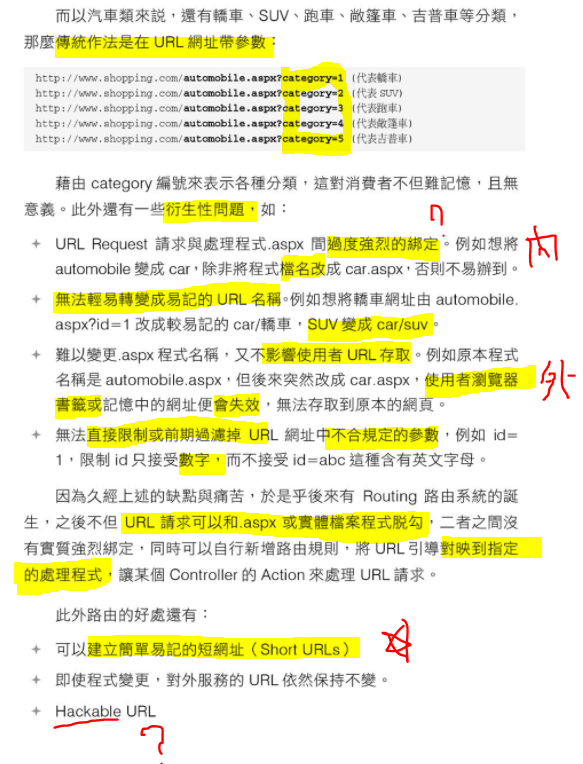
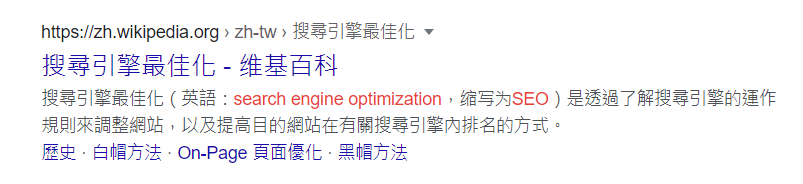
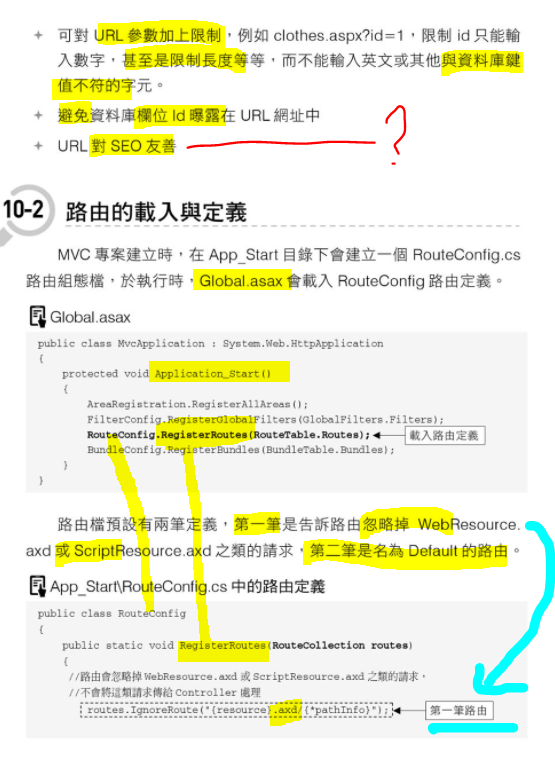
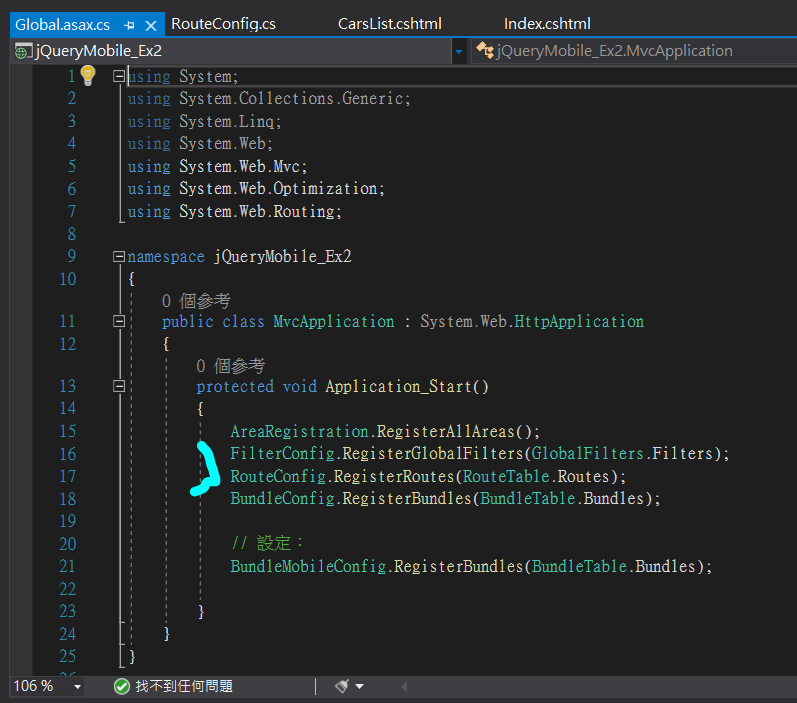
原因和優勢



