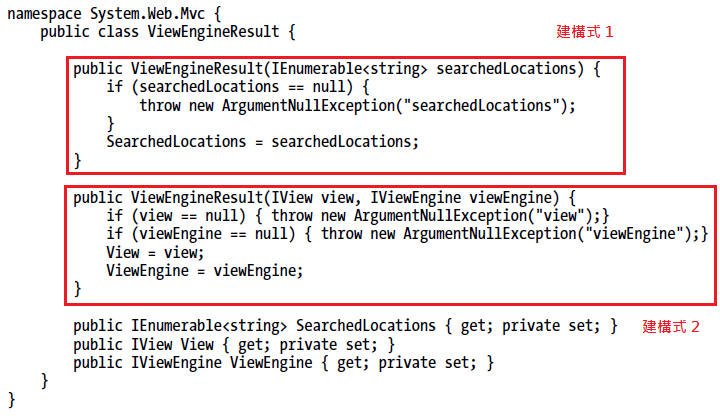
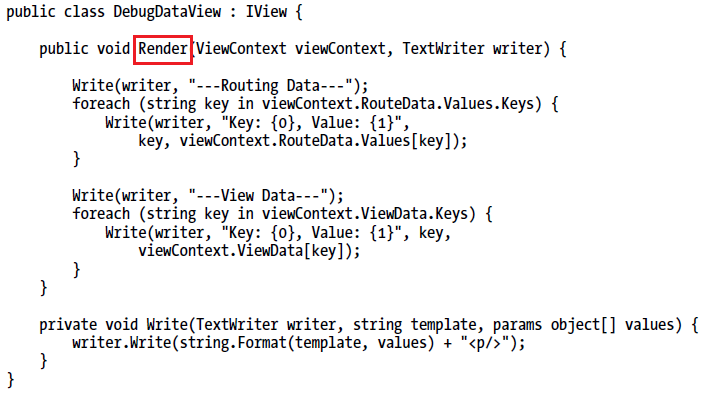
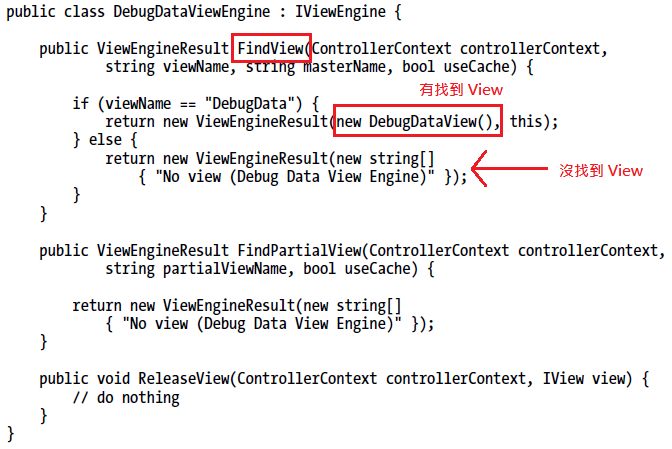
1. **建立自定義的View Engine**
   1. View Engine的角色就是把對於View的請求，轉化為ViewEngineResult物件，如下



* 1. 最主要的物件是IViewEngine以及IView
     1. IView：負責實際輸出HTML內容到頁面上的物件



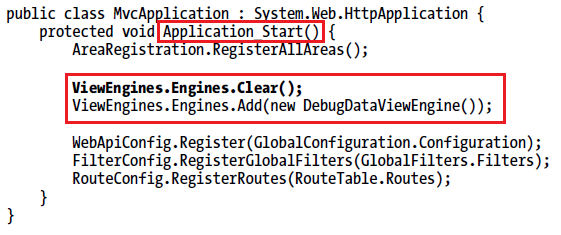
* + 1. IViewEngine：把View的請求轉化為ViewEngineResult物件



