

在 Windows 上安裝與使用 VS Code

鄭士康, 2/11/2021

這裡一步一步教大家如何安裝 Visual Studio Code (VS Code，或 VSC)，並且建立 C#專案，在螢幕印出 **Hello World!**。

1. 安裝 .NET SDK

SDK 是 Software Development Kit 的簡稱，是開發軟體的工具套件。所以 .NET SDK 便是撰寫 .NET 程式所需的底層工作套件，可以支援多種以 .NET 架構為基礎的程式語言，C# 是其中之一。

首先連線到微軟的快速教學網站

<https://dotnet.microsoft.com/learn/dotnet/hello-world-tutorial/intro>

進站畫面如圖 1 所示：

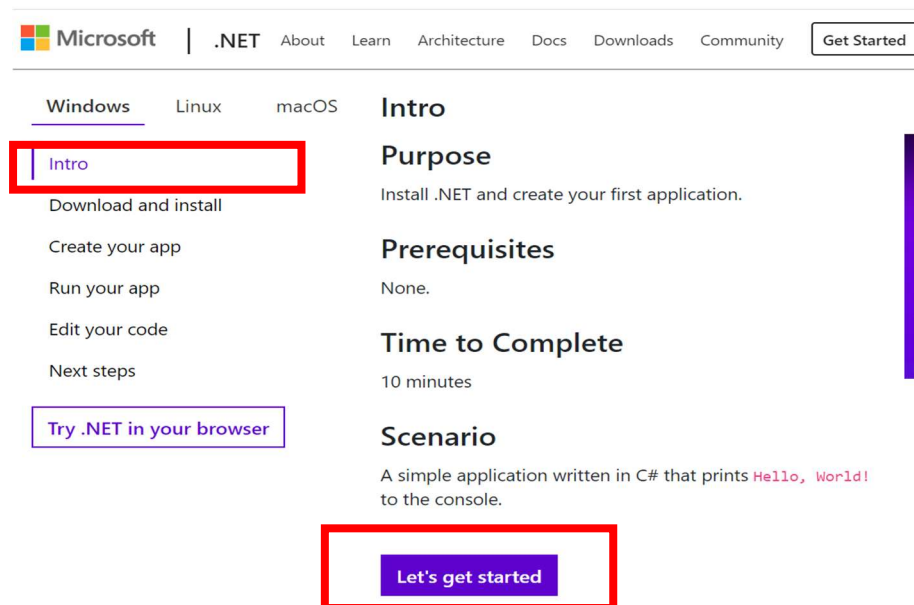


圖 1. 微軟 .NET 教學網站進站畫面

注意現在是在 intro 的網頁。按下我以紅框框起來的紫色按鈕 Let's get started，進入 Download and install 網頁如圖 2：

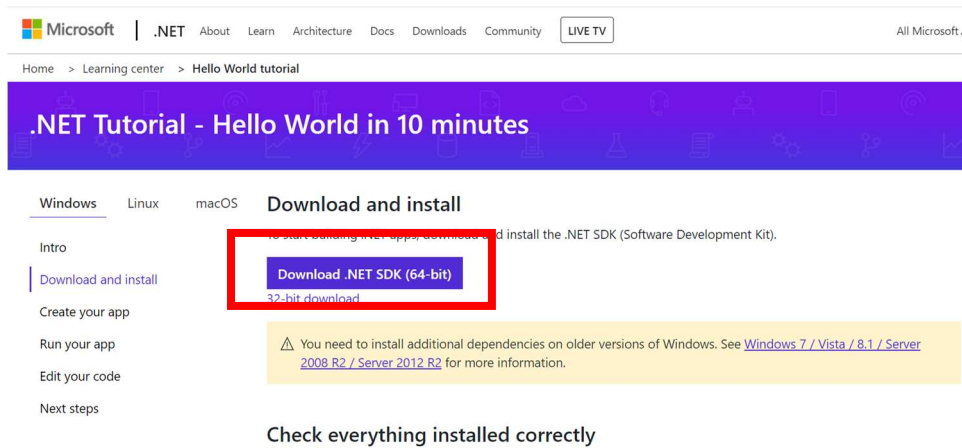


圖 2. 微軟 .NET 下載安裝畫面

現在一般電腦都是 64 位元，所以按下圖 2 中以紅框標示的紫色按鈕 Download .NET SDK (64bit)，開始下載 .NET SDK。此時電腦會跳出如圖 3 的視窗：

