
tw
MVC

Introduce twMVC

Who We Are and What We Want

Dino Wang

http://dinowang.blogspot.com

為何組成 twMVC

- 我們熱愛 Web 開發技術，並樂於分享 ASP.NET MVC 開發心得
- 已成功建立數個灘頭堡的經驗，深知 MVC 開發模式條理分明，有利於分工與管理，想推薦給尚未了解的開發人員
- 不願意看到 MVC 叫好不叫座，我們樂意透過經驗交換來破除導入障礙

~~當然也希望能趁機紅起來~~

twMVC 成員

- Bibby
- Demo
- Dino
- Jerry
- Kevin
- Wade

twMVC 資訊分享

- A quite simple http://mvc.tw/



- But, it’s still under-construction now ☺

twMVC 做哪些事推廣 ASP.NET MVC 技術

- 每週小型聚會，不特定主題

颱颱風千萬別來，~~我們沒那麼熱血~~

  - 固定於每週四 19:30-21:30，不改時間，逢國定假日暫停一次
  - 議程不會提前排定，以當天參加者最近的心得分享為主

- 舉辦不定期研討會

twMVC 不定期研討會

- 預先設定主題的中大型會議，主題提前於 mvc.tw 上公告
- 原則上一個月排定一次，期初因場地關係設定報名人數上限為 30 人

場場都是精心安排的實戰經驗

ASP.NET MVC 可以吃嗎？

tw MVC
demoshop
http://demo.tc
MVP Microsoft Most Valuable Professional

ASP.NET MVC 真的熟了嗎？

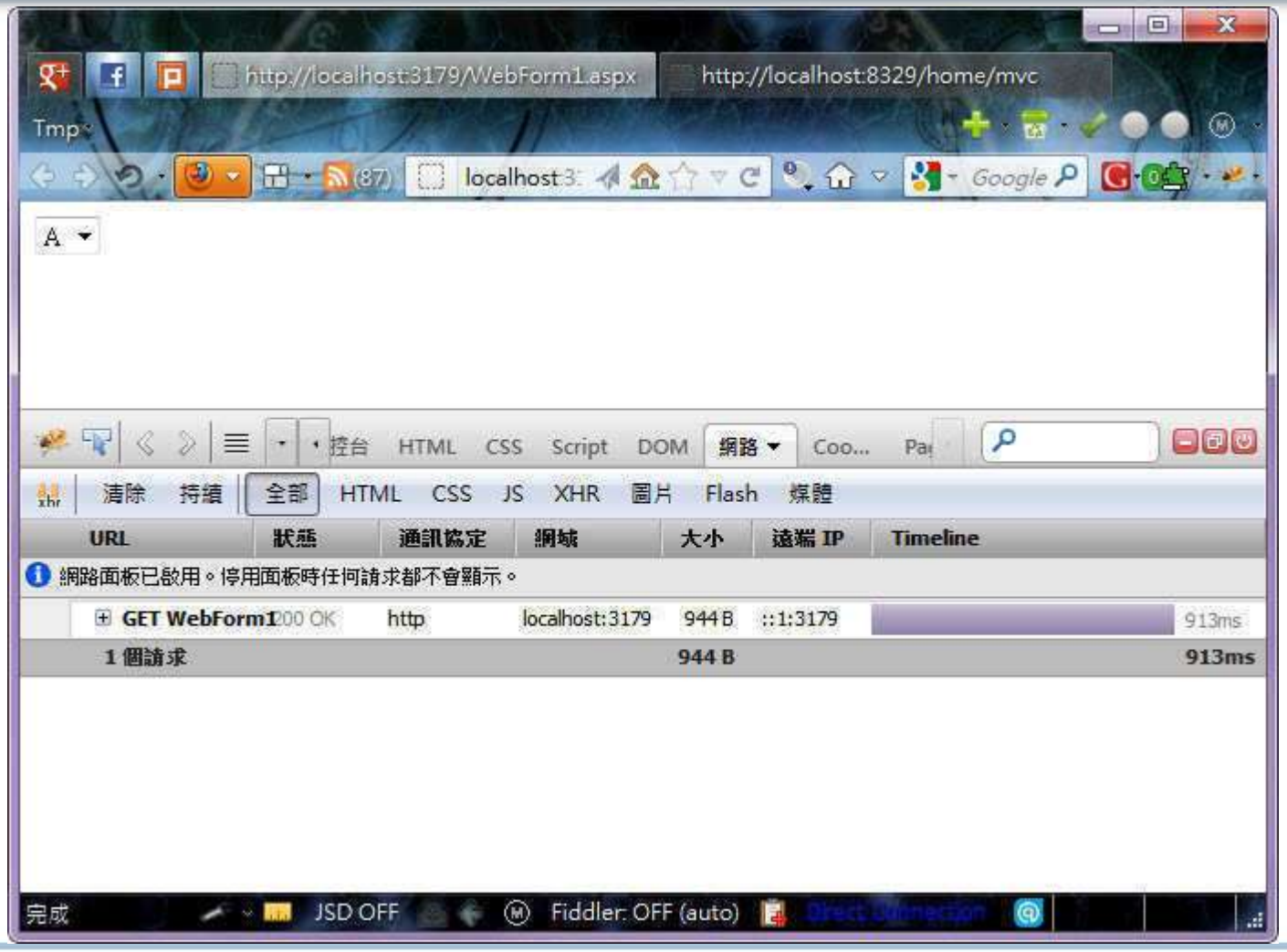
- 政府網站
  - 行政院人事行政局-人事服務網 http://ecpa.gov.tw
  - 行政院研究發展考核委員會民眾e管家 https://msg.nat.gov.tw
  - 中華民國外交部 http://www.mofa.gov.tw
  - 縮減產業數位落差計畫 http://e98.sme.gov.tw
- 商務網站
  - TutorABC http://www.tutorabc.com/aspx/mvc
  - 華人健康網http://www.top1health.com
  - 米格國際 http://www.lativ.com.tw

