* CSS (Cascadin Styling Sheet 階層式樣式表)

cascade是啥? 階層式，由上往下去做執行

下面的設定會蓋掉前面的設定

* CSS的選取器有很多種方式去找到HTML TAG:

Class selector、Group selector、Descendant selector、Attribute selector

* **Class selector (class選取器): 另可用 #id 或 tag name，但要注意**Specificity

h2*.bigfont*{ 【選CLASS都是使用dot notation而且沒有空格】

*/\*CSS會先去找所有h2元素*

*再從中找到cLass = "bigfont"的元素 \*/*

}

* **Group selector (群組選擇器)**

h1,h2,h3,h4{ 【Group selector 用逗號分隔】

    color: red;

}

* **Descendant selector (後裔(子元素)選擇器)**

div*.div1* h2{ 【Descendant selector用空格來分隔】

先使用class selector 找到class=”div1”的 div tag

再加空格 寫上子元素的tag name就可以找到該元素

color:red;

}

* **Attribute selector (屬性選擇器)**

input[*type*="text"] { 【Attribute selector: tagname[attribute] 都不用空格】

要選擇多個擁有相同attribute的HTML tag , 這樣很快

    color: red;

}

所有的input標籤中，type="text"的標籤都會被選擇起來改變

tag[ attr="text"]

* CSS Specificity 特定度
* Rule1. Cascading 下面的設定會蓋掉前面的設定
* Rule2. 優先度 id> class>tag 且無視Cascading
* Rule3. inheritance 繼承
  + CSS的屬性分為INHERIT PROPERTY <<父元素PARENT繼承>>
  + 以及NON-INHERIT PROPERTY <<標籤自帶>>

**CSS Specificity 特定度總結:**

**inline styling【直接寫在標籤內】>id>class>element selector>inheritance**

\*JS的.object也是一種inline styling Ex:

object.style = "background-color:lightblue; color:white;";

* Text styling 文字基礎設定
  + 文字的單位: (font-size)
* 瀏覽器預設的font-size是16px;【絕對單位，不建議使用!!!】
* 1rem是一個相對值 (相對於預設的16px)【相對單位，隨著網頁的縮放變化，建議使用】

(2rem = 16px\*2 =32px 依此類推)

\* 這邊的16px是指瀏覽器縮放比例為100%的狀況

使用者主動放大時基準的16px也會相對放大

因此通常不設定字大小為絕對單位，而是以rem來做開發

* + Text-align 文字位置

    p{

        text-align: center; //預設是left靠左；另可設定center、right等等

    }

* + Text-decoration 文字樣式

    p{

        text-decoration: line-through; 刪除線

        text-decoration:overline; 頂線

        text-decoration:none; 消去所有(預設)樣式 (幫anker tag 去除醜醜底線常用)

        text-decoration: underline; 底線

    }

\*有些tag element有自己預設的樣式 像是<a></a> anker tag

本身自帶醜醜底線跟紫色文字

(text-decoration: underline、cursor: pointor【小手】)

這時候就可以透過text-styling去做調整

    a{

        text-decoration:none; */\*去底線\*/*

        font-size: 3rem; */\*文字大小\*/*

        color:cornflowerblue; */\*文字顏色\*/*

    }

* + 另有line-height【行高】、letter-spacing【字母間距】、text-indent【段落縮排】、

font-family【設定字體】、等等

都可以找到直接複製貼上的字體

常用的可以找> <https://www.cssfontstack.com/>

https://fonts.google.com/

\* font-family: "Times New Roman" , Times , serif

後面逗號串接的,Times , serif 指的意思是要是前面的字體不同作業系統

無法正常顯示字體的話，會使用後面的 ,Times , serif 字體來當作備案。

* Background 背景
  + background-color: (250,250,0) 背景色

color: red 文字顏色

background-image: url("路徑") 背景圖片

* + background-size: contain 把圖片放到最大，如果填不滿頁面，他會repeat

【通常會搭配: background-repeat: no-repeat來使用】

* + background-size: cover將圖面延展至頁面填滿

【通常會搭配: background-position: center來使用】

* + background-size: 也可以設定30%、或是200px 200px
  + \*background: 可以代表background-image:或者是代表 background-color:

\*其他background的設定可以參照下方

<https://developer.mozilla.org/zh-CN/docs/Web/CSS/background>

\*還有炫砲的background: liner-gradient漸層設定可參照下方

https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\_Images/Using\_CSS\_gradients

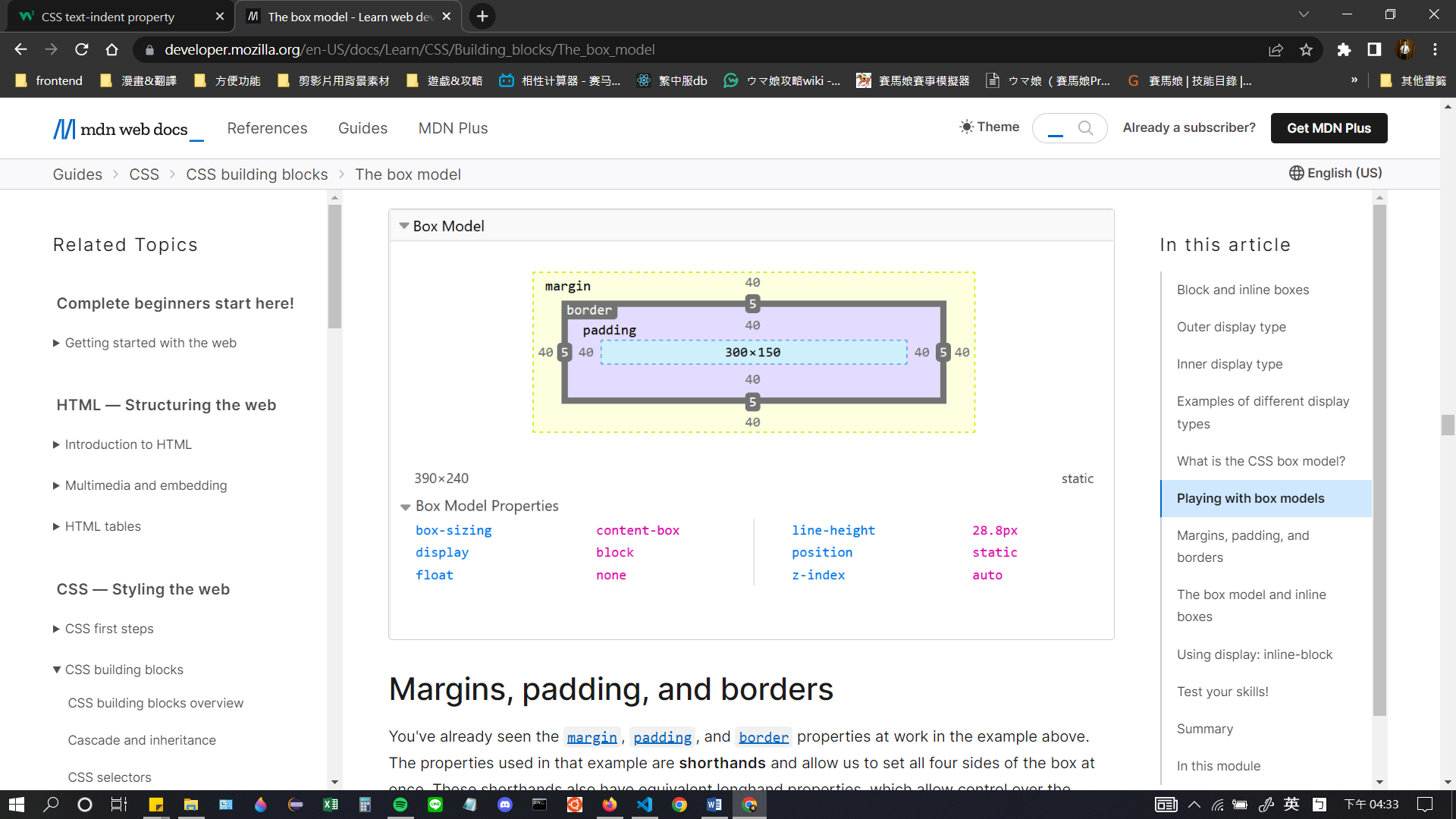
* Box model
  + ALL HTML elements are boxes and with 6 attr

[width, height, content, padding, border, margin]

* + Padding&Margin

border內的叫做padding、border外的叫做margin

整個element寬度都包含margin



* + padding: 2px 10px 6px 3px ; 【4個值狀態】

(上 右 下 左 順時針設定 )

* + padding: 2px 10px ; 【2個值狀態】

(上下 左右)

* + margin也是一樣設定方式
  + Padding&Margin

如果發現anker tag上margin卻無效是怎麼回事呢??

