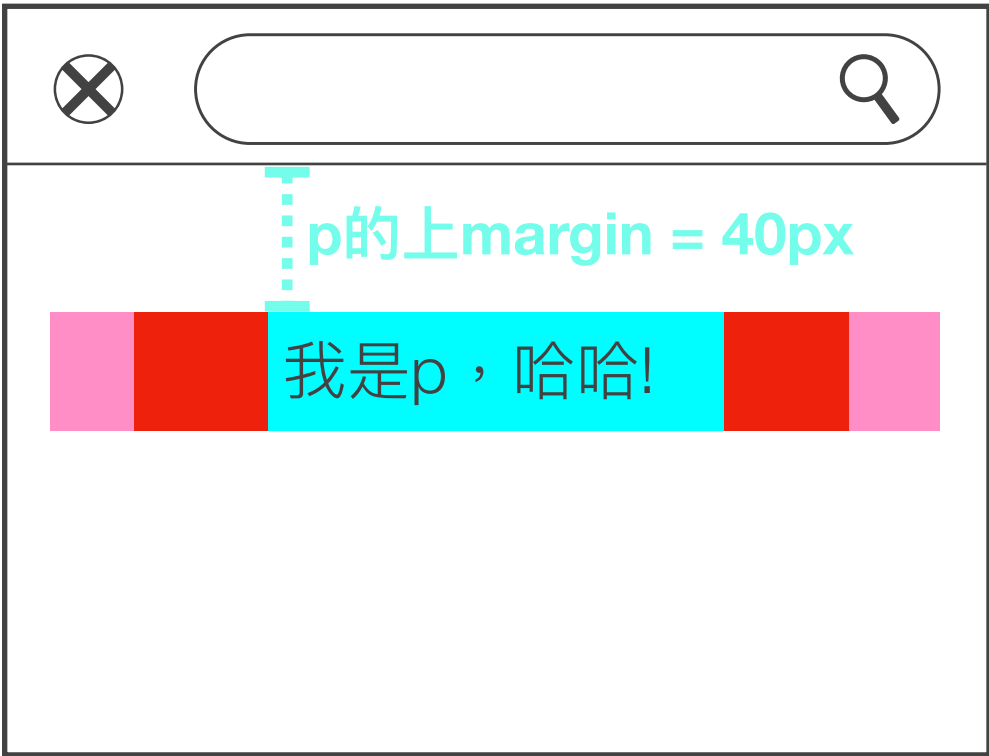
範例2: parent & child

```
<!DOCTYPE html>
<html>
<head>
<style>
  * { padding: 0; margin: 0; }
  div {
    margin: 10px;
    background: pink;
  }
  h1 {
    margin: 20px;
    background: red;
  }
  p {
    margin: 40px;
    background: aqua;
  }
</style>
</head>
<body>
<div>
  <h1>
    <p>我是p，哈哈!</p>
  </h1>
</div>
</body>
</html>
```

你想要的結果 (網頁畫面)



實際上的結果 (網頁畫面)



DEMO

參考解法：

- Step 1. 把div的padding改成20px
 - Step 2. 把h1的margin改成0，將h1的padding改為40px
 - Step 3. 把p的margin改成0
- 這樣就完成了。

Quiz 22

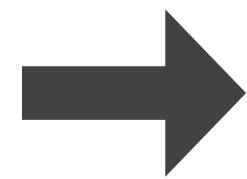
- (1) 請找找看專案1中有哪兩個元素出現margin collapse的現象。
- (2) 請說明以下程式是否有出現margin collapse；若有，如何解決？

```
<!DOCTYPE html>
<html>
<head>
  <style>
    * { padding: 0; margin: 0; }
    h1 {
      background: red;
    }
    div {
      margin: 20px 0 0 0;
      background: pink;
    }
    p {
      margin: 40px 0 0 0;
      background: aqua;
    }
  </style>
</head>
<body>
  <h1>我是標題</h1>
  <div>
    <p>我是文章段落</p>
  </div>
</body>
</html>
```


Quiz 22

- (1) 請找找看專案1中有哪兩個元素出現margin collapse的現象。--> Ans: <body>和<h1>
(2) 請說明以下程式是否有出現margin collapse；若有，如何解決？

```
<!DOCTYPE html>
<html>
<head>
<style>
  * { padding: 0; margin: 0; }
  h1 {
    background: red;
  }
  div {
    margin: 20px 0 0 0;
    background: pink;
  }
  p {
    margin: 40px 0 0 0;
    background: aqua;
  }
</style>
</head>
<body>
  <h1>我是標題</h1>
  <div>
    <p>我是文章段落</p>
  </div>
</body>
</html>
```