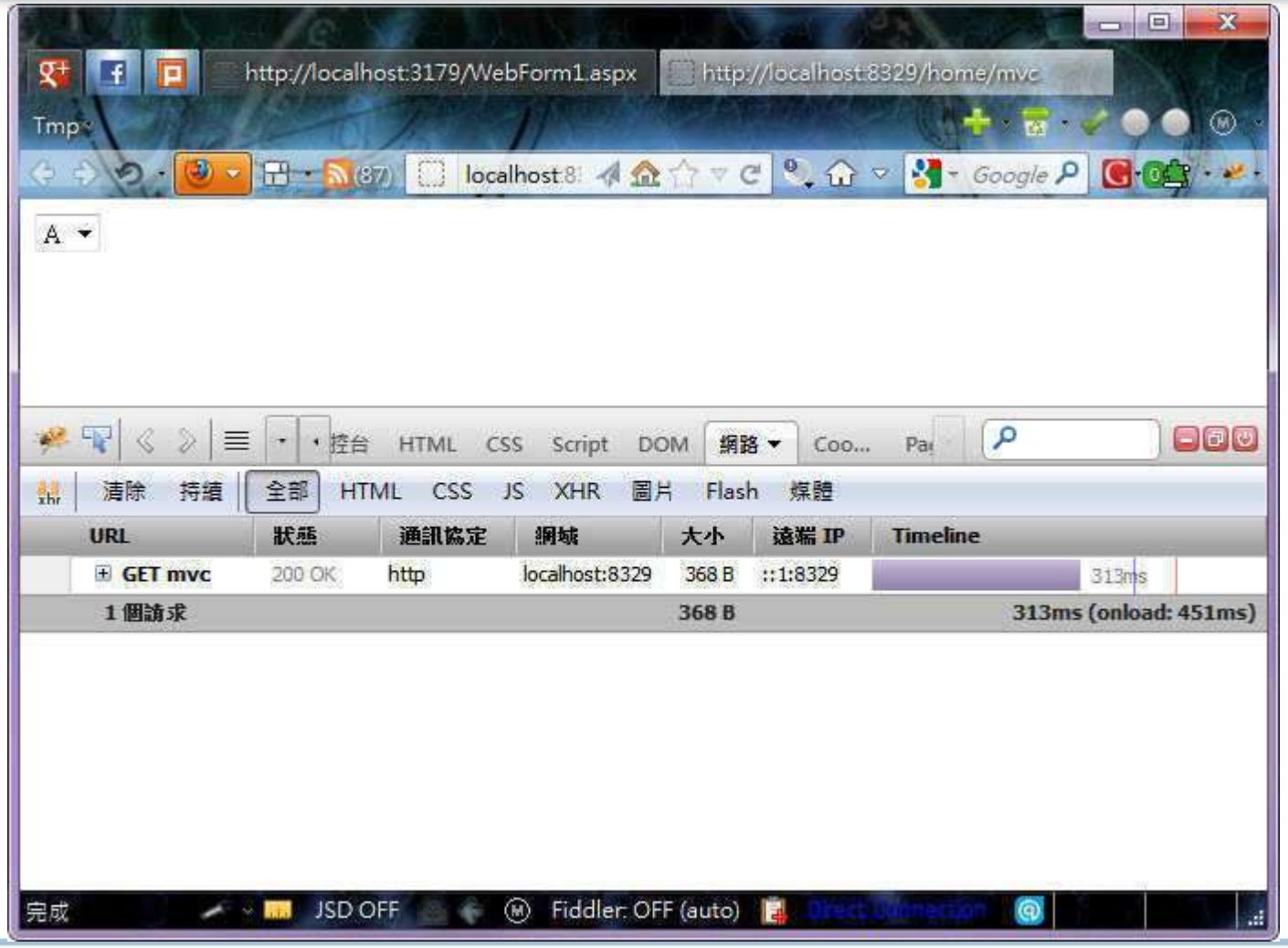
ASP.NET MVC 真的熟了嗎？

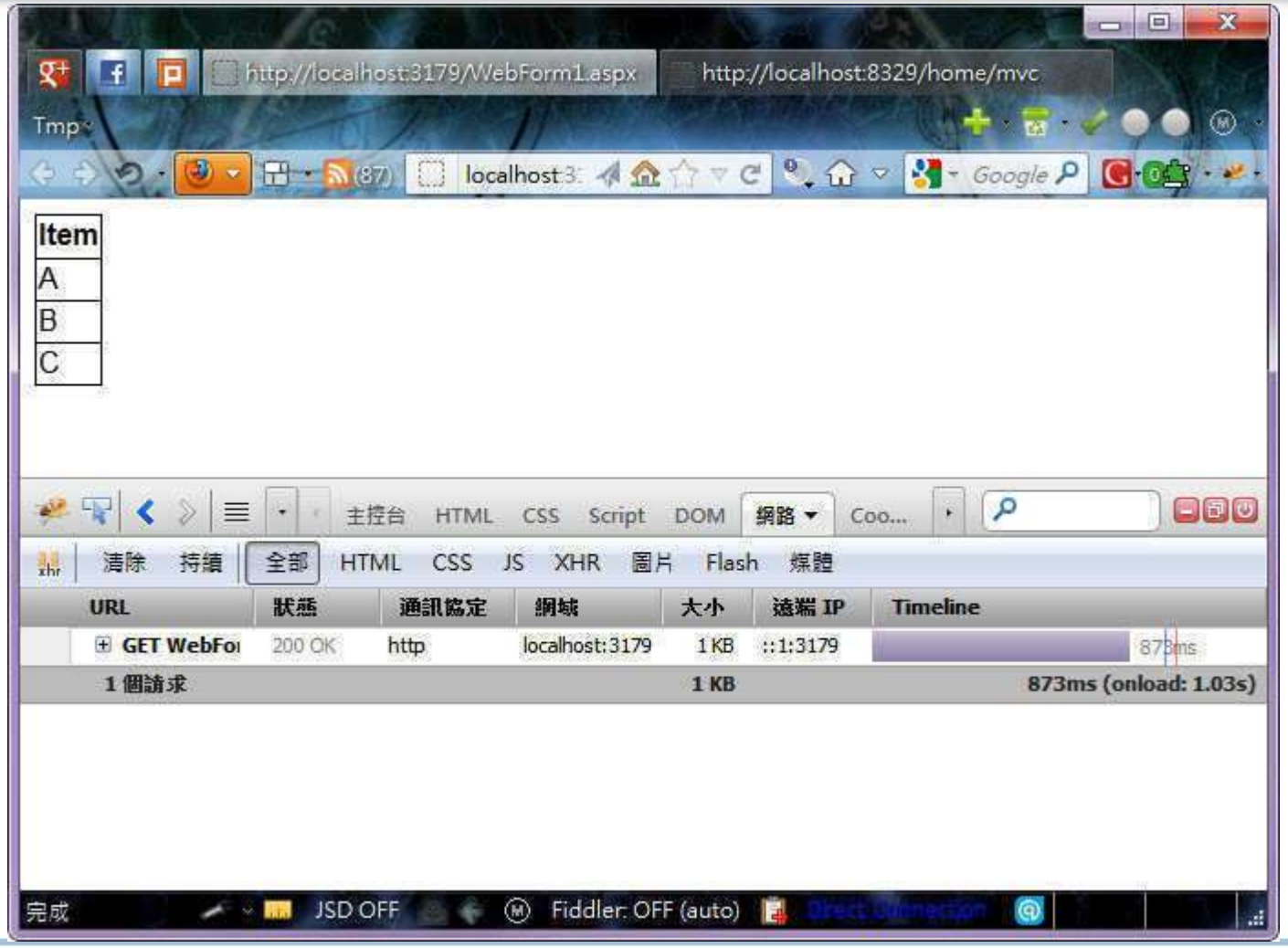
- Stack Over Flow http://stackoverflow.com/
- Code Plex http://www.codeplex.com/
- Docs http://docs.com/
- Market Watch http://www.marketwatch.com/
- Ruths Chris http://www.ruthschris.com/
- Dotnet Shout Out http://dotnetshoutout.com/