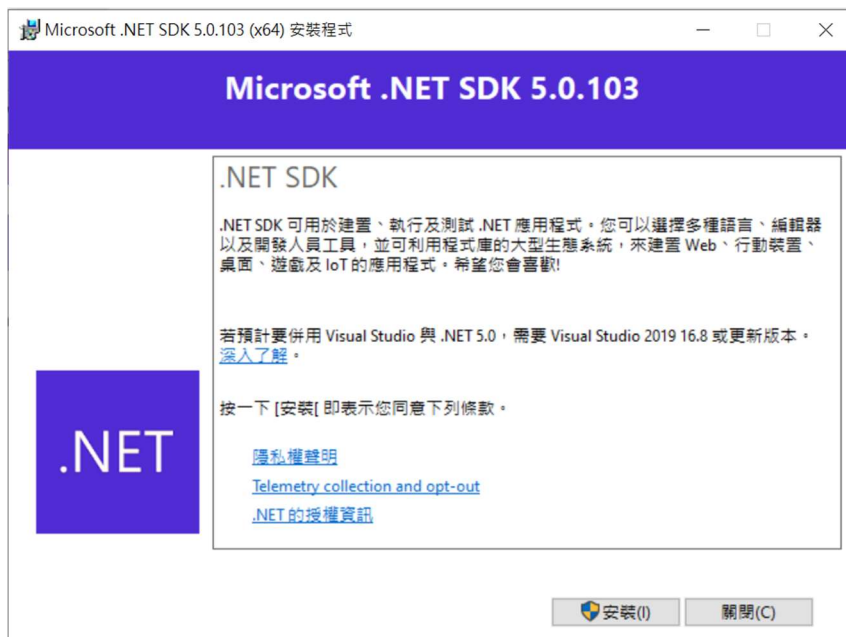


圖 3. 微軟 .NET SDK 安裝視窗

按下圖 3 的「安裝」按鈕，開始安裝，直到完成，顯示出圖 4 視窗：

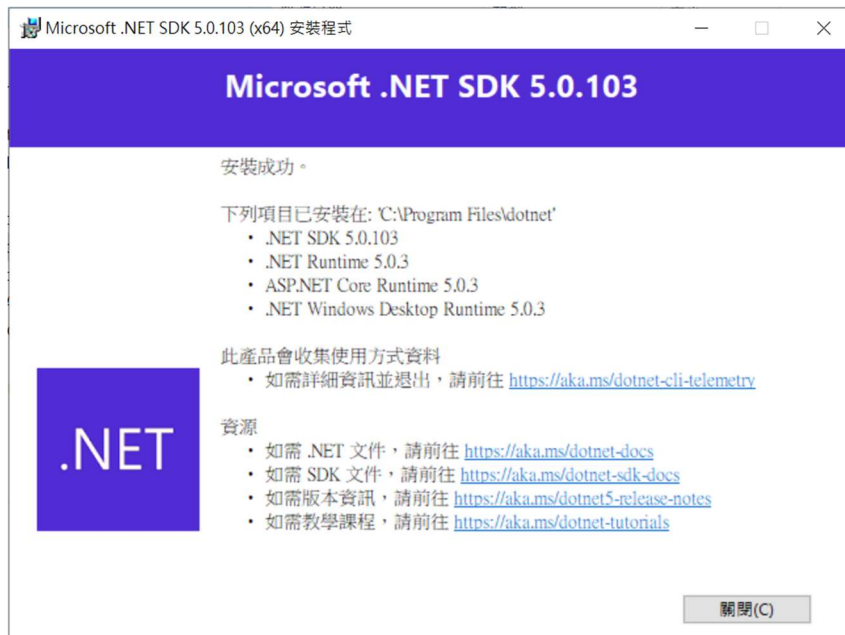


圖 4. 微軟 .NET SDK 安裝完成顯示的訊息視窗

關閉圖 4 視窗後，可以使用執行 Windows 系統的命令提示字元，輸入 `dotnet`，系統回應如圖 5，表示 .NET SDK 已經成功安裝：

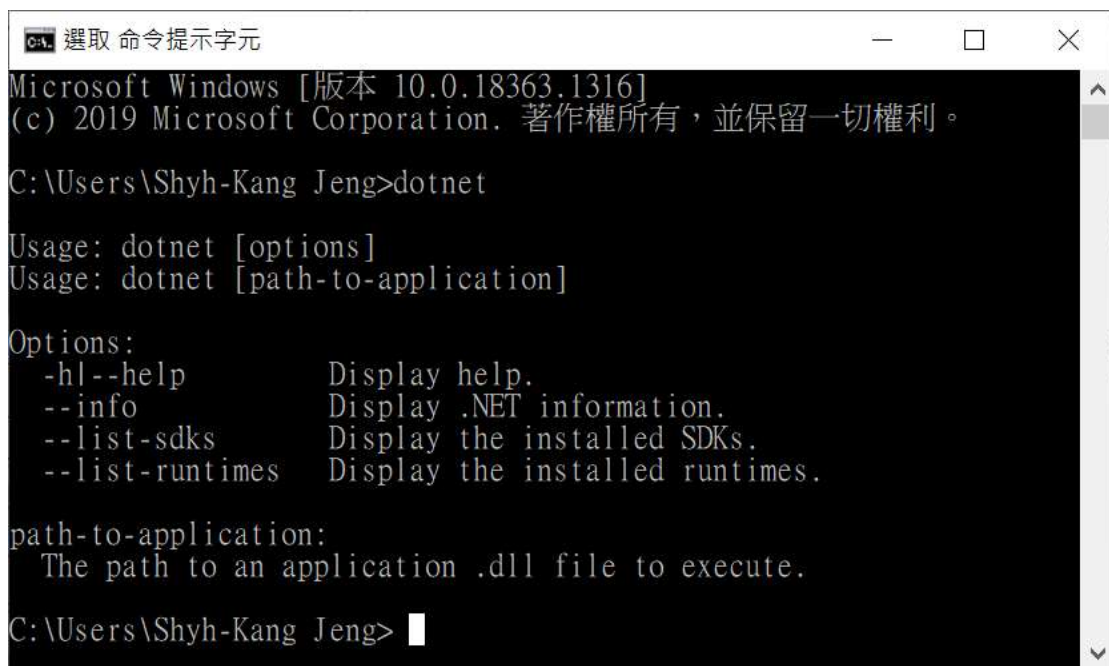


圖 5. Windows 作業系統命令提示字元的回應視窗

2. 安裝 Visual Studio Code

接著安裝 Visual Studio Code 程式整合開發環境(Integrated Development Environment · IDE)。

先到 VSC 的官網 <https://code.visualstudio.com/>，其首頁畫面如圖 6：