(2) Ans:
有，解法參
考右邊程式

```
<!DOCTYPE html>
<html>
<head>
<style>
  * { padding: 0; margin: 0; }
  h1 {
    background: red;
  }
  div {
    margin: 20px 0 0 0;
    background: pink;
    border: 1px solid black; /* 法1 */
    /* padding: 10px 0; */ /* 法2 */
  }
  p {
    margin: 40px 0 0 0;
    background: aqua;
  }
</style>
</head>
<body>
  <h1>我是標題</h1>
  <div>
    <p>我是文章段落</p>
  </div>
</body>
</html>
```

Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```



你以為的網頁畫面

✕

🔍

光速買

光速買

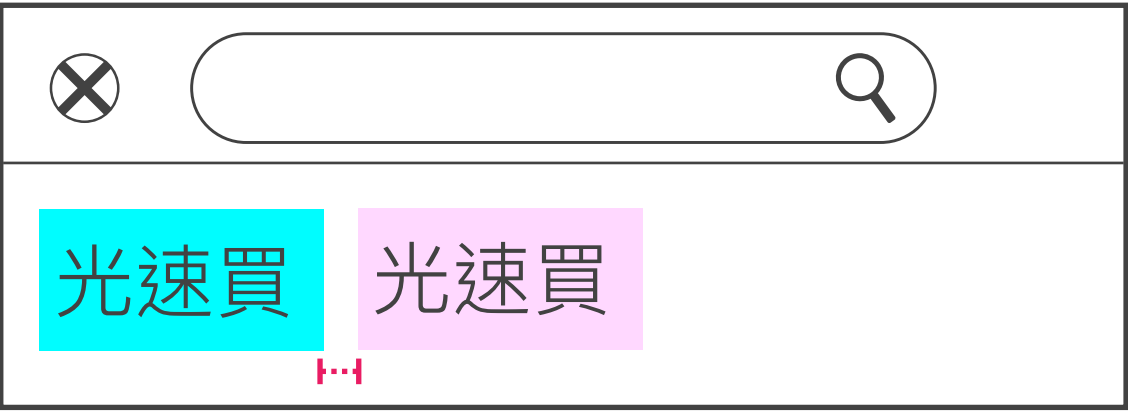
Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```



實際的網頁畫面



不知哪來的空白

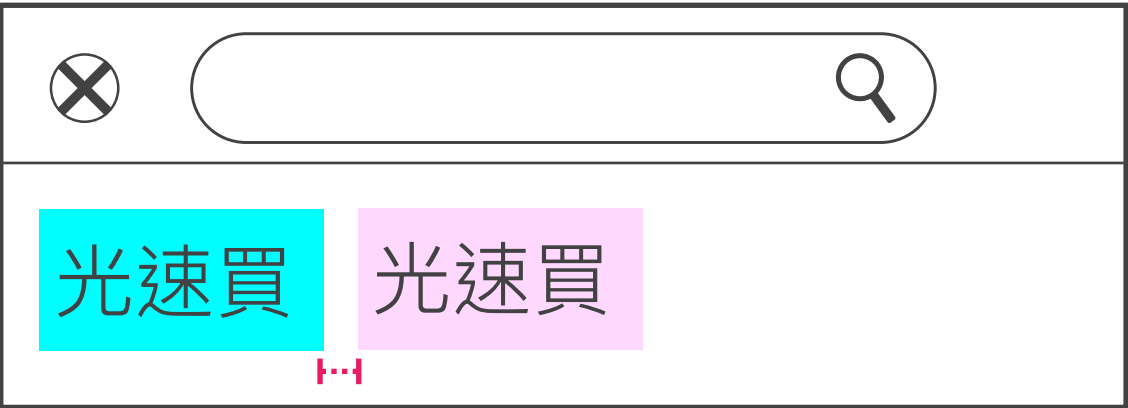
Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