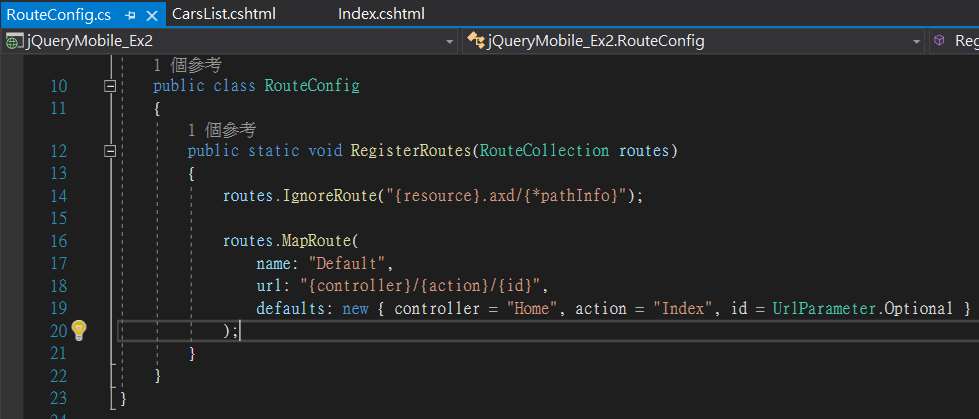


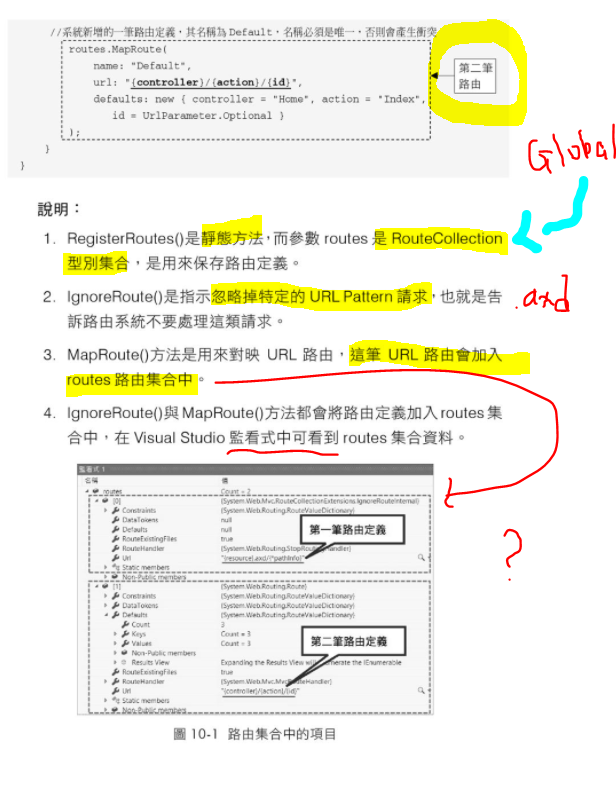


基本解析





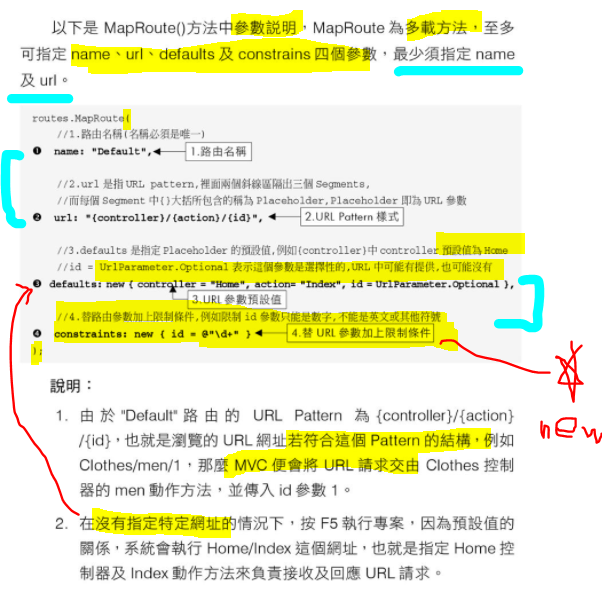




where? how?