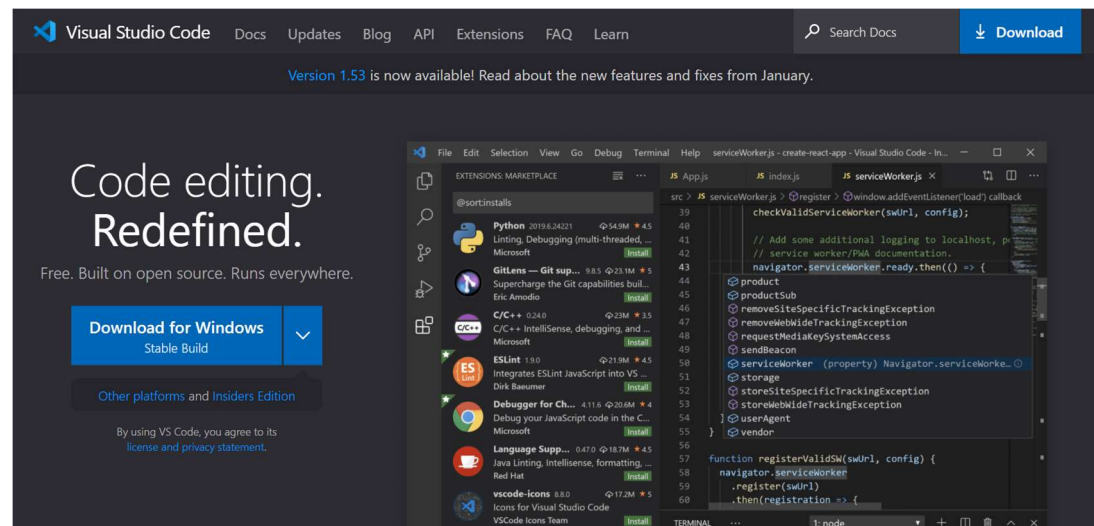


圖 6. Visual Studio Code 官網進站畫面

按下 Download，畫面如圖 7：

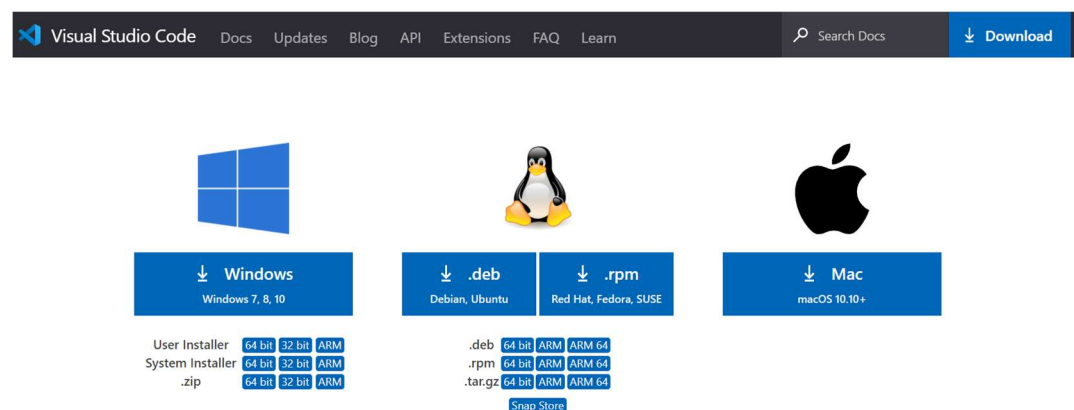


圖 7. 下載檔案選擇畫面

依照所作業系統，點選對應小按鈕，下載 Visual Studio Code。接著執行下載的 Set Up 檔案，都使用默認(default)選項。完成安裝後，應該可以直接啟動 Visual Studio Code。如果沒有，或者之後要重新啟動 Visual Studio Code，

可以由 Windows 系統「開始」按鈕 ，在其顯示的程式列表中，選擇 Visual Studio Code 即可。

啟動後，所顯示的使用介面，可以到

<https://code.visualstudio.com/docs/getstarted/userinterface>，參看如圖 8 顯示的 IDE 使用介面解說。圖中標示的 A、B、C、D、E 五個部分，可以先大致瀏覽一下。其中常用的部分，本文用到時自會說明。