發生了什麼事？

實際的網頁畫面



不知哪來的空白

Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

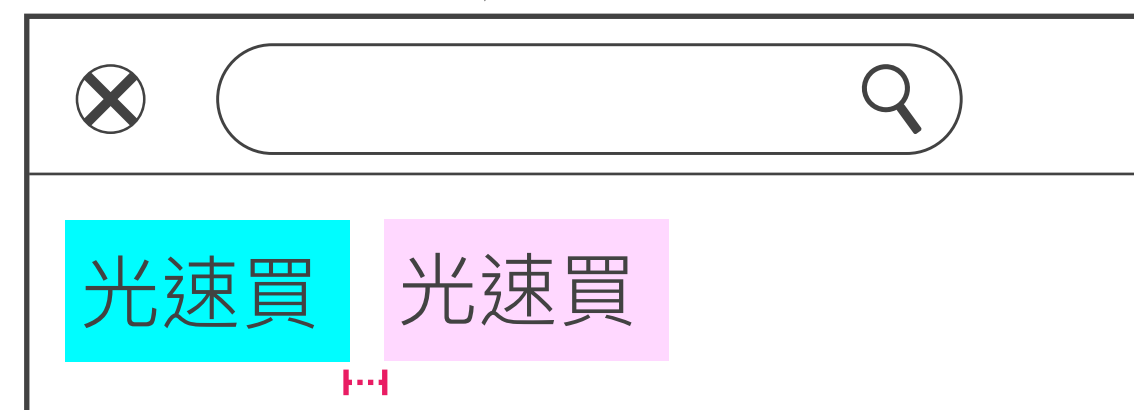
原因：HTML在render時會將文字間的斷行當成一個空白來render。

你應該有發現，在編輯HTML檔案時，若你在標籤內的文字內容中間，按下“Enter”鍵來斷行 (通常是為了排版好看一點)，則render時其實不會斷行，而是會產生一個空白。

那你一定想問，<p>和<div>又不是一般文字，它怎麼會符合這些規則呢？答案是，當元素被設為inline-block的時候，HTML就會把它當成一塊大文字。

以左邊的程式碼為例，<p>現在是inline-block，所以<p>是一塊大文字，<div>現在是inline-block，所以<div>也是一塊大文字，那我在這兩塊大文字的中間，按了一個“Enter”，增加了一個斷行，因此HTML，就將這個斷行render成空白。

