**Vue recap -0224**

**老師解說範例檔案：**<https://drive.google.com/drive/u/0/folders/1VjuRqIAYV-2zuVXKFX_D9ghgOi7ZuUYm>

## **Vue重要指令**

1. v-model：雙向綁定資料
2. v-on：綁定事件要用v-on，簡寫為@
3. v-bind：透過標籤改變屬性v-bind，把變數綁上去，簡寫為:
4. v-for：設定的資料[陣列]跑迴圈跑出來，v-for的v-bind:key一定要是唯一unique
5. component：會重複使用的HTML/UI樣式可以獨立做成元件，讓元件單純化(不需迴圈)，大幅提升可維護性

## **Vue Coding習慣**

1. 布林值→要用物件

直接丟一個值→用字串

1. <productItem name="Flash"></productItem> ←沒冒號丟字串，通常是丟變數

<productItem

:name="name"

:price=”price”></productItem> ←有冒號丟變數

1. 設定很多個變數時會分行，肉眼看比較清楚

## **CSS樣式**

1. 共用樣式可以貼到App.vue
2. border-box 訂閱老師頻道看影片
3. width: calc(25% - 10px) ←一排塞四個(25%)扣掉margin(10px)，要先border-box

## **Vue相關知識**

1. router-link用to ←SPA  
   用href會連結到新的頁面，就不是SPA了
2. 讓資料倒進元件裡面，怎麼切元件是前端工程師的重要功力
3. Vue是data binding，不是dom binding

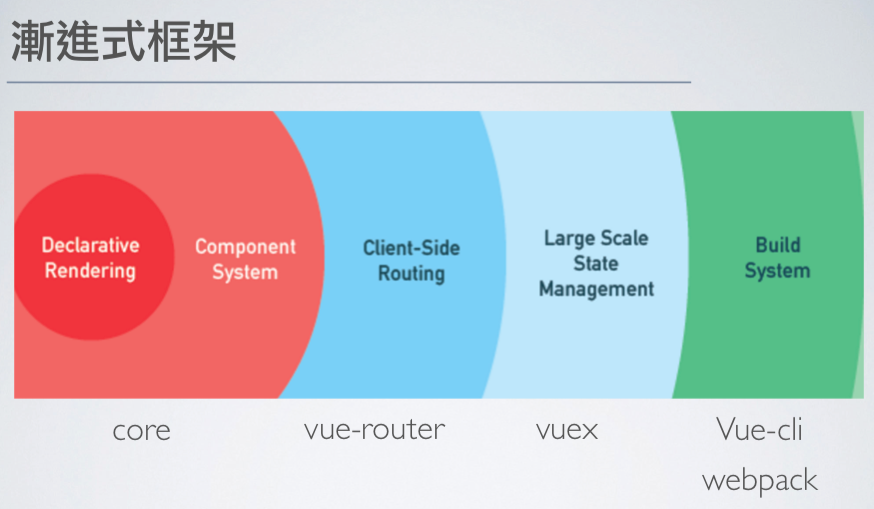
所以新增資料 要改的不是dom本身，是資料data ←在原始物件多push一筆資料