因為在w3c recommendation中有提到margin適用所有元素

但是是針對【垂直的margin屬性設定】在inline elements上面不會有任何的效果!!!

(inline element 不能設定長寬!!!，而<a>又剛好是inline element)

* Z-index 堆疊屬性

如果想要讓某個有重疊的element要做上下排序，可以分別設定z-index

值越小的越下層；反之值越大的越上層!

\*要讓z-index啟動它的效果，要重疊的物件<div>兩者

都必須是position: static之外才可以做堆疊!!!!!

https://developer.mozilla.org/en-US/docs/Web/CSS/z-index

* position: position你不設定的話會是static

>>static: non stacking context 不可堆疊(不可z-index) [其他的都可以]

>>relative: Stacking context 可堆疊(可用z-index)

>>absolute: Stacking context 以最接近的父元素作為設定基準

>>fixed: 固定於網站某的地方

>>sticky: 網站下拉某個div黏住

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

* List tag CSS設定
* ul部分:

list-style-type: disc (黑點)、circle(空心圓)、square(正方)、...還有很多

* ol部分:

list-style-type: upper-roman(羅馬字)、lower-alpha(小寫英文abcd)、upper-alpha(大寫英文ABCD)

}

* table tag styling {

border-collapse : table default border雙線改為單線

}

* cursor 滑鼠游標樣式{

放在img中可以改變游標樣式:

https://developer.mozilla.org/zh-TW/docs/Web/CSS/cursor

}

* Transition 轉場設定

3個屬性需要設定，最簡的轉場設定可以用以下的三個property達成

property name ex: background-color

duration ex: 2s

timing function ex: ease-in-out;

    div{

        transition:all 2s ease;

    }

其他詳細設定可參考這:

https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\_Transitions/Using\_CSS\_transitions

* 2D Transform 轉變

如果想要讓box移動，必須將box的父元素設position: relative;

本身設position: absolute;

盡量搭配transition使用，否則會閃現很奇怪

translate() 移動(換位子)

ex: transform:translate(100px,300px)

以物件左上方那個點為基準，x軸正向移動100px; y軸負向移動300px

ex:

transform:translate(-50%,-50%);

用%來表示就不一樣!! 這邊是將box本體以重心為基準移，

x軸負向移動50%本體長; y軸正向移動50%本體長;

transform: translateY(50%);

物件較原設定位置在下移50%(依據本體位置下移)

* 注意!! 後面設定的transform會把前面的transform給覆蓋掉喔!!

你想要同個物件有多個效果的話，可以用空格分開即可

ex: transform:rotate(90deg) scale(2);

* rotate()

ex: transform:rotate(90deg); 順時鐘回90度

* scale()

scale值 1 為一般大小

2.5>> 2.5倍 0.5>>0.5倍

ex: transform:scale(2.5)

scale也可以個別設定: (寬,高)

ex: transform:scale(2,3) 寬2倍 高3倍

* 注意: 如果想要讓物件放大為何不用絕對單位設定放大?

ans: 如果用絕對單位設定的話，放大的基準點不同!!

你用絕對單位設定的話，會以左上角為基準出發去放大

但是如果是以scale去設定的話，將會以物件重心為基準出發去放大!!

* 3D transform:

rotateX() x軸為方向旋轉

rotateY() y軸為方向旋轉

rotateZ() z軸為方向旋轉，通常我們想看到的平面旋轉都是這種!

* Flexbox 重要排版觀念

CSS網頁排版用工具，更快開發網頁

<div> tag，default setting its display:block;

div通通都不設定的時候，會自動設定display為block

【只要在整個container tag(div,section,header…)設定display=flex;

此時所有包在container tag中的物件都會變成flex items(範圍只有一層)】

Ex: 爸爸(父元素)是display:flex，那他的兒子(子元素)自動是flex items，但孫子不是!!

通常我們都是針對flex items來做設定，所以要清楚知道你要設定的tag是哪一層

再去把那個tag的爸爸抓出來display:flex一下

* + flex-direction: 可以設定所有flex items的排列方式(預設為row)

      div{

          flex-direction: row; */\*以 ”一列” 排列\*/*

          flex-direction: row-reverse; */\*以 ”一列” 排列，並且反序排列\*/*

          flex-direction: column; */\*以 “一行” 排列(但原本block就是以行來排列，沒實質意義)\*/*

          flex-direction: row-reverse;  */\*以 “一行” 排列，並且反序排列\*/*

      }

* + flex-items 寬度設定: 1. 讓content決定寬度 2. 自己設定最大寬度 3. flex-wrap

>不設定寬度 --> 物件寬度就由flex-items 的content寬度而定

>單位設定寬度 --> max-width or flex-wrap

flexitems可以隨著視窗大小改變寬度

如果對flexitems設定寬度，設定的寬度就是該物件的 **"最大寬度!!"**

那反過來要設定flexitems **”最小寬度”** 怎麼設定?

使用flex-basis即可(可固定那個flex item最小多寬，之後視窗再小都不會再變化)

* + flexitem高度設定:

>不設定高度 --> 物件高度就由flex-items 的content高度而定

>單位設定高度 --> 這個高度即為固定的高度!

* + RWD (Responsive Web Design 自適應網頁設計)
    - Flexitems 預設以一列做排列，items寬度將隨視窗大小變化，視窗要是比items預設的最小寬度還小時，那個items越擠越小道看不見，這不會是我們想要的!
    - 可以設定flex-basis:; 設定items的最小寬度
    - 再設定flex-wrap:wrap; 讓視窗小於items寬度時，items會自動換列往下
  + flex-wrap預設為(nowrap不換行!!)【這個wrap設定必須設在container上才生效】

ex: flex-wrap: wrap; 這麼一來不管你的視窗大小多少，那個物件都會是固定的寬度

【視窗裝不下的時候，物件會做自動換列】

* + flex-grow 縮放倍率

設定items寬度的成長倍率(隨著視窗縮放變化)，與flex-basis搭配使用!

* Flexbox 重要排版觀念

>使用flex之前的規劃:

1. flex container:

不用設定長寬度(因為都是由flexitem決定的)但是你要設定display: flex 以及flex-wrap: wrap;

2. flex item:

寬度: 設定單位寬度 以及flex-grow

高度: 設定單位高度即可

* + Flexitems position setting : (justify-content、align-items)
* justify-content : 控制與flex-direction方向(可能是column或row)平行的flexitems

      div{

          justify-content: flex-start; */\*靠左開始排列 【預設是flex-start】\*/*

          justify-content: flex-end; */\*靠右開始排\*/*

          justify-content: space-between;*/\*空白處均分，並且將物件等間距排列\*/*

*/\* (但左右物件貼齊container邊界)\*/*

          justify-content: space-around; */\*讓物件兩側都有固定間距\*/*

*/\*(左右物件固定保持相同間距)\*/*

          justify-content: space-evenly; */\*物件外的空白處均分，固定每個物件的間距\*/*

*/\*(跟apace-around蠻像，但不一樣，看實務需求)\*/*

}

* align-items: 控制與flex-direction方向(可能是column或row)垂直的flexitems

      div{

        align-items: flex-start; */\*靠上開始排列 【預設是flex-start】\*/*

        align-items: flex-end; */\*靠下開始排列\*/*

        align-items: center; ; */\*置中排列\*/*

*/\*【注意:  如果是center設定，他的flex item高度如果沒有指定，高度就是內容高度】\*/*

        align-items: stretch; */\*拉長*

*【預設是stretch, 如果你對flex item沒有設定height 他會自動拉成container的高度】\*/*

      }

flex:如果忘記可以來這邊看看

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

* Flexbox 重要排版觀念

flex-shrink、flex-grow、flex-basis 三兄弟(分別控制flex-items-縮小/放大比率、寬度最小值)

      div{

        flex-grow: 1; items放大比例

        flex-shrink: 1;

        flex-basis: 200px; 固定items最小就是200px寬，不會更小

      }

上面那三行設定的東西可以簡寫在同一行裡

      div{

        flex: 1 1 200px; /\* flex: (grow) (shrink) (basis);  \*/

      }

* + Flex-grow算法:

