

# CSS

## 網頁設計×視覺特效 專題班

Lecturer: LinJer (林哲)

evin92@gmail.com

網頁前端-進階

CSS



CSS

網頁設計 × 視覺效果



LJW

第一堂：初探CSS基礎**入門**

第二堂：進階CSS編寫**技巧**

第三堂：解構CSS網頁**排版**

第四堂：玩轉CSS視覺**特效**



# CSS屬性與值 - 字體 (font-\*)

[功能] 設定與字體相關的屬性



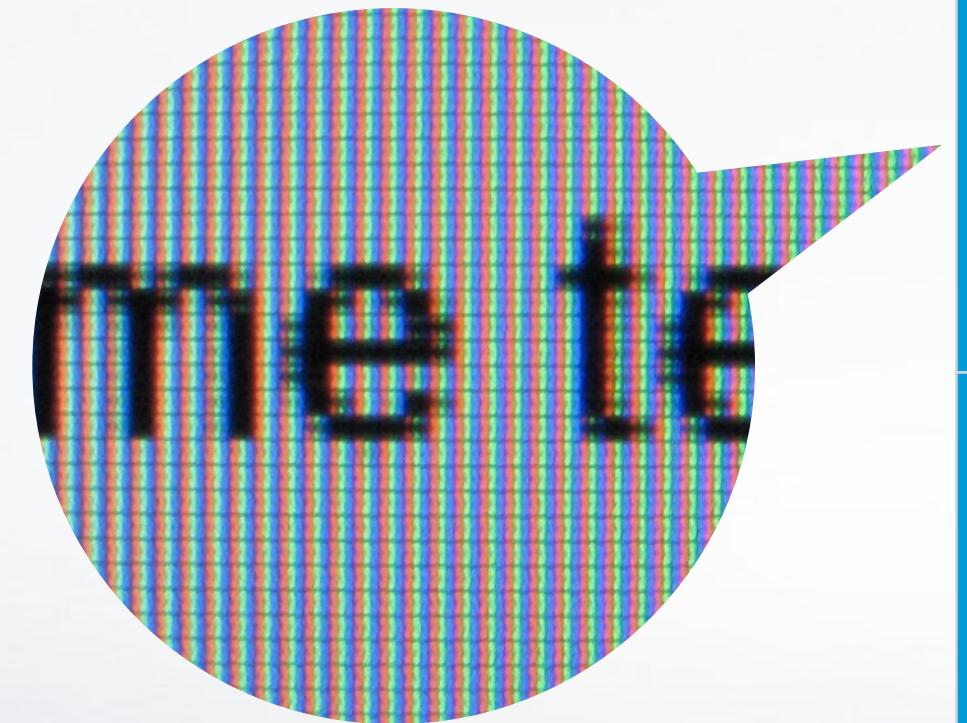
選擇器 {

font-*	功能
font-family	設定字體
font-size:	設定大小
font-weight	字體粗細

}

# 網頁設計小知識 - 字級單位

```
選擇器 {  
    font-size: (px, cm, em, %)  
}
```



表示法	舉例	意義
數字(px)	12px	電腦螢幕顯示的基本單位:像素 (Pixel)
數字(cm)	0.4cm	電腦螢幕顯示的單位:公分 (Centimeter)
數字 (rem, em, %)	0.75rem (=75%)	同百分比 $1em=100\% = 12pt=16px$

# 網頁設計小問答 -

## 字級單位: em

Q: 第一層(學&好好玩)設定為2em，其子階層(HTML)設定為 2em，那子階層(HTML)的實際尺寸為?

Ans: (A) 2em (B) 4em (C) 1em

```
div[align="center"] {  
    font-size: 2em;  
}  
div[align="center"] p {  
    font-size: 2em;  
}
```

```
<div align="center">  
    學  
    <p>HTML</p>  
    好好玩  
</div>
```

# 網頁設計小問答 -

## 字級單位: %

```
div[align="center"] {  
    font-size: 2em;  
}  
div[align="center"] p {  
    font-size: 2em;  
}
```

```
<div align="center">  
    學  
    <p>HTML</p>  
    好好玩  
</div>
```

Q: 第一層(學&好好玩)設定為200%，其子階層(HTML)設定為 50%，  
那子階層(HTML)的實際尺寸為？

Ans: (A) 1% (B) 10% (C) 100%

# CSS屬性與值 - 文字對齊 (text-align)

[功能] 設定文字水平對齊

真正的東西不是用眼睛可以看得到的

真正的東西不是用眼睛可以看得到的

[前往練習](#)

選擇器 {  
text-align:

}

left	向左對齊
right	向右對齊
center	置中
justify	使左右對齊本文
inherit	繼承父元素的 text-align 屬性



# CSS屬性與值 - 字母間距 (letter-spacing)

[功能] 設定每一個字母間的距離

目標: 字元間相隔1字元寬

```
真正的東西不是用眼睛可以看得到的
```

```
div {  
  letter-spacing: ?em;  
}
```

選擇器 {  
 letter-spacing:

}

表示法	舉例
數字(px)	12px
數字(cm)	0.4cm
數字(rem)	1rem (預設 16px)

# CSS屬性與值 - 段落間距 (line-height)

[功能] 設定每一段落間的距離

目標: 段落間相隔2字元寬

真正的東西

不是用眼睛可以看得到的

```
div {  
    line-height: ? em;  
}
```

選擇器 {  
 line-height:

}

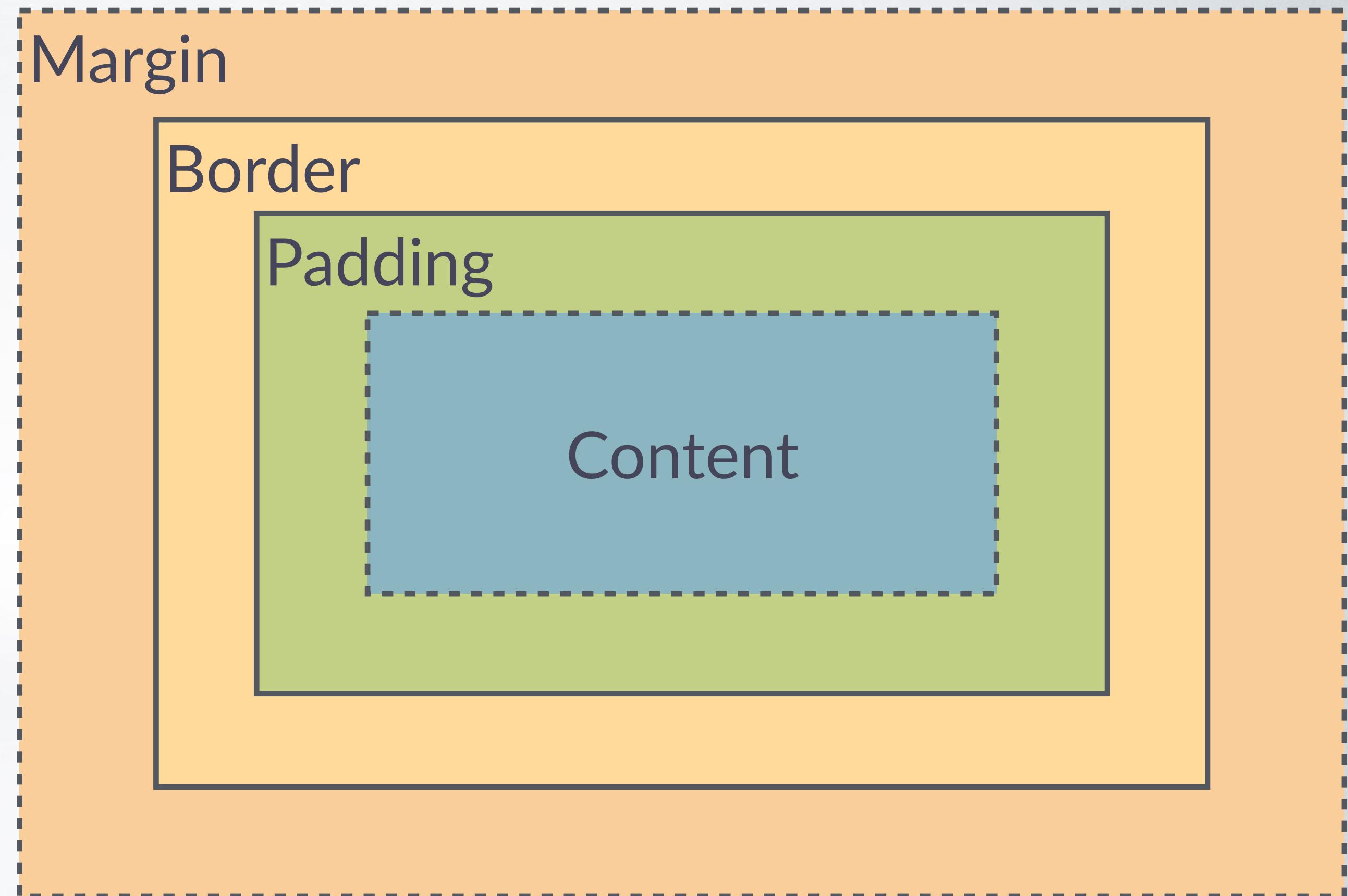
表示法	舉例
數字(px)	12px
數字(cm)	0.4cm
數字(rem)	1rem (預設 16px)