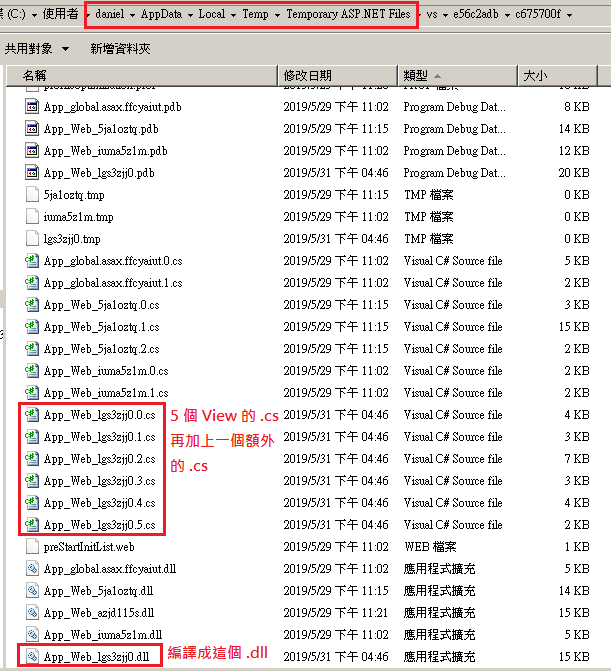
\* 有找到View：回傳ViewEngineResult的建構式2 (第一張圖)

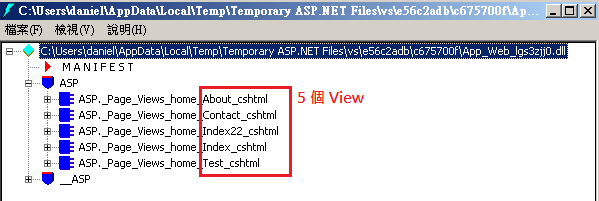
\* 沒找到View：回傳ViewEngineResult的建構式1

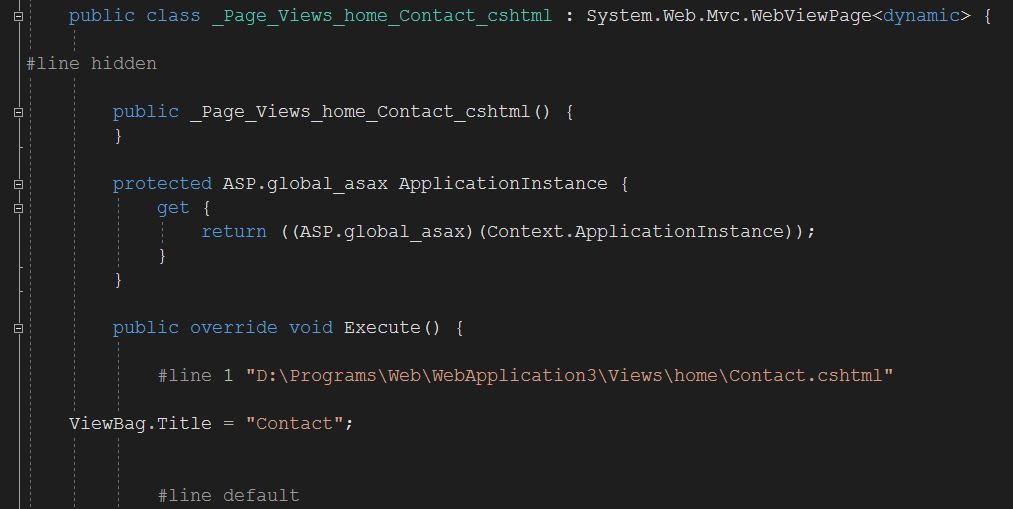
* 1. 到Global.asax註冊ViewEngine



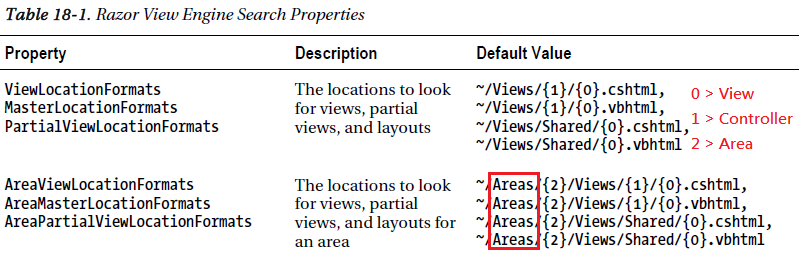
1. **使用Razor View Engine**
   1. Razor View Rendering
      1. View會被轉為C#類別，在編譯後輸出到以下路徑 (基本上是一個Controller底下的所有View被編譯為一個dll檔案，參考下面圖片)