將每個flexitems的數字都加在一起，再去平均分配

Ex: 下面的box1、box2 flex-grow和為3 >>代表將整個container的寬度分成3等份

Box1設定flex-grow: 1，他就佔有1/3的container寬度

Box2設定flex-grow: 2，他就佔有2/3的container寬度

*#content* {

          display: flex;

          justify-content: space-around;

          flex-flow: row wrap;

          align-items: stretch;

        }

*.box* {

          flex-grow: 1;

          border: 3px solid rgba(0, 0, 0, 0.2);

        }

*.box1* {

          flex-grow: 2;

          border: 3px solid rgba(0, 0, 0, 0.2);

        }

* Ａｎｉｍａｔｉｏｎ：

記得要做動畫的物件要將position設為relative

keyframes 設定動畫執行內容

*@keyframes* anime-name {

        from {

          background-color: aquamarine;

          top: 0;

          left: 0;

        }

        to {

          background-color: gold;

          top: 300px;

          left: 200px;

        }

      }

animation-name: {keyframe名稱} 輸入要執行的動畫名稱

animation-duration : 動畫播放時間

animation-delay: 重整後等待幾秒後執行

animation-iteration-count: 動畫迴圈次數 數字、infinite(無限次

animation-direction: 動畫播放方式

>>reverse ; 倒著播放

>>alternative:從頭播到尾，再從尾巴播回來。alternative-reverse反之亦然

animation-timing-function:; 加入轉場效果

>>ease-in

>>ease-out

animation-fill-mode:

forwards; 動畫結束時會停在最後部分

      div*.box2* {

        width: 100px;

        height: 100px;

        background-color: aquamarine;

        transition: all 1s ease-in;

        animation-name: anime-name;

        animation-duration: 2s;

        animation-iteration-count: 1;

        animation-timing-function: ease-in;

        position: relative;

        animation-fill-mode: forwards;

      }

* 相關CSS設定技巧
* anker tag裡面 讓href="#"的用意在於，回到同一頁面時不用再次重新整理，適用同一頁面

可以去w3school查到這些基礎元素的固有屬性

* 有關你的div寬度高度無法讓背景貼齊整個網頁頁面?

主要是因為body tag 自己預設有個margin:8px的設定!!!

\*其他HTML tag預設屬性可以去下面查

https://www.w3schools.com/csSref/css\_default\_values.asp

【網頁設計起初會優先用\*把所有的margin和padding通通削掉還有設定box-sizing】

* box-sizing有兩種: content-box(預設)和border-box

\*content-box : 你設定元素寬度為100%，再設定padding的時候，會在你的元素外再新增一個 寬度-->系統會產生horizontal scrollbar，不是我們想要的。

\*border-box: 所以通常會採用border-box，這樣一來你設定元素寬度100%，再設定他的padding的時候，它會讓你設定的元素寬度固定(不再額外生成一個寬度出來)

【keep the object same width】

* + 小結: 通常開發網頁換先針對每個tag element 先做以下CSS設定:

\*{

padding: 0;

margin: 0;

box-sizing: border-box;

}

* 程式碼:

<nav>

<ul>

<li><a href="#">首頁</a></li>

<li><a href="./1.html">農業改革</a></li>

<li><a href="./2.html">工業改革</a></li>

<li><a href="./3.html">經濟起飛</a></li>

<li><a href="./4.html">關於作者</a></li>

</ul>

</nav>

* 你如果要修改上面HTML的超連結文字顏色，你直接對nav做CSS設定是沒有用的，

因為inherit繼承部分的效力，<a>tag的default設定是大於父元素給的繼承設定的

【CSS Specificity :default styling > inherit style】

你要對default styling的 tag去做更改樣式CSS，直接對這個tag本身做CSS設定才有效!!

* 承上，你如果想讓li這個tag左到右排列，我們會想採用display: flex，但是如果你設定的是在nav上面做flex :

ex:

nav{

display:flex;

flex-direction:row;

}

這樣只會讓"nav下面第一層的子元素"變成"flex-item" 而這邊我們想要排列的東西

是li 不是ul所以我們要針對ul 來做display: flex，這麼一來

他下面的所有li 才會化為flex item，達成我們要的排列

* navbar有兩種做法，

一種是游標移到a tag文字上才可以進行超連結的

一種是只要移到a tag文字含padding範圍就可以進行超連結

>前者只要在li進行padding設定，並且把anker tag設定好即可 >> 文字部分超連結

>後者的話，你想讓文字padding範圍也屬於anker tag作動範圍 :

你要針對li 再進行設定display:flex 再把原本設定在li的padding

移到在anker tag 去做padding，這樣一來anker tag本身也會變成flex item

他的作動範圍就會增加!!

* 有關taiwan的icon 可以去找: G: TAIWAN FREE ICON

https://www.twicon.page/

內容排版:

可以想像成有一個大型的flex container

再把內容分成兩個子元素

navbar 位置顯示

讓使用者知道現在在哪裡!!

User Snippet

>snippet 片段

>>像是vscode裡面打驚嘆號會自動生成一個html框架一樣

>>user snippet就像是你已經預設好的腳本

>>要客製自己的snippet 去google: Snippet Generator

>>【https://snippet-generator.app/】

>>使用User snippet generator 你要先打好一段你想讓vscode記憶的code(html)

>>之後把整個html tag code通通複製 貼到左側空白處(Your Snippet)

>>記得左上角的Description 要打你對這個code的註解

>>右上角的Tab trigger指的是你要讓vscode記憶的關鍵字是啥 你打了他才會問你

>>都完成之後點選右下的copy snippet

>>回到VScode 左下角齒輪 > 設定使用者程式碼片段 >上方跳出的選單中搜尋HTML

>>點選後會跳出一個html.json檔，把你做好的snippet整個貼上在一個括弧內

>>有複數個snippet就用逗號分隔即可，儲存後就可以回html裡面試打看看你剛剛設定的snippet trigger是否作動!

>>trigger 關鍵字

HTML Bookmark

>同一個頁面中可以建立不同的書籤連結

>也可以實現一個按鈕ｆｉｘｅｄ點選之後回到目錄!

<base> tag

一個html中只能有一個base標籤

HTML <base> 標籤主要是控制網頁的連結屬性，可以指定網頁頁面的文檔與連結根目錄，也可以指定網頁超連結的開啟方式，HTML base 屬性有 href , target。

你在head中的base標籤設定 href的話，所有的超連結href屬性都會被導引到base設定的這個來

另外base還可以設定屬性target="\_self"; 讓超連結在跳轉時在同一頁

base還可以設定屬性target="\_blank"; 的話，超連結會在另一個分頁跳轉

ex:

<base target="\_self">

scroll-behavior可以設定HTML Bookmark連結在同一個頁面跳轉時，會友滾動的特效

但是注意，這個效果要寫在html的css裡面

另外chrome瀏覽器的smooth-scrolling 實驗性功能，你要複製以下網址

chrome://flags/#smooth-scrolling

進去之後把smooth-scrolling 的選項點選enabled (default不行)

字體設定方式: css內建文字、google fonts、local font

Local Font

ex: 找到自己要的字體.tff檔 透過@font-face去做 local-font

\*{

padding: 0;

margin:0;

box-sizing: border-box;

/\* font-family: 'Noto Sans TC', sans-serif; \*/

font-family: "my-font",sans-serif;

}

@font-face {

/\* 確認可商用就可以這麼做 \*/

font-family: "my-font";

src:url(./清松手寫體1.ttf);

}

Object-fit :cover

在css中設定

.pic {

width:200px;

height:200px;

object-fit:cover; <--可以讓你的圖片只顯示你設定的範圍，且不會做壓縮

}

https://www.yesharris.com/crawl-and-index/

https://www.yesharris.com/meta-robots-and-robots-txt/

https://www.yesharris.com/page-speed-seo/

<meta name="robots" content="index, follow" />

meta robots以及robots.txt的工作分別是阻止Google 檢索、索引你的頁面。在這個年頭，只是SEO排名/流量高是沒有用的，流量必須對企業有價值、必須要能夠創造轉換，所以使用者體驗相對重要，如果你有特定頁面會傷害使用者體驗（UX），你可以透過這些方式去阻止該頁面出現在Google 搜尋結果中。

robots.txt可以阻止搜尋引擎檢索你的資料，如果你使用了robots.txt來阻擋搜尋引擎，那麼搜尋引擎將會略過你所阻擋的頁面，不去做檢索。

但meta robots 就不同了，他在索引層面阻止搜尋引擎索引你的頁面，但Google 還是有爬你的網站資料的，但究竟為什麼我們要這樣做？後面我將娓娓道來。

盡可能避免在導覽列、麵包屑、網站側欄、商品/文章列表這些重要的地方使用JavaScript以及AJAX會比較保險（在這篇文章我先列出幾個大方向的常見重點，未來我會在看大家回饋狀況各自拉出來寫成獨立的文章，JavaScript與AJAX這個議題有太多層面要探討，但簡單總結的話就是不要太過度使用AJAX）。