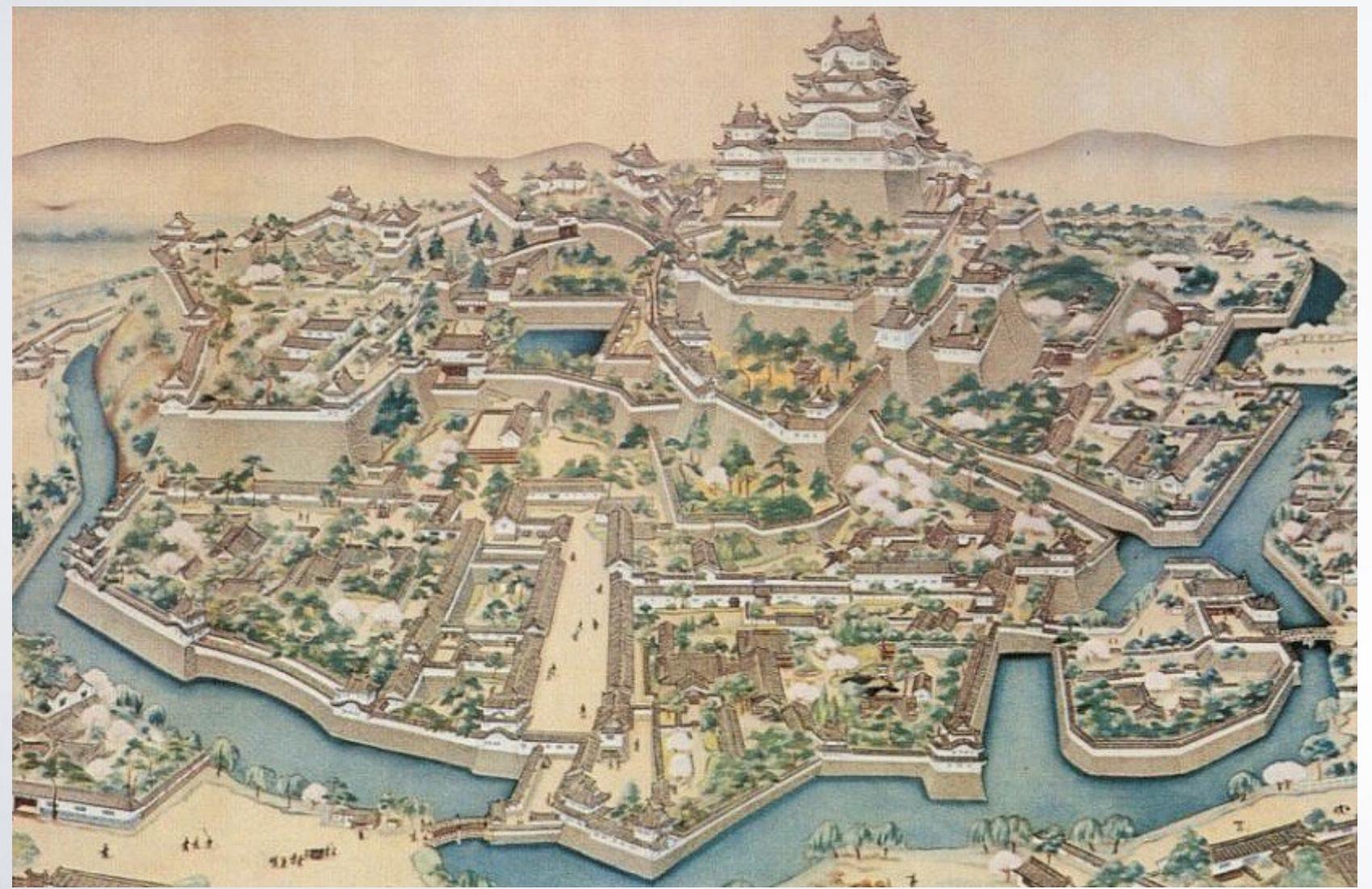
# Box Model

所有的HTML元素可以被視為一方塊(Box)，方塊內由四個子框組成。

從內到外分別是：

**內容(Content)**、**空白區(Padding)**、  
**邊線(Border)**、**邊境(Margin)**





Margin

Border

Padding

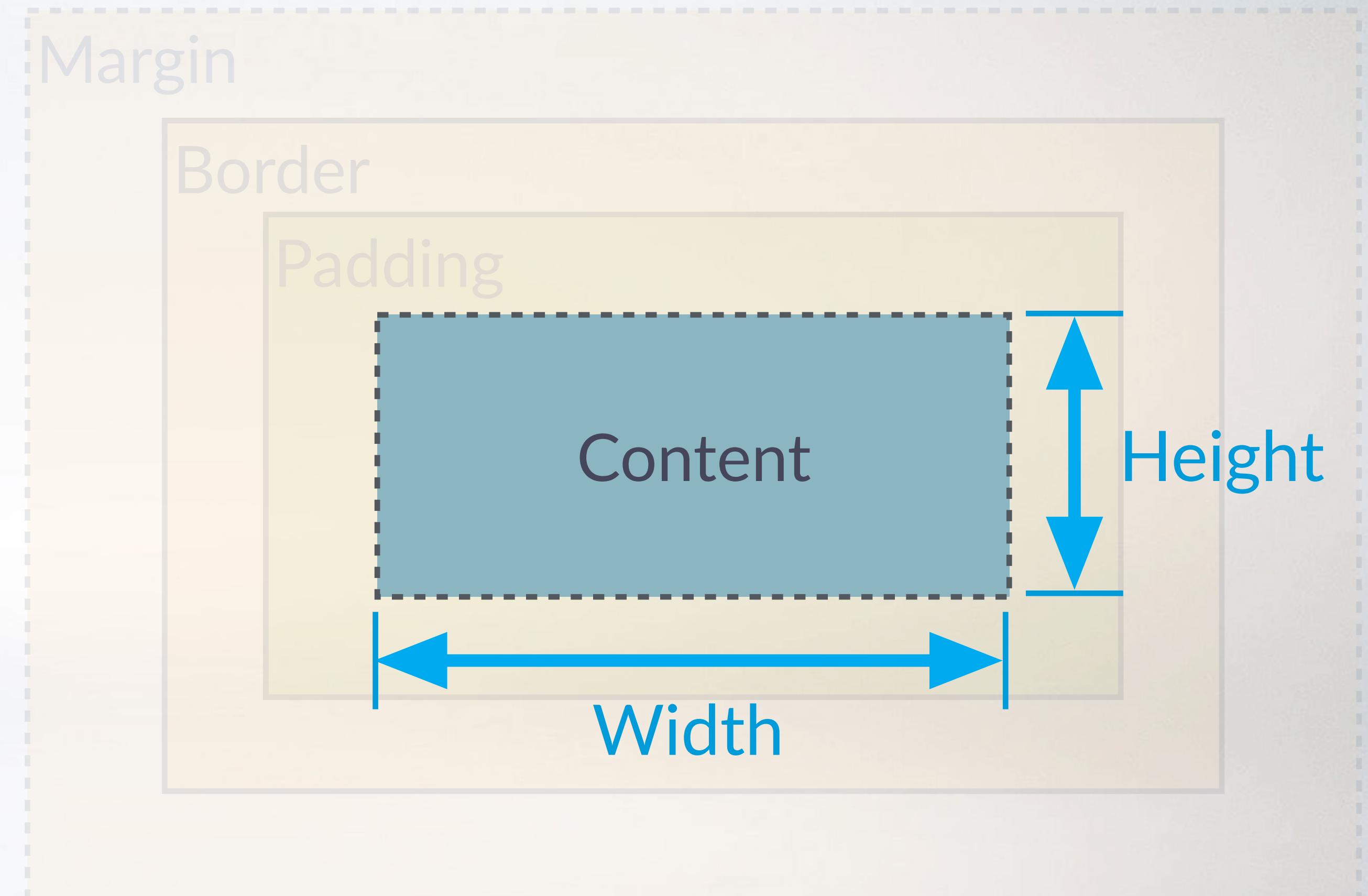
Content

## 內容(Content) - 城堡

文字或圖片出現的方框

- width
- height
- min-width
- min-height
- max-width
- max-height

```
width : 400px;  
min-width: 700px;  
max-width: 1000px;
```



## max-width (不會比這再更寬了)

定義元素「最大寬度」(不會比這再更寬了)，  
這時就沒有限定元素要多寬(width)，  
因此，此元素寬度具有可調整的彈性

max-width: 500px;

[按我連結](#)

## min-width (不會比這再更窄了)

定義元素「最小寬度」(不會比這再更窄了)，  
這時就沒有限定元素要多寬(width)，  
因此，此元素寬度具有可調整的彈性

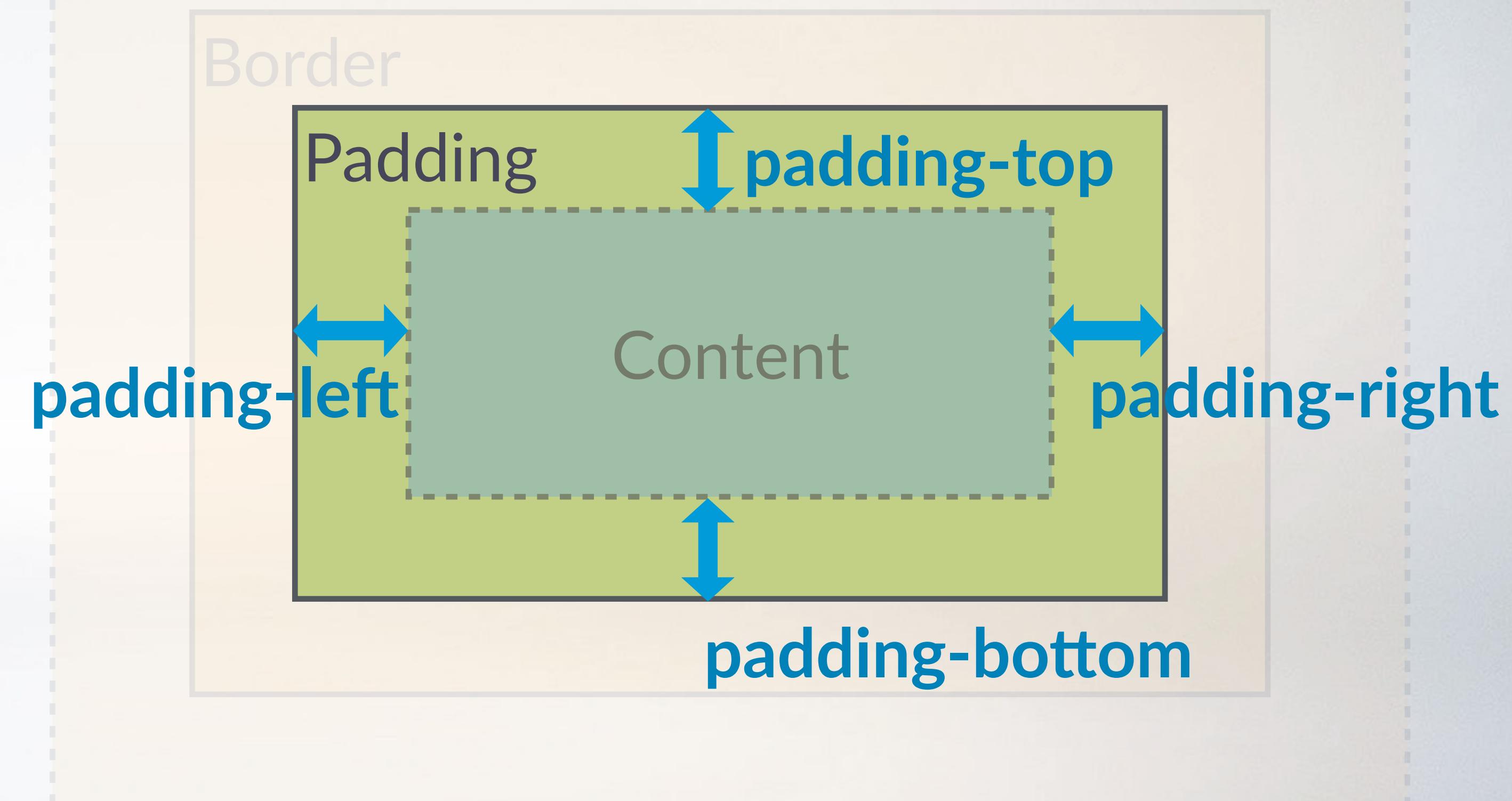
min-width: 500px;

[按我連結](#)

## 空白區(Padding) - 城堡至城牆區域(城內設施)

Content外圍的空白方框，  
背景受background影響

- padding-top
- padding-bottom
- padding-right
- padding-left
- padding



# —“條”成型(簡寫)

[貼貼看]

空白區(Padding) - 城堡至城牆區域(城內設施)

(1)padding: 上下左右;

(2)padding: 上下 左右;

(3)padding: 上 左右 下;

(4)padding: 上 右 下 左;

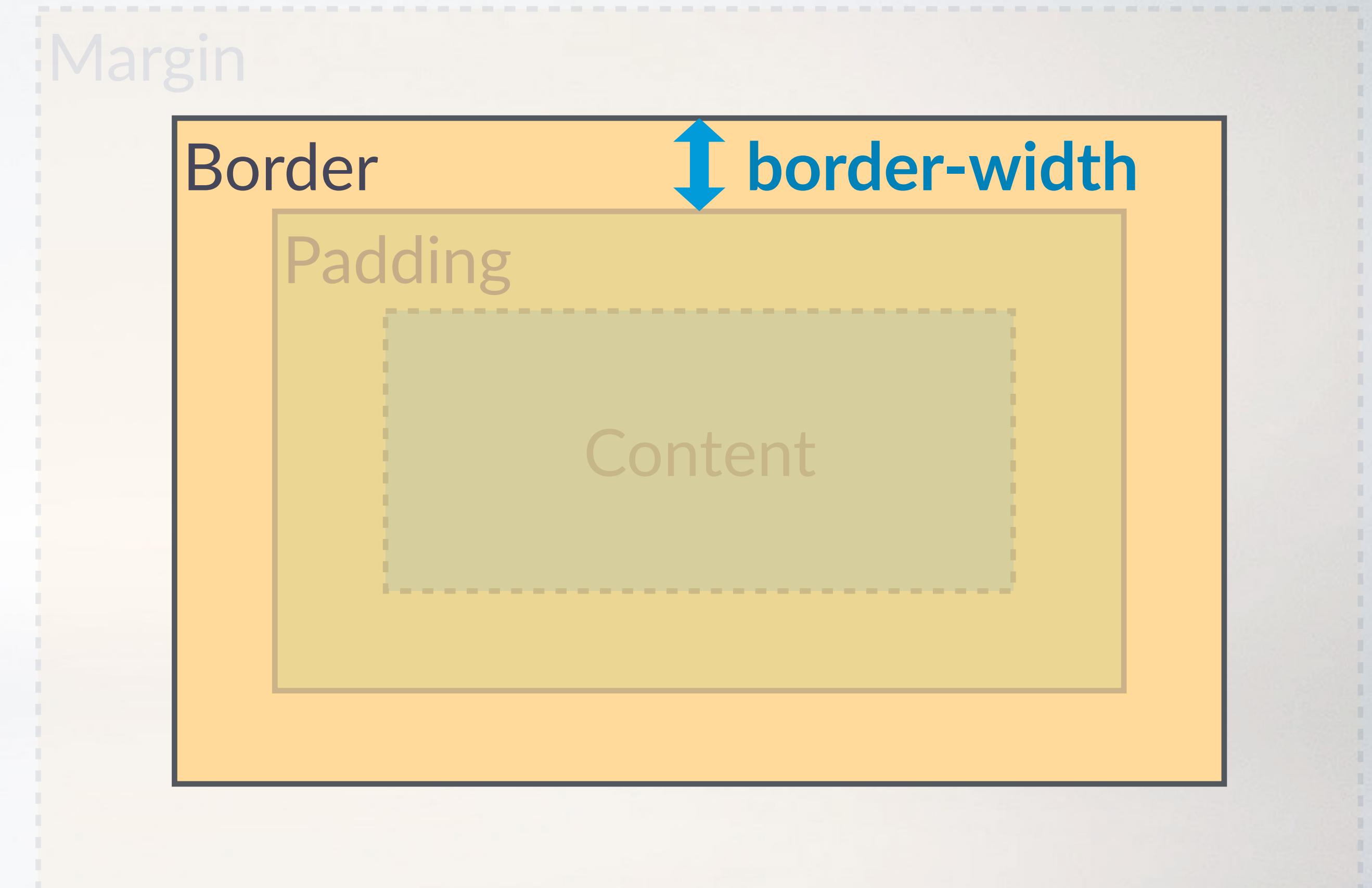
```
div{  
background: pink;  
width: 5em;  
  
/* padding: 1em; */  
/* padding: 2em 1em; */  
/* padding: 2em 1em 5em; */  
/* padding: 4em 1em 3em 9em; */  
}
```

```
<div> Content, period. </div>  
<div> Content, period. </div>  
<div> Content, period. </div>  
<div> Content, period. </div>
```