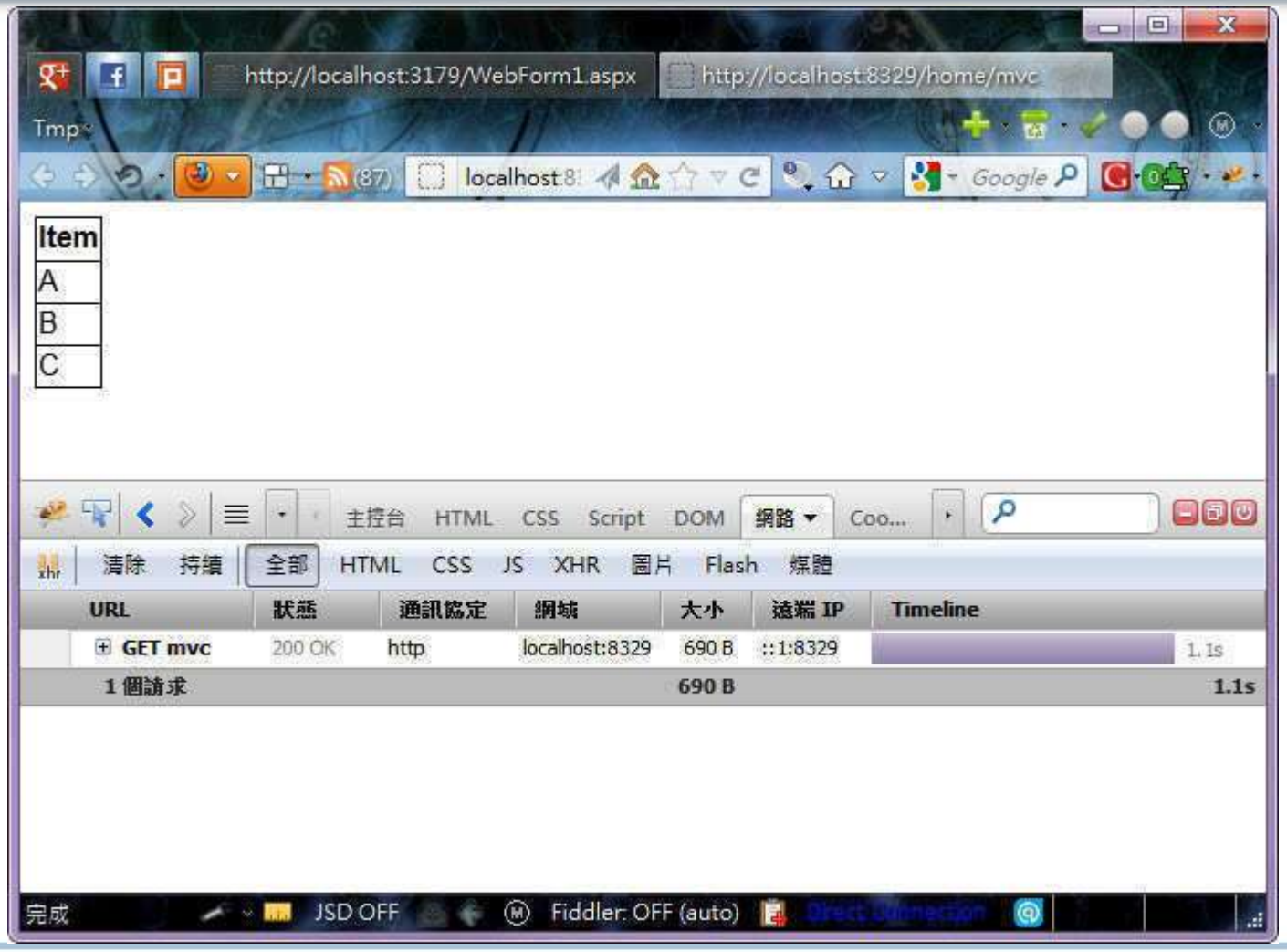
不要為改變而改變，是因為值得而改變

- ASP.NET Web Form 的初衷
- 肥大的 ViewState


http://localhost:3179/WebForm1.aspx
http://localhost:8329/home/mvc
Tmp
localhost:3
Google
A
控台
HTML
CSS
Script
DOM
網路
Coo...
清除
持續
全部
HTML
CSS
JS
XHR
圖片
Flash
媒體
URL
狀態
通訊協定
網域
大小
遠端 IP
Timeline
網路面板已啟用。停用面板時任何請求都不會顯示。
GET WebForm1
200 OK
http
localhost:3179
944 B
::1:3179
913ms
1 個請求
944 B
913ms
完成
JSD OFF
Fiddler: OFF (auto)

