4/29 學習 Angular 概念

簡介

Angular是一個應用設計框架與開發平台,旨在建立高效而精緻的單頁面應用。

使用 TypeScript 來充分發揮在強型別程式設計環境中工作的優勢。強型別檢查降低了應用程式中的一個元素向另一個元素傳送錯誤格式資料的可能性。

Angular包括：

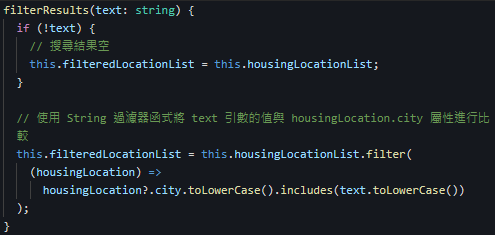
1. 一個基於元件的框架，用於建立可伸縮的 Web 應用
2. 一組完美整合的函式庫，涵蓋各種功能，包括路由、表單管理、用戶端-伺服器通訊等
3. 一套開發工具，可幫助你開發、建立、測試和更新程式碼

核心： 1.[元件](#元件) 2.[樣板](#樣板) 3.[指令](#指令) 4.[依賴注入](#依賴注入)

Angular優點

1. 宣告式樣板，讓應用邏輯與外觀完全分開，易於建立和維護。-from [指令](#指令)
2. 自帶函式庫，讓開發應用時新增其他功能更容易。一張含有 文字, 螢幕擷取畫面, 字型, 軟體 的圖片

   自動產生的描述
3. Angular CLI工具，快速建立應用程式一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

   自動產生的描述
4. TypeScript 設計程式，降低錯誤型別發生率(指定參數型別)，EX:  
   

和Django比較

1. 前後端:  
   Django比較像是前後端整合；  
   Angular則是前端框架，故Angular在加載頁面速度來說相對較快
2. 前端開發方式:  
   Django開發時通常一頁的東西都寫在一起，各種功能也就隨之寫在同個JavaScript區塊中，導致複雜不好維護；  
   Angular則是分成多個元件，每個元件有類似像”元件名稱”，可直接在template引用此元件，而引用同時可傳遞資料到另一元件上，再讓另一元件的template展示該元件想展示的內容。【分離功能】  
    

知識要點

一個Angular應用由許多**元件**構成。

**˙ 元件**

**一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述 - 元件**

** - HTML**

**↓**

**一張含有 文字, 字型, 螢幕擷取畫面, 圖形 的圖片

自動產生的描述 - 生成的DOM**

元件分為**”帶有@Component()裝飾器的TypeScript類”**、  
**”HTML樣板”**、**”樣式檔案”。**

**@Component()指定以下訊息: (描述元件的使用、渲染模樣及外觀)**

* **一個CSS選擇器，定義如何使用元件**
* **一個HTML樣板，指示如何渲染此元件**
* **一組可選的CSS樣式，定義HTML元素外觀**

˙ 樣板(.html)

1. 支援**內嵌**：{{ message }}指示Angular進行”內嵌”。值由元件類別定義。

Ex:   
一張含有 字型, 圖形, 文字, 印刷術 的圖片

自動產生的描述

1. 支援**繫結**：[attribute] =“value” 指示如何設定HTML屬性，繫結到元件類別中的屬性(value)

Ex:

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

1. 宣告事件監聽器：( click ) 指定操作。例子中呼叫的func由元件類別中定義

Ex:

一張含有 文字, 字型, 螢幕擷取畫面, 設計 的圖片

自動產生的描述

而元件(.ts)類別定義如下：

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

˙ 指令&樣板指令

在類別中使用if、for等等的指令，動態修改DOM。  
樣板指令: \*ngIf、\*ngFor

一張含有 文字, 螢幕擷取畫面, 字型, 設計 的圖片

自動產生的描述 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

˙ 依賴注入(DI)

一張含有 文字, 筆跡, 字型 的圖片

自動產生的描述

宣告TypeScript類別的依賴項就可使用，不需操心如何實例化。

(OS: 有點像是C++中import函式庫，就可以直接呼叫func使用)