要中斷點？

這個集合是從哪來的？該Object呢？

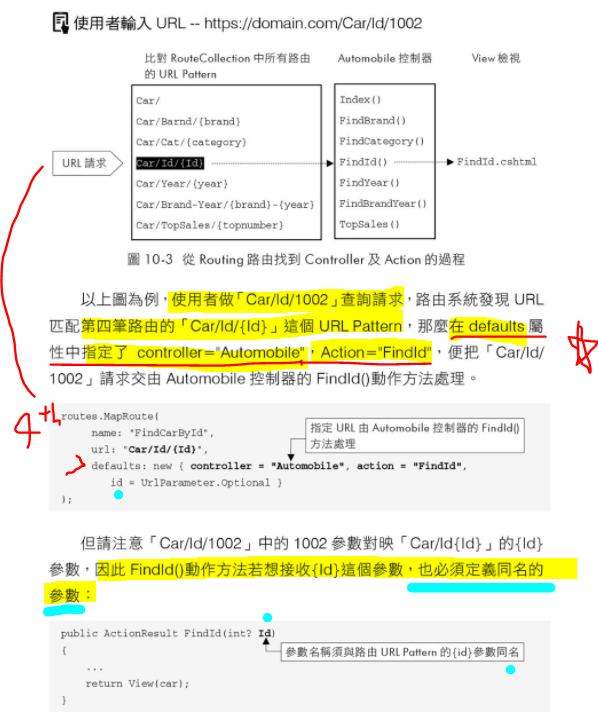
注意︰

##### url 和 defaults 的分開，這個就是重點！

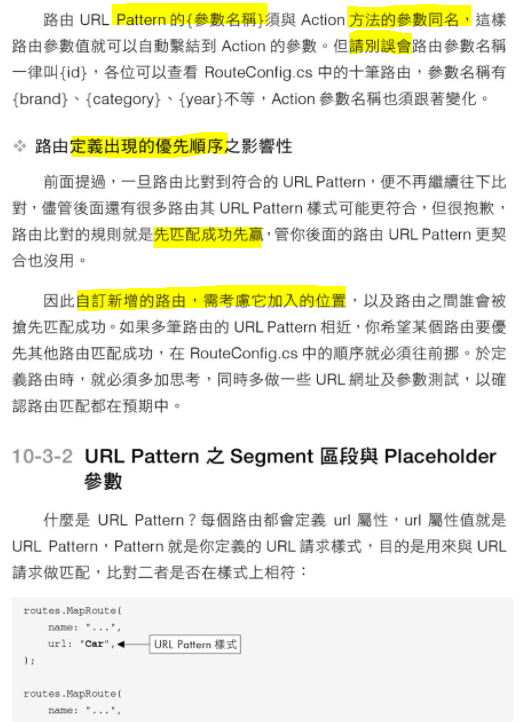
及

##### 如何下參數格式限制

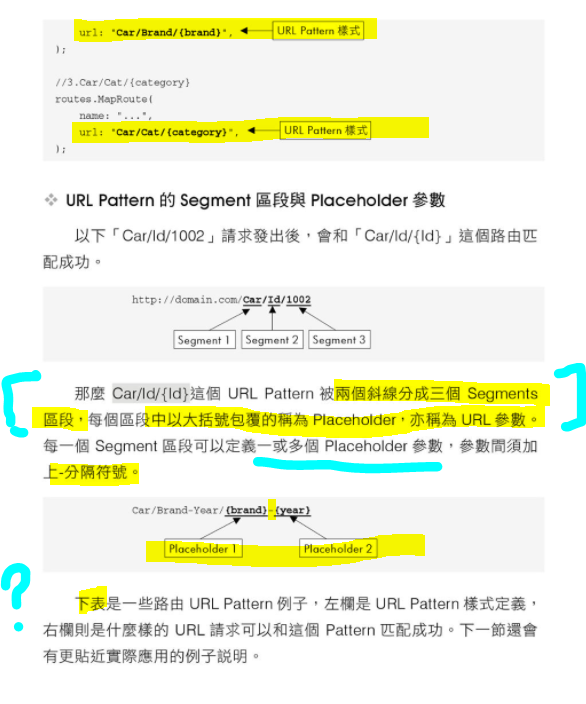
對比流程

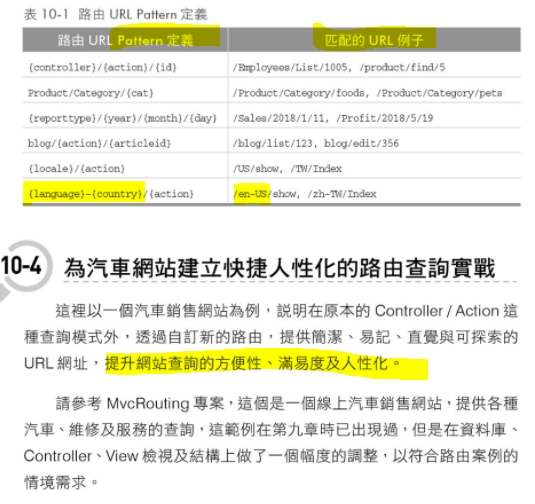


##### Action接收參數的命名限制



Segment 和 Placeholder

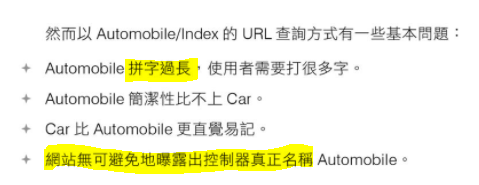




實作︰

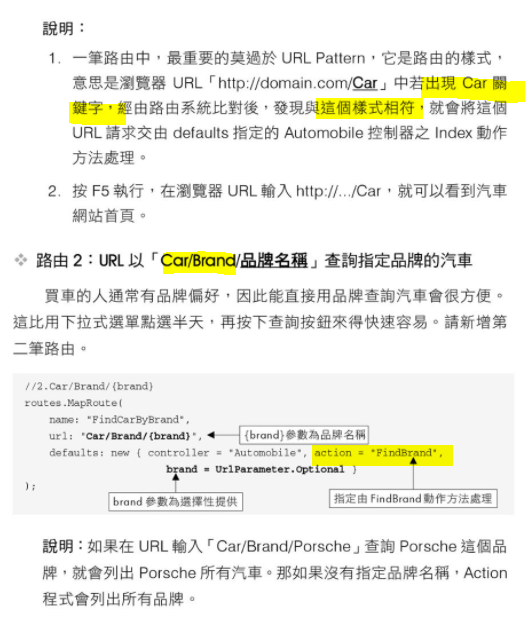
在上一章的實作上修改。

## 預設的局限



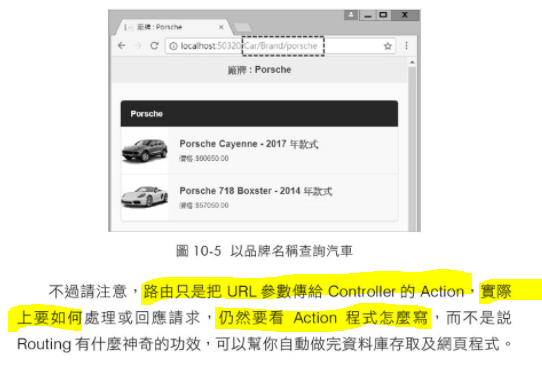
## 什麼是理想的URL

# 預設和自設的對比！

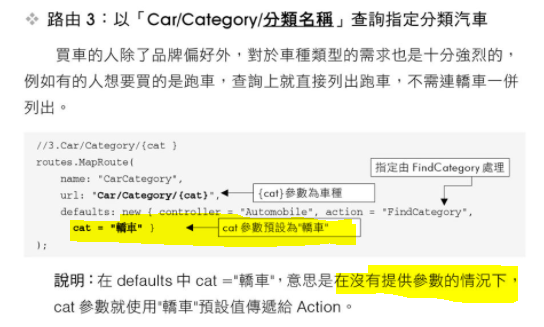


輸入參數作查詢

### 

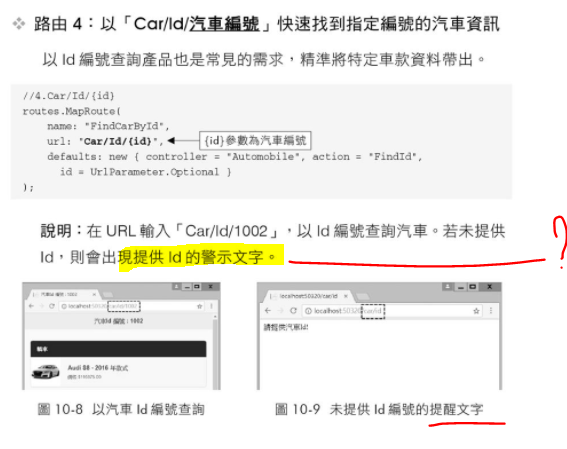


## 預設參數

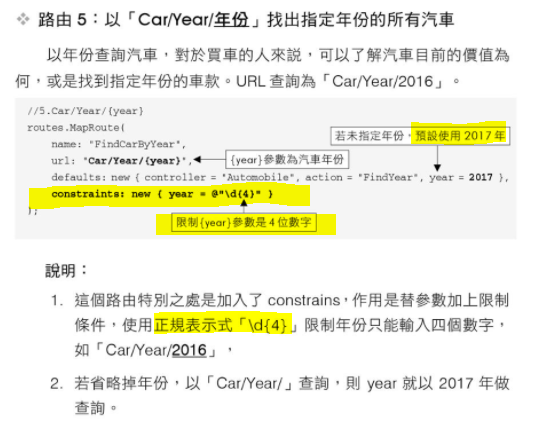