通常我會建議你找到一個＂解析度不會失真的最低限度＂為多少，並以此為標準即可，以我的文章來說，我也是從400 x 400、 500 x 500、 600 x 600來逐一嘗試，最後我選擇圖片盡可能最大不要超過600，寬度600在我的網頁上是清晰且容量不會太大的大小。你可以透過 Photoshop調整圖片的品質、解析度、大小，也可以使用Google官方的免費圖片壓縮工具『Squoosh』來壓縮自己的圖片。

https://pjchender.blogspot.com/2015/04/css-3vh-vw.html

vw view width = 視窗寬度

100vw =100%視窗寬度 >>>>6vw= 6%視窗寬度

https://stackoverflow.com/questions/37143046/flexbox-layout-keeps-overlapping-footer-in-mobile-view

當你發現有些flex-items元素在flex-wrap: wrap的時候，會蓋掉你下面的元素，很有可能是你有針對flexitems有做fixed height的設定!!!!!

千萬不要對flex items做height設定，會產生上出問題!!!!!!

如果希望超連結是放在圖片上的，只要用 a 標籤把 img 標籤包起來即可。

ex:

<a href="url"><img src="pic-url"></a>

box-shadow範例

<https://getcssscan.com/css-box-shadow-examples>

使用FIGMA技巧

G: figma download >> windows>> 登入

新增New Design File >> Frame >> 拉出一個想要的大小(大小不對沒差，右側可調width height

)

開始新增>> 如果有很多個文字方塊與長方形圖形想做group時>>可以透過ctrl+alt+點選

>>選完之後 ctrl+g 完成群組 【注意: group中有很多個物件，由上到下代表堆疊的層數，月上面代表越上層，如果發現東西不見，很有可能是被蓋掉!】

ctrl +c ctrl+v 都可以使用，但是貼上的時候他會在原地貼上，點選移動直接左鍵點選拖拉即可

RWD 響應式網頁設計

可以參考HTML UP5的範本去做設計

RWD原則:

1. 能用flexbox處理就用flexbox處理

2. 不能用flexbox處理，就用mediaquery處理

@media screen and (max-width: 800px) {

background-color: orange;

}

只要寬度比800px小(滿足max width)的時候，改變顏色!!

Bootstrap 網頁設計的library

Accessibility 可接觸性的 火狐F12後右上角有 (中文叫做輔助功能)

給有障礙人士的人物使用

可透過這個工具檢查你的網站是否有符合大眾使用 EX: 配色 鍵盤等等等

CONTRAST 對比: 對比色 檢查是否可以給色盲使用

KEYBOARD 鍵盤: 如果你沒有滑鼠 讀取後可用TAB遍歷你的超連結 原則你有照HTML去找他就會有

TEXT LABELS 文字標籤:

所謂的文字標籤指的是圖片中 要是link壞掉，你要顯示一個文字出來

也就是alt屬性的功用!! 他會幫忙檢查你的alt有沒有設定 建議要設定!!

<img ="./..../..." alt="picture">

另外建議可以在img 的屬性新增title 可以讓使用者在滑鼠mouseover 幾秒之後會跳出你打的註解!!

<img ="./..../..." alt="picture" title="car picture">

網頁使用優化:

1. nav bar 吸頂:

position: sticky; <----跟 top是搭配的要一起使用

top:0; <---與上面搭配

z-index:5; <----避免輩網頁其他元素蓋掉

background-color: white; <------ navbar 範圍顏色

box-shadow:3px 3px 5px 1px rgb(71, 71, 71); <--------------- 讓navbar跟網頁內容有所區分可以用box-shadow這個效果來呈現

可以到下面的網站搜尋box-shadow

他就會有別人寫好的樣式範例給你看

這邊也還有很其他的css設定 可以去看看

https://css-tricks.com/

2. scrollbar 可以客製化側邊的拖曳列!!

注意: firefox、edge有版本看不到，可以去w3school那邊去看看

https://www.w3schools.com/howto/howto\_css\_custom\_scrollbar.asp

ex:

/\* width 側拉寬度\*/

::-webkit-scrollbar {

width: 10px;

}

/\* Track 側拉底色\*/

::-webkit-scrollbar-track {

background: #f1f1f1;

}

/\* Handle 側拉拉條顏色\*/

::-webkit-scrollbar-thumb {

background: #888;

}

/\* Handle on hover 側拉拉條hover的顏色\*/

::-webkit-scrollbar-thumb:hover {

background: #555;

}

3. %的設定方法

寬度可自由設定

但是高度要用%去設定不行，因為原本height:的 default數值為auto

當你要設定的物件，他的父元素是沒有固定高度的時候，%會起不了作用

唯一例外是html這個root元素，他可以設定%作為高度

所以如果你想要用%設定高度可以用以下設定 【html, body必須同時設定height:100%】

html{

height: 100%;

}

body{

height:100%;

}

div{

height: 100%;

}

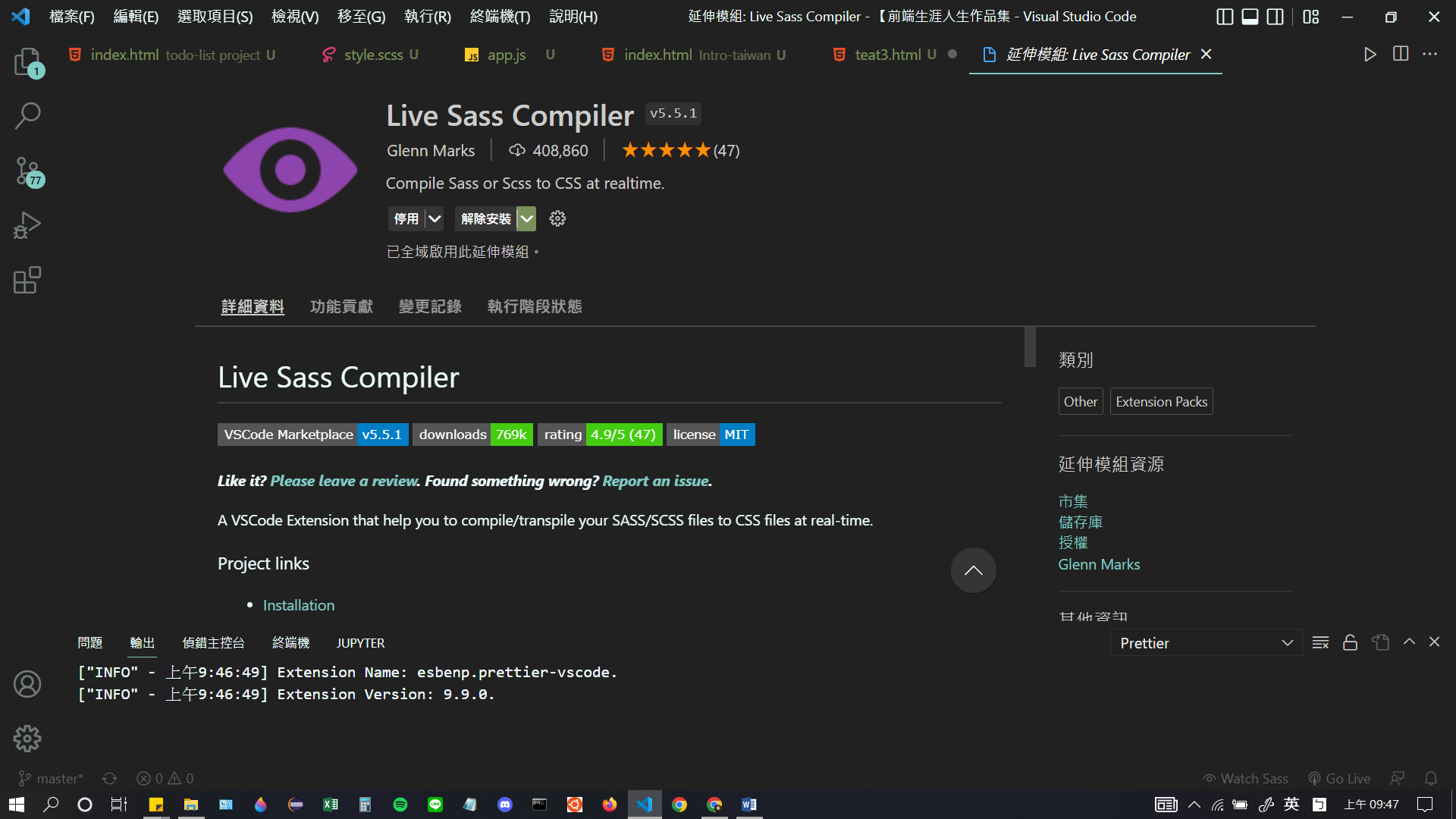
* SASS CSS Extension : 提升穩定性，寫更少的CSS code