## 邊線(Border) - 城牆

Padding外圍的方框(線條)

- **border-width**
- **border-style (必填:預設none)**
- **border-color**
- **border: width style color**



# —“條”成型(簡寫)

邊線(Border) - 城牆

●**border: width style color**

●**border-top: width style color**

●**border-right: width style color**

●**border-bottom: width style color**

●**border-left: width style color**

[按我連結]

p{width: 200px;}

p.dotted {border-style: dotted}  
p.dashed {border-style: dashed}  
p.solid {border-style: solid}  
p.double {border-style: double}  
p.groove {border-style: groove}  
p.ridge {border-style: ridge}  
p.inset {border-style: inset}  
p.outset {border-style: outset}

<p class="dotted">A dotted border</p>  
<p class="dashed">A dashed border</p>  
<p class="solid">A solid border</p>  
<p class="double">A double border</p>  
<p class="groove">A groove border</p>  
<p class="ridge">A ridge border</p>  
<p class="inset">An inset border</p>  
<p class="outset">An outset border</p>

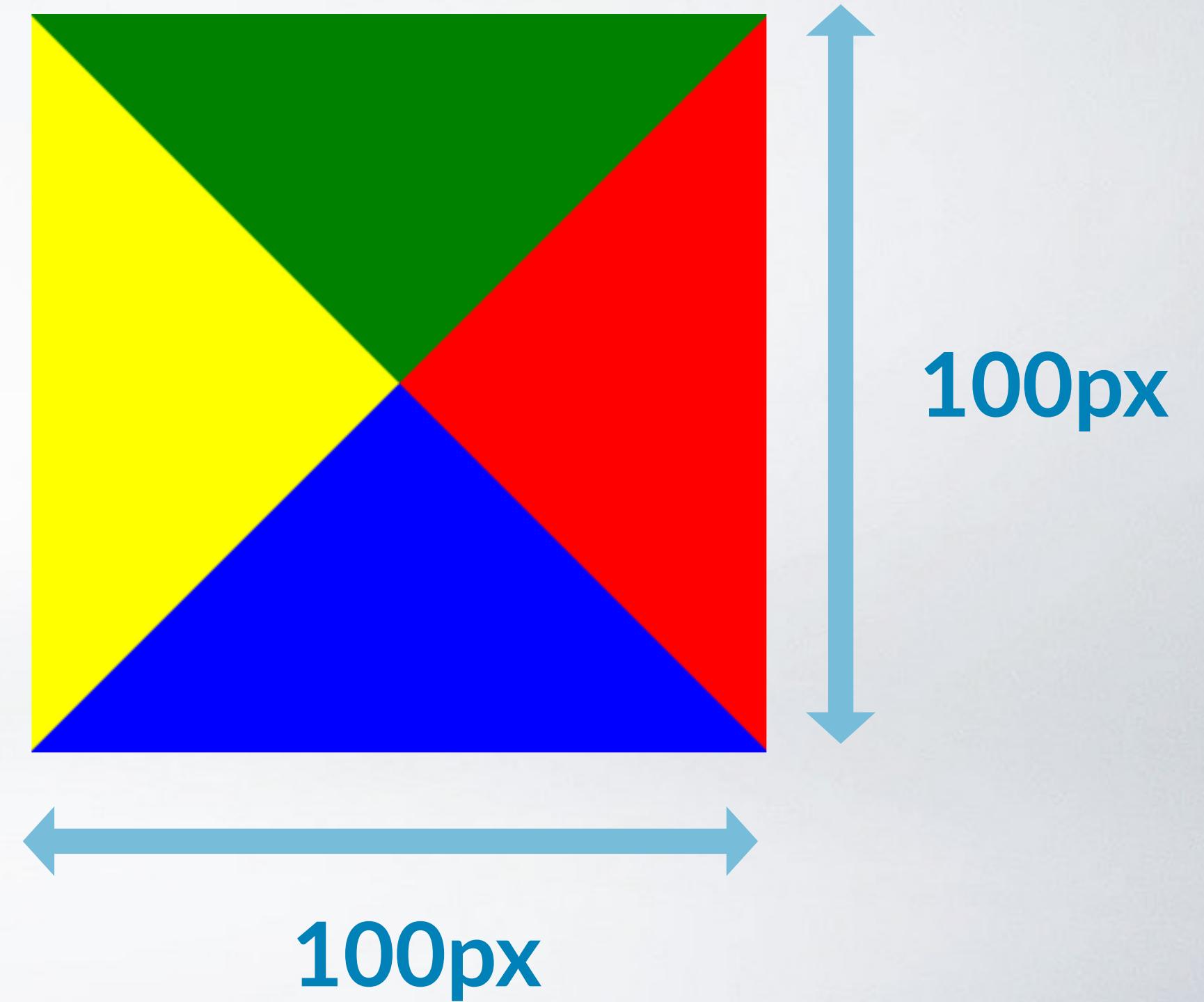


[溫馨小提示]

- 1.用border
- 2.content寬高都為0

# 動手時間

能做出這樣的方塊嗎？



# 你答對了嗎？

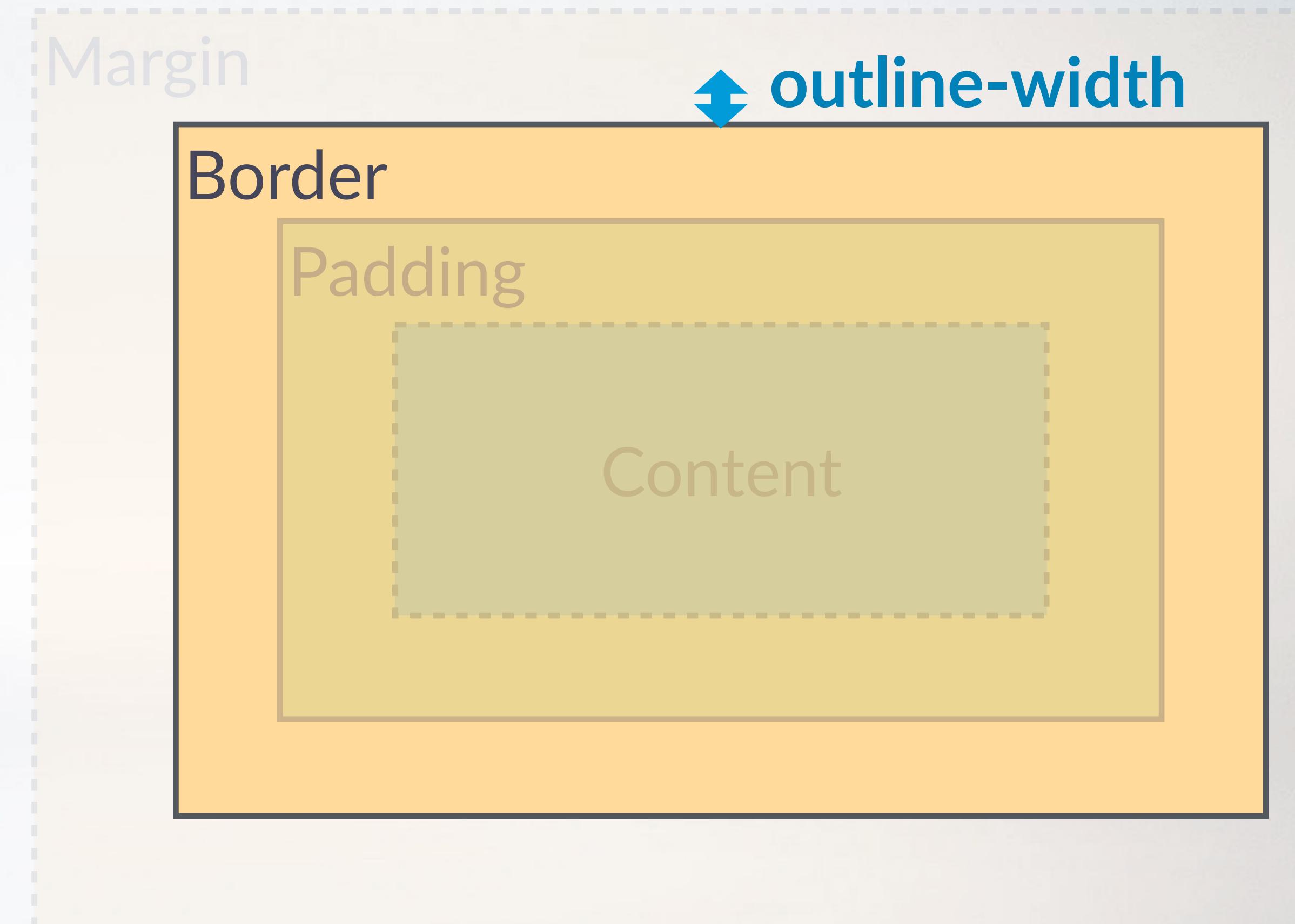
```
<div style="background-color:lightblue;width:0px;height:0px;border-top:50px solid green; border-bottom: 50px solid blue; border-right: 50px solid red; border-left: 50px solid yellow;"></div>
```

## 邊線的輪廓線(Outline) - 城牆上的雜草

border外圍的方框

(不增加額外空間，只佔用Margin區域)

- **outline-color**
  - **outline-style (必填:預設none)**
  - **outline-width**
- 
- **outline: color style width**

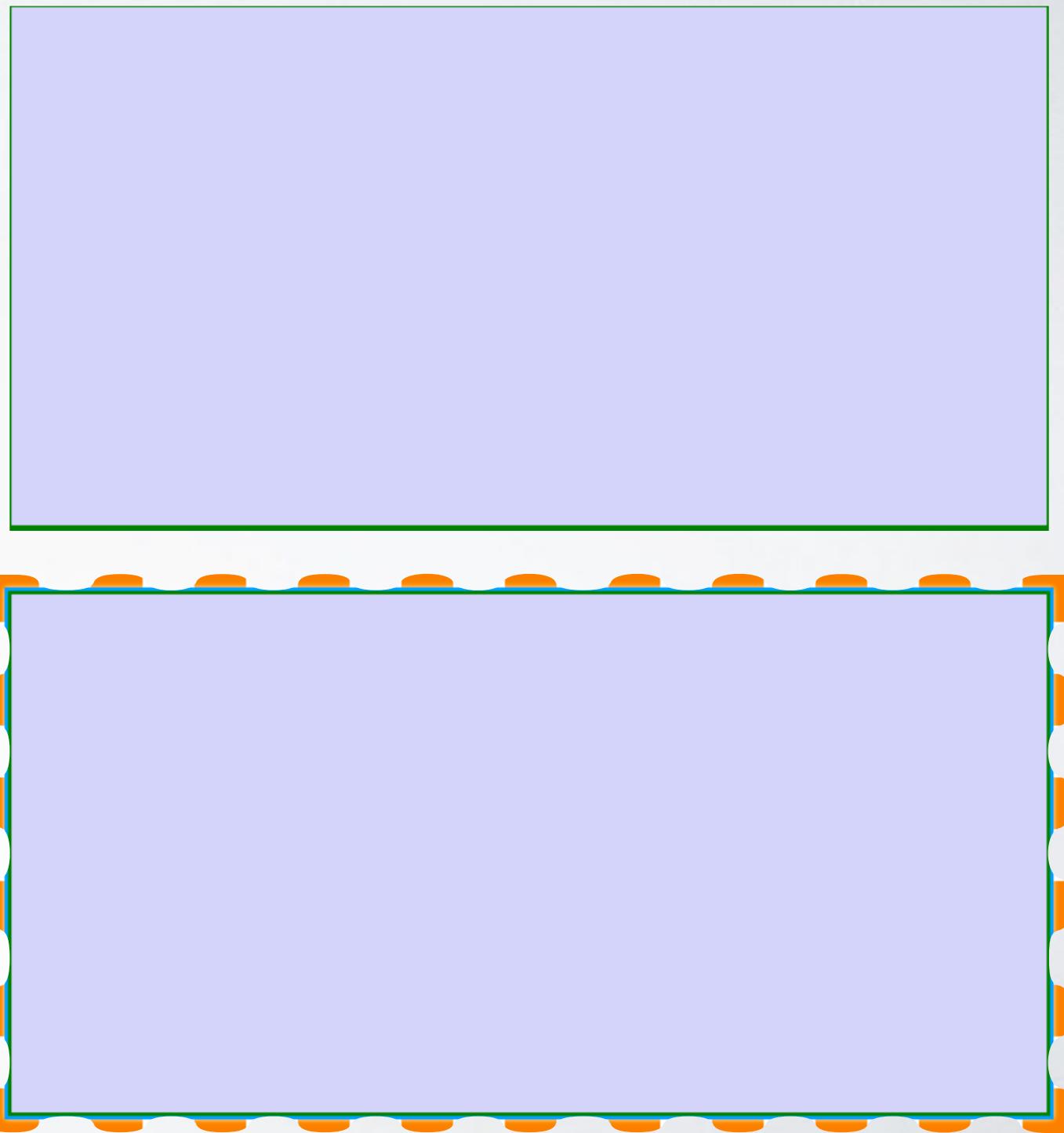


## 邊線的輪廓線(Outline) - 城牆上的雜草

border外圍的方框  
(不增加額外空間，只佔用Margin區域)