運作：在(1)定義writeCount()，在(2)呼叫建構子，將服務注入到(2)即可使用

(此時，(1)稱為提供依賴項，(2)稱為注入依賴項，  
providedIn: ’root’表明從應用程式根級別提供服務，  
也就是根下所有類別皆可使用)

(1) logger.service.ts (2) hello-world-di.component.ts

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

教程-工具指令

1. 啟動開發伺服器: ng serve
2. 建立元件: ng generate component Home   
   　 --standalone --inline-template --skip-tests
3. 建立介面: ng generate interface housinglocation
4. 建立可注入服務: ng generate service housing --skip-tests

教程-知識描述

1. 元件中的”元資料” (簡略列出)

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

˙selector: 如何在樣板引用本元件

˙standalone: 元件是否需要ngModule

˙imports: 元件依賴關係

˙template: 元件的HTML標籤和佈局

˙styleUrls: 列出元件使用的CSS檔案的URL(陣列形式)

1. 介面: 定義新資料型別(HousingLocation)-它定義單個物件的資料屬性及資料型別，在其他地方可以使用HousingLocation型別的資料。
2. @Input() 裝飾器: 在子元件將需要共享資料的變數以@Input()裝飾，如此就可以將資料由父元件共享到子元件。
3. 繫結(傳資料)與內嵌(接收資料)：

繫結(傳資料) ：

在模板中鍵入<app-housing-location>標籤，並在其中”繫結”，也就是將housingLocation這個屬性的資料傳給housing-location元件。

內嵌(接收資料) ：

在housing-location元件中就可以使用內嵌{{housingLocation.name}}來插入值。

一張含有 文字, 螢幕擷取畫面, 軟體, 陳列 的圖片

自動產生的描述 一張含有 文字, 螢幕擷取畫面, 字型, 軟體 的圖片

自動產生的描述

1. Angular指令: \*ngFor

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

// \*ngFor="let housingLocation of housingLocationList"

// 宣告housingLocation為housingLocationList的每一項

// [housingLocation]="housingLocation"右側值為ngFor使用的變數

1. 路由: 是從應用程式中的一個元件導向到另一個元件的能力。

EX: 指定請求該路由(urls)時應在螢幕上顯示哪個元件(頁面)。

* 1. Main.ts匯入路由檔案和函式  
     一張含有 文字, 字型, 螢幕擷取畫面, 圖形 的圖片

     自動產生的描述
  2. Main.ts新增路由設定  
     
  3. Routes.ts 定義兩個路由  
     一張含有 文字, 螢幕擷取畫面, 陳列, 軟體 的圖片

     自動產生的描述
  4. App.component匯入函式，並在@Component中import，並更新模板  
     

  
一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

* 1. Housing-location.ts 匯入函式、imports並將模板新增連結  
       
       
     

1. 一張含有 文字, 螢幕擷取畫面, 行, 字型 的圖片

   自動產生的描述表單(form)  
   一張含有 文字, 螢幕擷取畫面, 字型 的圖片

   自動產生的描述   
   一張含有 文字, 字型, 螢幕擷取畫面, 數字 的圖片

   自動產生的描述 一張含有 文字, 螢幕擷取畫面, 字型 的圖片

   自動產生的描述

FormGroup():整個表單  
FormControl(‘’):建立表單輸入內容的型別(也提供預設值)

1. 樣板變數

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

#filter變數會=Input輸入的值，透過button將變數值傳入func，  
進行篩選房屋判斷(字串比對)

教程-架構描述

一張含有 文字, 螢幕擷取畫面, 軟體 的圖片

自動產生的描述

Index.html:HTML樣板，根模板

App.component:應用程式元件，這裡當作全部元件的容器

Housinglocation:介面

Housing.service:可注入服務(提供服務)

Routes.ts:定義路由處

Home.component:Home元件，提供輸入表單及顯示結果，房屋資料傳遞

Housing-location:房屋資料顯示

Details:展現房屋位置的詳情頁面