【往後的CSS code都在SASS裡寫就好】

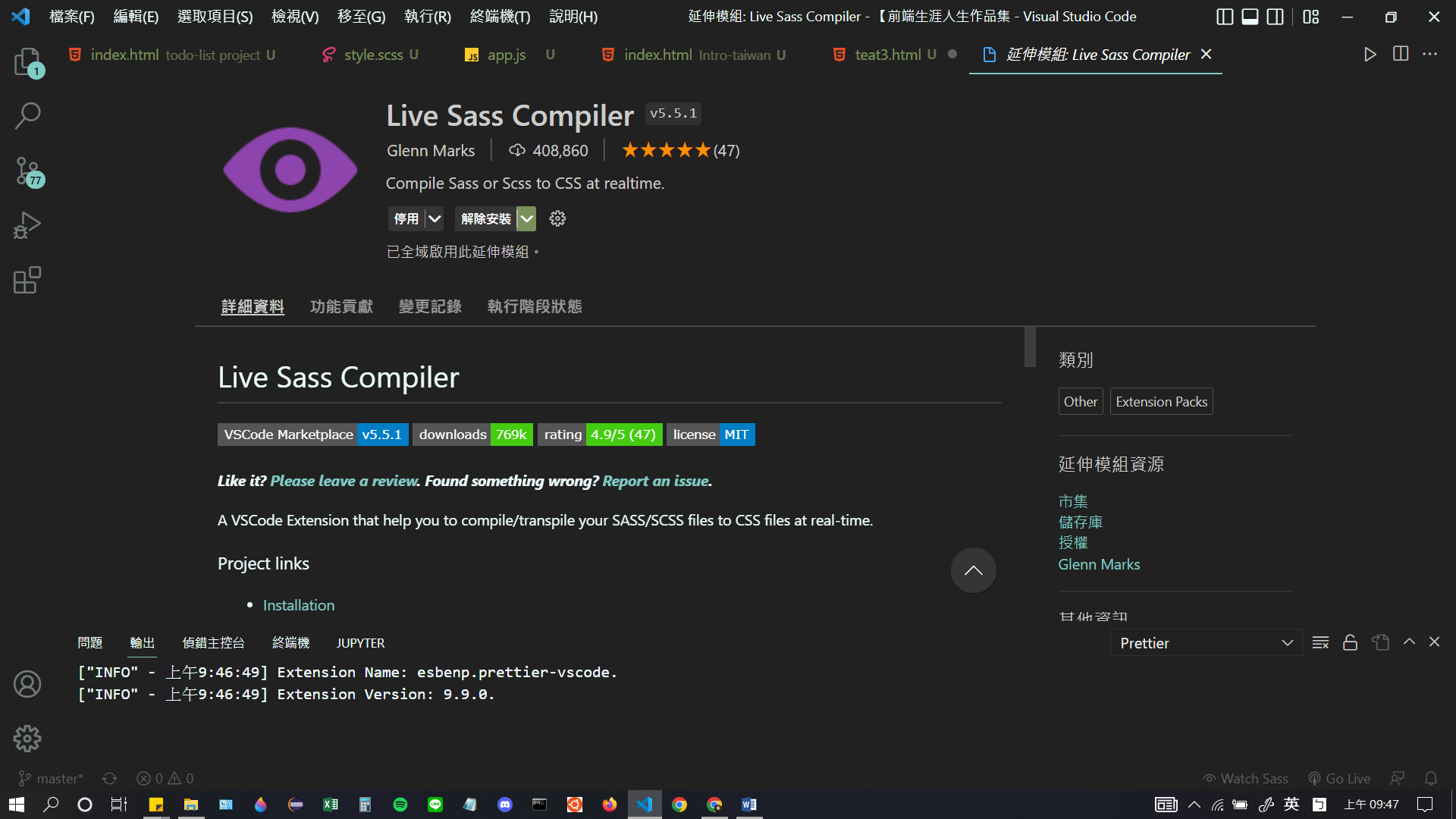
透過SASS生成的CSS會被編譯成另一個CSS，使網站在每種瀏覽器能夠正常的瀏覽

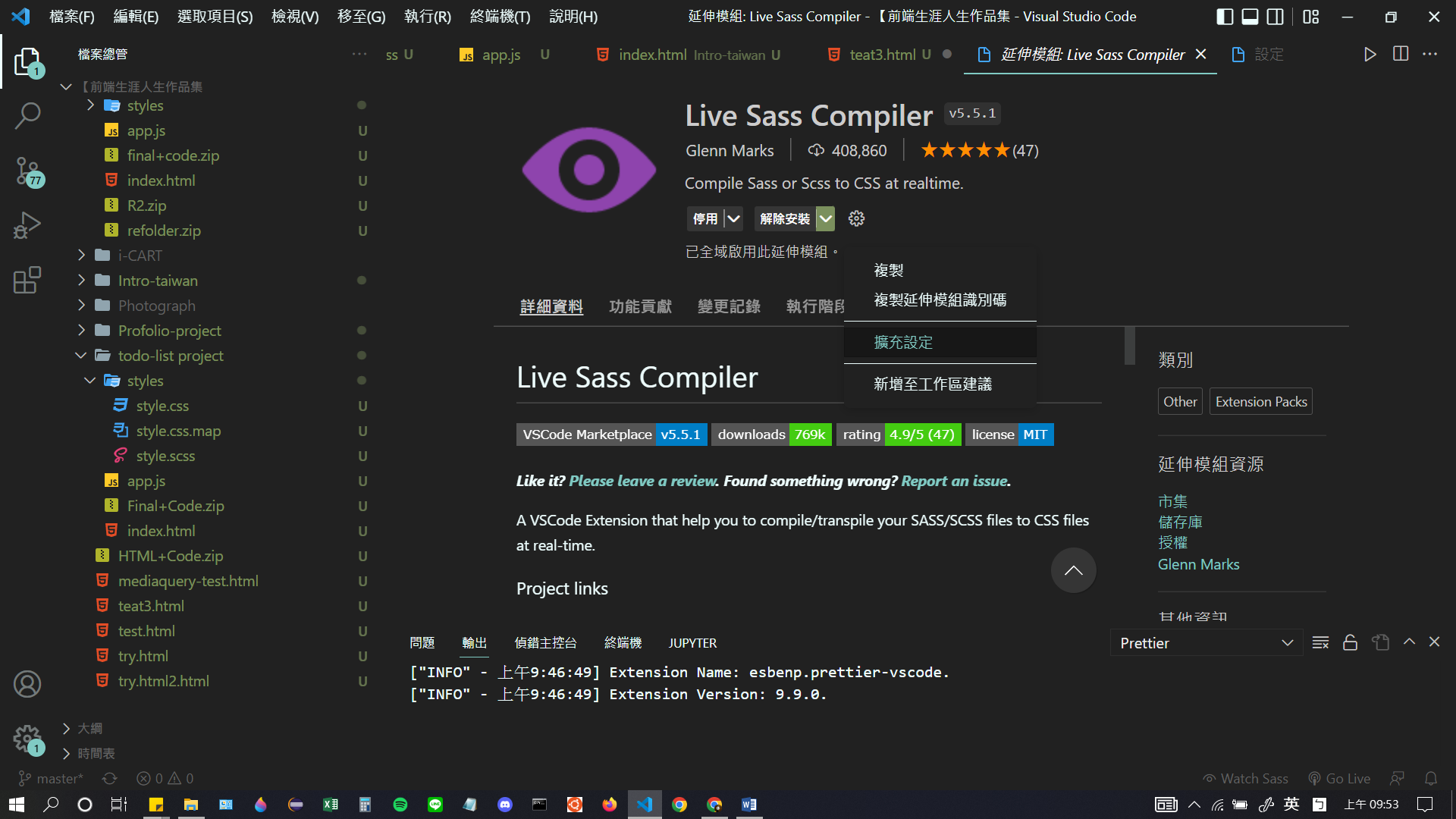
* SASS環境設定:

1. VScode下載延伸模組 Live Sass Compiler

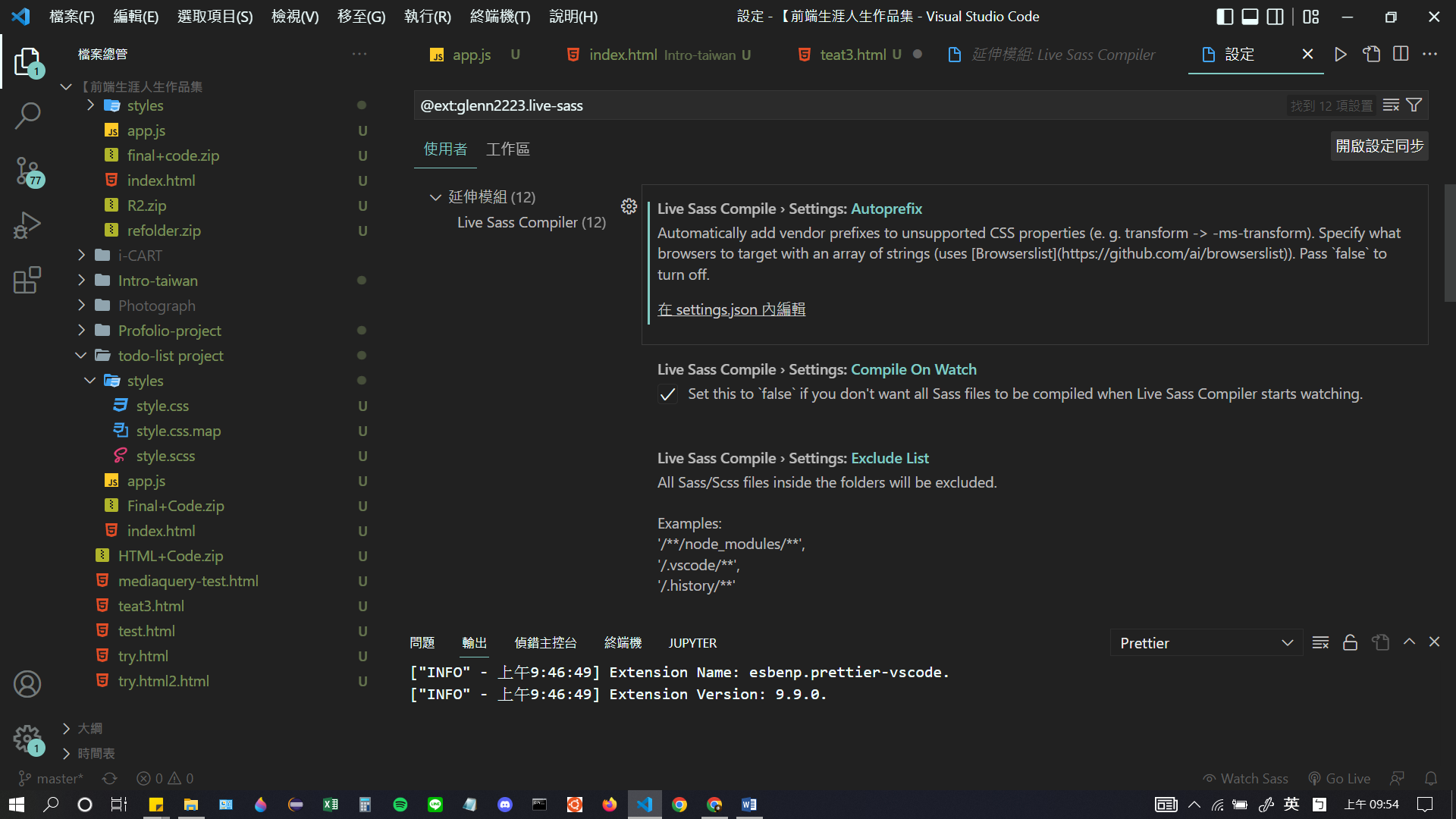


1. 基礎設定 : 延伸模組 點選小齒輪> 擴充設定 >找到第一個Live Sass Compile › Settings: Autoprefix





1. 點選下方的 在settings.json內編輯



* + SASS CSS Extension : 提升穩定性，寫更少的CSS code

4.在json檔下找到autoprefix裡加入 "> 1%", "last 10 versions"

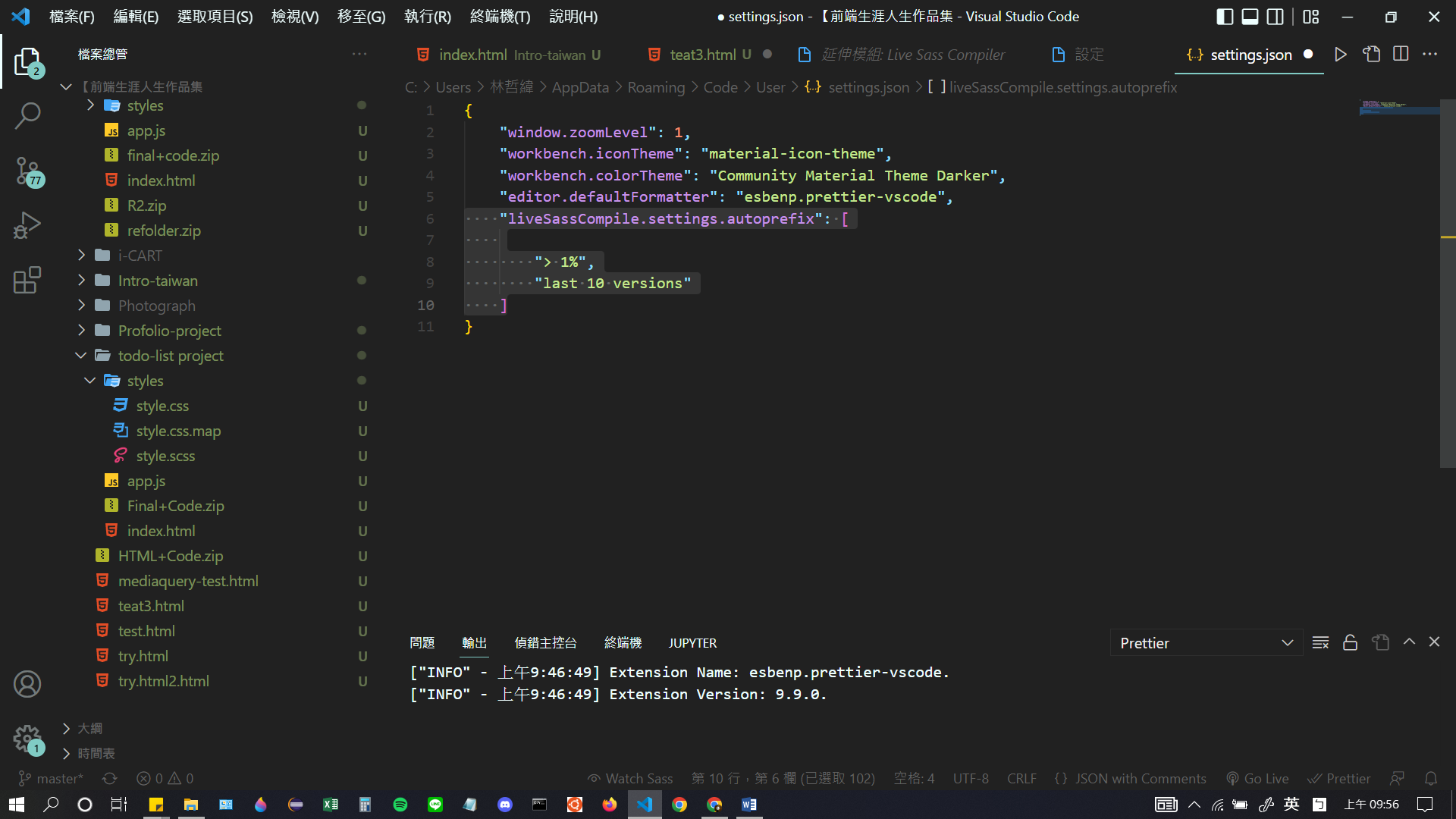
"liveSassCompile.settings.autoprefix": [