[按我連結](#)

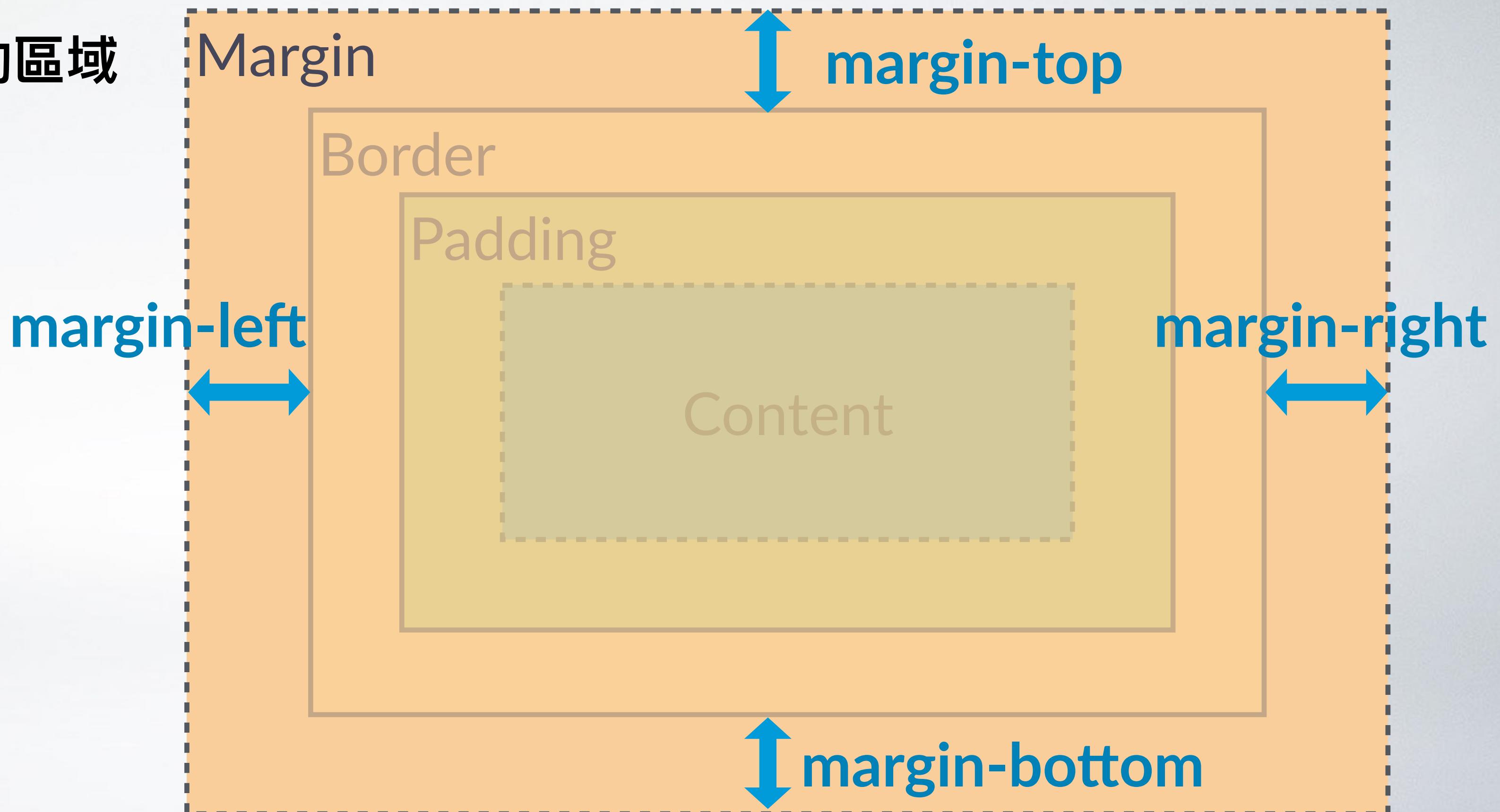
```
<div style="background-color:#D4D4F8;width:300px;height:150px;outline:#FA8000 dashed 5px;"></div>
<br>
<div style="background-color:#D4D4F8;width:300px;height:150px;outline:#FA8000 dashed 5px;"></div>
```



## 邊境(Margin) - 城牆至國界的區域

Border外圍的空白區域，  
背景必為透明

- margin-top
- margin-bottom
- margin-right
- margin-left
- margin



# 一”條”成型(簡寫)

邊境(Margin) - 城牆至國界的區域

(1)margin: 上下左右;

(2)margin: 上下 左右;

(3)margin: 上 左右 下;

(4)margin: 上 右 下 左;

[試試看]

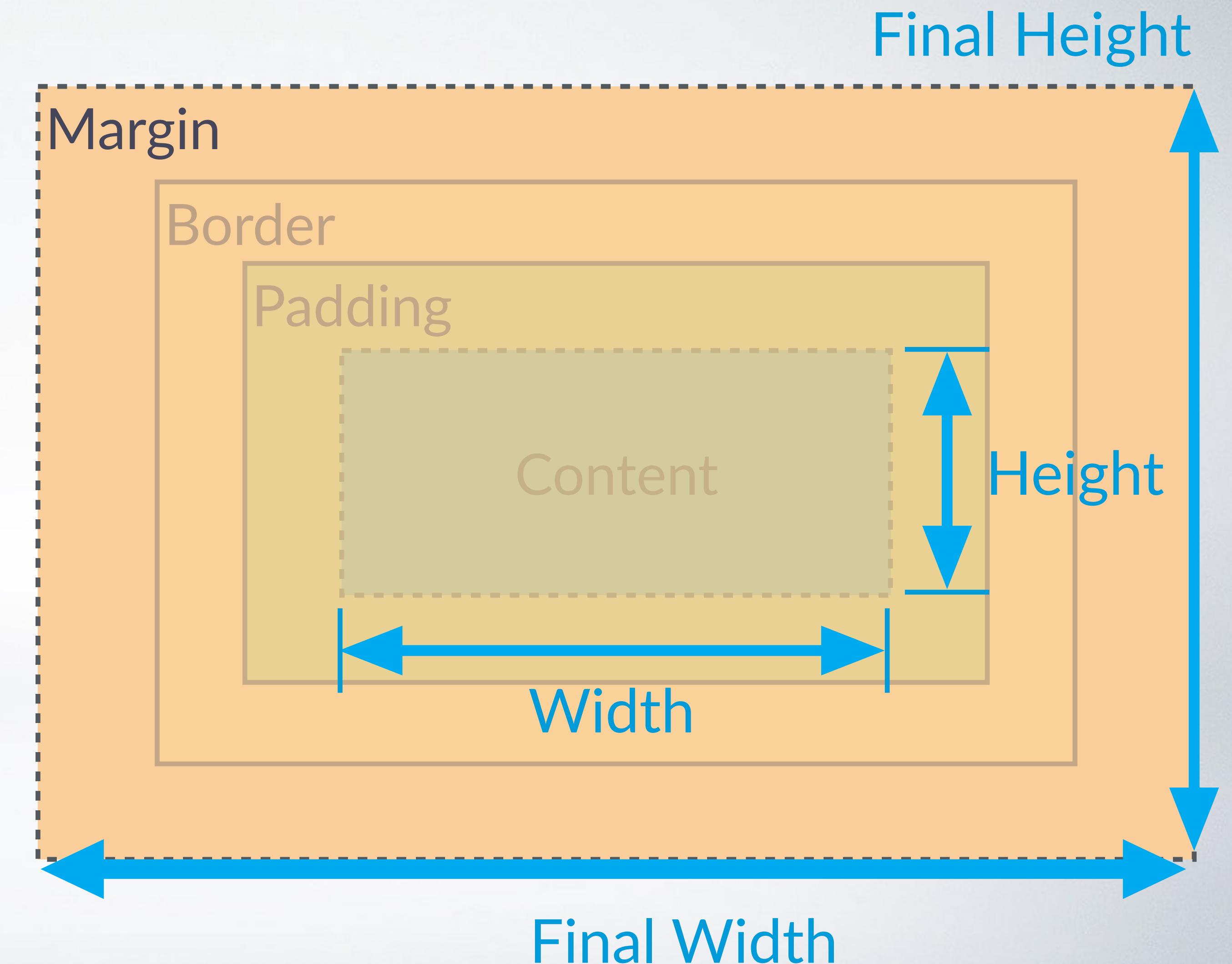
```
div{  
background: pink;  
width: 5em;  
  
/* margin: 1em; */  
/* margin: 2em 1em; */  
/* margin: 2em 1em 5em; */  
/* margin: 4em 1em 3em 9em; */  
}
```

```
<div> Content, period. </div>  
<div> Content, period. </div>  
<div> Content, period. </div>  
<div> Content, period. </div>
```

# 計算元素寬&高 content-box(預設)



```
<div style="width: 250px;  
height: 50px;  
padding: 10px;  
border: 9px solid green;  
margin: 40px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```



# 計算元素寬&高 content-box(預設)



```
<div style="width: 250px;  
height: 50px;  
padding: 10px;  
border: 9px solid green;  
margin: 40px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```

Final Width:  
250(content width)+  
10x2(padding left& right)+  
9x2(border left& right)+  
40x2(margin left& right)= 369

Final Height:  
50(content height)+  
10x2(padding top& bottom)+  
9x2(border top& bottom)+  
40x2(margin top& bottom)= 169

# 練習時間

```
<div style="width: 960px;  
height: 180px;  
padding: 5px;  
padding-left: 20px;  
margin-right: 20px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```

```
<div style="width: 440px;  
height: 270px;  
padding: 12px;  
border: 3px 4px 0px 1px;  
margin: 10px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```

Final Width?

Final Height?

# 練習時間

```
<div style="width: 960px;  
height: 180px;  
padding: 5px;  
padding-left: 20px;  
margin-right: 20px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```

Final Width:  
 $960(\text{content width}) + 5 + 2 \times 20(\text{padding left\& right}) + 20(\text{margin right}) = 1005$

Final Height:  
 $180(\text{content height}) + 5 \times 2(\text{padding top\& bottom}) = 190$

```
<div style="width: 440px;  
height: 270px;  
padding: 12px;  
border: 3px 4px 0px 1px;  
margin: 10px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```

Final Width:  
 $440(\text{content width}) + 12 \times 2(\text{padding left\& right}) + 1 + 4(\text{border left\& right}) + 10 \times 2(\text{margin left\& right}) = 489$

Final Height:  
 $270(\text{content height}) + 12 \times 2(\text{padding top\& bottom}) + 3 + 0(\text{border top\& bottom}) + 10 \times 2(\text{margin top\& bottom}) = 317$

# 練習時間 (Answer)

```
<div style="width: 960px;  
height: 180px;  
padding: 5px;  
padding-left: 20px;  
margin-right: 20px;  
background: rgb(255, 155, 55);>  
    Content  
</div>
```

Final Width:  
 $960(\text{content width}) + 5 + 2 \times 20 + 20 = 1005$

Final Height:  
 $180(\text{content height}) + 5 \times 2 = 190$

```
<div style="width: 440px;  
height: 270px;  
padding: 12px;  
border: 3px 4px 0px 1px;  
margin: 10px;  
background: rgb(255, 155, 55);">  
    Content  
</div>
```