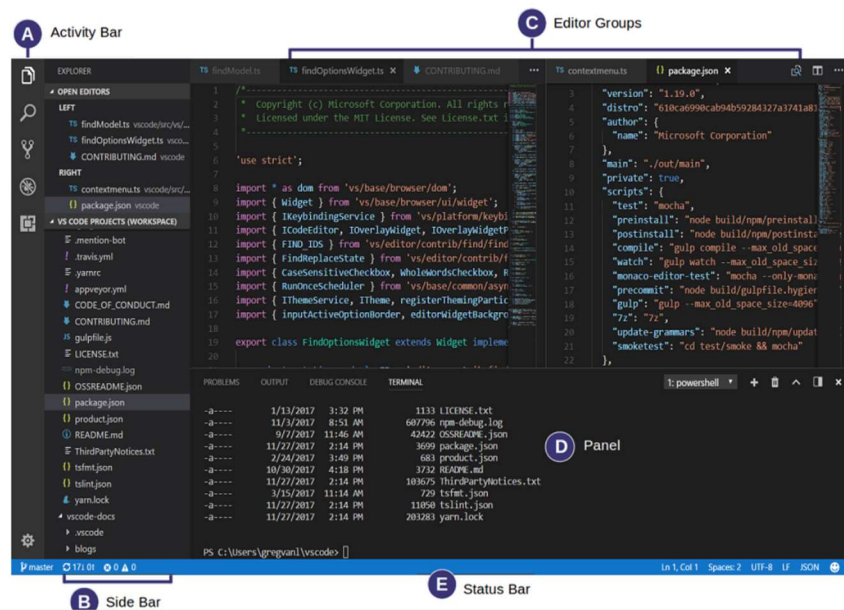


圖 8. Visual Studio Code 的使用介面說明網頁

3. 調整色彩佈景主題設定

由 Windows 作業系統最左下方的執行工作按鈕，在其顯示的程式列表中，選擇 Visual Studio Code，就能進入 VSC 的起始畫面如圖 9：

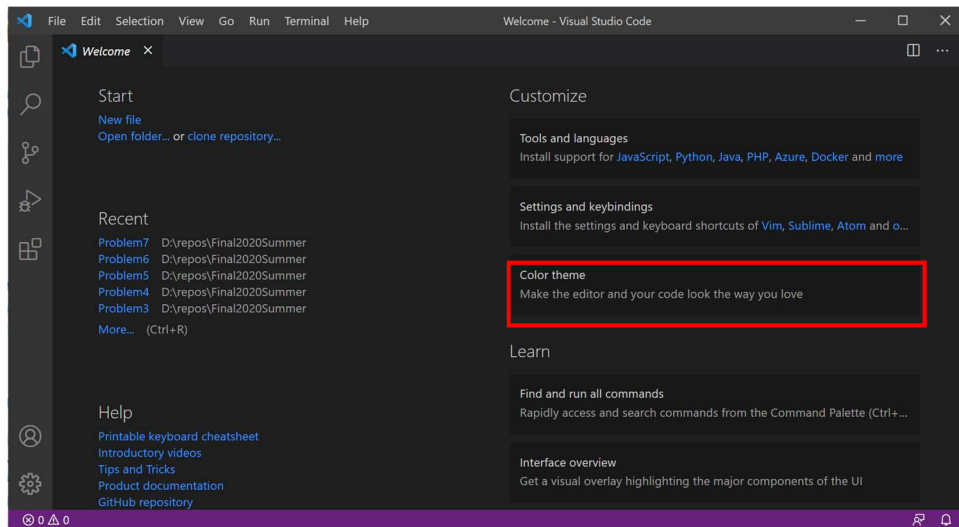


圖 9. Visual Studio Code 的起始畫面

Visual Studio Code 和當今許多 IDE 一樣，設定工作畫面為黑底彩字，可以節省能源，也可以避免眼睛受到強烈螢幕光線刺激。但是用於解說或簡報時，黑色背景中，低對比彩色的文字內容很難看得清楚，所以我們這裡要調整一下背景色彩。當然，我們工作時，還是可以設定黑色為底。

此處我們要修改色彩背景主題為明亮對比，淺黃色系的 Solarized Light。在圖 9 起始畫面右方 Customize 所列的項目中下方，紅色框起來的位置，找到 Color theme。點進去後，在下拉式選單中選擇 Solarized Light，就可以把 VSC 的背景和文字顏色調成如圖 10 的畫面。

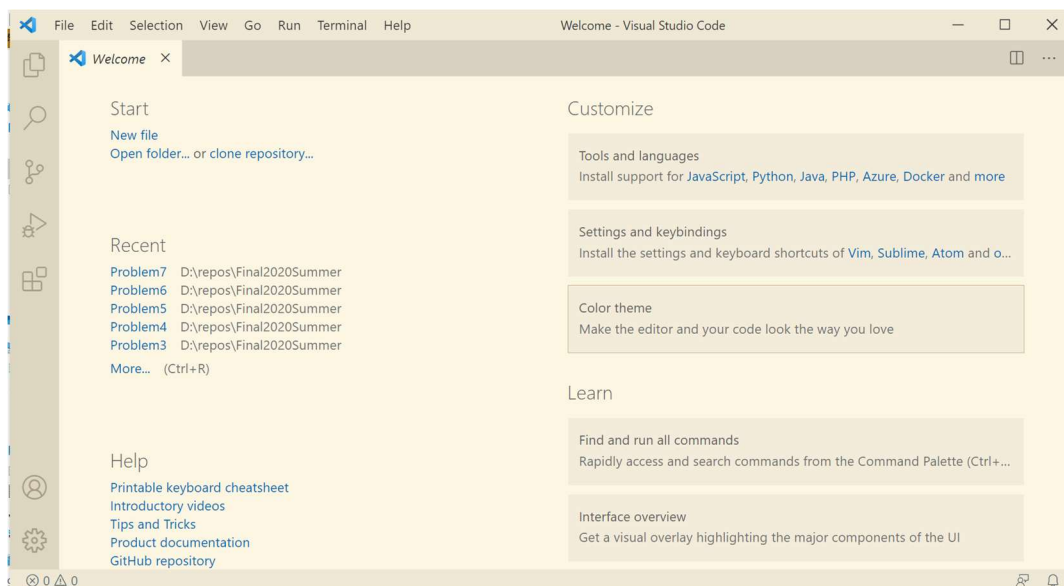


圖 10. VSC 在 Solarized Light 彩色背景主題下的起始畫面

4. 安裝 C# 語言 Extension

接下來我們要在 VSC 中，安裝針對 C# 的語言延伸。選擇如圖 10 起始畫面，右邊 Customize 中的 Tools and languages。在如圖 11 出現的選項列表中，選 C# for Visual Studio Code。按下同一圖中紅框所示的 Install 按鈕，電腦隨即安裝 C# 語言延伸，並顯示如圖 12 的說明。

