**ASP.NET-錢老師**

**Part\_1**

* **MVC** Lab\_Controller－在 ASP.NET Core 寫作 Controller **範例Return的討論**
* **MVC** Lab\_DI－註冊登記服務：呼叫生命週期的不同　**範例：現在時間為何？**
* **API** Lab Middleware－使用 ASP.NET 的 Middleware / Pipeline 功能
* **MVC** Lab MVC ToDo－以 ASP.NET Core 建立 MVC Web App
* **API** Lab\_WebAPL\_News－以 ASP.NET Core 建立 WebAPI Serve

**Part\_2**

* **MVC** Lab\_Controller（**Part\_1 SAME**）
* **MVC** Lab\_Controller\_PassDataToView－Controller 如何傳遞 / 繫結資料到 View
* **MVC** Lab\_View\_Helper－使用 ASP.NET Core 的 Helper 建構 View。

**範例：Model資料夾內的欄位名稱與欄位錯誤訊息**

* **MVC** Lab\_View\_Layout－使用 View 的版面構成功能
* **MVC** Lab View Razor －使用 Razor 語法

**Part\_3**

* **API** Lab\_LINQ－以 LINQ 語法實作 Web API
* **MVC** Lab\_MultiUsers－多人共用衝突調解
* **MVC** Lab MVC ToDo HandMake－以 ASP.NET Core 建立 MVC Web App
* **API** Lab WebAPI TODo-以 ASP.NET Core 建立 Web API 應用程式

**(配合 Vue.JS??)**

**Part\_4**

* **MVC** Lab\_Identity－使用 ASP.NET Core 的 Identity 進行驗證與授權。
* **API** Lab\_LINQ - 本專案採用 Database-first 作法- 說明/練習 LINQ 語法

（之前教過）

* **MVC** Lab\_Login\_Cookie –以 Cookie 實作使用者登入系統。設定幾天保存。
* **MVC** Lab\_Login\_Session－以 Session 實作使用者登入系統。
* **MVC** Lab\_ MultiUsers（之前教過）

**Part\_1**

**MVC Lab\_Controller**－在 ASP.NET Core 寫作 Controller

**範例Return的討論**

1. View()

一張含有 文字 的圖片

自動產生的描述

1. Content() 不用在View資料夾裡面有.cshtml

一張含有 文字 的圖片

自動產生的描述

1. return result

一張含有 文字 的圖片

自動產生的描述

1. return Redirect("http://www.hinet.net")
2. return RedirectToAction("Privacy", "Home"); 資料夾裡的檔案開啟

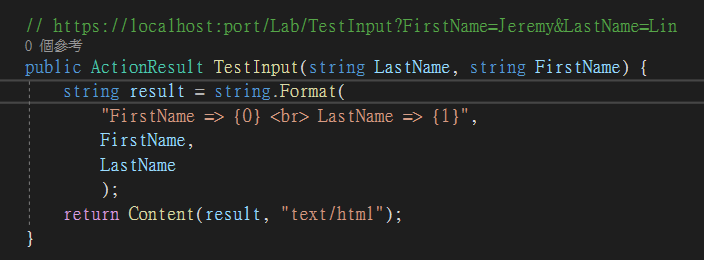
一張含有 文字 的圖片

自動產生的描述

1. return Content(result, "text/html");　用text/html的格式把string result用出來

一張含有 文字 的圖片

自動產生的描述



1. return Content();

一張含有 文字 的圖片

自動產生的描述

※[ActionName( )]，另取網址名。

一張含有 文字 的圖片

自動產生的描述

**MVC** Lab\_DI－註冊登記服務：呼叫生命週期的不同

範例：現在時間為何？

services.AddSingleton<IDateTime, SystemDateTime>();

services.AddScoped<IDateTime, SystemDateTime>();

**API** Lab Middleware－使用 ASP.NET 的 Middleware / Pipeline 功能

**MVC** Lab MVC ToDo－以 ASP.NET Core 建立 MVC Web App

1. 先建MODEL－要自己打String student 之類的
2. 用語法快速建置控制器CREATE GET DELETE EDIT **SCRU**

※延伸－探討兩者的差別：

options**.UseSqlServer(**Configuration.GetConnectionString("TodoContext"))