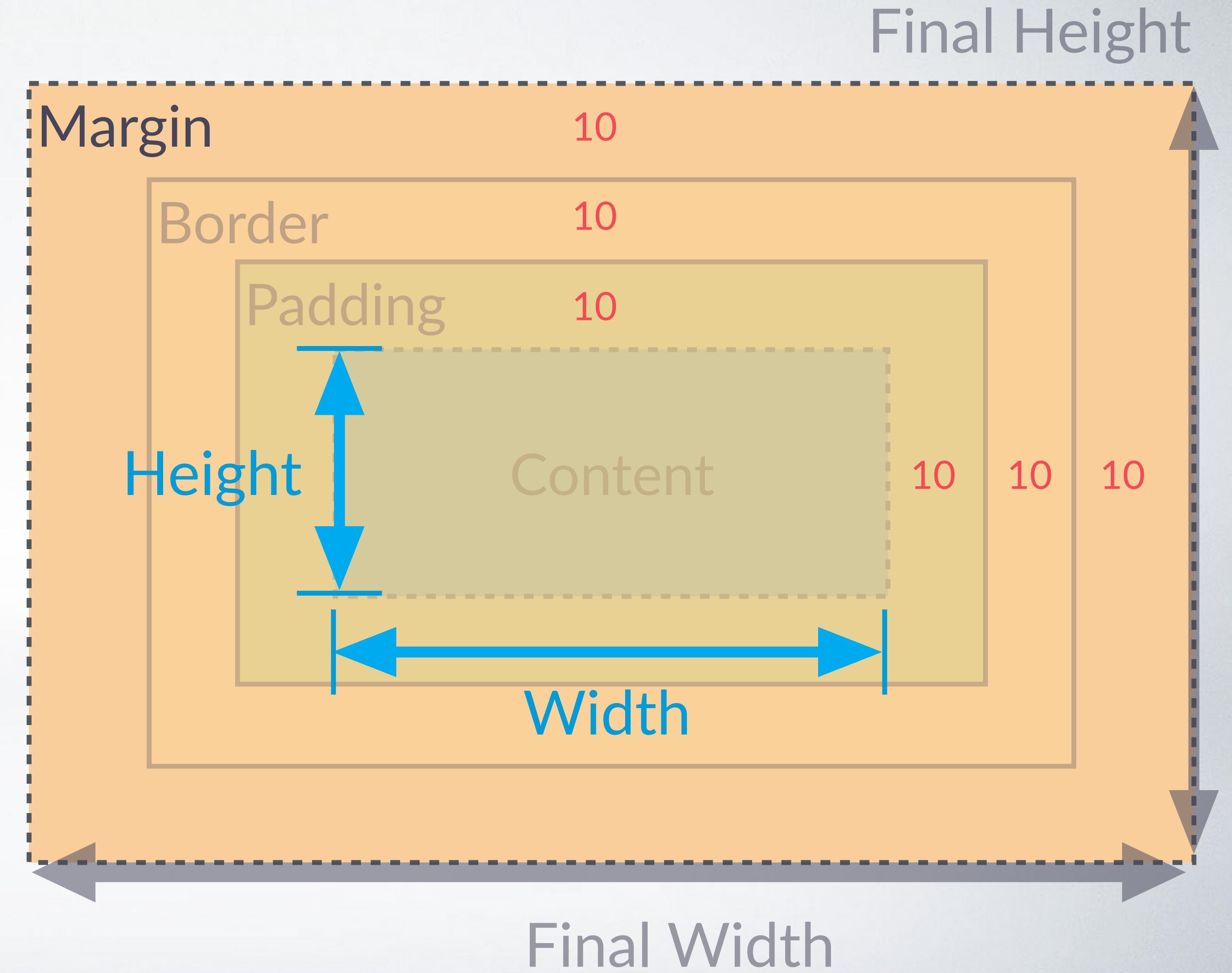
Final Width:  
 $440(\text{content width}) + 12 \times 2 + 1 + 4 + 10 \times 2 = 489$

Final Height:  
 $270(\text{content height}) + 12 \times 2 + 3 + 10 \times 2 = 317$

# 計算元素寬&高 content-box(預設)

- width
- min-width
- max-width
- height
- min-height
- max-height

```
width : 400px;  
height: 200px;
```



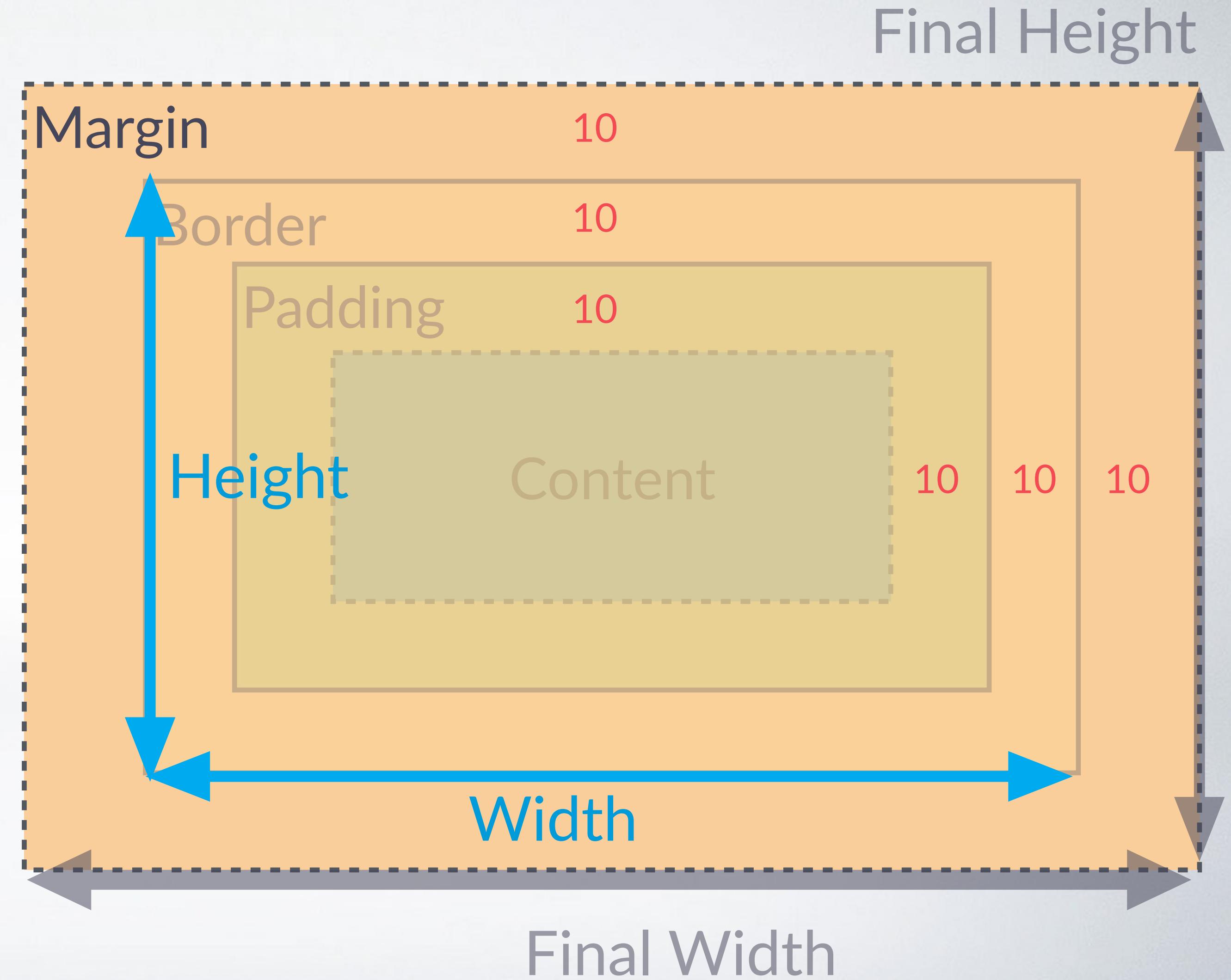
# 計算 元素寬&高 border-box

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

- width
- min-width
- max-width
- height
- min-height
- max-height

```
width : 440px;  
height: 240px;
```

[按我連結](#)



# CSS重要屬性: display

設計 CSS 版面配置中最重要的屬性

## [提示]

- 1.每個元素都有一個預設的 display 屬性，但你可以把它改掉！
- 2.display : inline 元素不該包住 display : block 元素
- 3.display : inline 元素 不能使用height等相關宣告

## block: 區塊元素

- ☆ 不管 display : block 元素的前面後面是什麼，碰到 display : block 元素就是會換行，並盡可能的撐滿容器 **Ex. div, p, h1**

## inline: 行內元素

- ☆ 可以在段落中包裹一些文字片段，不會打亂段落原本的版面配置 **Ex. span, a, img**

## none: 完全消失

- ☆ 完全消失，毫無容身之處 **Ex. script**

[按我連結](#)

# CSS重要屬性: display

設計 CSS 版面配置中最重要的屬性

**block:** 區塊元素

- ☆ 不管 display : block 元素的前面後面是什麼，碰到 display : block 元素就是會換行，並盡可能的撐滿容器 **Ex. div, p, h1**

**inline:** 行內元素

- ☆ 可以在段落中包裹一些文字片段，不會打亂段落原本的版面配置 **Ex. span, a, img**

**inline-block:** 外行內區元素

- ☆ 外面是 inline ，裡面是 block。所以碰到 display : inline-block 元素不會換行，但是又可以設定 padding-top 、 padding-bottom 、 width 、 height 、 background-image

# 網頁設計小知識： 區塊元素(block) 要如何水平置中

```
<!DOCTYPE html>
<ul>
<li>HTML</li>
<li>CSS</li>
<li>JavaScript</li>
</ul>
```

```
ul{
    /* margin: 上下 左右 */
    margin: 10px auto;
    width: 200px;
    background: lightgreen;
    border: 1px solid red;
}

li{
    /* li 間距 20px，第一個 li 的上邊界會頂出 ul */
    margin-top: 20px;
    margin-bottom: 10px;
    border: 1px solid green;
}
```

[Ans]  
**margin-left: auto;  
margin-right: auto;**

- HTML
- CSS
- JavaScript

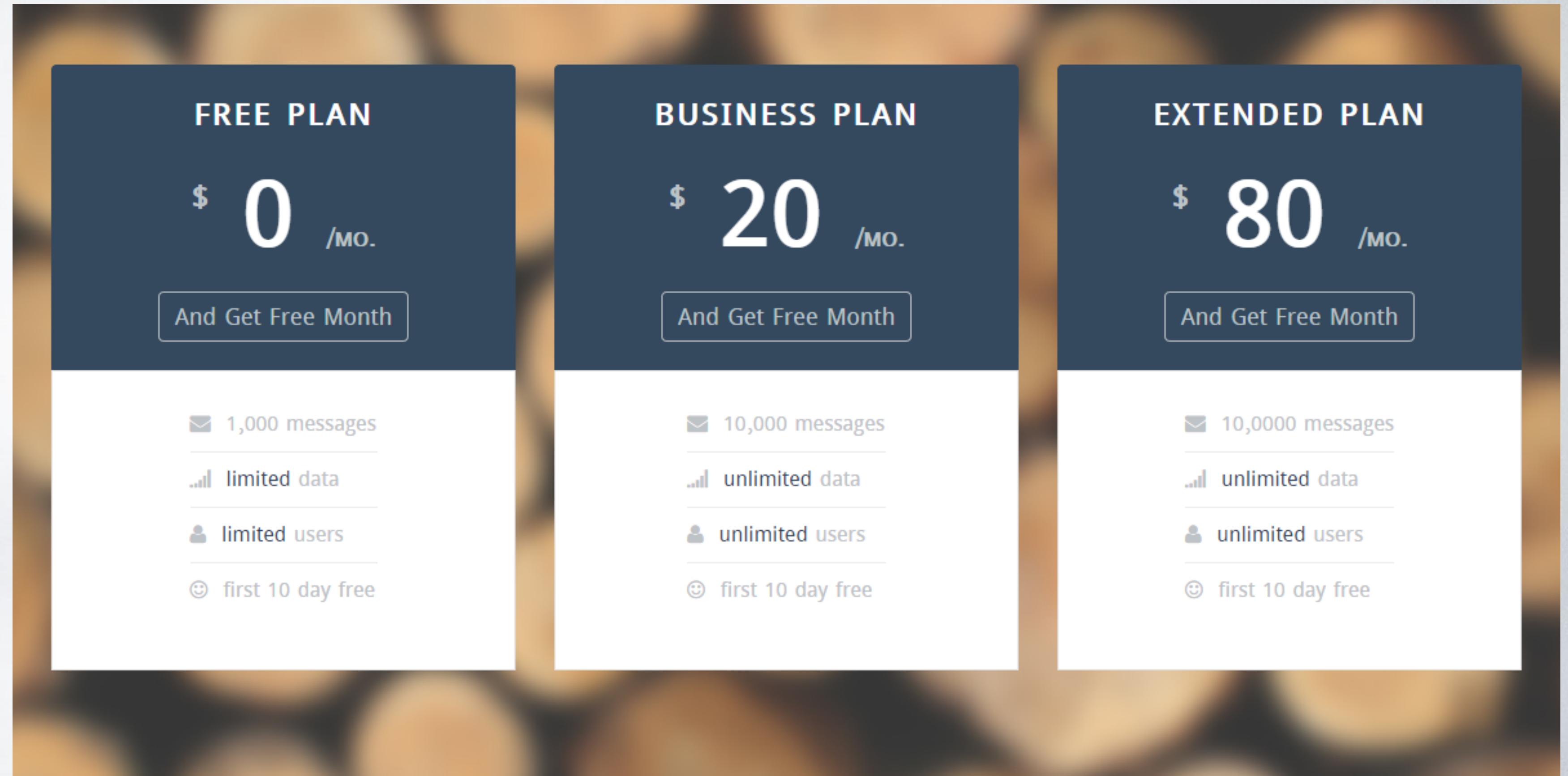
[按我練習](#)