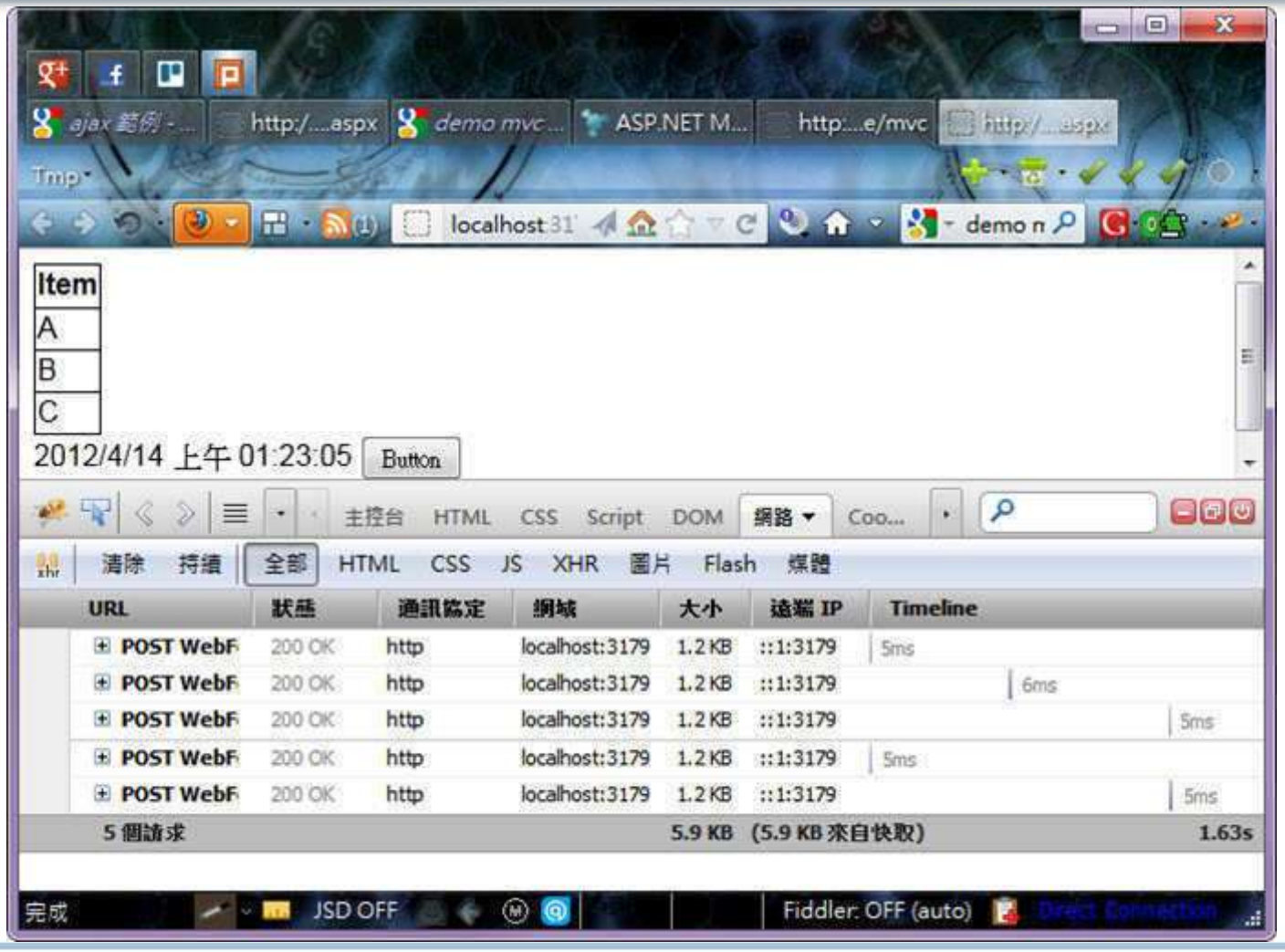

http://localhost:3179/WebForm1.aspx
http://localhost:8329/home/mvc
Tmp
localhost:8
Google
A
控台
HTML
CSS
Script
DOM
網路
Coo...
清除
持續
全部
HTML
CSS
JS
XHR
圖片
Flash
媒體
URL
狀態
通訊協定
網域
大小
遠端 IP
Timeline
GET mvc
200 OK
http
localhost:8329
368 B
::1:8329
313ms
1 個請求
368 B
313ms (onload: 451ms)
完成
JSD OFF
Fiddler: OFF (auto)

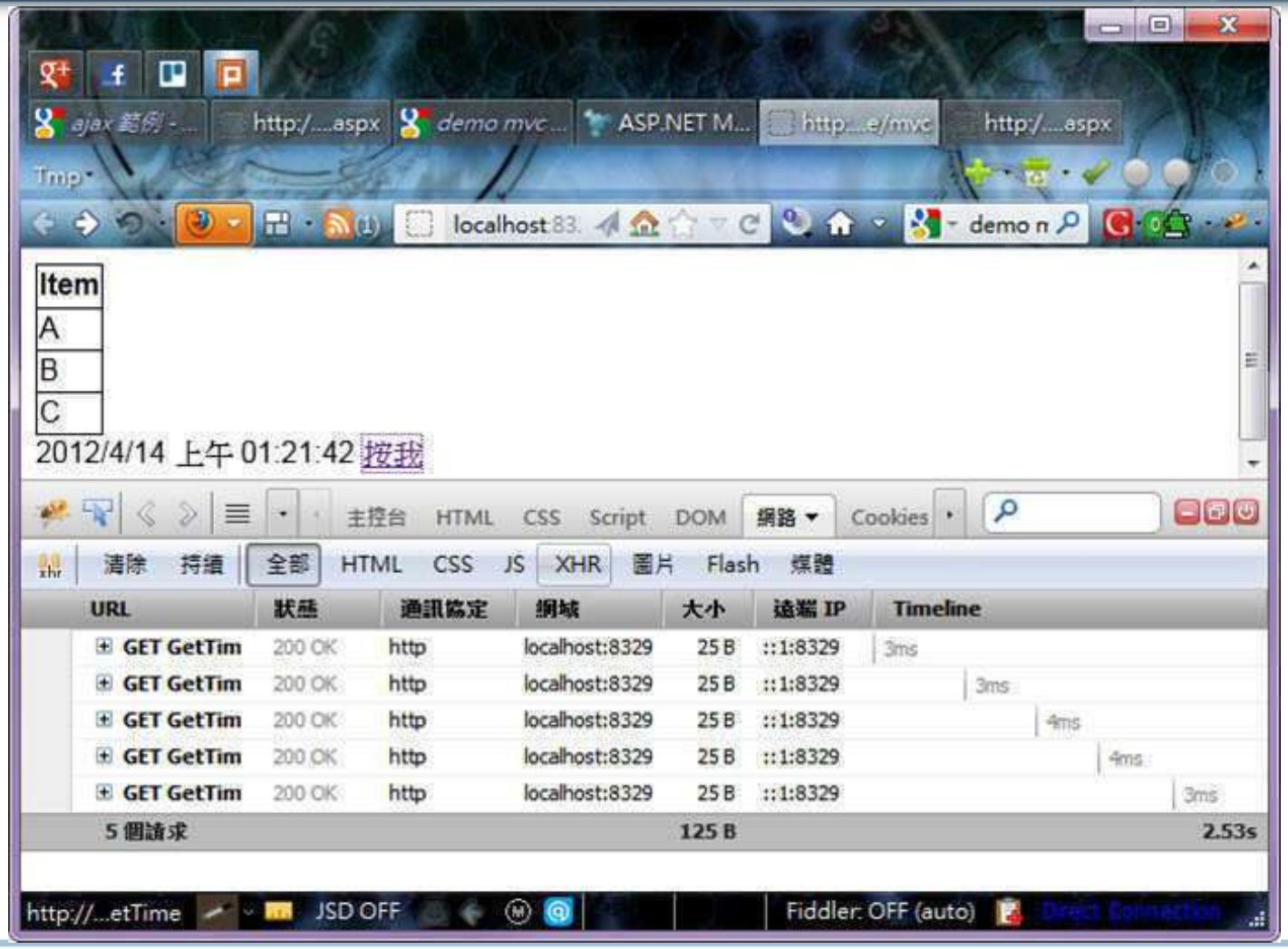

http://localhost:3179/WebForm1.aspx
http://localhost:8329/home/mvc
Tmp
Google
Item
A
B
C
主控台
HTML
CSS
Script
DOM
網路
Coo...
清除
持續
全部
HTML
CSS
JS
XHR
圖片
Flash
媒體
URL
狀態
通訊協定
網域
大小
遠端 IP
Timeline
GET WebFo
200 OK
http
localhost:3179
1 KB
::1:3179
873ms
1 個請求
1 KB
873ms (onload: 1.03s)
完成
JSD OFF
Fiddler: OFF (auto)


http://localhost:3179/WebForm1.aspx
http://localhost:8329/home/mvc
Tmp
Google
Item
A
B
C
主控台
HTML
CSS
Script
DOM
網路
Coo...
清除
持續
全部
HTML
CSS
JS
XHR
圖片
Flash
媒體
URL
狀態
通訊協定
網域
大小
遠端 IP
Timeline
GET mvc
200 OK
http
localhost:8329
690 B
::1:8329
1.1s
1 個請求
690 B
1.1s
完成
JSD OFF
Fiddler: OFF (auto)