### 

## 無輸入警示文字

從何而來？來自Action中的判斷嗎？對，稍後會有說明。

## 格式限制



## 多參數及其格式限制

有格式限制就不可以不填？不，RegEx 是可以設定，不填也通過…

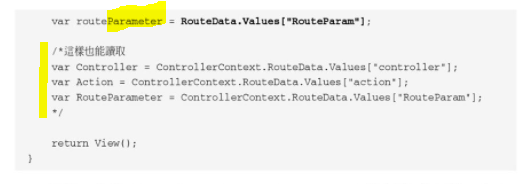
但要有預設的話，也可以在 defaults 的後方設置

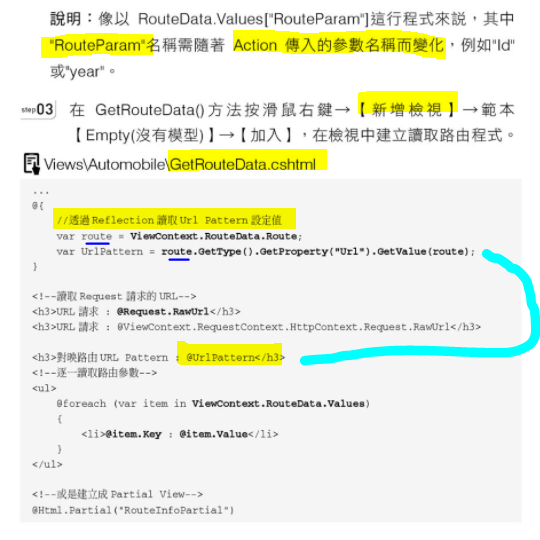
## 參數限範圍

原理一樣

Controller和View倒回去讀URL的資料

# 實作一，建路由︰

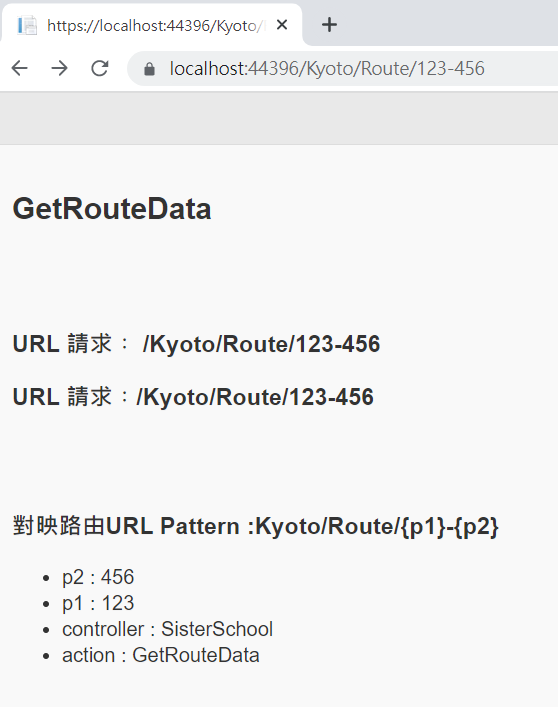


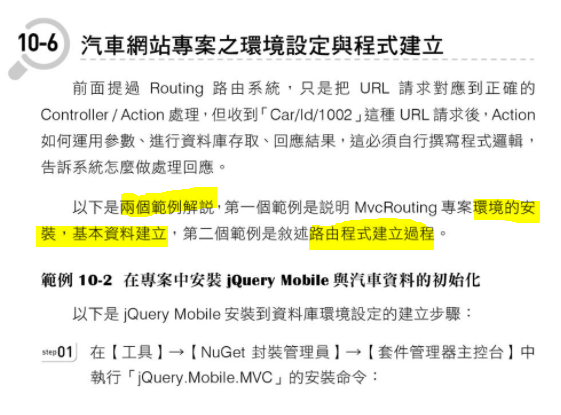




要溫習在Action中，呼叫PartialView 是會產生什麼後果？

我只記得前者(@Html.Partial())是鑲在View中，還可以傳入參數。

實作成功，但我發現多個參數的話，好像就會變成必填了，因為沒有了{p1}-{p2}，會讓路由找不到。



呃…我承上一章的實作，所以這個應該可以省了吧？

# 實作二，設環境︰

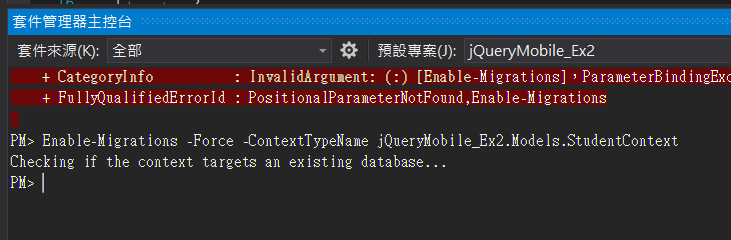


上章本來偷懶，省了這一步，現在正好，可以建個比較多欄位的…