# Homework : CSS-part2

## [要求]

- 1.不需要按鈕，border即可
- 2.不可用table，試著用  
inline-block&margin等排版

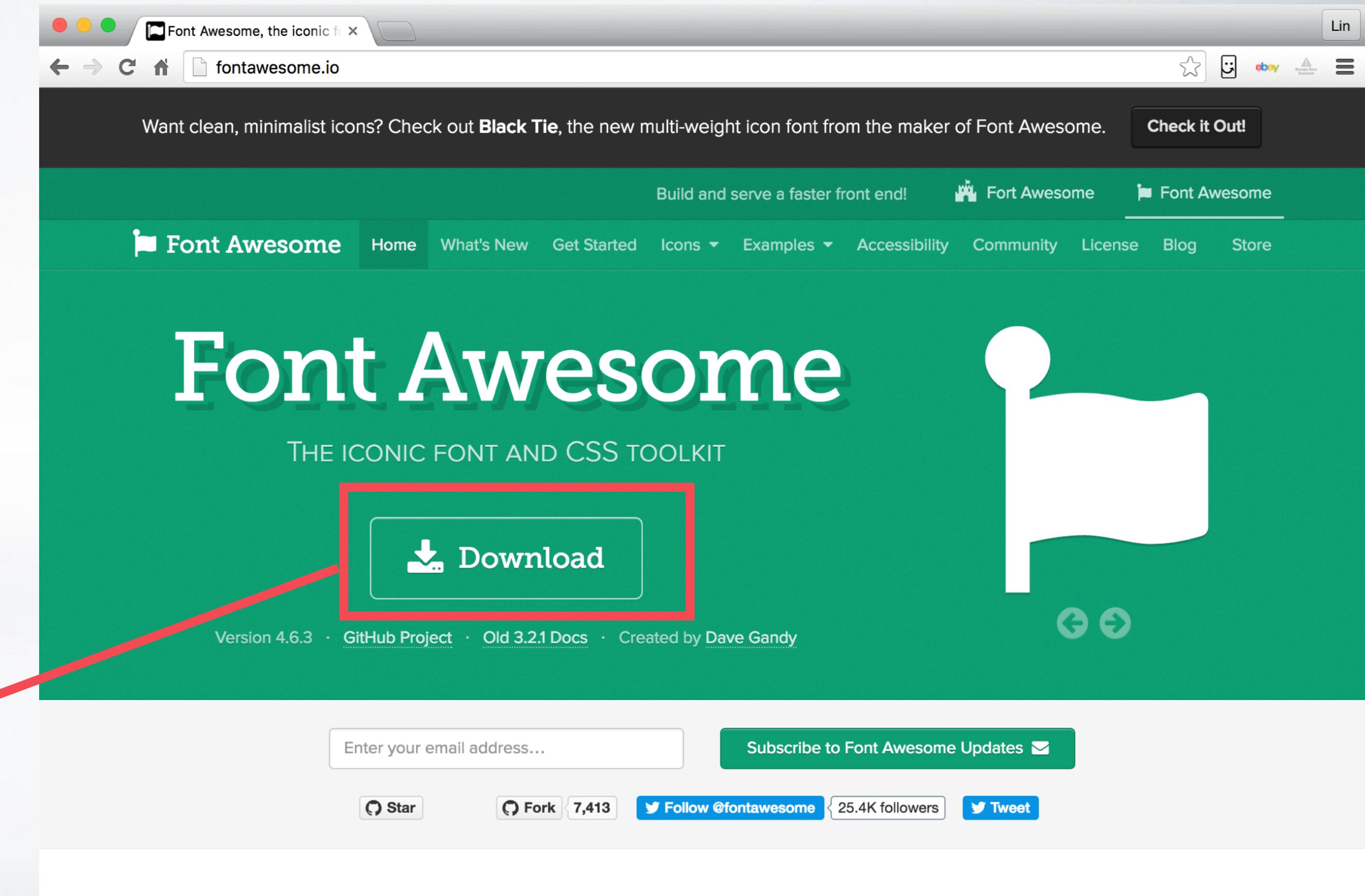


# 如何加入圖示(icon) ?

## 來試試Font Awesome吧

連結：<http://fontawesome.io>

Step 1: 下載專案



# 在自己的專案中啟動 Font Awesome

Step 2: 把css & fonts資料夾放自己的專案夾裡

Step 3: 在html裡加入：

```
<link rel="stylesheet" type="text/css" href="font-awesome.min.css">
```



The screenshot shows the Brackets code editor with the file 'index-3.html' open. The code editor has a dark theme. The code is as follows:

```
index-3.html (Class-01-HTML) — Brackets
1  <!doctype html>
2  <html>
3
4  <head>
5      <meta charset="UTF-8">
6      <title>Homework 1</title>
7      <link rel="stylesheet" type="text/css" href="main.css">
8      <link rel="stylesheet" type="text/css" href="font-awesome.min.css">
9  </head>
10
11 <body class="root">
12
13     <div align="center">
14         學
15         <p>HTML</p>
16         好好玩
17     </div>
18     <hr>
19     <div>
```

The line '8 <link rel="stylesheet" type="text/css" href="font-awesome.min.css">' is highlighted in green, indicating it has been selected or is being edited.

Step 4: 試試看！

```
<i class="fa fa-camera-retro"></i>
fa-camera-retro
```

Step 5: 定義id或style改變外觀

# 如何做長方形？



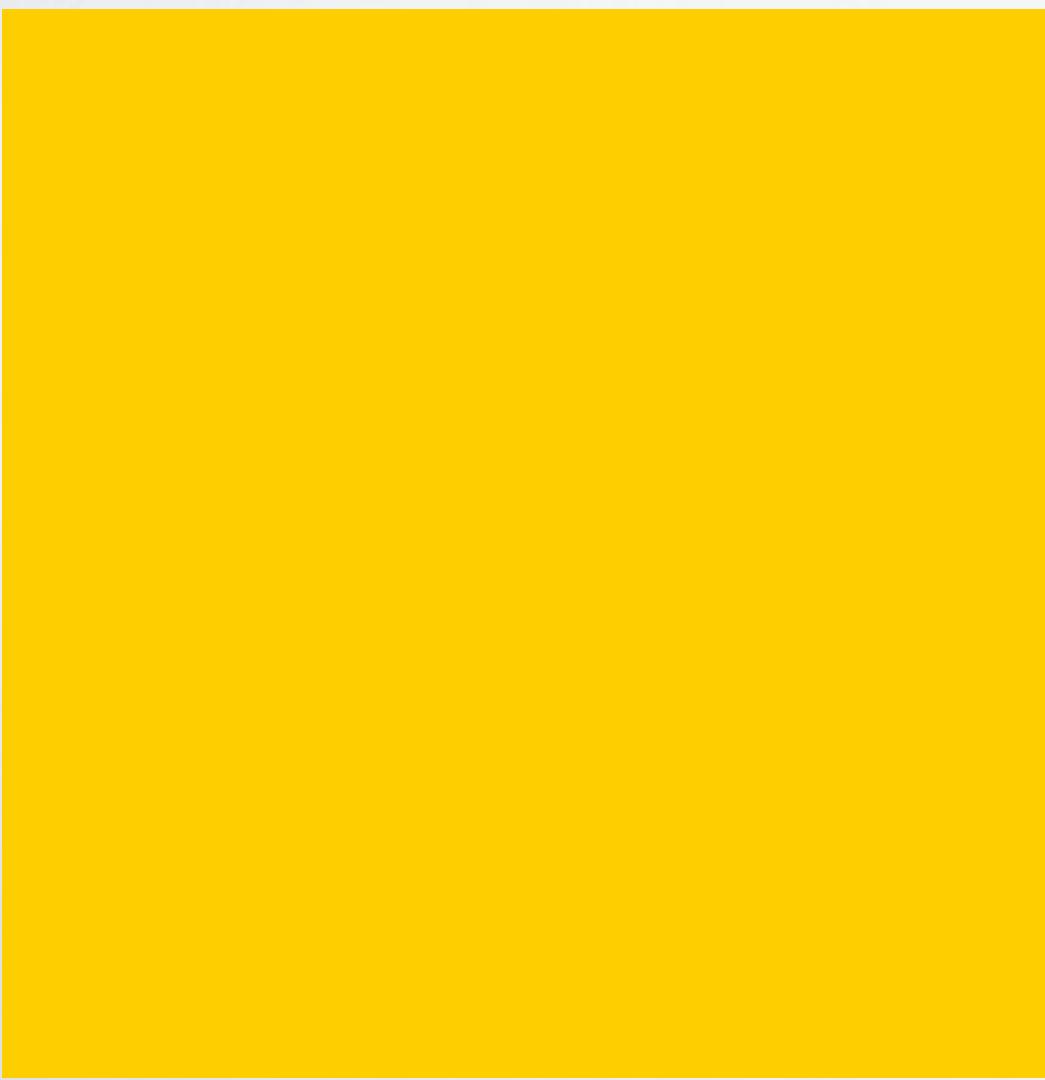
CSS

```
.rect{  
    width:300px;  
    height:150px;  
    background:gold;  
}
```

HTML

```
<div class="rect"></div>
```

# 如何做正方形？



CSS

```
.square{  
    width:300px;  
    height:300px;  
    background:gold;  
}
```

HTML

```
<div class="square"></div>
```

# 如何做有圓角正方形？



## CSS

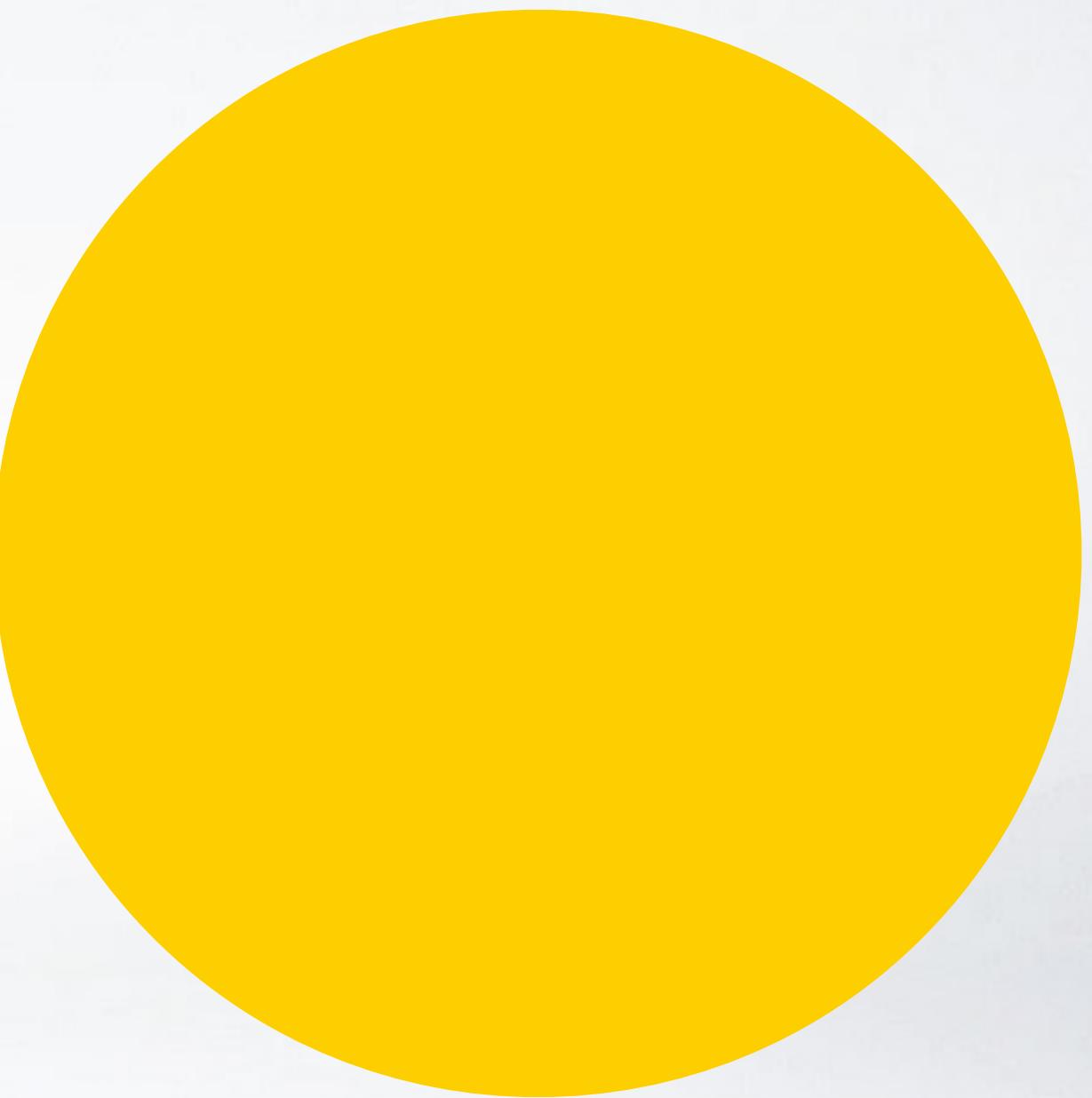
```
.radius-square{  
    width:300px;  
    height:300px;  
    border-radius:8px;  
    background:gold;  
}
```

## HTML

```
<div class="radius-square"></div>
```

# 小小測驗時間

其實你會做一個圓形 (?)