不要為改變而改變，是因為值得而改變

- ASP.NET Web Form 的初衷
- 肥大的 ViewState
- 瘋狂的閃爍（Postback）
- 神奇的 UpdatePanel？


Item
A
B
C
2012/4/14 上午 01:23:05
Button
主控台
HTML
CSS
Script
DOM
網路
清除
持續
全部
HTML
CSS
JS
XHR
圖片
Flash
媒體
URL
狀態
通訊協定
網域
大小
遠端 IP
Timeline
POST WebF
200 OK
http
localhost:3179
1.2 KB
::1:3179
5ms
6ms
5ms
5ms
5ms
5 個請求
5.9 KB
(5.9 KB 來自快取)
1.63s
完成
JSD OFF
Fiddler: OFF (auto)


ajax 範例 - ...
http:/....aspx
demo mvc ...
ASP.NET M...
http:...e/mvc
http:/....aspx
Tmp
localhost:83...
demo n
Item
A
B
C
2012/4/14 上午 01:21:42 按我
主控台 HTML CSS Script DOM 網路 Cookies
清除 持續 全部 HTML CSS JS XHR 圖片 Flash 媒體
URL 狀態 通訊協定 網域 大小 遠端 IP Timeline
GET GetTim 200 OK http localhost:8329 25 B ::1:8329 3ms
GET GetTim 200 OK http localhost:8329 25 B ::1:8329 3ms
GET GetTim 200 OK http localhost:8329 25 B ::1:8329 4ms
GET GetTim 200 OK http localhost:8329 25 B ::1:8329 4ms
GET GetTim 200 OK http localhost:8329 25 B ::1:8329 3ms
5 個請求 125 B 2.53s
http://...etTime
JSD OFF
Fiddler: OFF (auto)

不要為改變而改變，是因為值得而改變

- ASP.NET Web Form 的初衷
- 肥大的 ViewState
- 瘋狂的閃爍（Postback）
- 神奇的 UpdatePanel？

坑爹阿

不要為改變而改變，是因為值得而改變

- ASP.NET Web Form 的初衷
- 肥大的 ViewState
- 瘋狂的閃爍（Postback）
- 神奇的 UpdatePanel？
- 難以操作的 HTML
- 超高難度的客製化

回到 Web 的本質

- ASP.NET MVC 為什麼要出現 回歸 Web 的本質
- 丟棄不要的包袱 ViewState Web 是無狀態的
- Web中沒有 Postback HTTP動詞你一定要知道
- 隨心所欲的操控 HTML 凍結表頭、表格排序、套版
- 純正的 AJAX？ jQuery 是你的好幫手
- 輕鬆搞定客製化 那麼多東西都是自己生的，還有什麼搞不定？

使用者專訪

使用者專訪



使用者專訪



使用者專訪



什麼是MVC



MR.KT

的程式學習筆記

tw MVC

什麼是 MVC

- MVC是一種「設計模式」不是程式語言也不是框架
- 最早於1979年時由Trygve Reenskaug所提出
- 一開始是應用於SmallTalk的程式語言之中

什麼是 MVC

- 把應用程式架構給切分為三個部分，分別為：
  - Model（模型）
  - View（視圖）
  - Controller（控制器）
- MVC的三個組成部分所負責的功能不同，相互獨立
- MVC並不等於三層式架構

Model

- 「資料模型」用於封裝與應用程序的業務邏輯相關的數據以及對數據的處理方法
- 有關「資料處理」的範圍都可屬於Model，包含：
  - 資料庫的存取操作（讀取、寫入）
  - 資料結構的定義
  - 資料格式的驗證

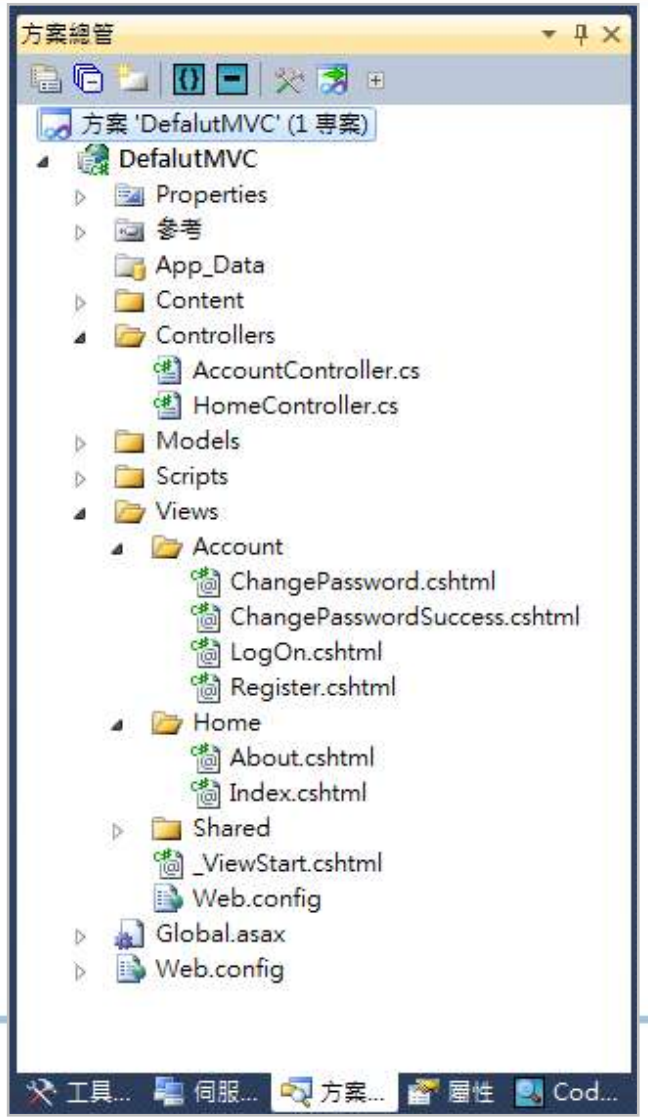
View

- 在頁面上顯示或是經由頁面傳回後端都是View所負責
  - 輸出：把資料輸出並顯示到使用者介面上
  - 輸入：把使用者所輸入的資料傳回到伺服器端
- 有關View的技術還包含：
  - HTML / CSS
  - Javascript(jQuery, MoonTool, ExtJS …etc)
  - JSON / XML
  - AJAX
  - RIA(Rich Interactive Application, ex:Flash,Silverlight)