這個[DatabaseGenerated(DatabaseGenerateOption.None)]是什麼意思，不加會怎樣？



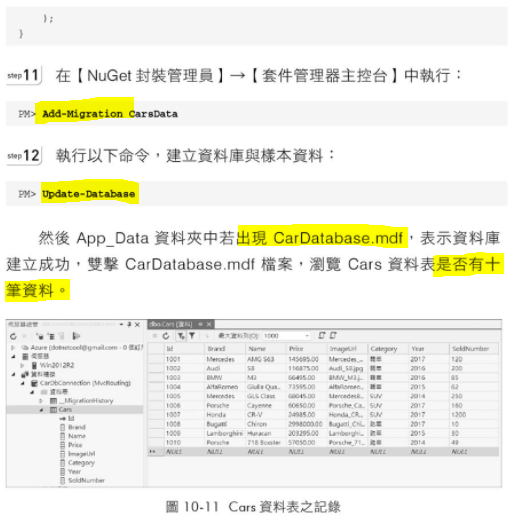


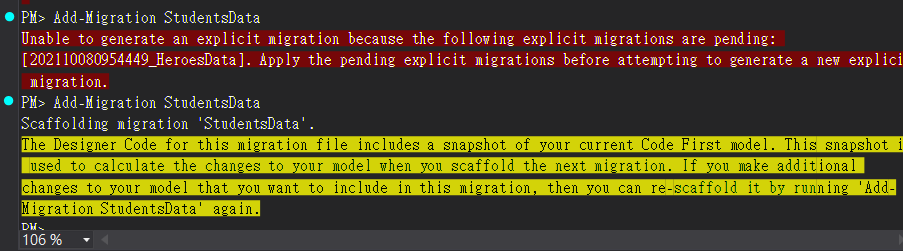


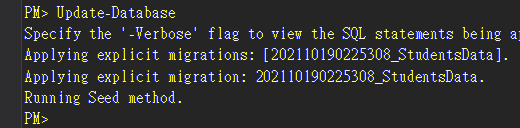
如果在同一個專案，要建立多於一個的資料庫，

第9個步驟要輸入以上的指令，洗去之前的 Configuration.cs

第一次建立的那個資料庫會還在的，不用擔心。



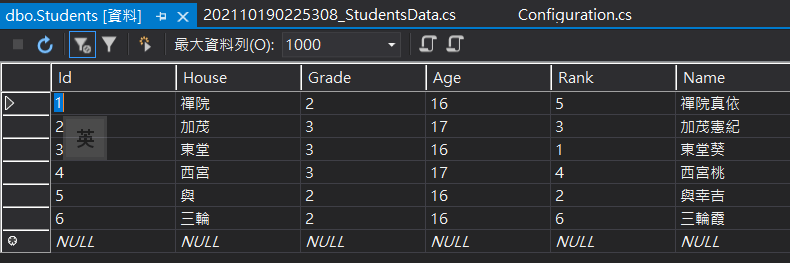




我先把之前的那個預備指令檔(上一個實作，就是那個有年月日時分秒的檔案)刪了，

才可以建到新的。

反正我覺得那個檔案，在資料成功進入資料庫後也不重要。



建立成功



# 實作三，建MVC︰



## 路由一﹑二



原來訊息果然是由Action判斷的(回應之前的疑問)

看到這裡，我突然覺得不能承續，而是要複製一個專案來修改。

因為上一章的實作例子，我都標明了檔案的名稱。

所以路由的修改…咦？

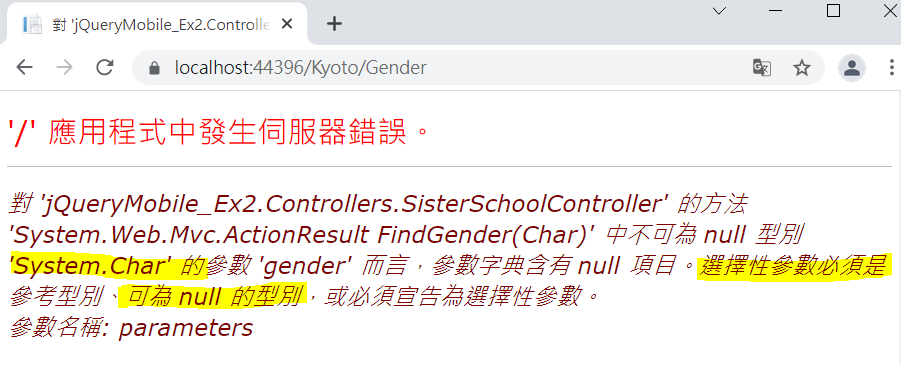
不對，開了個新的控制器，而且定了和之前都不同的 url pattern (/Car/)的話，