# 

# **Vue特色**

## **漸進式框架**

<https://ithelp.ithome.com.tw/articles/10216661>



## **SPA (Simple Page Application)**

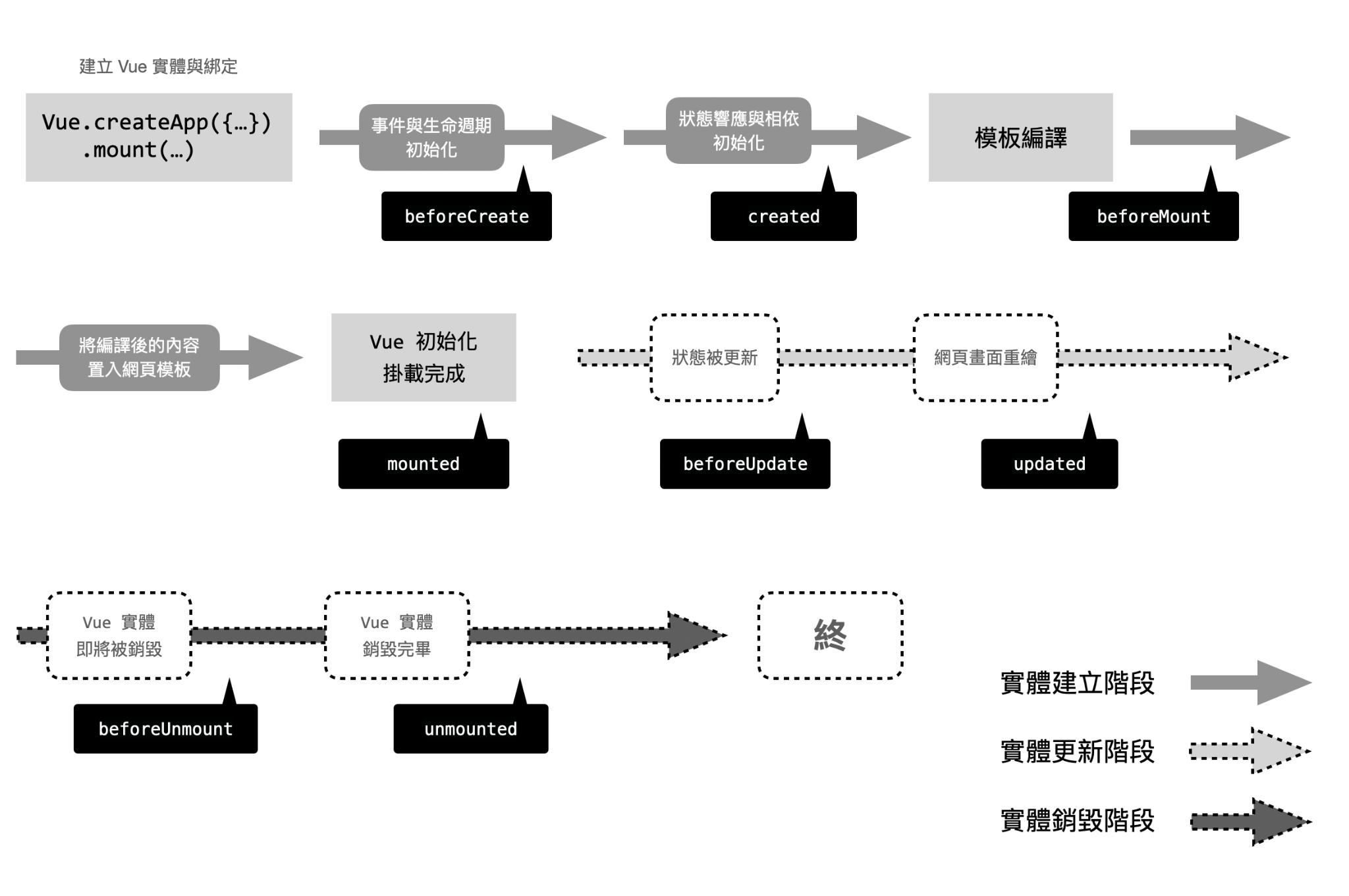
Push State: 透過router，局部更新內容，非整頁刷新，網址不會換都是同一個

以前網頁是MPA，每個網頁都是獨立網址

理論上效能: SPA > MPA，但要注意生命週期堆積太多事件效能會變差

## **生命週期**

<https://book.vue.tw/CH1/1-7-lifecycle.html>



## **SSR & CSR**

SSR 透過後端語言渲染，Nuxt.js is fpr Vue, Next.js is for REACT

CSR 透過前端語言JavaScript渲染，SEO不利，前端三大框架目前都是基於CSR所以建議多做SSR(Nuxt.js)