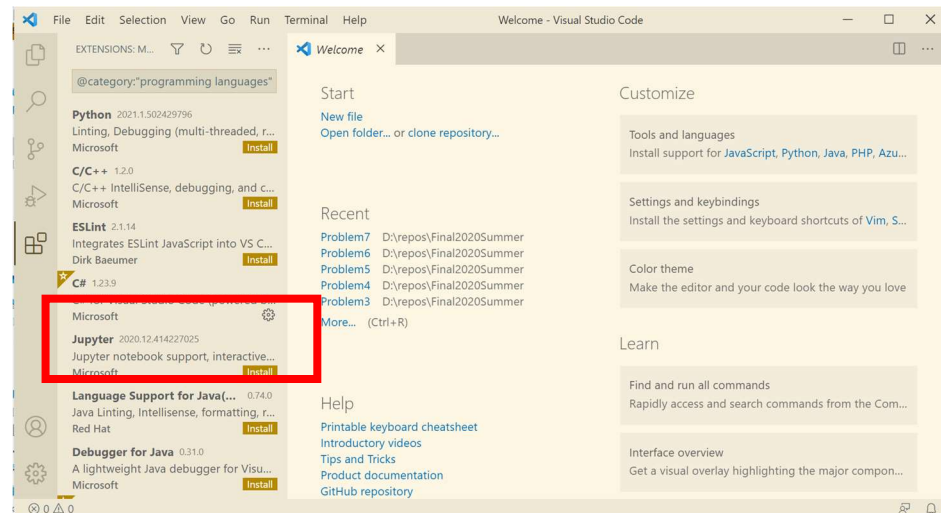


圖 11. VSC 語言工具延伸列表，以及 C# 語言延伸選項

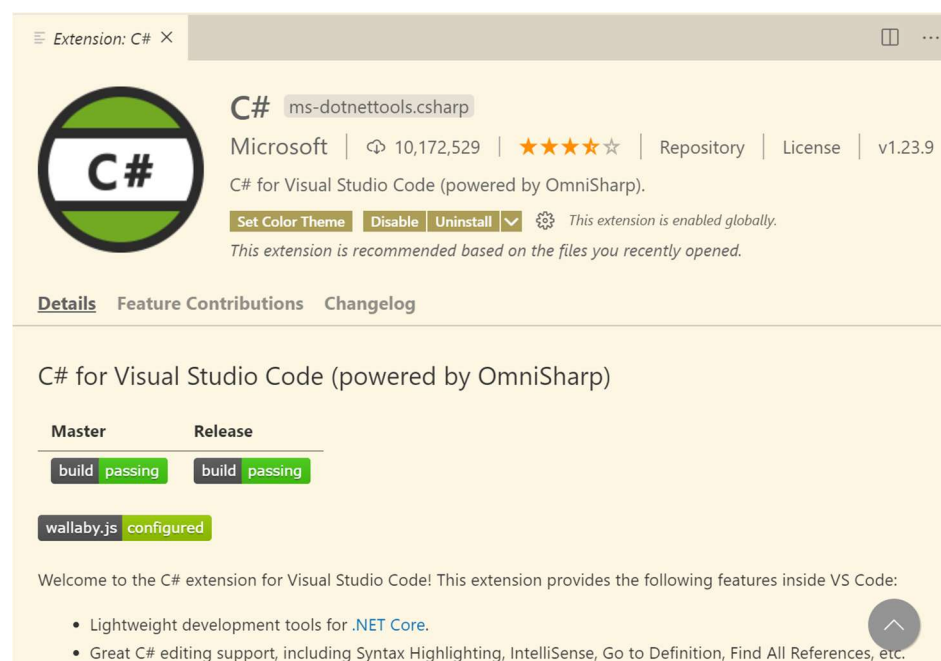


圖 12. VSC 中，C# 語言延伸的說明畫面

5. 建立 C# 專案

終於要開始使用 Visual Studio Code 了。一套彼此相關的程式及檔案，構成一個專案。專案的建立，如同許多 VSC 的應用，需要在終端機(terminal)介面，輸入文字指令完成。

在 VSC 起始畫面上方的選單(Menu)選取 View，由展開的選項列表選擇 Terminal，畫面右方就會出現可以鍵入指令(command)，以及顯示電腦回應訊息的區域，如圖 13 所示。

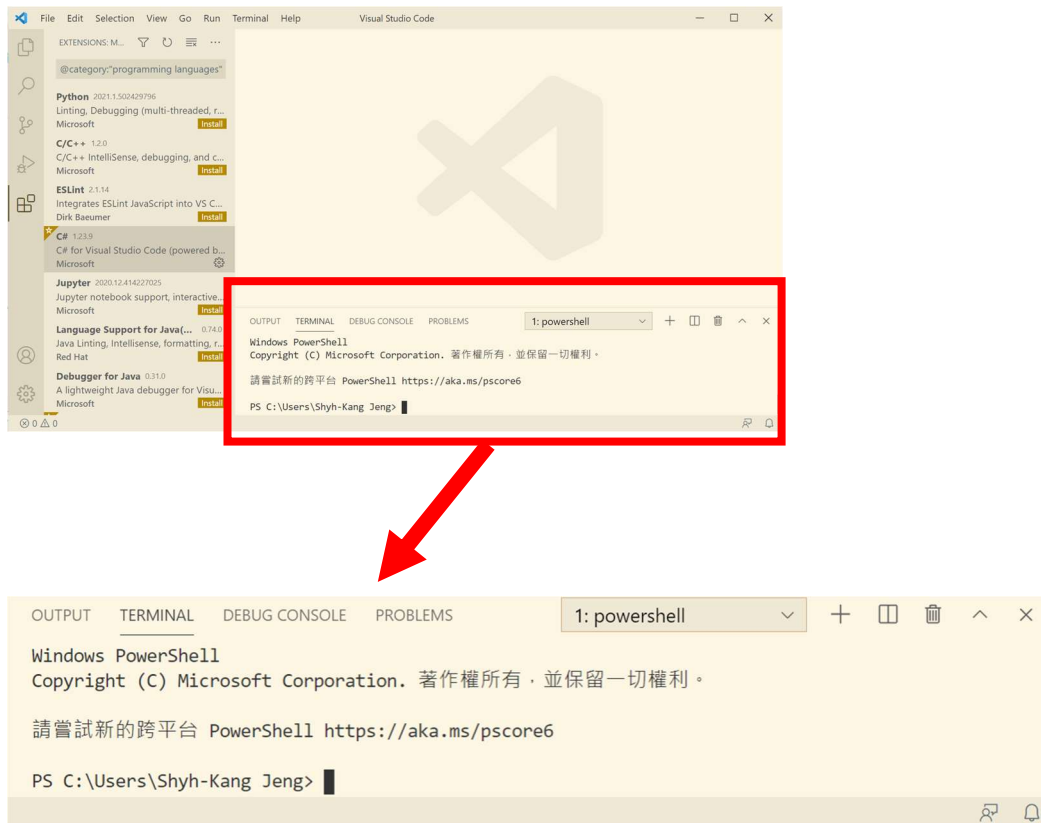


圖 13. VSC 畫面右下方出現的終端機(terminal)區域

接下來我們希望建立一個 C# 程式專案：放置此專案的檔案夾路徑假定已建好為 D:\2020Spring\VSCodeProjects\Chapter2。

在圖 13 中的黑色小方塊是游標(cursor)，在它的後方輸入文字指令
dotnet new console -o D:\2020Spring\VSCodeProjects\Chapter2\Hello

如圖 14。同一圖也顯示電腦的回應，看到

The template "Console Application" was created successfully.