就不會影響到之前的實作的路徑了。

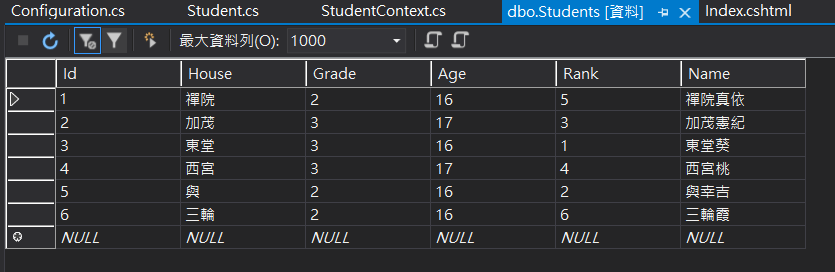
沿用 jQueryMobile\_Ex2 專案…

我改編為Gender來取代品牌，但原來只有string可以有null，char沒有

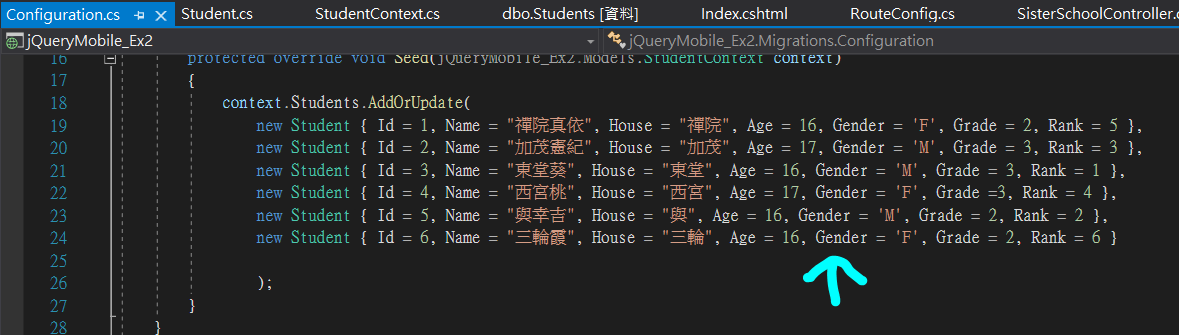




咦？我還發現char有更奇怪的地方︰

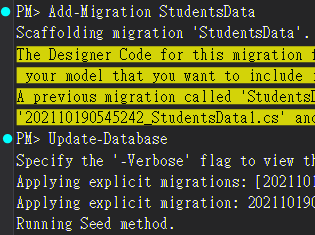


細看下，沒有Gender這個欄目？

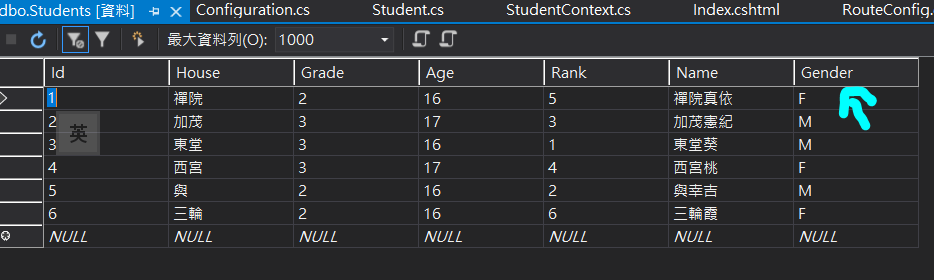


明明有設？

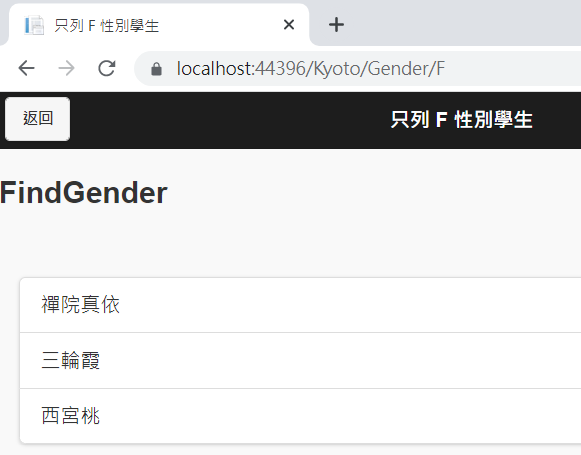
所以我把char 都改成了string，再跑一次下面兩句︰



這次有出來了︰

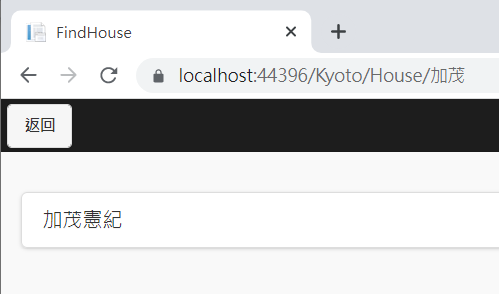
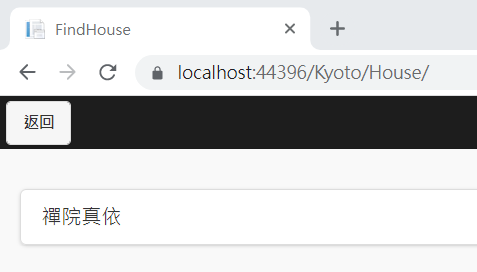


實作成功︰

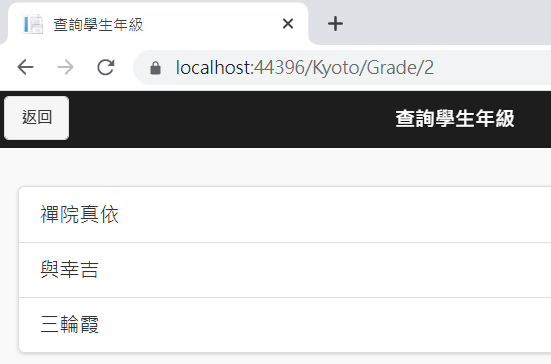


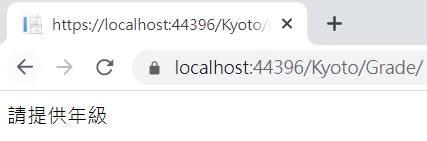
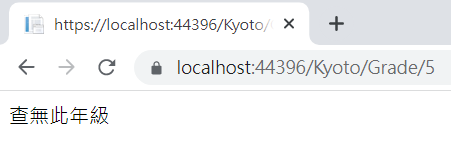
## 路由三