🡪直接影響資料庫

opt**.UseInMemoryDatabase**("TodoList")

🡪存在站存記憶體，不會影響到資料庫，網站關閉之前的修改也不會更動到資料庫。

**API** Lab\_WebAPL\_News－以 ASP.NET Core 建立 WebAPI Server

1. 建立MODEL－直接用連線字串連SQL
2. 用語法快速建置控制器**SCRU**
3. 加入前端畫面

Client framework: Vue.JS 說明在哪~~?

**Part\_2**

**MVC** Lab\_Controller－在 ASP.NET Core 寫作 Controller（跟Part1資料一樣耶）

**MVC** Lab\_Controller\_PassDataToView－Controller 如何傳遞 / 繫結資料到 View

1. 使用 ViewData["welcome"] = "Hello! world."; / @ViewData["welcome"]
2. 使用 ViewBag.Welcome = "Welcome!"; / @ViewBag.Welcome
3. 使用 ViewModel

string dataToModel = "Welcome!!", return View("index", dataToModel);/

@model string / @Model

1. 使用 TempData[]，讓資料活到下一頁面。

搭配TempData.Peek()可活到在下一層頁面。

TempData["prompt"] = $"Today is : {DateTime.Now.ToString("yyyy-MM-dd")}";

一張含有 文字 的圖片

自動產生的描述

* TempData是將資料存放在 Session當中，而ViewData是存放在

Controller的物件內。然而TempData的生命週期是在一個Request的範圍內，若是透過RedirectToAction導頁到不同的Action時，則可以取得到上一個Action所存放在TempData的資料。

**MVC** Lab\_View\_Helper－使用 ASP.NET Core 的 Helper 建構 View。

Model資料夾內的欄位名稱與欄位錯誤訊息

* [Required(ErrorMessage = "產品名稱是必填資料")]
* [DisplayName("價格")]
* [Range(1, int.MaxValue, ErrorMessage = "價格請大於零")]

**MVC** Lab\_View\_Layout－使用 View 的版面構成功能

不使用 Layouot 版面配置

@RenderSection("MyStyle", required: false) -JS

@RenderBody() -JS

@await RenderSectionAsync("MyScripts", required: false)

@section MyStyle {<style>h1 {color: blue; }</style>} -CSS

@section MyScripts {<script>alert("@(TempData["prompt"] ?? "Hello!")") </script>}-JS

**MVC** Lab View Razor－使用 Razor 語法

ViewData["twoLines"] = "Line1<br>Line2";

@Html.Raw(ViewData["twoLines"])

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

**Part\_3**

**API** Lab\_LINQ－以 LINQ 語法實作 Web API

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

**MVC** Lab\_MultiUsers－多人共用衝突調解

簡易實作購買流程

**MVC** Lab MVC ToDo HandMake－以 ASP.NET Core 建立 MVC Web App

- 之前 Lab\_MVC\_ToDo 的續集，

再做一遍待辦事項清單的增查修刪

- 我們會透過

- 放慢速著眼細節

- 嘗試不同作法

- 手作（不用 Scaffold 快速建立控制項與檢視）

學習如何從無到有建構出 MVC Web 應用程式

-顯示待辦事項清單

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

**API** Lab WebAPI TODo-以 ASP.NET Core 建立 Web API 應用程式 (配合 Vue.JS)

以 Vue.js 搭配 Axios 實作 Client App???

**Part\_4**

**MVC** Lab\_Identity－使用 ASP.NET Core 的 Identity 進行驗證與授權。

**API** Lab\_LINQ - 本專案採用 Database-first 作法- 說明/練習 LINQ 語法

（之前教過）

**MVC** Lab\_Login\_Cookie –以 Cookie 實作使用者登入系統。設定幾天保存。

**MVC** Lab\_Login\_Session－以 Session 實作使用者登入系統。

**MVC** Lab\_ MultiUsers（之前教過）