以及

Restore succeeded.

就表示專案已經成功建立。



```
OUTPUT  TERMINAL  DEBUG CONSOLE  PROBLEMS  1: powershell
請嘗試新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\Shyh-Kang Jeng> dotnet new console -o D:\2021Spring\VSCodeProjects\Chapter2\Hello
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on D:\2021Spring\VSCodeProjects\Chapter2\Hello\Hello.csproj...
正在判斷要還原的專案...
已還原 D:\2021Spring\VSCodeProjects\Chapter2\Hello\Hello.csproj (158 ms 內)。
Restore succeeded.

PS C:\Users\Shyh-Kang Jeng>
```

圖 14. 在 Terminal 輸入建立專案的指令

上述指令中，dotnet 代表後面是指揮 .NET 系統的命令。new console 表示我們要建立一個新的主控台(console)程式專案。

-o C:\2020Spring\VSCodeProjects\Chapter2\Hello

則要求產生一個檔案路徑

C:\2020Spring\VSCodeProjects\Chapter2\Hello

對應一個專案所在的新檔案夾 Hello，同時也將專案命名為 Hello。

隨後在 VSC 畫面上方的選單(Menu)選取 File，由展開的選項列表選擇 Open folder，再從開啟的檔案選擇視窗，點選剛剛產生的專案檔案夾 Hello。此時圖 15 VSC 畫面左邊紅色框中，出現對應圖 8 A 的 Activity Bar 和 B 部分顯示的 Side Bar。Activity Bar 包含數個工作按鈕圖示，Side Bar 則對應 Explorer，列出專案所包括的檔案名稱。同時，圖 15 右下方，在另一個紅框區域，常會出現一個小視窗，顯示

Required assets to build and debug are missing from 'Hello'. Add them?

表示需要為專案加入一些內容，以便建置(build)及偵錯(debug)。我們的答覆當然是 Yes，所以按下 Yes 按鈕。

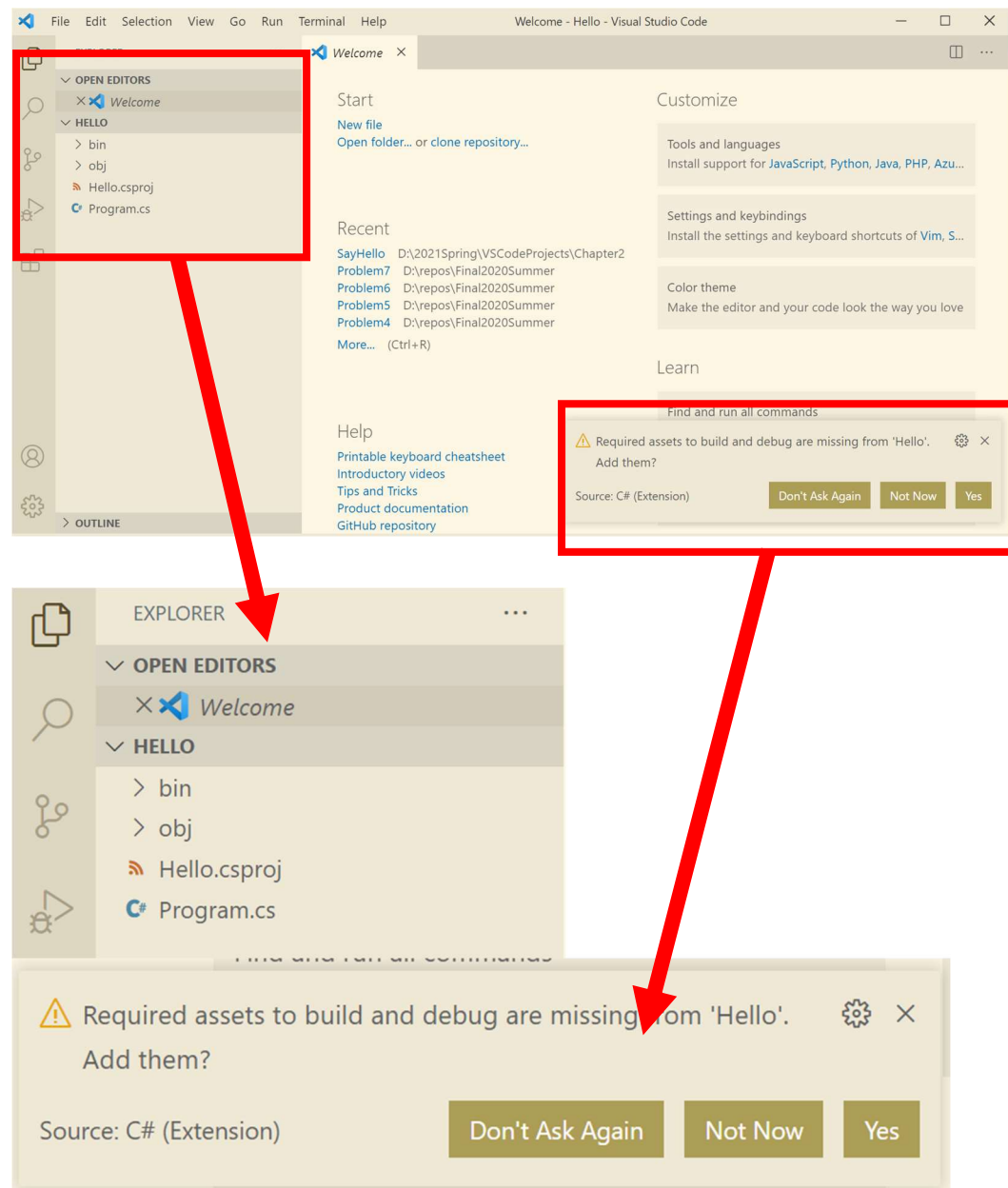
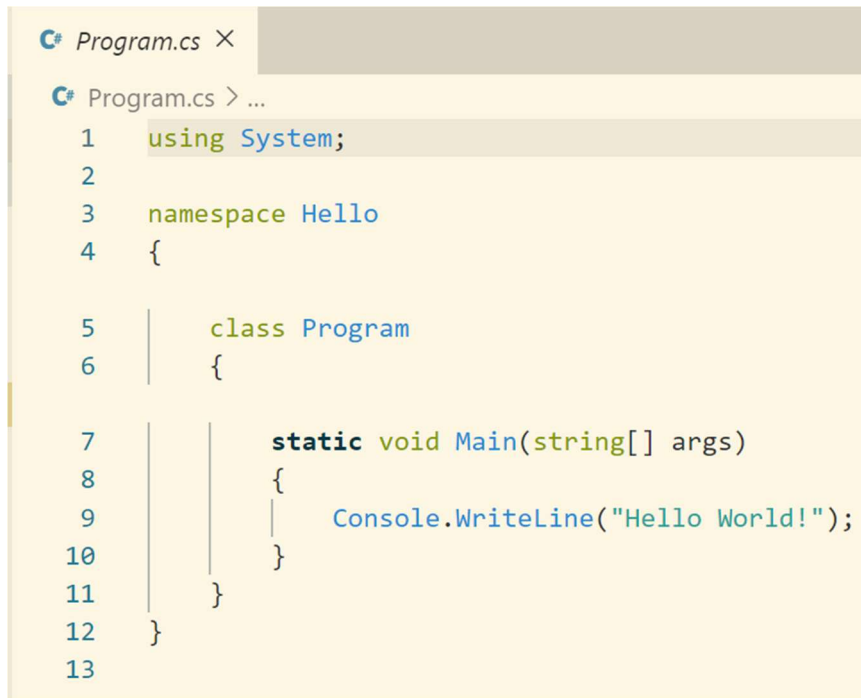