Controller

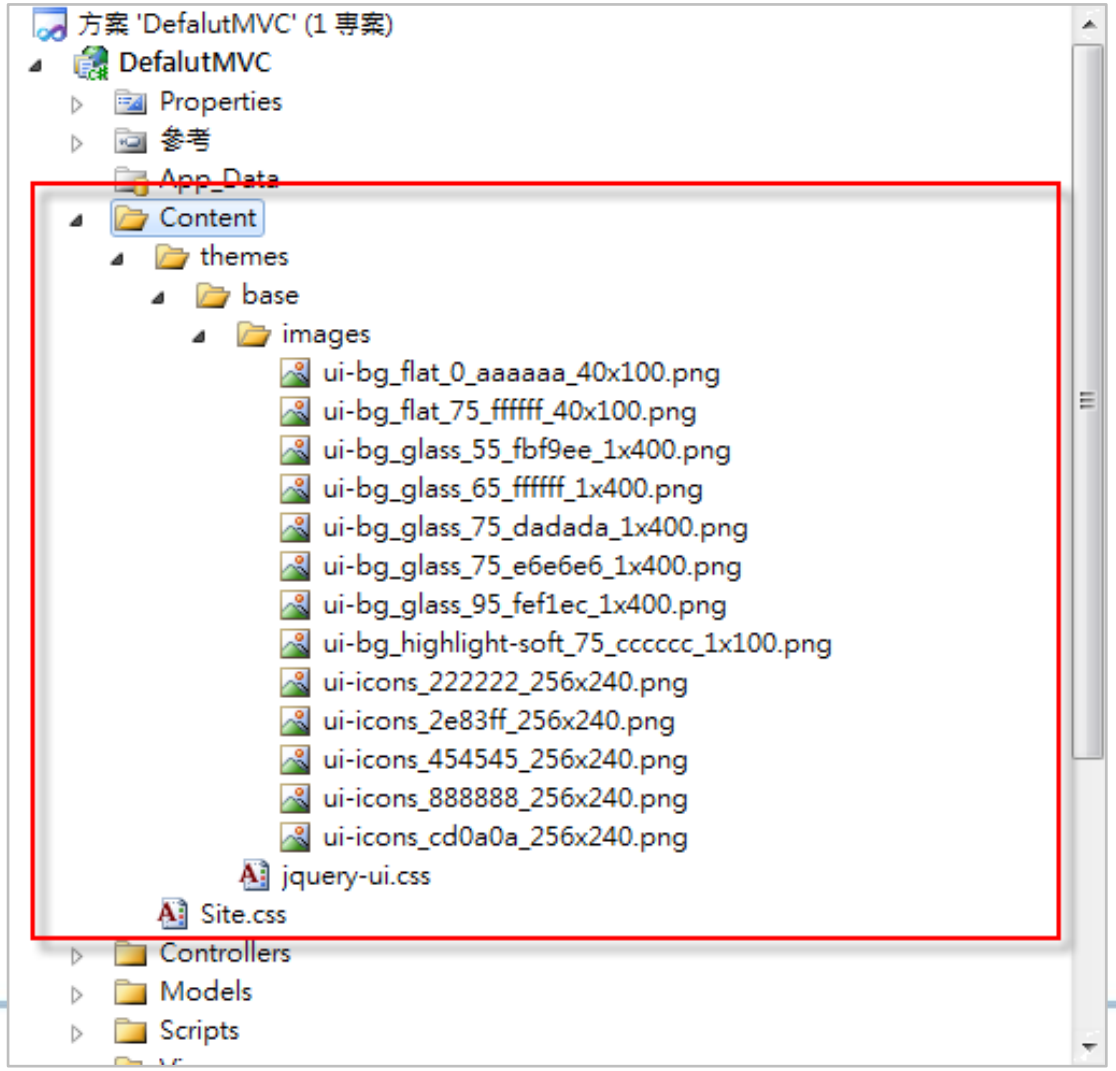
- 控制整個系統的業務流程
- 依據制定好的規則進行對資料處理以及View的使用
- 控制器接受外部傳入的資料並與Model進行資料處理，最後再由View回應外部的傳入並做顯示