# 你答對了嗎？

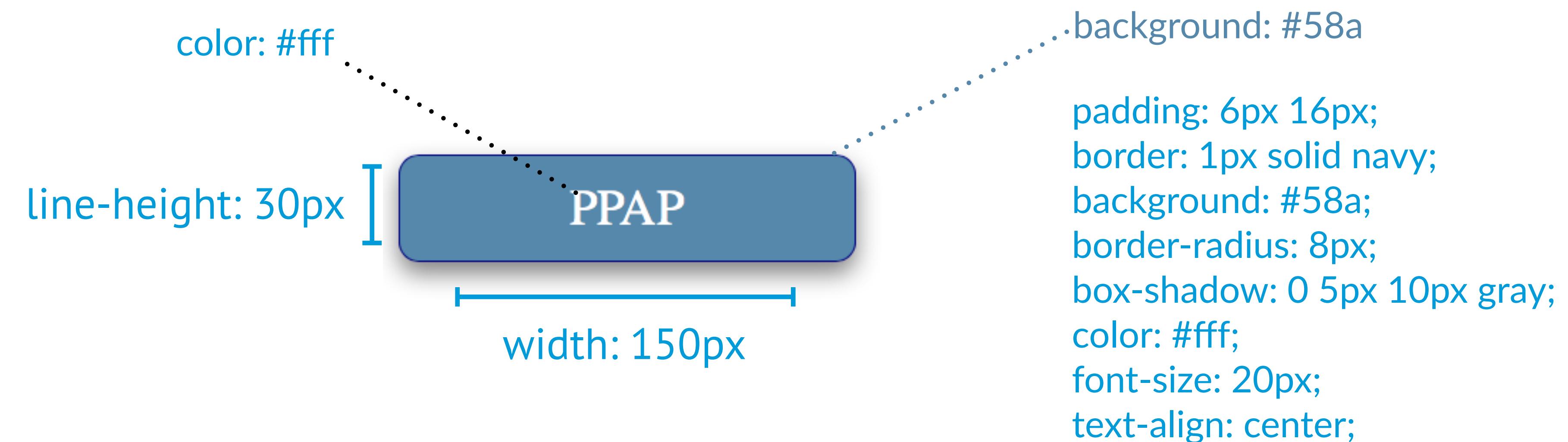
CSS

```
.circle{  
    width:300px;  
    height:300px;  
    border-radius:50%;  
    background:gold;  
}
```

HTML

```
<div class="circle"></div>
```

# 來試做個按鈕吧！



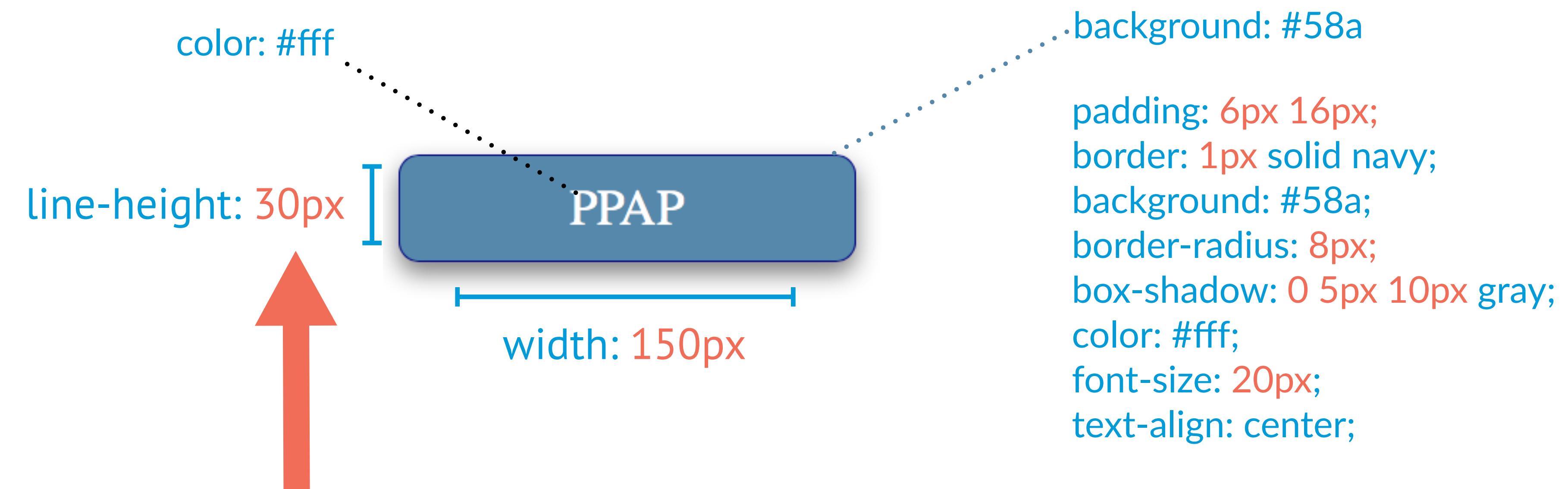
：「我老花，這個按鈕的字可以再大一點嗎？」



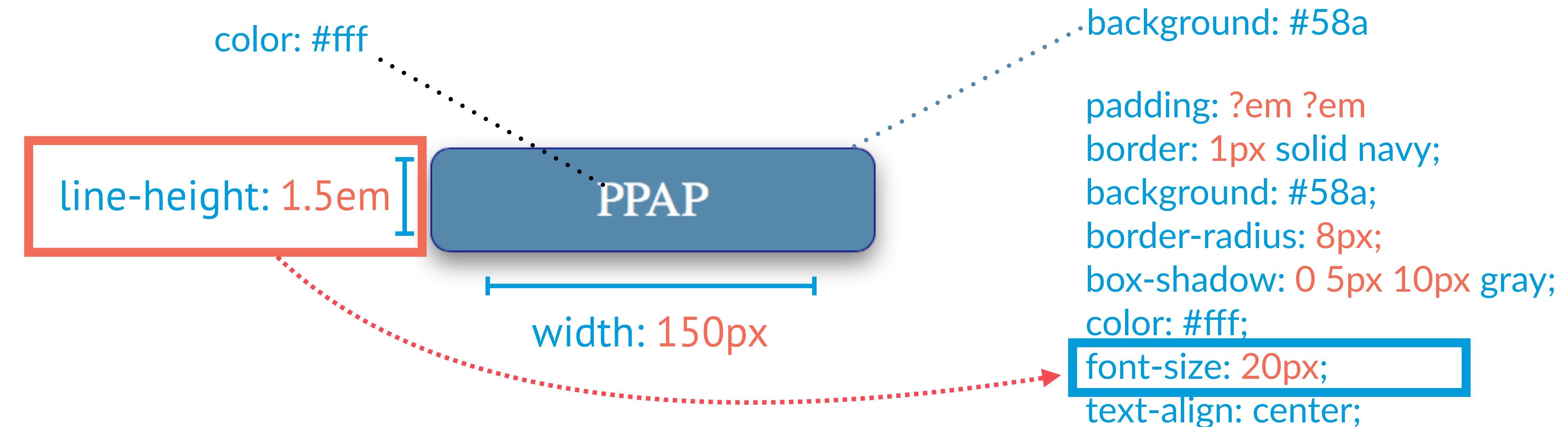
「大給你看！」：

[上一頁答案](#)

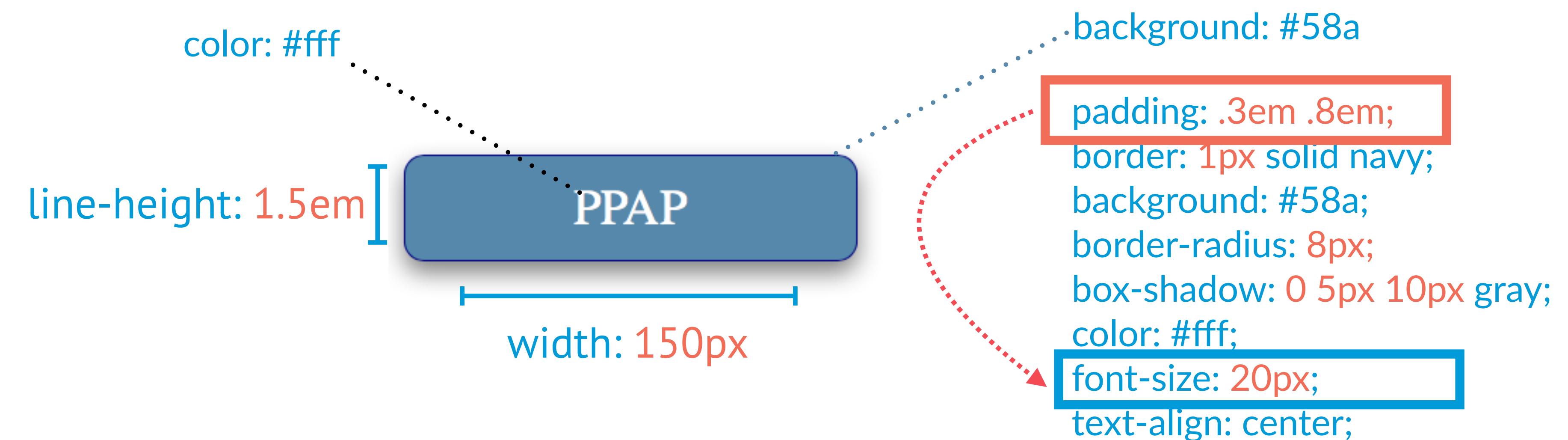
# 哪裡可能需要調整？



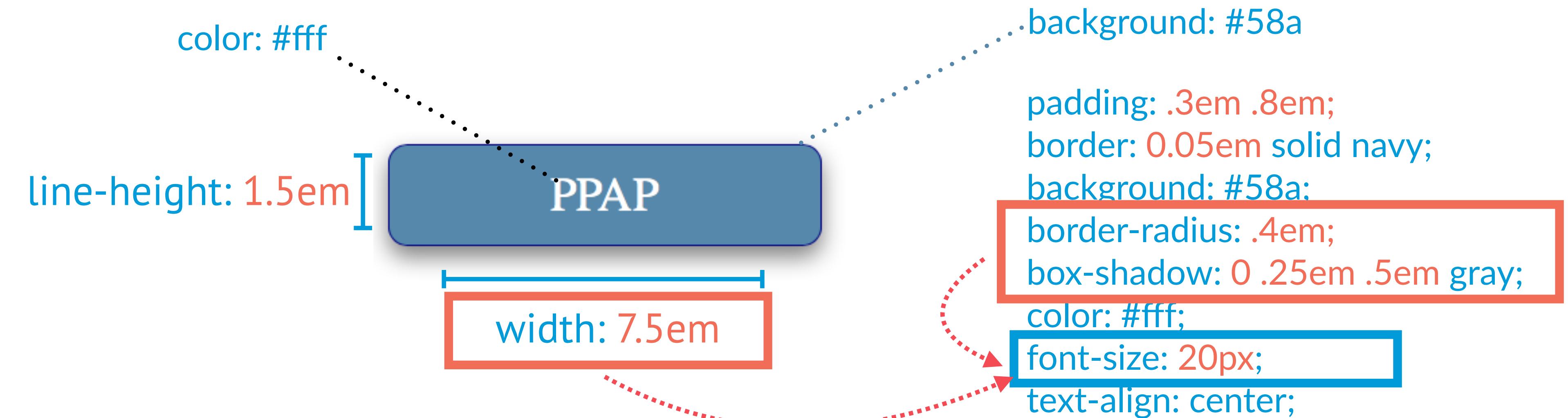
# 哪裡可能需要調整？



# 哪裡可能需要調整？



# 哪裡可能需要調整？



「要多大就有多大喎！  
夠字大嗎？」：

PPAP

[答案在這裡](#)

# 連結HTML與CSS方法一：<link>外部CSS檔案

index-3.html (Class-01-HTML) — Brackets

```
1  <!doctype html>
2  ▼ <html>
3
4  ▼ <head>
5    <meta charset="UTF-8">
6    <title>Homework 1</title>
7    <link rel="stylesheet" type="text/css" href="main.css">
8  </head>
9
10 ▼ <body>
11
```

<link rel="stylesheet" type="text/css" href="main.css">

```
15      好好玩
16
17    </div>
18    <hr>|
19  ▼ <div>
20    <h1>小王子</h1>
21    <p>然後他又回到狐狸那裡。</p>
22    <p>「再見了！」小王子說。</p>
```