圖 15. VSC 專案開啟後的工作畫面

專案中已經預設一個稱為 Program.cs 的 C# 程式。由上方選單選取 File，再從展開的選項列表選擇 Open file...，打開程式碼檔案。另一個開啟檔案的方式，就是以滑鼠直接雙按 Explorer 中的 Program.cs 檔案名稱。程式碼檔案開啟後，VSC 畫面右方便顯示出如圖 16 的一個完整 C# 程式，這便是我們課程的第一支範例應用程式，將在課程中解說。



```
1  using System;
2
3  namespace Hello
4  {
5
6      class Program
7      {
8          static void Main(string[] args)
9          {
10             Console.WriteLine("Hello World!");
11          }
12     }
13
```

圖 16. 第一支範例應用程式

6. 執行與儲存 C# 專案

前述程式已經備便，接著要執行專案程式。確認程式無誤後，儲存更新正確版本的專案。專案程式可以先建置(build)，產生 .NET 的共同中介語言碼，再執行。也可以要求 VSC 自行建置直接執行。

建置專案過程如下：在上方選單選擇 Terminal，由展開的選項中選取 Run Build Task...，繼續選擇 build。電腦接著開始建置，並且在終端機介面，顯示如圖 17 的成功訊息。



```
1: Task - build
> Executing task: C:\Program Files\dotnet\dotnet.exe build D:\2021Spring\VSCodeP
rojects\Chapter2\Hello\Hello.csproj /property:GenerateFullPaths=true /consolelog
gerparameters:NoSummary <

Microsoft (R) Build Engine for .NET 16.8.3+39993bd9d 版
Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。

正在判斷要還原的專案...
所有專案都在最新狀態，可進行還原。
Hello -> D:\2021Spring\VSCodeProjects\Chapter2\Hello\bin\Debug\net5.0\Hello.dll

Terminal will be reused by tasks, press any key to close it.
```

圖 17. 專案程式建置成功的訊息

建置過後，點選上方選單中的 **Debug** 選項，接著由其選項列表選擇 **Run Without Debugging**，就可以看到右下方的區域標籤由 **TERMINAL** 變成 **DEBUG CONSOLE**，並且如圖 18，出現

Hello World!

字樣，這便是我們第一個範例程式的輸出。也可以在 **Terminal** 介面，按鍵盤上的 **Enter** 鍵，在游標後鍵入 **dotnet run** 指令，程式便又執行一次，在 **TERMINAL** 區域顯示輸出內容 **Hello World!**。

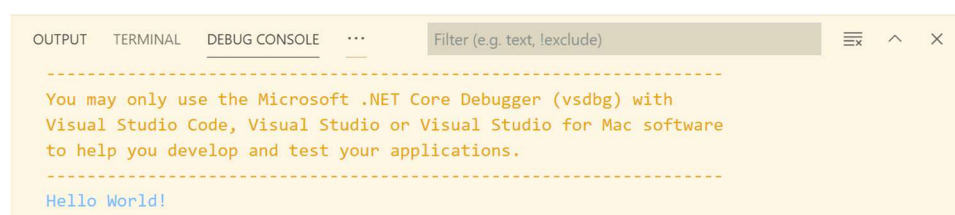


圖 18. 執行 Hello 專案中，程式 Program.cs 得到的輸出

在這同時，如果我們希望程式輸出到另外一個獨立的主控制台視窗，以便調整位置大小，可以用以下方法設定：在畫面左方 **Explorer** 中，點按 **.vscode**，出現兩個檔案：**launch.json** 及 **tasks.json**，其中儲存程式專案相關的程式執行設定資料。雙按打開 **launch.json**，把其中的 **"console" : "internalConsole"** 改成 **"console" : "externalTerminal"**，如圖 19 所示；接著按下 **Add Configuration** 按鈕，並且以 **File** 選單中的 **Save all** 儲存。關閉 **launch.json**。