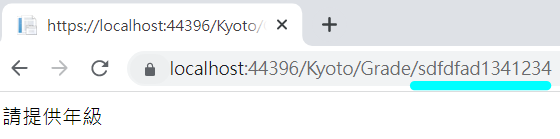


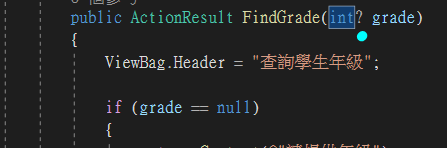


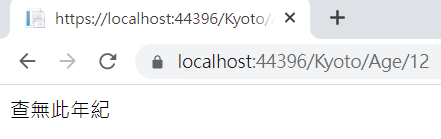
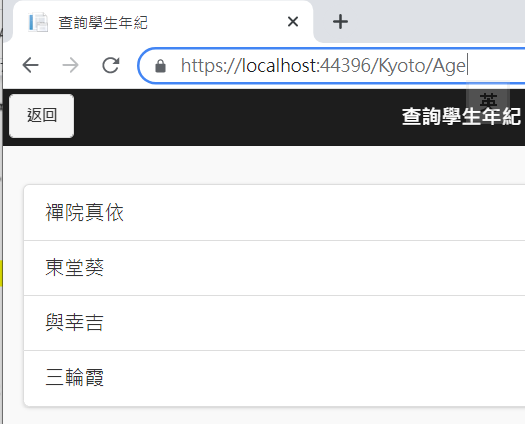
## 路由四﹑五

