看過E-book 後，看恒毅的教材，整理出來可以參考的東西。

恒毅教材的第一部份是「理論」，很多PowerPoint的關係圖，

內容大致和E-book差不多，看不懂的地方也沒有解釋，不想深究。

第二部份是實作(佔8成)，我分開了顏色，

沒有紅色的，可以不特別注意(因為Ebook那邊已經涵蓋)。

有紅色的部份，我會查看實作，再製作筆記。

紙本部份，我就不掃描進電腦了，以免有版權(或內容太過冗長)的問題。

----------------------------------------------------------------------------------------

# 第一章︰

比較三種網站架構︰

Web Page (Razor3) -- 特色，使用Razor的 HTML Page <- 一閃而過，沒前沒後

Web Forms Site -- 特色，使用控制項 <- 一閃而過，沒前沒後

MVC -- 這個課程的主題，特色︰ Model-View-Controller

# 第二章︰

Model

01 - 建立基本Model, Controller, View

02 - DataAnnotations , DisplayNameFor

03 - DataAnnotations , 自訂驗證(未有效果顯示), Nullable參/變數, QueryString輸入Object

04 - DbContext , 資料庫連接，Seed ← (懷疑過時，還是跟Ebook那邊做就夠了。)

# 第三章︰

Controller

01 - RedirectToAction

02 - ViewData, ViewBag 使用上的分別

03 - Detail頁的機制，HttpStatusCodeResult( HttpStatusCode.BadRequest ), HttpNotFound()

View with Model (Linq in HTML Helper) , @HTML.ActionLink()

04 - Create頁的機制，HttpPost，if (ModelState.IsValid)，context存入，AddView > Create >

Reference script libraries (前端驗證) ， HTML.BeginFrom() ，

05 - 自訂驗證(效果顯示)，HTML.ValidationMessageFor() ，jquery的前端驗證套件

06 - ValidationSummary , 顯示錯誤的Css設計參考

07 - Edit頁機制，context.Entry( opera ).State = EntityState.Modified; ，

Html.HiddenFor (隱藏ID，但在Post回去的時候，會用得到)

08 - 刪除頁機制，[ HttpPost , ActionName(“”) ] 和 method 命名差別，onclick comfirm

09 - 用Scaffold 一次產生 CRUD

10 - 用Scaffold 一次產生 CRUD ← 勾 async

<https://blog.darkthread.net/blog/async-aspnet/>黑大解釋為什麼非同步可以加速效能

11 - System.Diagnostics ， Debug.WriteLine() ，

ActionFilterAttribute 繼承，[ 標記 ] Controller中的方法。

Filter攔RouteData來寫Log

<- (我覺得這個東東是用來在Debug的，在output的視窗紀綠用到了哪個方法，但是我們現在並不是寫大型，多分工，又複雜的公開線上網頁。所以Debug不致於這麼麻煩？所以也不實用。但我還是做了一下筆記。)

# 第四章︰

Razor

01 - HTML.Raw

02 - for loop

03 - foreach

04 - if , @:

05 - return File( path , format)，<video src =”Controller / Method”>，@Url.Action(“”)

06 - 查詢頁機制 - BeginForm

07 - Html.EditorFor 和 Model 之間的關係

08 - PartialView

# 第五章︰

01 - 自訂錯誤頁面

02 - 按不同HTML Error Code 來指定錯誤頁 (這個我WebFrom時就做過了，只是提醒)

# 第六章︰

01 - 自訂路由

02 - routes.MapMvcAttributeRoutes() 用 [] 來訂路由

03 - RouteData.Values 取路由資料

04 - Html.MvcSiteMap().Menu() 和 Html.MvcSiteMap().SiteMapPath()

# 第七章︰

01 - Layout

02 - Layout + Section

03 - Bootstrap的下載和引用

04 - Bootstrap之中的Grid使用(一行12格，除非寛小於某值，視乎md/xs設定)

05 - HTML表單 input type , 使用 Bootstrap之中的 class = “form-control/ -group/ inline”

06 - Bootstrap之中 不同 button 的式樣

07 - Navbar (在Ebook那邊研究過了)

# 第八章︰

01 - PartialView 和 ActionLink + @AJAX.ActionLink( )

02 - Ajax.BeginForm()

# 第九章︰

01 - JavaScript

02 - jquery

03 - jquery + css

04 - Request.IsAjaxRequest()

05 - Web Service (.asmx) + ajax

06 - jquery Datepicker

07 - jquery UI Accordion (和bootstrap 的navbar 類似，但是更好寫)

# 第十章︰

01 - Identity - 預設資料庫和系統

02 - 頁面的授權機制，使用[Attribute]

# 第十一章︰

01 - TempData + RedirectToAction

02 - Session

03 - StateServer + Session ← (在工作上不實用，因為我們沒有辨法改做伺服器的權力？)

04 - SQLSever + Session ← (在工作上不實用，目前沒有大規模用戶(至少數十人以上))

# 第十二章︰

01 - Web API - Get

02 - Web API - Get + Static List + 準備新增﹑刪除﹑修改(但未用)

03 - Web API - Get + Ajax + @Url.Content()

04 - Web API - Post + .serialize() : 新增

05 - Web API - Put(自訂方法名) : 修改

06 - Web API - Delete : 刪除

# 第十三章︰

01 - HttpApplication.Context.Respon.Redirect() 用QueryString 存取值並轉頁

(雖然沒有見過，但感覺更加的麻煩和不好用？意義何在？用來過濾？)

02 - HttpHandler 自訂 URL 副檔名 (也不是很實用的感覺…)

03 - WebSocket 聊天室的多人多視窗同步功能 !

# 第十四章︰

01 - 部署