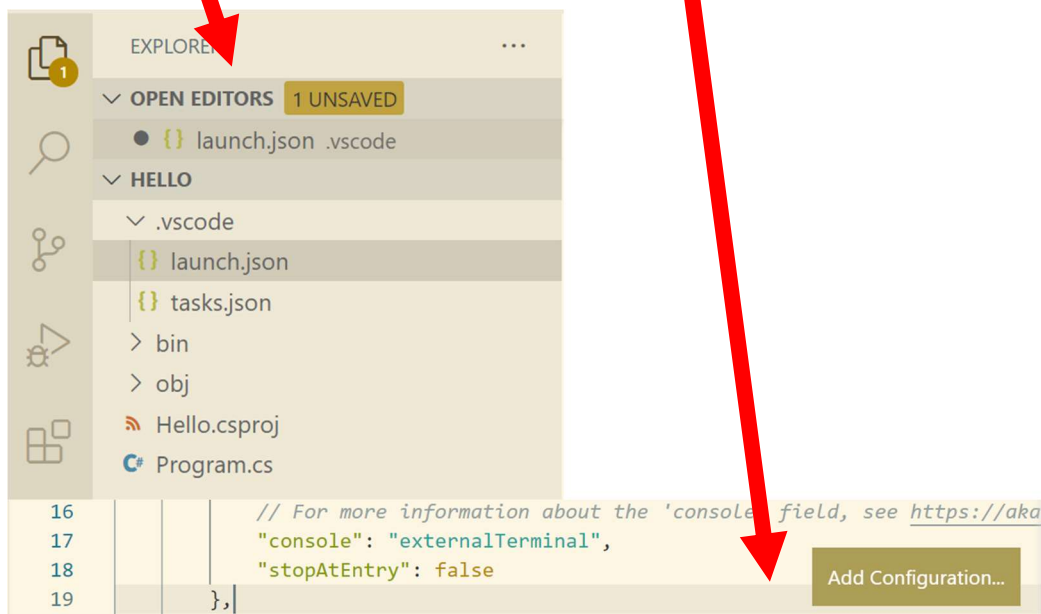
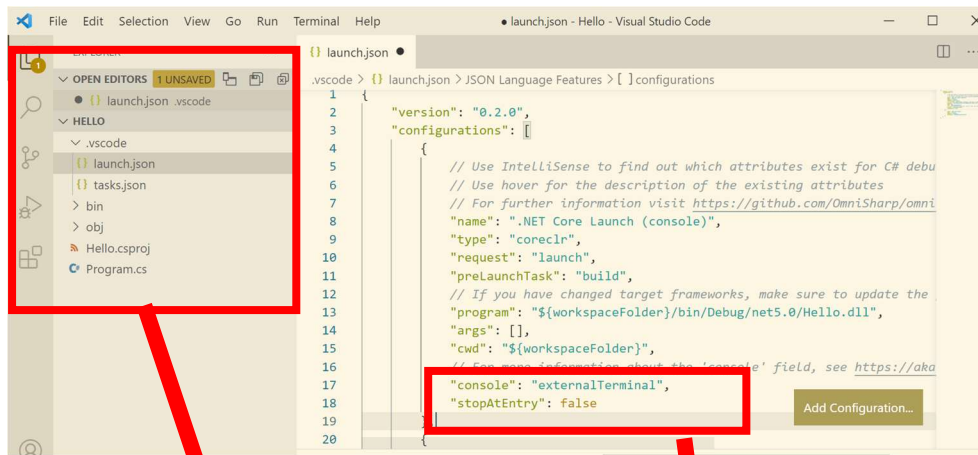


圖 19. 主控台螢幕改為外界視窗

接下來可以再執行程式，發現有一個黑影在眼前一閃而過。這是因為程式結束時，同時也關閉了外界主控台視窗。我們可以在程式碼最後加上如下敘述(意義在課程開始時便會說明)：

```
// ending the program
Console.WriteLine();
Console.WriteLine("按 enter/return 鍵結束");
Console.ReadLine();
```

注意不要打錯字。重新執行程式後，電腦便出現如圖 20 的黑底主控台視窗，按 enter 鍵後結束程式執行及主控台視窗顯示。同樣，黑底文字不利於解說。所以

我們在結束程式前，將滑鼠移到主控台視窗左上角，按住滑鼠左鍵，向右向下，選取同時反白主控台視窗有文字呈現的部分，一起按 `ctrl – Alt – PrtScn` 三個鍵，主控台視窗便被輸入視窗系統的剪貼工作記憶中，可以貼入 Power Point 或 Word 檔內，再利用圖形工具修剪大小如圖 21。當然，也可以使用 Power Point 或 Word 內的螢幕擷取畫面功能，直接取得主控台視窗畫面。

程式告一段落後，可以由 VSC 上方選單的 File 選項，依次執行 Save all、Close folder、Exit，儲存完此次的變動後、離開 Visual Studio Code。



圖 20. 執行 Hello 專案，程式 Program.cs 得到的外界主控台視窗輸出



圖 21. 反白剪貼後修剪過的主控台視窗畫面

7. 線上參考資源

更多的 VSC 資訊，包括一些視訊短片，可以參考前面提到的 Visual Studio Code 說明文件網址: <https://code.visualstudio.com/docs>

關於 C# ，則可以參見 微軟 C# 文件
<https://docs.microsoft.com/zh-tw/dotnet/csharp/>

此外，以前修課的學長姐推薦：用 C#, XXX 做為關鍵字 (XXX 代表想問的問題)，到 Google 搜尋，通常就能獲得解答。當然，向助教請教，也是不錯的方法。