整個網站同步的東西都可以寫到state，Vue cli適合需要整個網站同步的專案

## **Vite**

<https://vitejs.dev/>

# **Vue學習資源**

1. <https://book.vue.tw/>
2. <https://ithelp.ithome.com.tw/users/20114645/ironman/2811?page=1>
3. 六角學院

## 

## **Windows指令權限問題**

<https://israynotarray.com/other/20200510/1067127387/>

el = element

el:  
data: 跟哪些資料綁定

Vue 選擇器通常是選ID

Ctrll+Shift+I：召喚開發者工具，可以看console

Ctrl+K+Z：畫面簡潔

Ctrl+P：查找整個專案中的檔案

右上角→file→fork sandbox

expression = any valid code can be resolved a value，任何有效程式可以變成數值（數字/布林/字串）

**任何一個程式碼最後會丟出一個數字/字串，用console檢查**

**[JavaScript] Javascript 的表達式 (Expressions)和陳述式 (Statements)：一個用說的，一個用做的**

<https://medium.com/itsems-frontend/javascript-expressions-and-statements-7bd374effc95>

v-model 只會用在input/需要輸入資料的欄位

v-on:event="method / statement" 可以簡寫成@event="method / statement"  
  
event.target 那個觸發事件的物件本身

Vue語法

computed(){　}

<span v-if="!hasData">目前還沒有資料。</span>

!放在變數前面，就把ture變成false

JS存資料用object，object再存到資料庫

# **Vue語法問題整理，對照講義頁數**

1. p.9 v-bind的簡寫 = 空一格:

2. p.8 parentText, parent-text：Vue如果有大寫，要用橫槓-連結

3. p.11 type: Object Q：為什麼這裡的資料型態Object不用中括號[]？

- A：沒有用到validation不用中括號[]