實際的網頁畫面



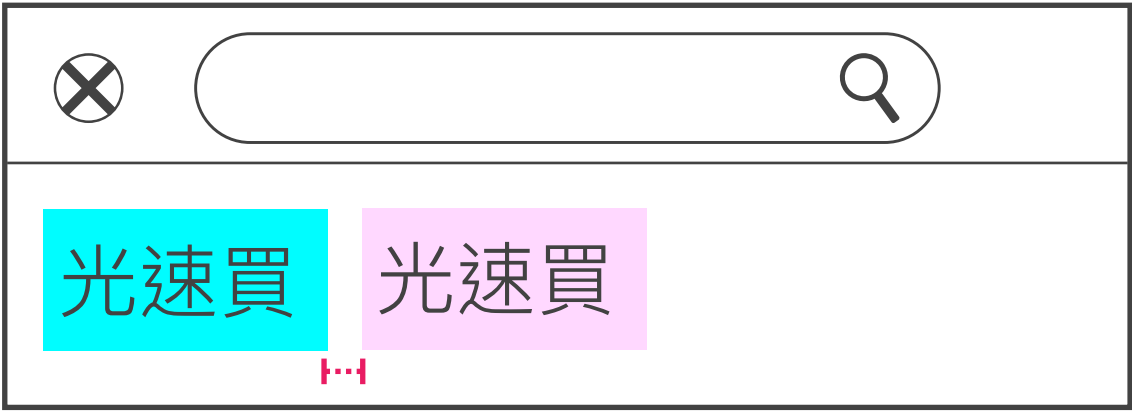
從斷行來的空白

Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

實際的網頁畫面



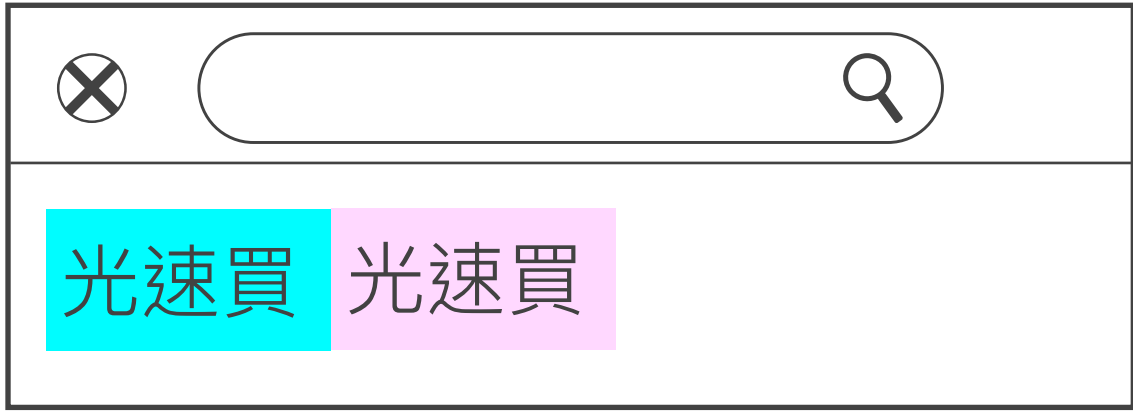
從斷行來的空白

解決法1

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p><div>
      光速買</div>
  </body>
</html>
```

實際的網頁畫面



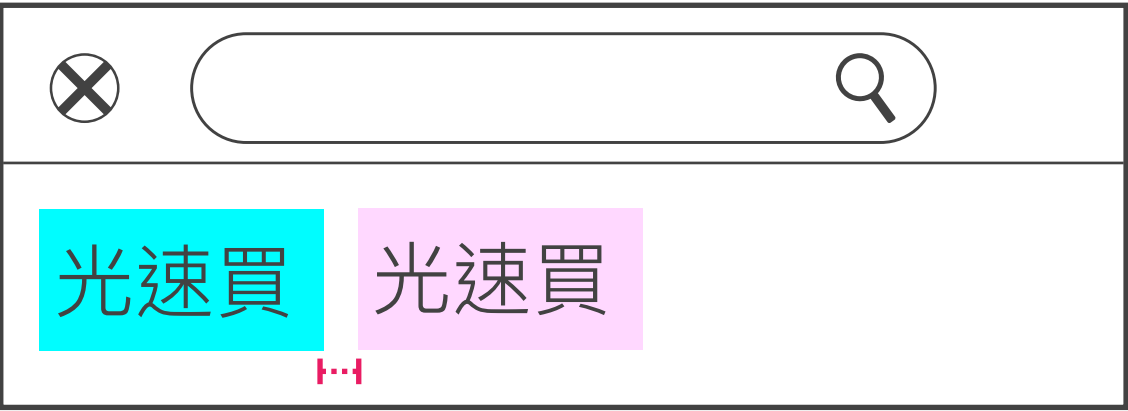
空白消失了

Inline-block的排版小陷阱 - 無中生有的空白

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

實際的網頁畫面



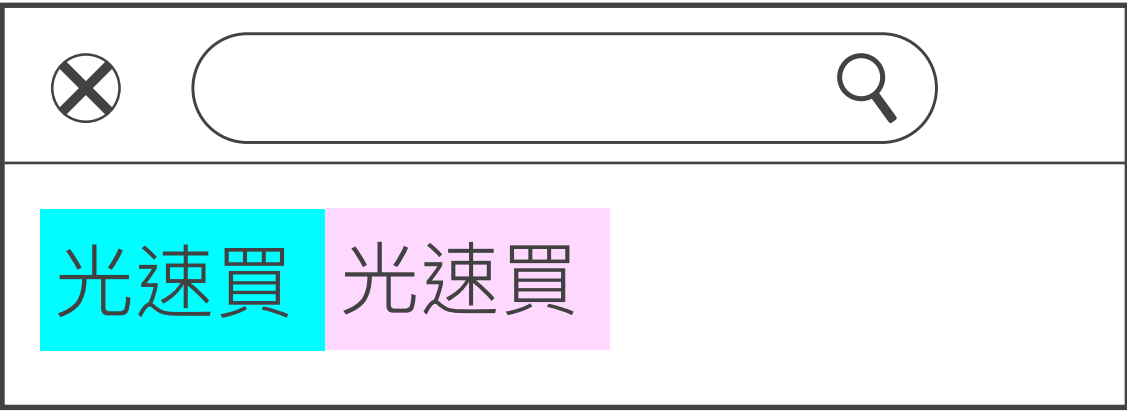
從斷行來的空白

解決法2

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        letter-spacing: -0.34em;
      }
      p {
        margin: 0;
        background: aqua;
        display: inline-block;
        letter-spacing: 0;
      }
      div {
        margin: 0;
        background: pink;
        display: inline-block;
        letter-spacing: 0;
      }
    </style>
  </head>
  <body>
    <p>光速買</p>
    <div>光速買</div>
  </body>
</html>
```

實際的網頁畫面



空白消失了

Quiz 23 (1) 請找找看專案1中有哪兩個元素之間出現無中生有的空白。
(2) 如何把這個空白消除？

Quiz 23 (1) 請找找看專案1中有哪兩個元素之間出現無中生有的空白。
(2) 如何把這個空白消除？

Ans:

(1) `<input type="text">`和`<input type="button">`

(2) 請見DEMO

Inline元素的排版小陷阱 - 無中生有的空白

inline元素其實也會和inline-block一樣，有同樣的排版小陷阱，那解法就跟我們剛剛介紹的都一樣，在此不再贅述。