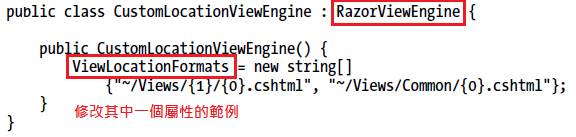






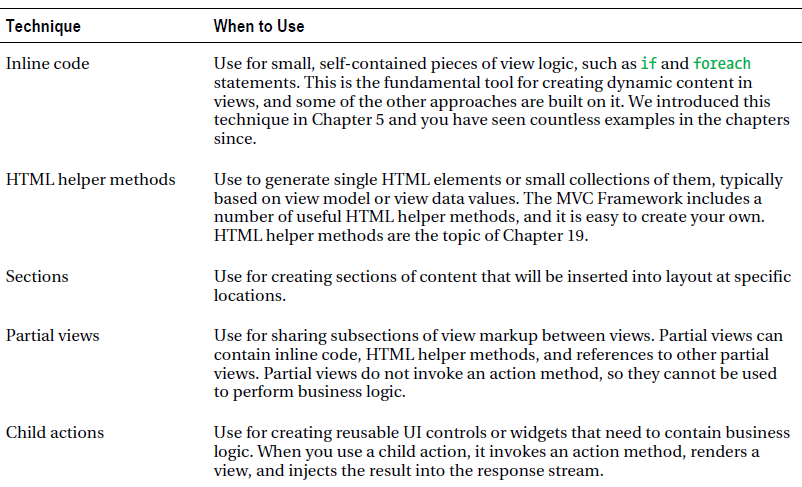
* 1. Configuring the View Search Locations
     1. 搜尋View的位置預設是以下的路徑，可以再更改屬性就好



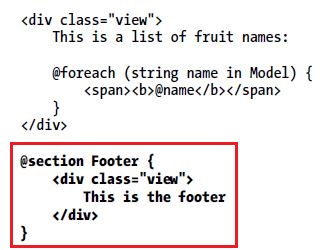




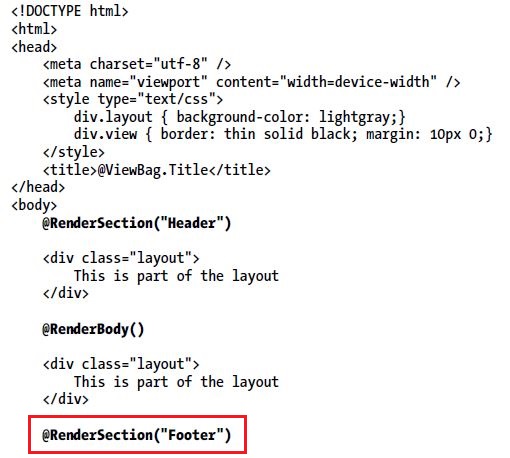
1. **Adding Dynamic Content to a Razor View**
   1. 預設有以下幾種方式可以在View裡面加入資料



* 1. Inline code：直接在View裡面加入C#程式碼
  2. Html Helper Methods：下一章會講到
  3. Section
     1. 用來輸出特定的內容到Layout的頁面上 (不是所有頁面都會共用的資料，若是會共用，應該放在Layout頁面就好)
     2. 先在View裡面自己定義一個或多個Section

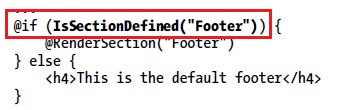


* + 1. 然後在Layout中決定是否要輸出Section



* + 1. 也可以透過以下其他方法來判斷是否要輸出Section

Section有被定義才輸出



也可以告知該Section不是必要項，有定義就自動輸出，沒有定義就不會輸出

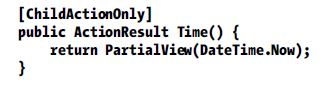
4.JPG

* 1. Partial View
     1. 可以把很多頁面共用的部分抽離出來，放到Partial View，例如Footer
     2. 跟一般的View一樣，可以在Partial View設定Model的型別 (下面範例傳入字串陣列)
     3. 透過以下方式來輸出Partial View

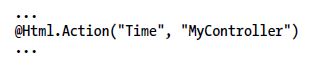


* 1. Child Action
     1. 可以在View裡面主動呼叫某個Action Method (View載入時執行一次，跟Ajax不一樣)，而Method可以是別的Controller裡面的Method (還不確定什麼情境下會需要用到這個做法去解決什麼問題)

Method上面加上Attribute，讓該Method無法直接透過瀏覽器來請求



呼叫其他Controller裡面的Child Action



或是傳遞資料到Method裡面也可以

3.JPG