5. v-bind解構，想再請老師補充

6. p.12 語法糖的根結點(root)只能有一個

7. p.15

- emit(發射)：由內而外，用@指定，通常寫在子component裡面觸發母component的東西

- props：由內而外，用:指定

8. p.15 watch監聽跟v-on差別：非監聽行為，而是監聽更細的資料

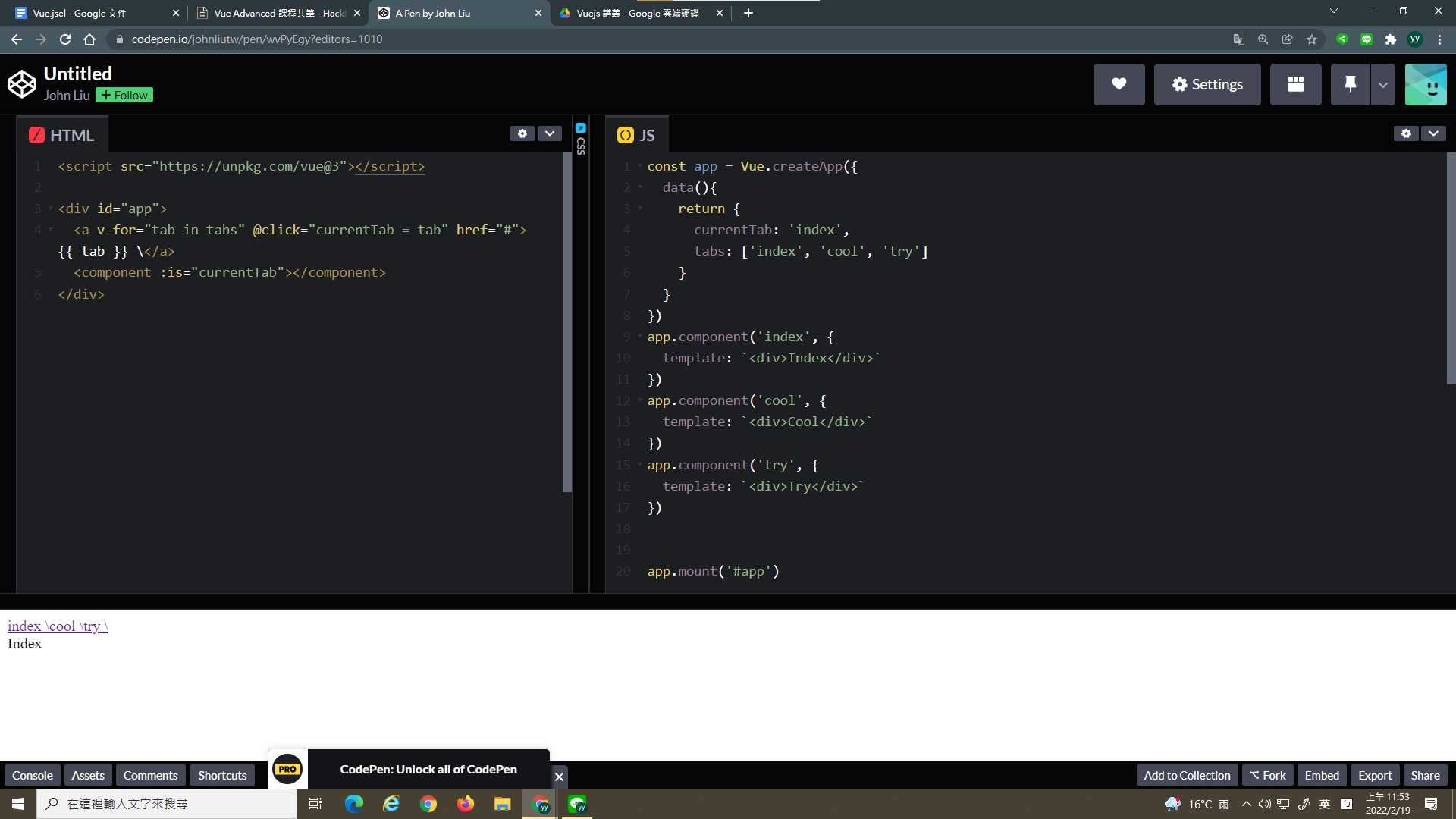
- 在這邊是監聽上面data裡的number(= v-model的number)

- watch 與 handler 固定一組

9. $emit $表示通用的函式，跨component大家都會用到

p.16 Provide + Inject 方便但也是很多bug來源，暴力解決留技術債

p.18 Component switch



<component :is="currentTab"></component> 這邊的 :is 就是"是"的意思 = 這個component是currentTab

<!-- v-text 設定tag裡面的純文字，像jQuery的text() -->

p.19 Vue 元件切換,為什麼資料遺失? A: 做component switch時是把把原本的component銷毀

p.20 Keep alive：component雖然切換但不會被砍掉

p.22 Keep alive life cycle 如果有上百個component的東西要記得，瀏覽器會變很慢

p.21 解法用include包含/exclude排除

Vue Life Cycle:

