



Open in app

Get started



Published in UI/UX練功坊 · Follow



Sandy · Follow

Nov 22, 2018 · 4 min read



# CSS的優先順位

載入位置 & 撰寫順序 & 選擇器類型都會影響 css 的優先順位



在介紹 CSS 的優先順位之前，  
先講一下在 HTML 中套用 CSS 的方法，共有三種：

## ① 行內(inline)套用：

直接在 HTML 標籤中使用 style 屬性指定 CSS 樣式

```
<h1 style="color: #ff0000;">主標題</h1>
```

雖然這個方法最直覺，但通常我們在撰寫 CSS 時使用的樣式繁多，為了保持 HTML 的結構乾淨清楚，原則上不會採用這種方法。且 CSS 選擇器有許多進階的用法必須另



[Open in app](#)[Get started](#)

```
<head>
  <style>
    h1 { color: #ffffff; }
  </style>
</head>
```

這麼做雖然可以將 CSS 的內容與 HTML 原始碼分開，但寫在 head 元素中的 CSS 就只能在這一個網頁使用，一般是建議少用這種方式，除非是有特殊的狀況需要覆蓋某些樣式。

### ③ 外部載入CSS：

將 CSS 程式碼統一存成一個外部檔案(.css)，再到 HTML 的 head 元素中載入這個檔案。可使用 link 元素或是 @import 這兩種方法來載入，**通常使用 link 元素載入。**

```
<head>
  <style>
    @import url(外部css檔案的路徑);
  </style>
</head>
```

或是

```
<head>
  <link href="外部css檔案的路徑" rel="stylesheet" media="all">
</head>
```

在 HTML 中套用 CSS，最常見的就是這個方式。這個方法的優點是多個網頁可以共用同一個 CSS，這樣維護檔案相對比較容易。

一般來說，CSS 的管理可分割為 網站共通的 CSS、首頁專用的 CSS、下層子頁面另外使用的 CSS 等，再依狀況分別載入需要的 CSS 檔案。

**不過要注意，建立太多 CSS 檔案可能會造成顯示速度變差，所以若非網站的規模非常龐大或 CSS 的資料量相當多，大致上不會分割太多太細的 CSS 檔案，主要目的還是取出可共用的 CSS 檔案一起管理。**



[Open in app](#)[Get started](#)

那麼，這三個套用方法，CSS 讀取時的優先順位是怎麼排的呢？答案是以越直接的方式套用 CSS 優先權就越大：

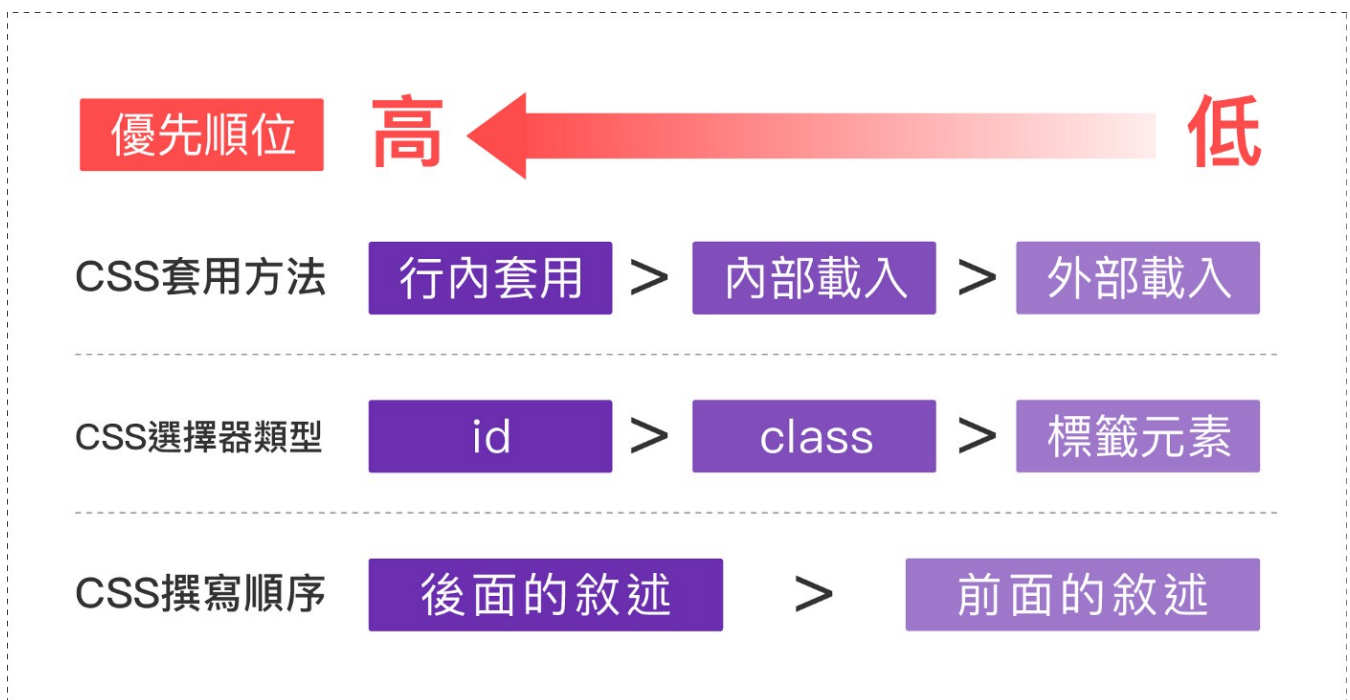
### 行內(inline)套用CSS > HTML內部載入CSS > 外部載入CSS

除了套用方法之外，還有 CSS 的撰寫順序以及選擇器的類型也會影響 CSS 的優先權：

CSS 的撰寫順序基本上是以「寫在後面的敘述，優先於寫在前面的敘述」為原則，只要後面衝突到同一個位置的值就會覆寫過去。

另外，CSS 選擇器類型的明確度(Specify)優先順位是：id選擇器 > class選擇器 > 標籤元素

總結以上，我們可以將剛剛說到的 CSS 優先順位整理為底下這張圖：



但在無法使用以這些順位修改覆蓋 CSS 時，有一種寫法可以直接忽略 CSS 的明確度，直接指定為最優先，就是在屬性後面加上「!important」

```
<head>
  <style>
    #main-title { color: red; }
    .title { color: blue !important; }
  </style>
</head>
```



[Open in app](#)[Get started](#)

標題會變成藍色的。

[查看範例 Demo →](#)

想要覆蓋選擇器後面加上的 **!important**，唯一的方法就是在行內套用的 CSS 中直接加上 **!important**，例如下列這個範例，標題就變成紫色的了

```
<head>
  <style>
    #main-title { color: red; }
    .title { color: blue !important; }
  </style>
</head>

<h1 id="main-title" class="title" style="color: purple !important;">
  主標題
</h1>
```

[查看範例 Demo2 →](#)

要注意的是 **!important** 是在沒有更好的辦法時才建議使用的，若胡亂使用，可能會破壞 CSS 原始的規則，導致要修改時找不到屬性撰寫的位置，CSS 看起來也會很雜亂，原則上應儘量避免使用才是比較優良的撰寫方式。

