



## ch2.1\_Vue\_Template

### 指令(directives)將資料render到畫面上

<b>v-bind</b>	動態綁定屬性值從vue取得 縮寫為:[:]	 (如 src、href、class、style 等)
<b>v-model</b>	實現表單元素和組件的雙向綁定 (修飾符:lazy, number, trim)	<input v-model="message"> <p>{{ message }}</p>
<b>v-if</b>	元素僅在條件為真時渲染，將物件從DOM新增刪除。	<p v-if="isVisible">這段文字僅在 isVisible 為 true 時顯示。</p>
<b>v-else</b>	配合 v-if 使用，當 v-if 為 false 時顯示的元素	<p v-if="isVisible">顯示內容</p> <p v-else>隱藏內容</p>
<b>v-else-if</b>	配合 v-if 使用，處理多重條件渲染	<p v-if="x">A</p> <p v-else-if="y">B</p> <p v-else>C</p>
<b>v-show</b>	根據修改 display 屬性來隱藏元素，而不刪除 DOM )	<p v-show="isVisible">這段文字會根據 isVisible 顯示或隱藏。</p>
<b>v-for</b>	用於遍歷Array或Object，渲染列表 不要使用index當key值。	<ul><li v-for="item in items" :key="item.id">{{ item.name }}</li></ul>
<b>v-on</b>	用於綁定事件處理器 縮寫為:[@] 修飾符:很多，上網找。 //如果action函數不傳參數，不用寫小括號。	<button @click="increment">Click me</button> <button v-on:click="action">Click me</button>
<b>v-cloak</b>	防止未編譯的 Vue 模板顯示在頁面上，直到 Vue 編譯完成	<div id="app" v-cloak><p>{{ message }}</p></div>

- v-mode 表單元件和組件做雙向綁定
  - lazy : trigger when onblur(發生change事件)

- `number : parseInt()|parseFloat()` 這樣就不會變成字串串接
- `trim` : 排除輸入時的前後空白
- `v-pre` : **跳過這個元素的編譯**，直接在 HTML 中顯示原始的 Mustache 語法 或 Vue 指令，而不會被 Vue 解析。
- 透過屬性 `loading` 的 `true false` 調整。值為 `false` 時有差異。
  - `v-show` = 添加標籤 `style='display : none'` 將物件的可視性調整為無，標籤存在。
  - `v-if` = 直接將物件從 DOM 新增或刪除，變成註解，標籤不存在。

v-on 修飾符	
<code>.stop</code>	停止觸發上層 DOM 元素事件。
<code>.prevent</code>	避免瀏覽器預設行為。
<code>.capture</code>	不管觸發事件的目標是否是下層，設定 <code>capture</code> 的事件一定會先觸發。
<code>.self</code>	只有觸發此 DOM 元素本身才會觸發 <code>self</code> 事件。
<code>.once</code>	此事件只觸發一次。
<code>.passive</code>	無視 <code>prevent</code> 功能。

## CSS related 設定

### 綁定 class

標籤裡可以設定好幾個 class，可以由 (`v-bind`) 來決定在什麼狀況下要出現那些 class。

```
<p v-bind:class="active bold"></p>
```

### 綁定 style

賦予標籤 `style` 屬性，而 `style` 內的 CSS 屬性由 Vue 的計算屬性 (`computed`) 決定。

```
<p :style="changeSize"></p>
```

### css 裡面的標籤

`#app>div{}` 下一層

`#app div{}` 下面全部的層

## Template

在 Vue 中，`template` 是一個用來定義組件 UI 結構的區塊，它包含了組件的 HTML 結構。在 Vue 3 中，`template` 主要用於定義組件的模板，這些模板會被 Vue 轉換成虛擬 DOM，並根據數據的變化動態更新 UI。

簡單來說，`template` 是組件的視圖部分，它會渲染組件的界面，並根據 Vue 的反應性系統自動更新。

在 Vue 中，`template` 是組件的一部分，通常在 Vue 實例或組件的選項中作為屬性來使用。在 Vue 3 的 Composition API 中，`template` 會放在組件的模板標籤內。

## 將 `font-size` 設為 0

將 `font-size` 設為 0 會將該元素及其所有子元素的文字大小設置為零，從而隱藏文字。這在某些情況下可以用來隱藏文字內容，但保持元素的結構和佈局不變。

- 隱藏文字。
- 去除行內元素之間的空隙或間距。
- 控制排版時的微小細節。

## Else

`${ expression}` 裡面放的是表達式，不是變數，主要處理多行字串。String template