"> 1%",

"last 10 versions"

]

完成設定

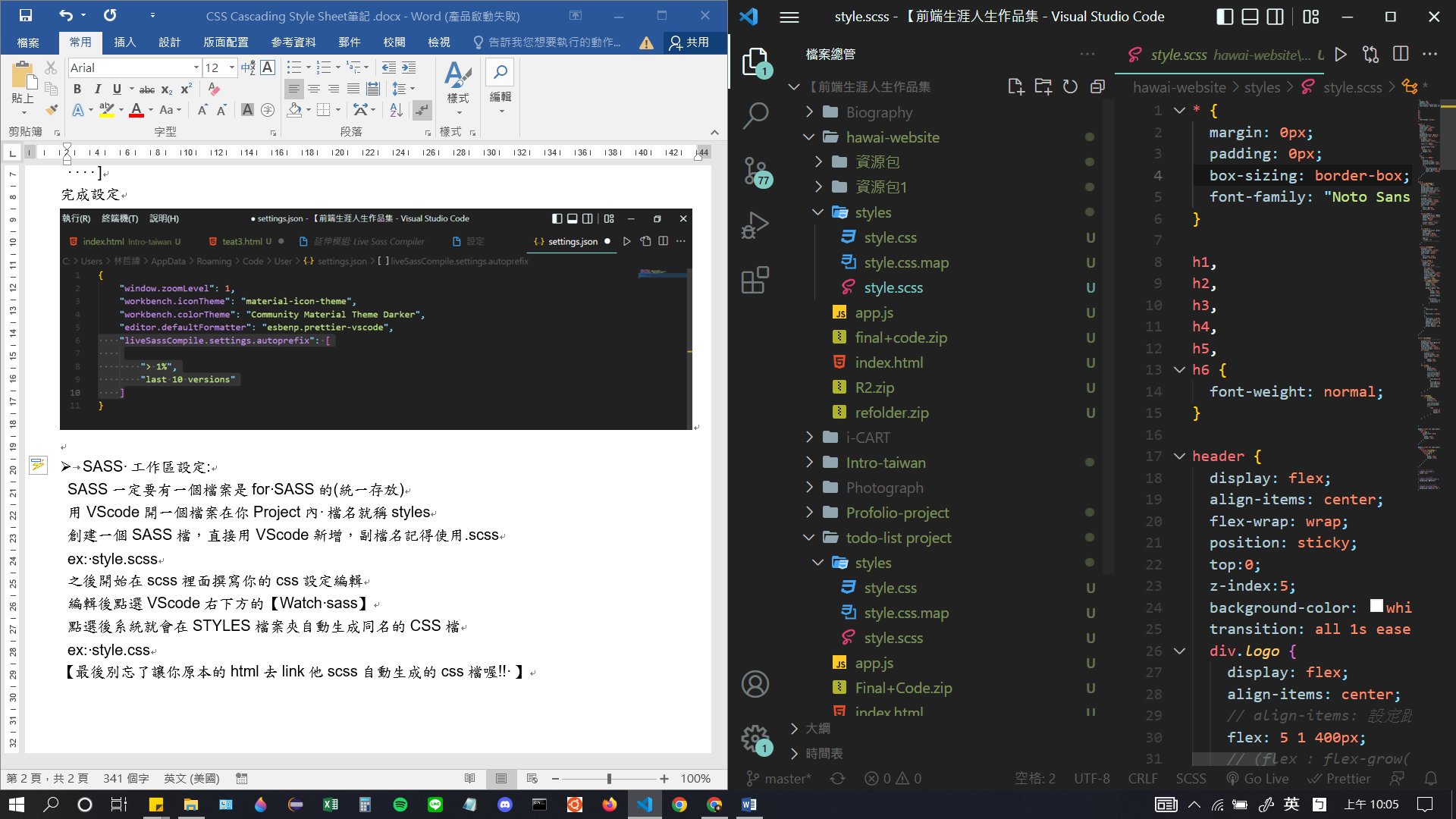


* SASS 工作區設定:

SASS一定要有一個檔案是for SASS的(統一存放)