<https://learnvue.co/2020/12/how-to-use-lifecycle-hooks-in-vue3/>

$emit $表示通用的函式，跨component大家都會用到

data與computed的差別：

- data：靜態資料

- computed：會被運算過會變化的資料，實際顯示出來會是運算後的樣子，像沒有用到函式的methods

特定的邏輯才會用到form，所以呼叫

template是Vue.js專用的HTML tag，不會出現在畫面上

忽視HTML結構，專注於template裡面包的東西就好

Vue CLI要先裝npm

檔案結構 main是設定檔，通常不去動

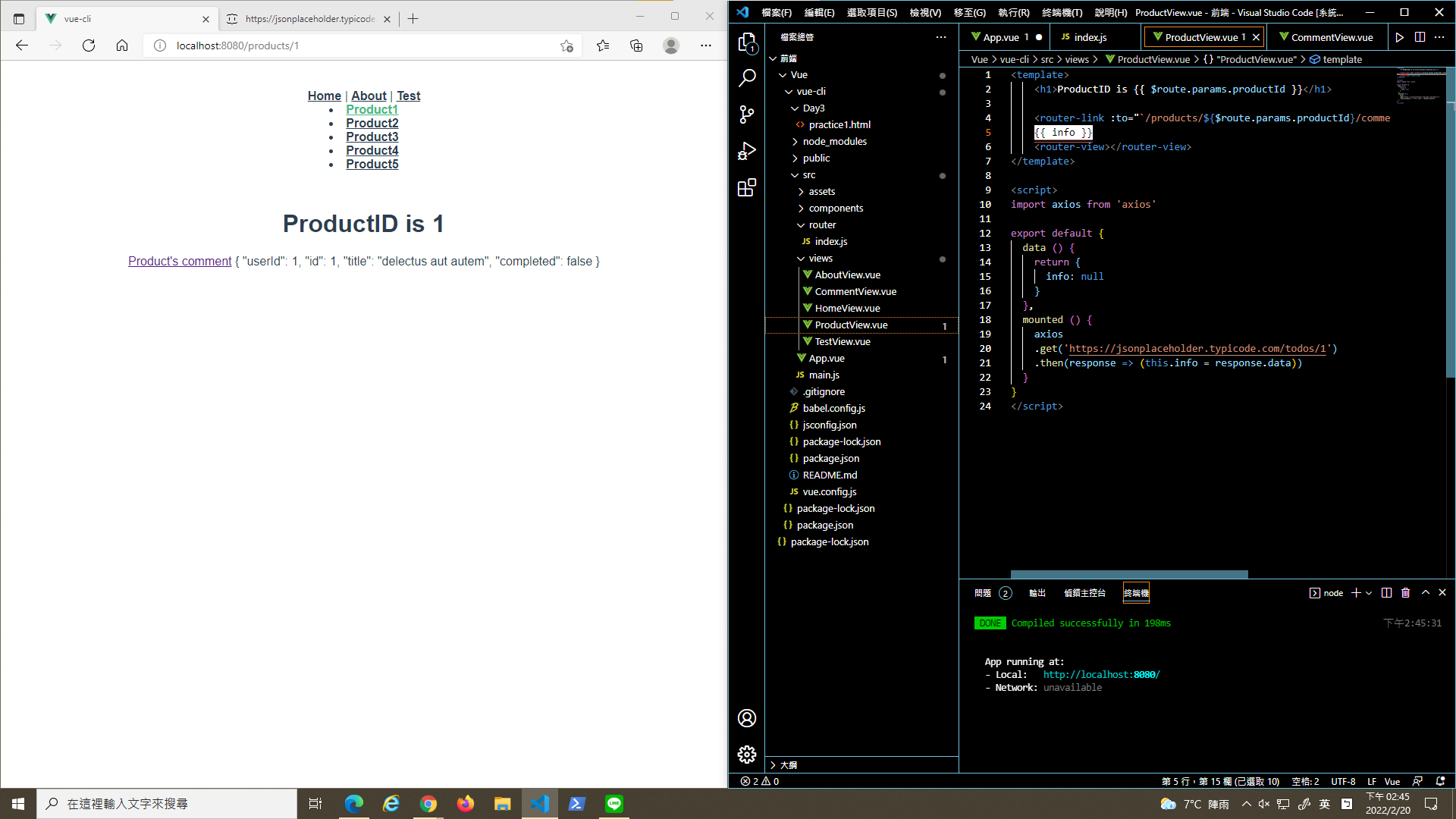
key是Vue裡面比較嚴格的檢查 先想成ID

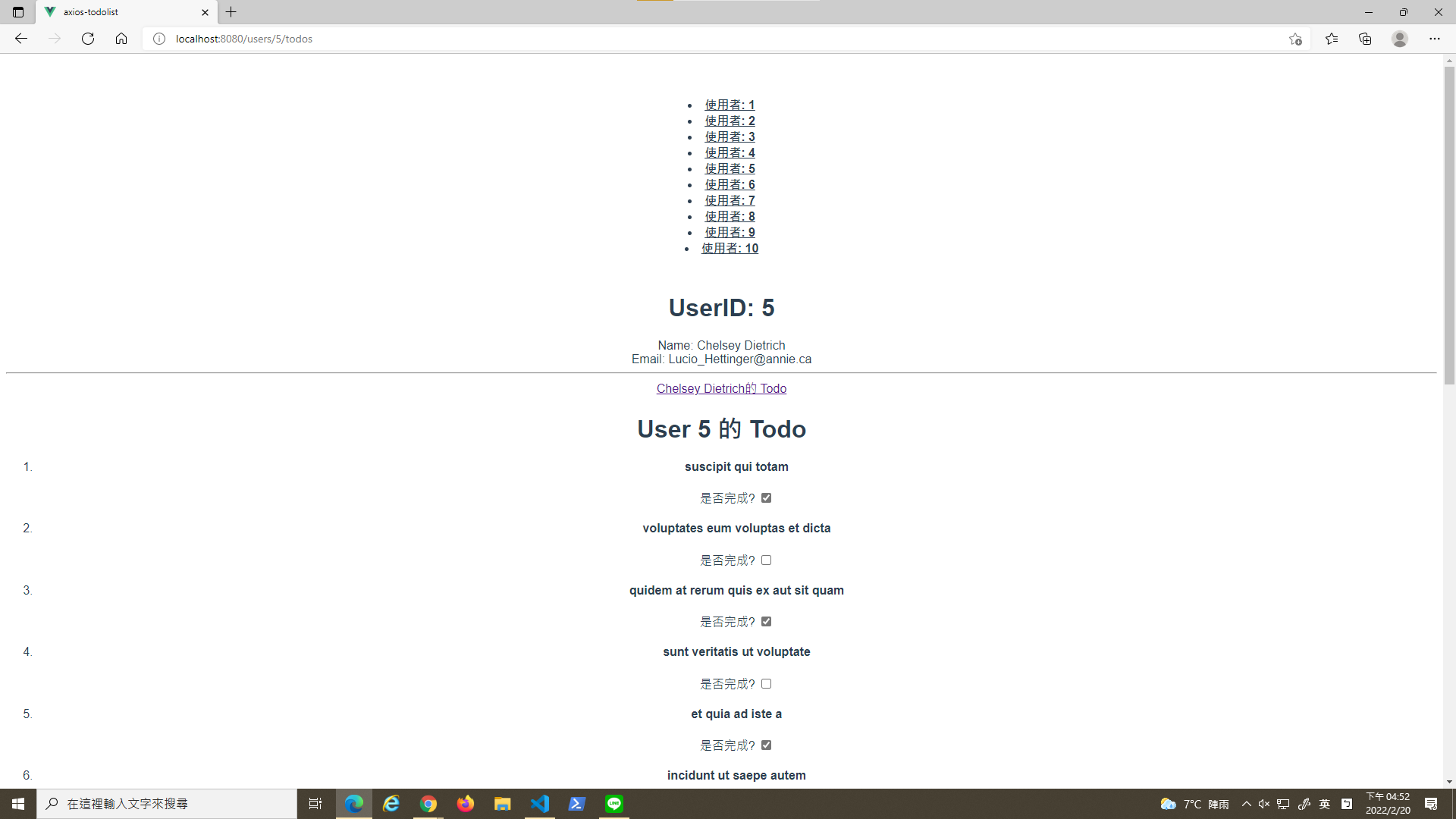
空白就是路由沒有對到

Redirect實用

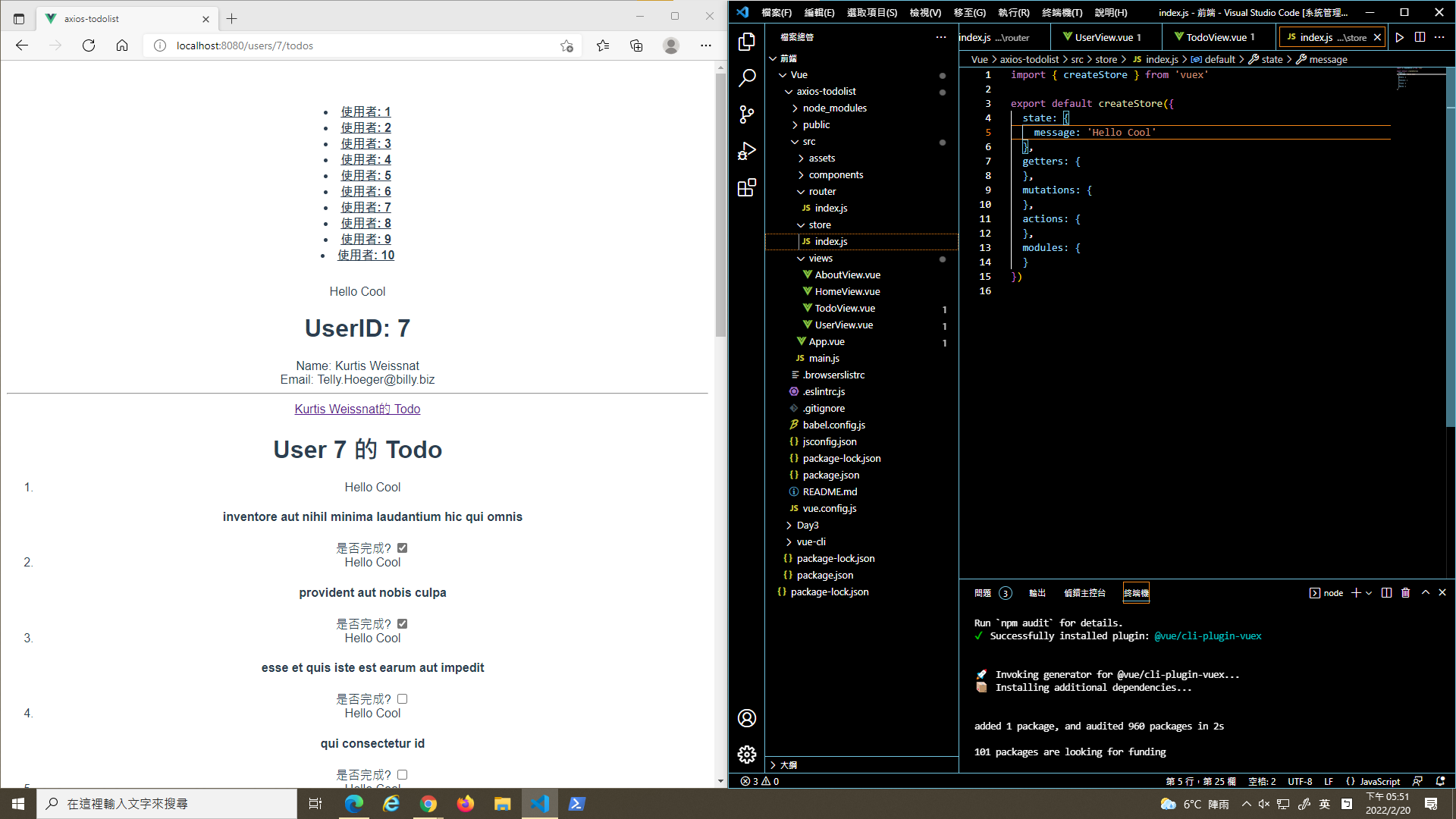
## **Axios**

1. script放到ProductView 2. template裡面要加一個{{ info }}





## **Vuex**



# **前端工程師職涯發展**

薪水：第一年雙北30k~50k

成長幅度：去面試就會知道

加班：觀察客戶會不會在非上班期間使用產品, 產品是否穩定

工作與學習時間分配：

工作要寫程式每天1小時學習練習時間

工作不用寫程式轉職每天2~3小時

學習資料：官方文件, 鐵人賽 的文章

轉職工程師的優勢：原本產業的知識knowhow

side project 3個左右，求深不求多

推薦學習Vue.js, Git, Linux指令

先有工作經驗，需要演算法等研究所的知識時再去補學歷