

codingstyle標準

目錄

- codingstyle標準
 - 目錄
 - html標準
 - 骨架
 - 可使用元素表
 - 但如果有需要SEO開放下列標籤
 - 可用屬性表
 - 書寫規範
 - css標準
 - 命名規則
 - 基本架構
 - js/php標準
 - 命名規則
 - 基本架構
 - 其他規範
 - 其他
 - 其他範例及規範
 - 範例1: 網頁骨架

html標準

骨架

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- meta tag -->
    <title>Document</title>
    <!-- style sheet -->
    <!-- js script -->
    <!-- plugin -->
  </head>
  <body>
    <!-- 網頁內容 -->
  </body>
</html>
```

可使用元素表

元素	元素	元素
<a>	<audio>	
<body>	 	<canvas>(X)
	<div>	<form>
<h1>(X)	<h2>(X)	<h3>(X)
<h4>(X)	<h5>(X)	<h6>(X)
<hr>	<html>	<i>
<iframe>(X)		<input type="button">
<input type="checkbox">	<input type="color">	<input type="date">
<input type="datetime-local">	<input type="email">	<input type="file">
<input type="hidden">	<input type="image">(X)	<input type="month">
<input type="number">	<input type="password">	<input type="radio">
<input type="range">	<input type="reset">	<input type="search">
<input type="submit">	<input type="tel">	<input type="text">
<input type="time">	<input type="url">	<input type="week">
	<link>	<meta>

元素	元素	元素
	<option>	<p>(X)
<script>	<select>	
	<sub>	<sup>
<table>	<tbody>	<td>
<textarea>	<tfoot>	<th>
<thead>	<title>	<tr>
<u>		<video>

但如果有需要SEO開放下列標籤

元素	元素	元素
<article>	<aside>	<footer>
<header>	<label>	<legend>
<main>	<nav>	<summary>

可用屬性表

屬性	屬性	屬性
accesskey(X)	autocapitalize	autofocus
alt(SEO開放)	base	class
data-*	draggable	hidden
id	lang(SEO開放)	slot
spellcheck	src	tabindex(X)
title(SEO開放)	accept	autocomplete
capture	disabled	max
maxlength(X)	min	minlength(X)
multiple	pattern(X)	readonly
required	size	step

書寫規範

在屬性內將分為"有值"及"無值"兩種

- 有值屬性: 如class="example"
- 無值屬性: 如disabled
- 最後不可以有多餘的空白或 />

架構如下:

<主標籤 必要屬性 有值次要屬性(以開頭字母由小到大排列) 無值次要屬性(以開頭字母由小到大排列)>

範例:

```
<input type="text" class="input" id="name" name="name" placeholder="請輸入您的姓名" value="abc" required>
```

CSS標準

命名規則

- 選擇器名稱: 全小寫,單字間不分隔
 - 但如果樣式只有一行可以使用元素名(小駝峰或全小寫(以小駝峰為主))-值(空白使用_分隔,括號使用__分隔,逗號使用___分隔)
 - 例如: `display-none`
 - 例如: `background-rgb__12__34__56__`
- 不得使用#選擇器
- 盡量不用偽類選擇器及偽元素
- 不得使用除rgb或rgba或顏色名以外的色碼表示法

基本架構

- 首先是content
- 接者是border
- 接者是定位
- 接者是寬高
- 接者是背景
- 接者是文字
- 接者是內外距
- 接者是display
- 最後是其他(以字母順序排列)

js/php標準

命名規則

- 如果為const使用全大寫命名法
- 如果為let或函式使用全小寫命名法

基本架構

```
// import 區塊
import XXX from "XXX"
// 換一行

// 全域變數定義區塊
// 先const再let
const XXX="XXX"
// 換一行
let xxxaa="XXX"
/*
先字串
再數字
再布林值
再陣列
再物件
最後null、undefined、空值
*/

// 換一行

// 函式定義區塊
function xxx(){
    // 函式內容(與外層架構相同)
}

// 換一行

// 初始化區塊
// 換一行

// 主區塊
// 換一行

// 事件監聽區塊
// 換一行

// 全域函數呼叫或需初始化的函數呼叫區塊
```


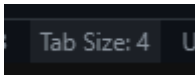
其他規範

- 字元間不加多餘空白

- 但json格式會使用正規json表示
- 結尾不加上分號
- 先雙引號再單引號
- 基本上不用foreach
 - 除了dom控制
- 不使用eval
- 不用switch-case
- 不用class
- 盡量不用while loop
- 能不用function就不用function
- 遇到`,`、`[]`務必換行並加tab
 - 但如果只有一行可以放在同一行但前後加空格(如: `{ "type": abc }`)
- 不用簡化表示法(如: `i++`等)

其他

- 請將space改為tab size,值設為4 如圖:



- 不要寫可重用區塊 範例:

```
<div class="footer-container margin-top-5vw" id="footer"></div>
<div class="footer-container-copy margin-top-5vw" id="footer-copy"></div>
```

應該改為

```
<div class="footer-container" id="footer"></div>
```

其他範例及規範

範例1: 網頁骨架

錯誤示範:

```
<div class="width-100vw flex justify-center items-center margin-top-5vw about-
block"></div>
```

應該改為

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

說明:

about-block這個class名稱不符合命名規則所以應該改為aboutblock.

而因為你都用了aboutblock了就將他都和到aboutblock內就好就不要再寫width-100vw flex justify-center items-center margin-top-5vw了。

v001003000 / 20241019