用VScode開一個檔案在你Project內 檔名就稱styles

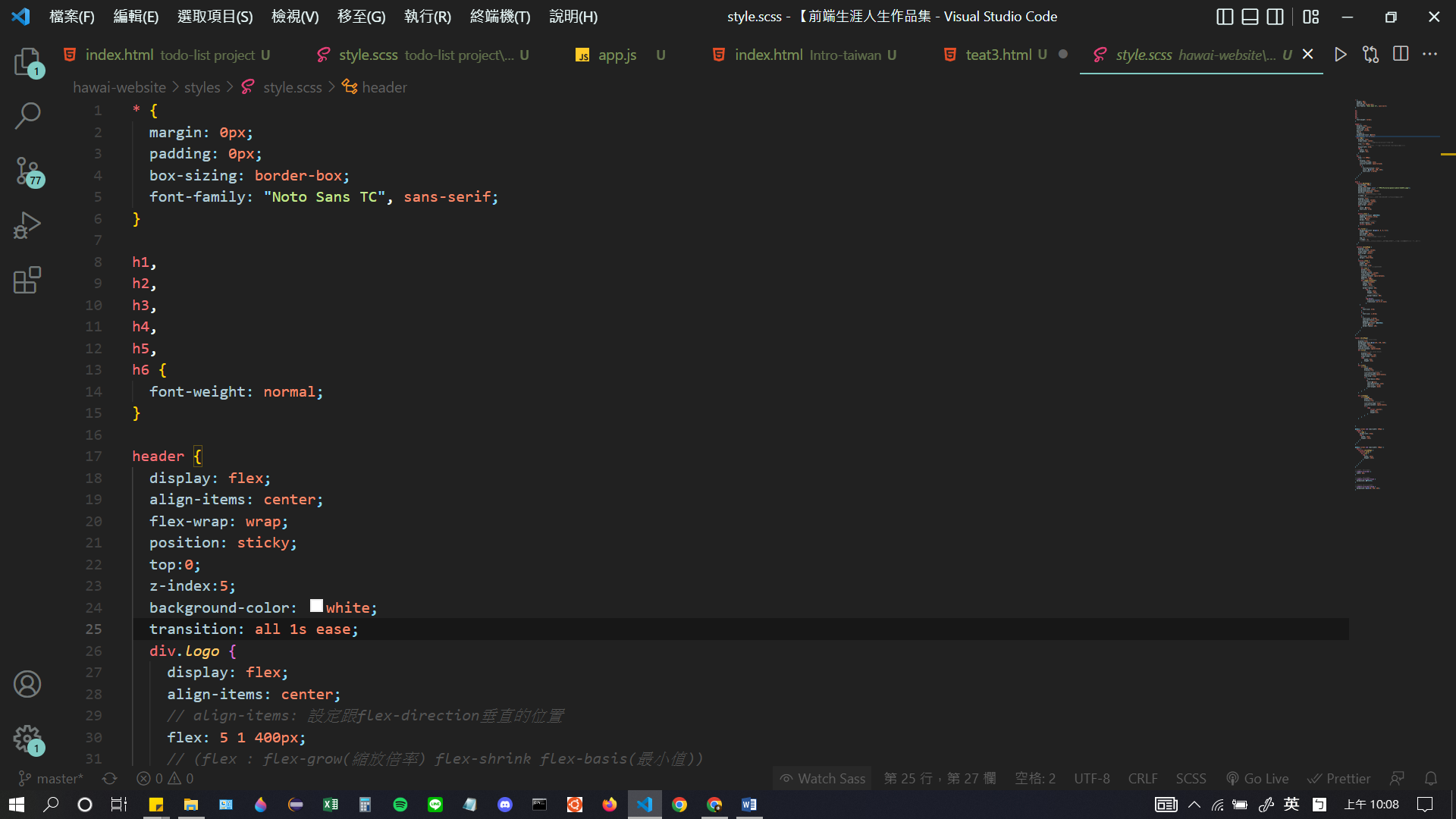
創建一個SCSS檔，直接用VScode新增，副檔名記得使用.scss



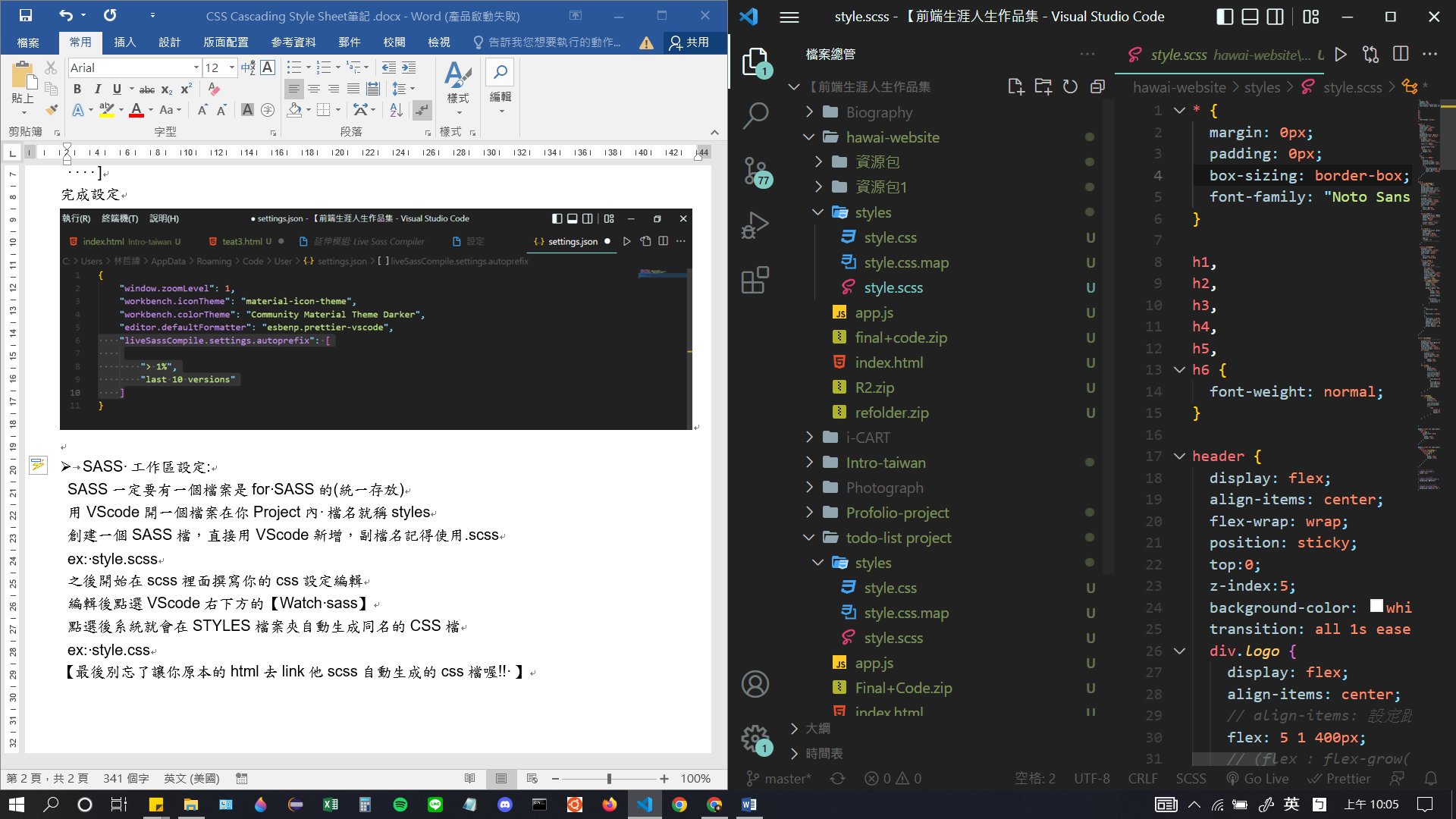
* SASS CSS Extension : 提升穩定性，寫更少的CSS code
* SASS 工作區設定:

之後直接開始在.scss裡面撰寫你的CSS code(寫法完全相同)

CSS code編輯完成後點選VScode右下方的【Watch sass】



點選後系統就會在STYLES檔案夾自動生成同名的CSS檔



ex: style.css

【最後別忘了讓你原本HTML去link SASS自動生成的css檔喔!! 】

    <link *rel*="stylesheet" *href*="./styles/style.css">

* SASS CSS Extension : 提升穩定性，寫更少的CSS code
* nested CSS: 巢狀式CSS寫法!

取代header h1 {} 這樣的寫法，也可以單獨針對header做設定，不用再寫兩次

ex: 在SCSS裡面 code更直觀好懂

header {

background-color: antiquewhite;

h1 {

color:chocolate;

}

}

* HTML對應nested CSS

HTML端:

    <header>

      <div *class*="logo">

          <img *src*="./資源包/Icons/black logo.svg" *alt*="logo">

          <h1>台灣旅遊租借服務</h1>

      </div>

      <nav>

          <ul>

              <li><a *href*="#">首頁</a></li>

              <li><a *href*="#">關於我們</a></li>

              <li><a *href*="#">會員登入</a></li>

              <li><a *href*="#">聯絡我們</a></li>

          </ul>

      </nav>

    </header>

SCSS端:

header {

  display:flex;

  div*.logo*{

      display:flex;

  }

  nav{

  }

}

* SASS CSS Extension : 提升穩定性，寫更少的CSS code
* variables: 遇到多個相同的顏色(主題色)，或者是多個相同屬性需要設定

就可以先設定一個variables 類似變數宣告，後續做維護時相當方便

ex: 在SCSS裡面

$themecolor: green;  <---先做一個宣告  記得要掛$

header{

color: $themecolor; <---使用變數時也需要掛$

}

* self & : Self Ampersand

在nested CSS中 【&符號】代表著這個【物件本身】!!!

header {

  background-color: antiquewhite;

  h1 {

      color:chocolate;

  }

 &*:hover*{   <---------這邊”&”*就代表header，所以要在後面加上* :hover時，可以直接這樣寫!!

background-color: yellow;

}

}

* import

用來分類、重複利用 (有些頁面沒有header、有些沒有footer)

你可以另外設定一個 .scss 專門寫網頁的某個部分

(例如: header、nav、main、section、footer等等)

分別獨立撰寫scss，最後統整成一份scss去做輸出css !!同時也更好維護!!

語法:

@import './header'

* SASS CSS Extension : 提升穩定性，寫更少的CSS code
* mixin

似程式語言的function  可以定義名稱，要先*@mixin* 函數名稱 (attr) { 函數內容 }

　之後再你要使用的物件中加上*@include* 加上函數名稱 並且帶入attr

*@mixin* flexbox ($direction){

    display:flex;

    flex-direction:$direction;   <---可以設定屬性給後續的物件去做設定

　}

　div*.link*{

*@include* flexbox(column); <---用@include來做maxin呼叫

　}

* Pseudo Classes 事件

div *:hover*{

    background-color: antiquewhite;

}

跟JS 的addEventListener 很像，將tag element綁定某個事件，定義事件觸發時要做什麼

語法:

Tagname + :事件{}

Ex: img:hover 🡨都不用加空格

:hover 滑鼠移過去觸發

:focus 點擊到該input區塊時觸發

:active 事件被觸發時

* 其他事件可以這邊找

<https://developer.mozilla.org/zh-TW/docs/Web/CSS/Pseudo-classes#%E5%9F%BA%E6%9C%AC%E8%99%9B%E6%93%AC%E9%A1%9E%E5%88%A5%E7%9A%84%E7%9B%AE%E9%8C%84>

* Pseudo Element

::before 可以在內容前方加上東西(content)，會自動判別段落

::after 可以在內容後方加上東西(content)，會自動判別段落

::first-letter 在每個段落第一個字生效CSS調整

::selection 可以讓使用者點選選取後的內CSS容作調整

記得tag name跟Pseudo Element 不用空格喔!!

*p::before* {

        content: ">>";

        color: red;

      }

      p*::first-letter*{

        text-transform: uppercase;

        font-size: x-large;

      }