認識ASP.NET MVC網站專案的目錄結構



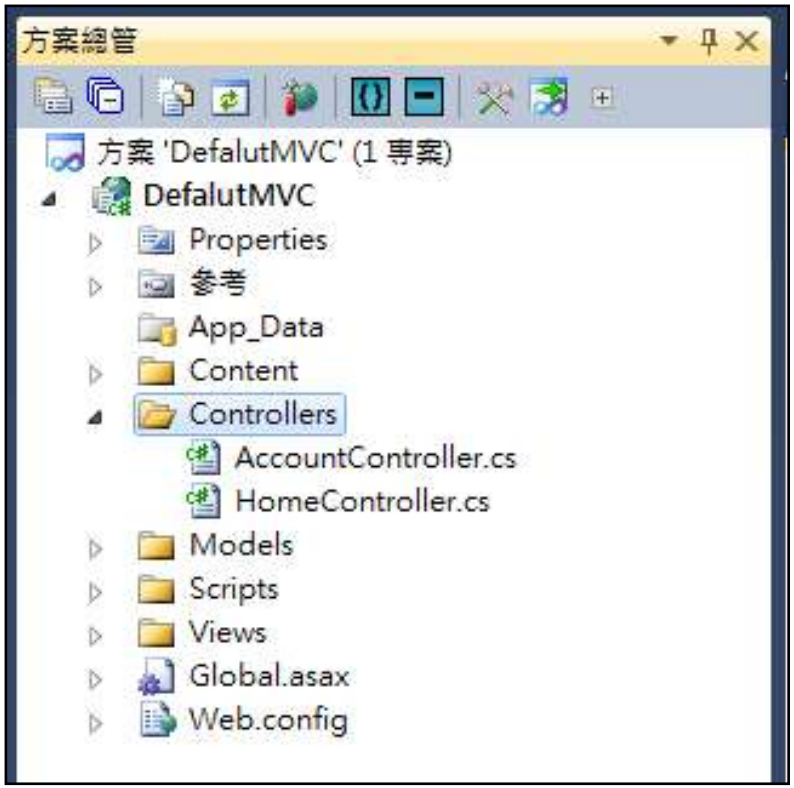
認識ASP.NET MVC網站專案的目錄結構

- Content



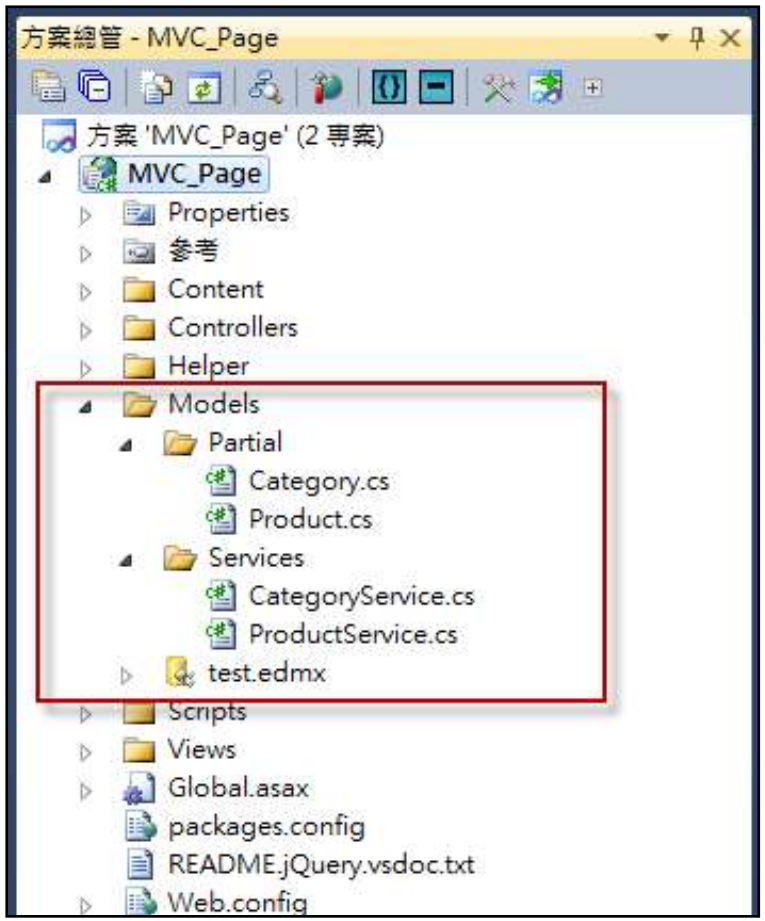
認識ASP.NET MVC網站專案的目錄結構

- Controller



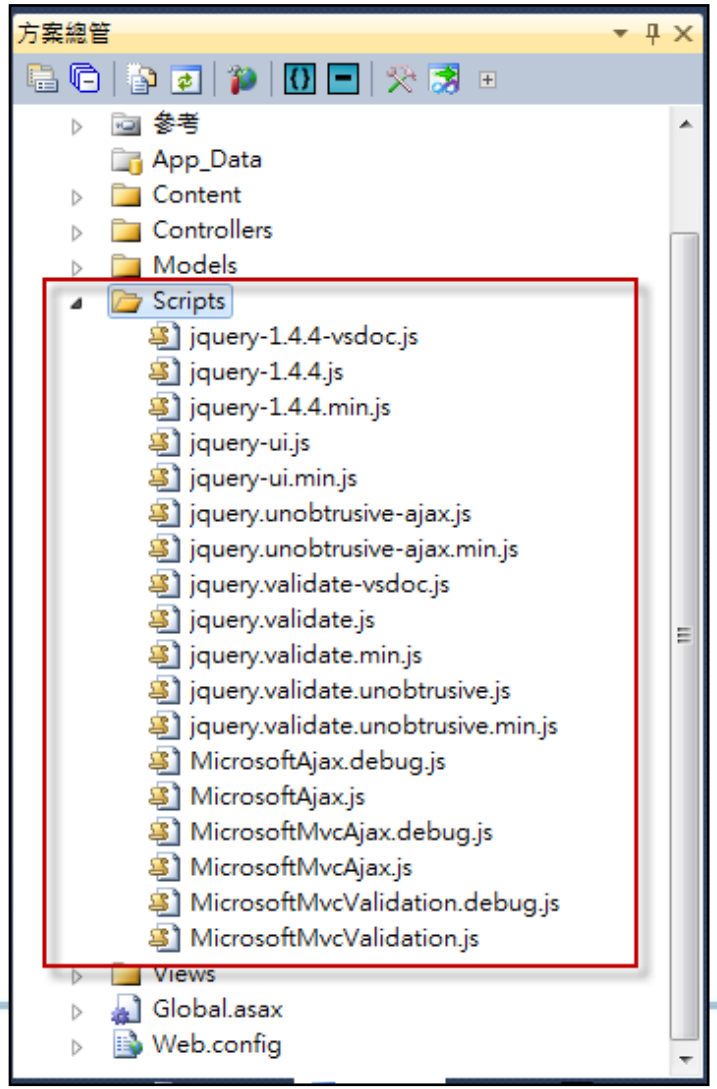
認識ASP.NET MVC網站專案的目錄結構

- Models



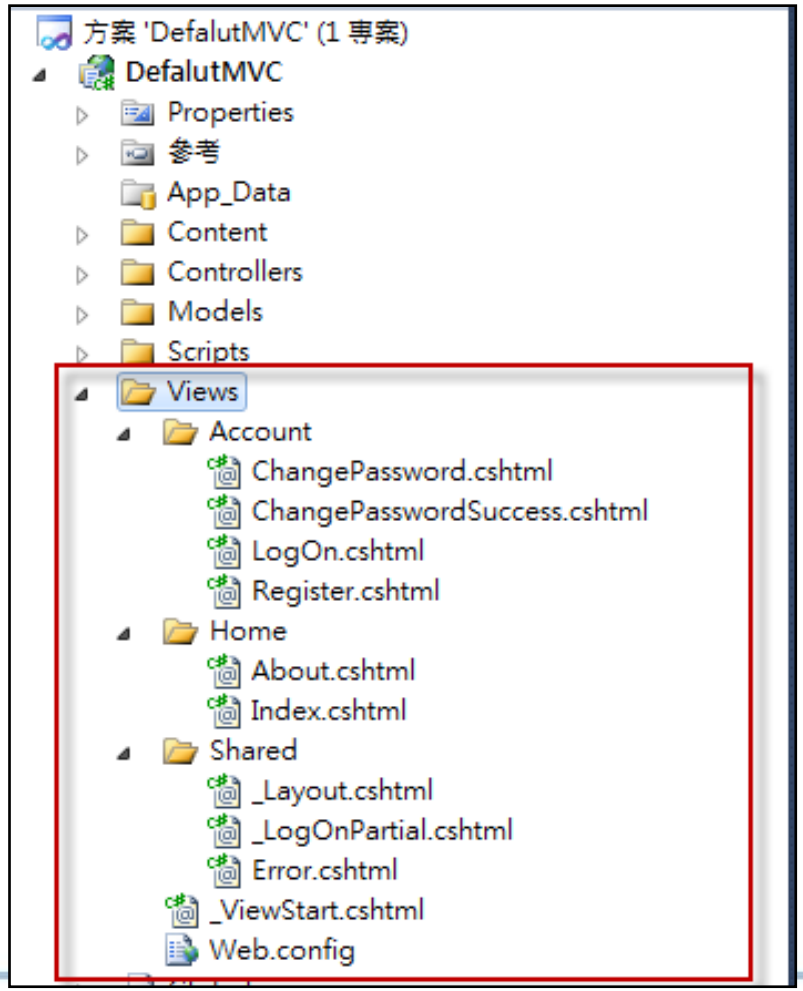
認識ASP.NET MVC網站專案的目錄結構

- Scripts



認識ASP.NET MVC網站專案的目錄結構

- Views



以習慣取代配置
Convention over Configuration

- 何謂「配置」?
- 配置所帶來的缺點
  - 減少開發人員在架構時決策的時間
  - 降低軟體設計過於彈性而導致過於複雜
- 以「開發習慣」共用同一套設計架構
- 所謂共同的「習慣」能可以自行客製以及擴充