## 連結HTML與CSS方法二：`<head>`裡嵌入`<style>`樣式表

```
• index.html (0629-gh-pages) — Brackets

1  <!doctype html>
2  <html>
3  <head>
4      <meta charset="UTF-8">
5      <title>Rayson's 06/29 Homework</title>
6      <style>
7          .circle{
8              width:50px;
9              height:50px;
10             border-radius:50%;
11             font-size:23px;
12             font-family:arial;
13             font-weight:900;
14             color:black;
15             line-height:50px;
16             text-align:center;
17             background-color:rgb(255,191,17);
18         }
19     </style>
20 </head>
21
22 <body>
```

## 連結HTML與CSS方法三：標籤裡增加style屬性

```
index.html (0629-gh-pages) — Brackets
55 <span style="font-size:14px;">
56   
```

# CSS-優先權

## 誰說定義不會打架

CSS :

```
.loud{  
    color: red;  
}  
  
div{  
    color: blue;  
}
```

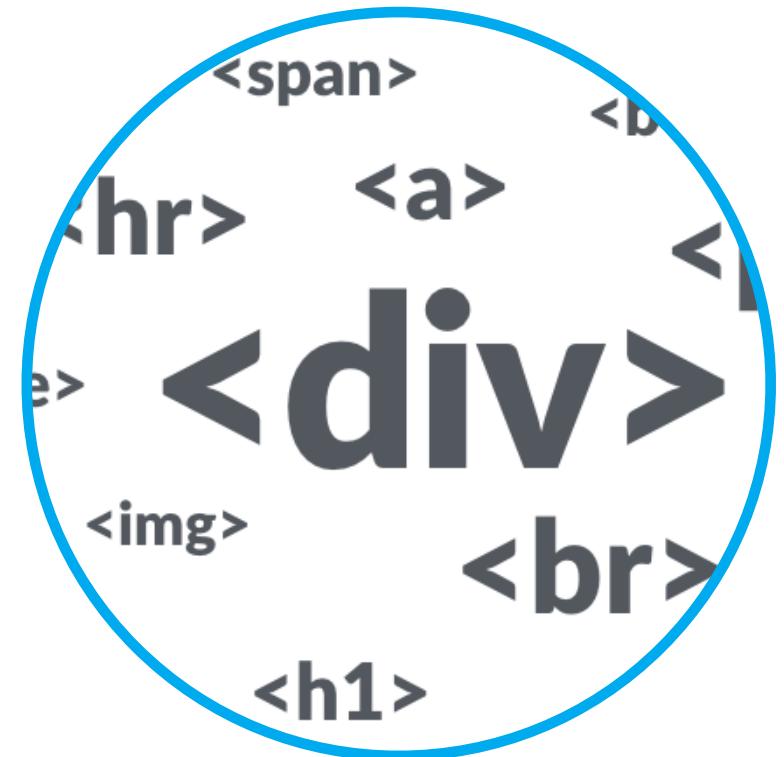
HTML :

```
<div class="loud">  
    What color is this?  
</div>
```



# 最基本的4類 選擇器

1



標籤 選擇器

2



類別(Class) 選擇器

3



ID 選擇器

4

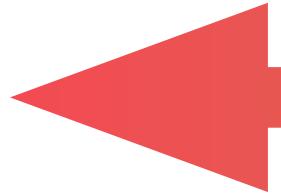


屬性 選擇器

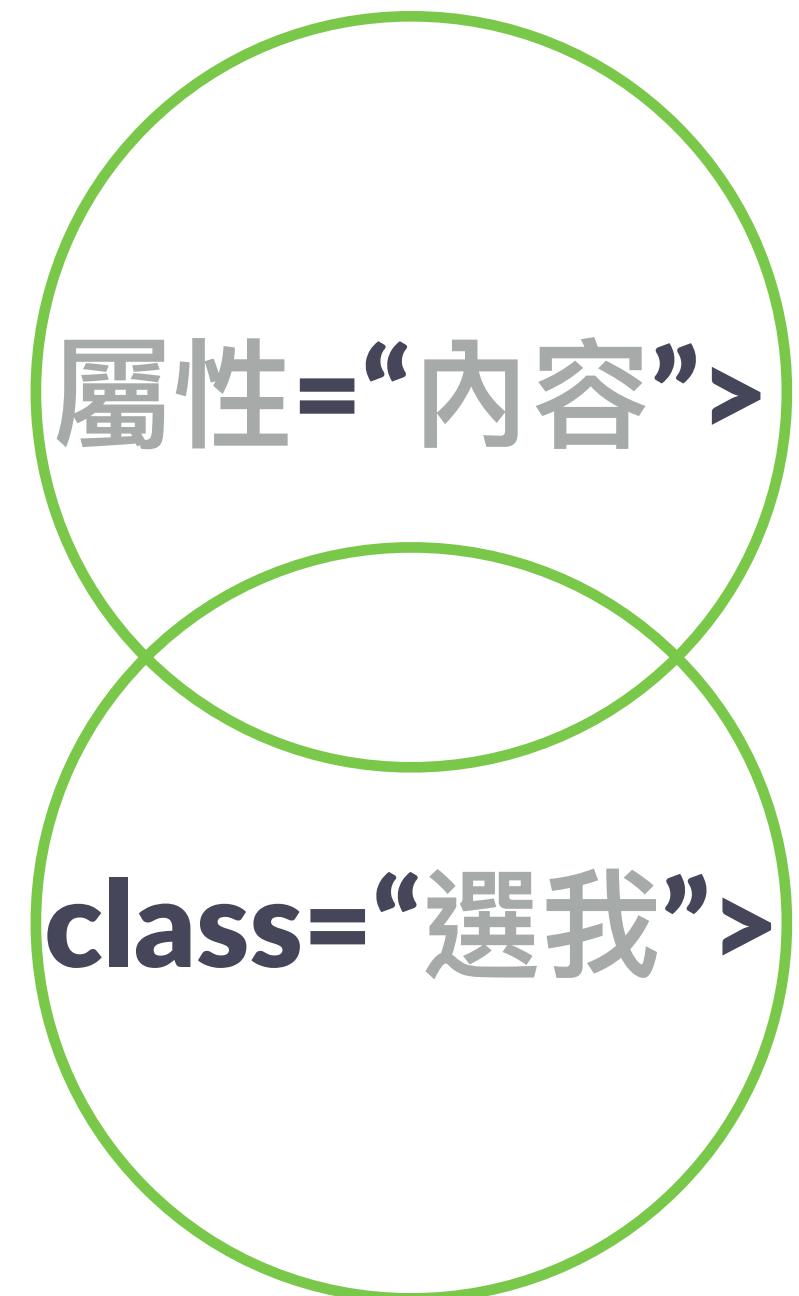
# 最基本的4類 選擇器 按照專一性排列



ID 選擇器

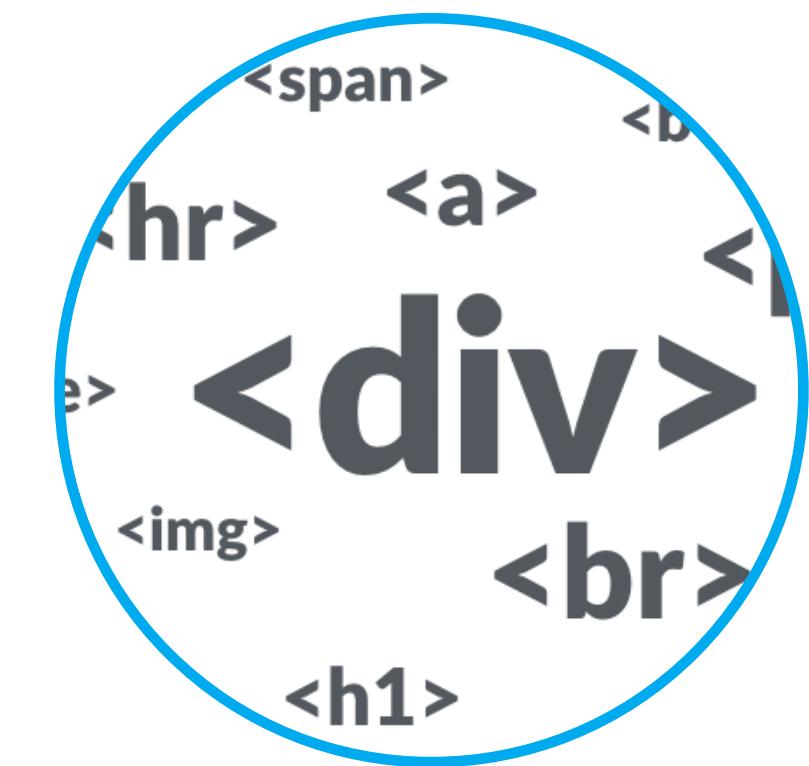


專一

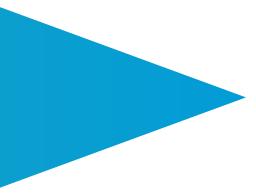


屬性 選擇器

類別(Class) 選擇器



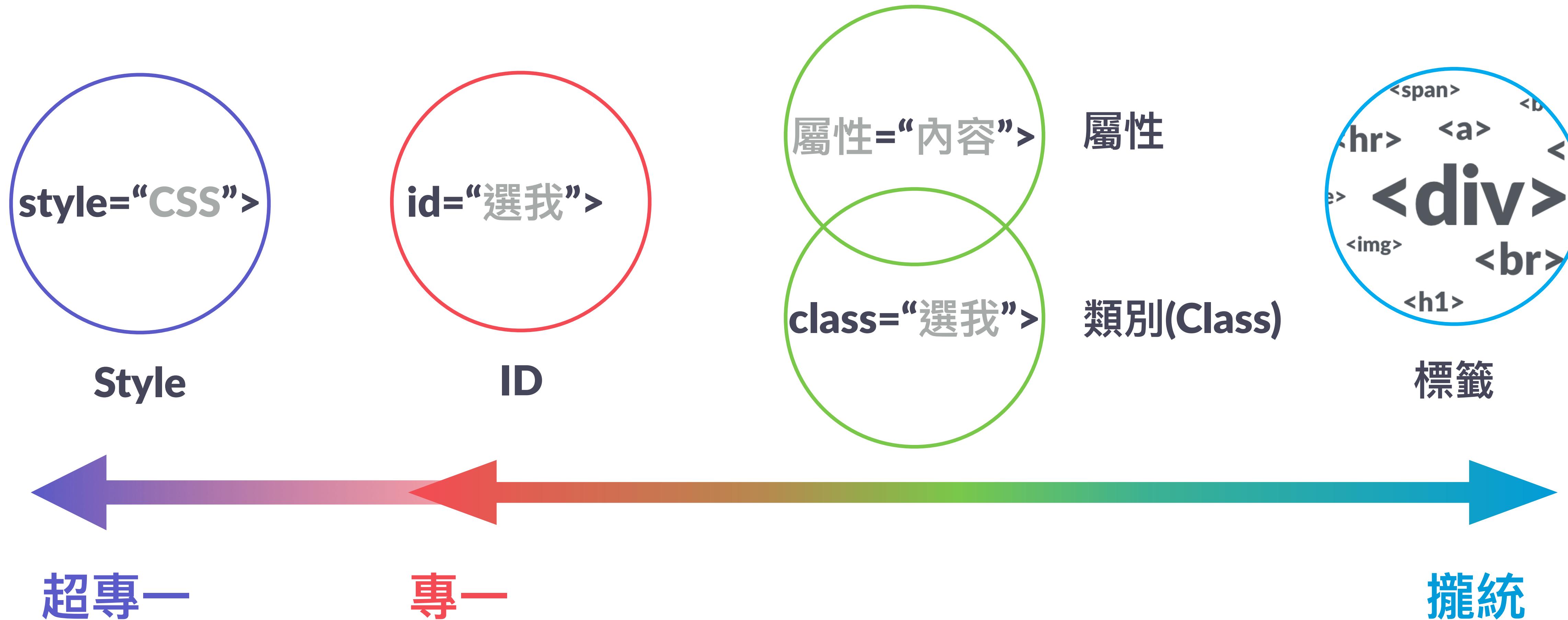
標籤 選擇器



攏統

# 最基本的4類 選擇器 + Style

## 按照專一性排列



# 優先權規則



# 優先權規則

超專一(最優先) 專一(優先) 捏統



# 知道答案了嗎？

吵架時就需要有原則的調解

解答: What color is this?

CSS :

```
.loud{  
    color: red;  
}  
  
div{  
    color: blue;  
}
```

HTML :

```
<div class="loud">  
    What color is this?  
</div>
```

# 你也可以是稱職的調解委員

Style	ID	類別(Class), 虛擬類別, 屬性	標籤
(A)	(B)	(C)	(D)

.circle	h1	#bar1	?
li:active	ul#nav	td[colspan="2"]	?
[colspan="2"]	span	li:focus	?
div	ol	strong	?
#target	<li style="color: red;">	li:hover	?

# 你也可以是稱職的調解委員

Style	ID	類別(Class), 虛擬類別, 屬性	標籤
(A)	(B)	(C)	(D)

.circle	h1	#bar1	B
li:active	ul#nav	td[colspan="2"]	B
[colspan="2"]	span	li:focus	C
div	ol	strong	D
#target	<li style="color: red;">	li:hover	A