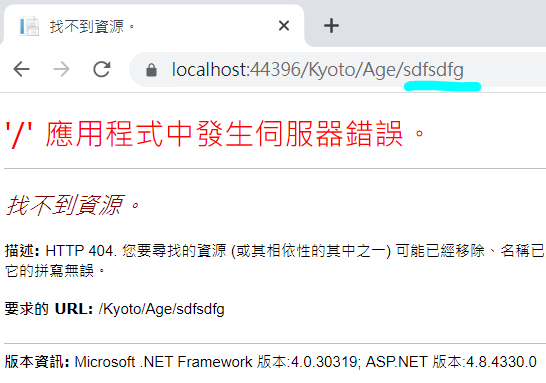


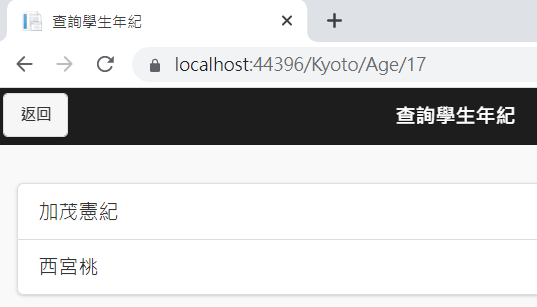




int 的 null 記得問號

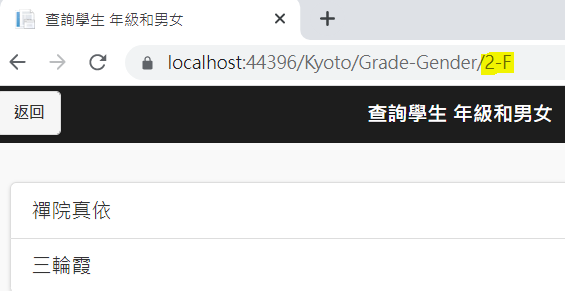


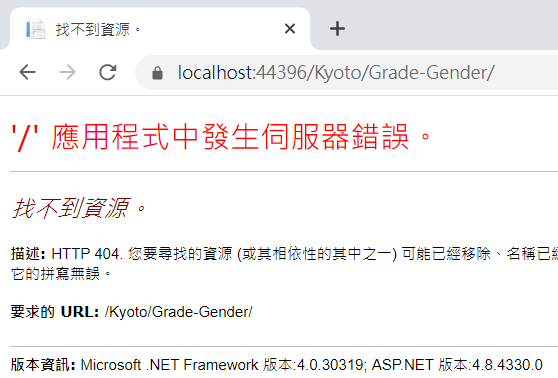
<-- 錯格式的效果



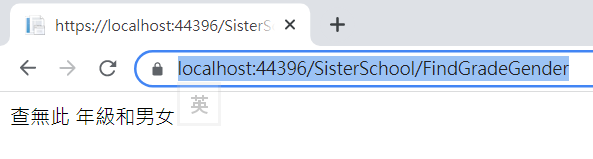
## 路由六﹑七

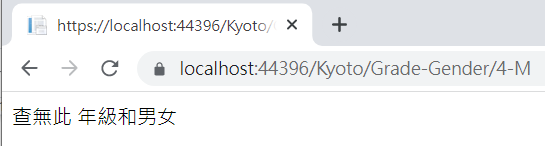
路由X這個部份，就是運用LinQ的方法大全吧…



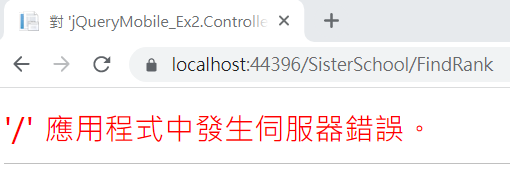
不填，也無預設，會Error。

但是由預路由的話，就還是會進入到方法，以參數為 null 進行搜尋︰

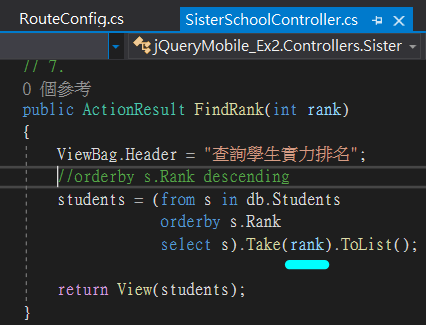




-------



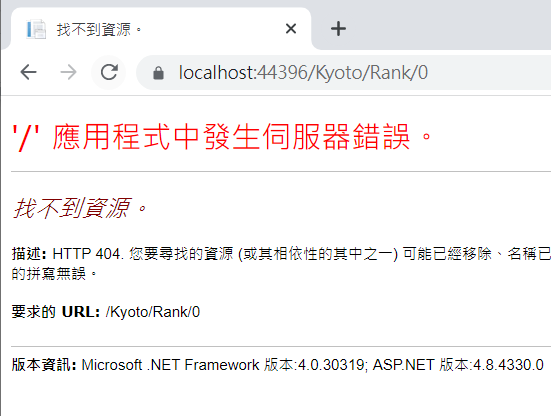
實力排名不能由預設的路口進入，因為會沒有輸入到參數給︰





不填=預設的實力頭三名



限了1-6



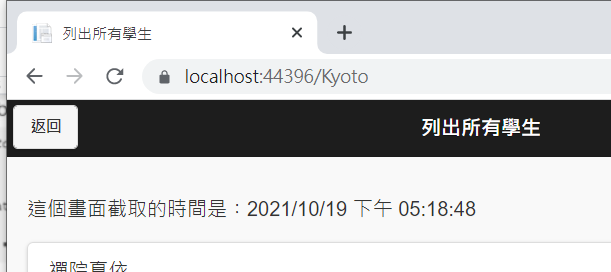
OutPutCache

MVC 的 Cache 十分方便

### 基本設定與如何觀測



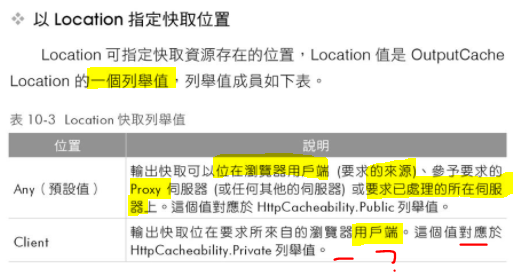
這只是一種觀察方法，不用太執著，最重要的是那個[ ] Action上方的屬性。

我沒有實作第二個方法，因為原理一樣。只是寫法不同。那個屬性已經擋下了60秒之內，任何人的請求了，自然不會跑入Action



### 進階屬性





#### Location



#### VarByParam

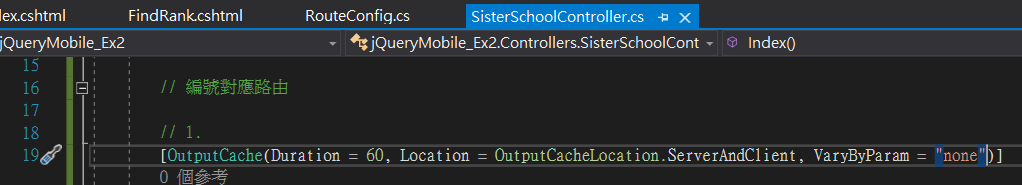


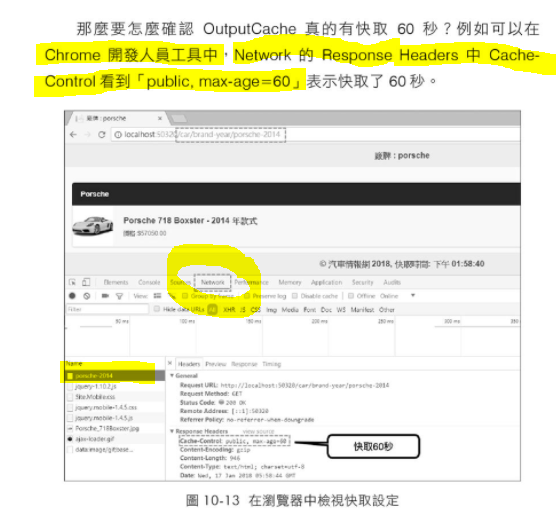


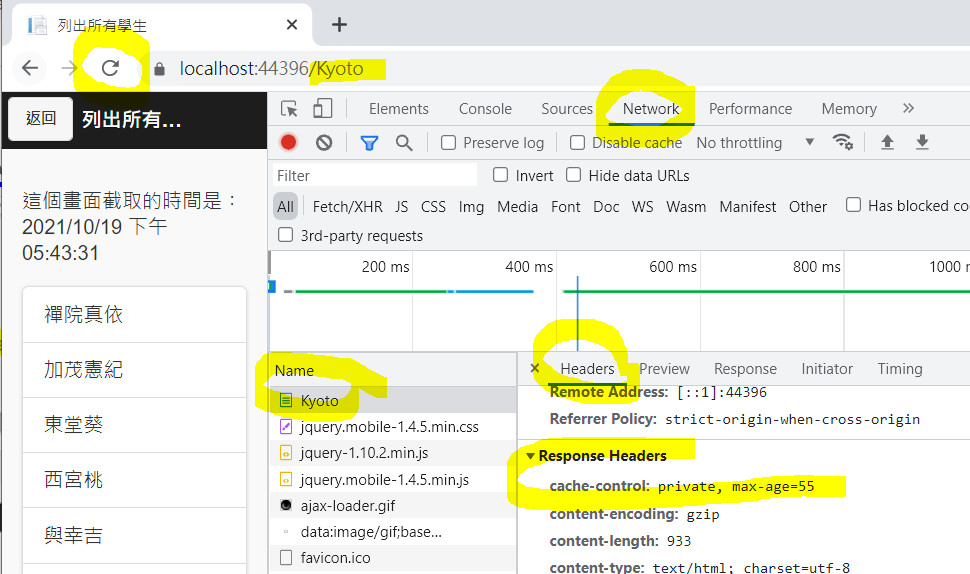
以上的兩個進階，我都是單寫，但沒有深究。只是知道有而已…

第一個進階，我是沒有另外一個Location的伺服器，也沒有Proxy 之類，寫了也沒辨法測。

第二個進階，是因為用了MVC，基本上有了路由，QueryString就會被淘汰了？所以看過就好？實作來是幹什麼？







我在第5秒更新，所以max-age = 55。max-age 是倒數時間。

Yeah，又搞定一章。