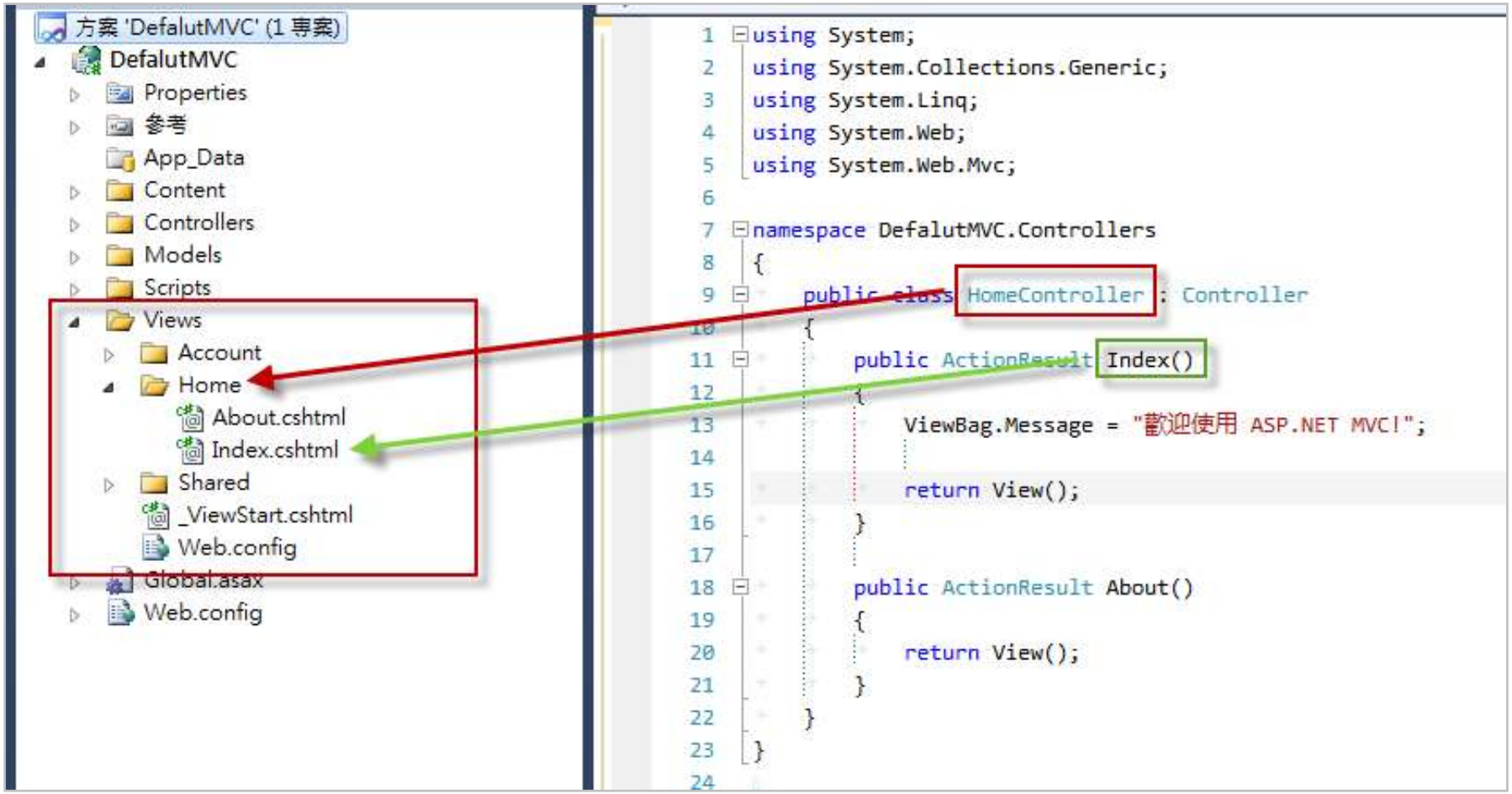
以習慣取代配置
Convention over Configuration

```
namespace DefalutMVC.Controllers
{
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            ViewBag.Message = "歡迎使用 ASP.NET MVC!";

            return View();
        }

        public ActionResult About()
        {
            return View();
        }
    }
}
```

以習慣取代配置
Convention over Configuration



ASP.NET MVC 對物件導向的潛移默化

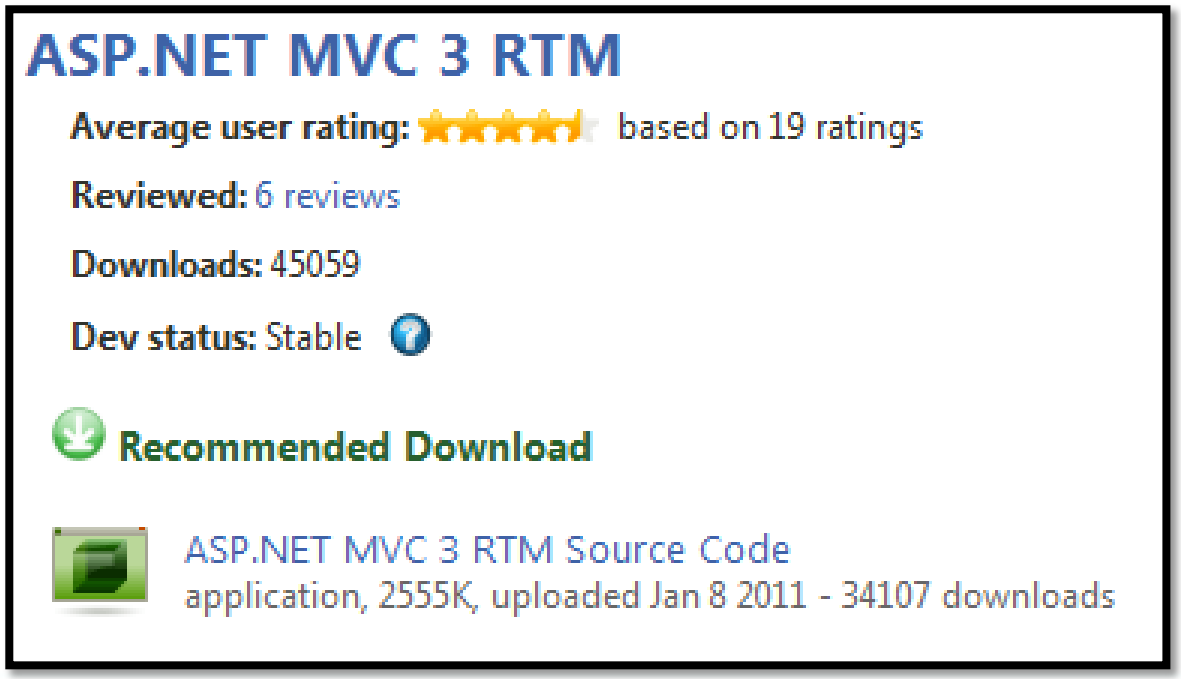






ASP.NET MVC 對物件導向的潛移默化

- Open Source
  - http://aspnet.codeplex.com



ASP.NET MVC 對物件導向的潛移默化

- 物件
  - 強型別物件
  - Model Binding

```
//在Mvc中都會使用強型別的Model，做為不同層的資料交換
public class SampleModel
{
    public string Name { get; set; }

    public string Description { get; set; }
}


 public ActionResult Add()
 {
     //從服務層取出資料，直接傳給View
     SampleModel sample = SampleService.Get(10);

     return View(sample);
 }
```

```
//View回傳的資料,因為有ModelBinding可以Bind到物件
[HttpPost]
public ActionResult Add(SampleModel sample)
{
    var name = sample.Name;

    return View();
}
```

```
@model SampleModel
@using (Html.BeginForm())
{
    <div class="editor-label">
        @*View也是使用強型別的Model來操作*@
        @Html.EditorFor(model => model.Name)
    </div>

    <div class="editor-field">
        @Html.EditorFor(model => model.Description)
    </div>
}
```

ASP.NET MVC 對物件導向的潛移默化

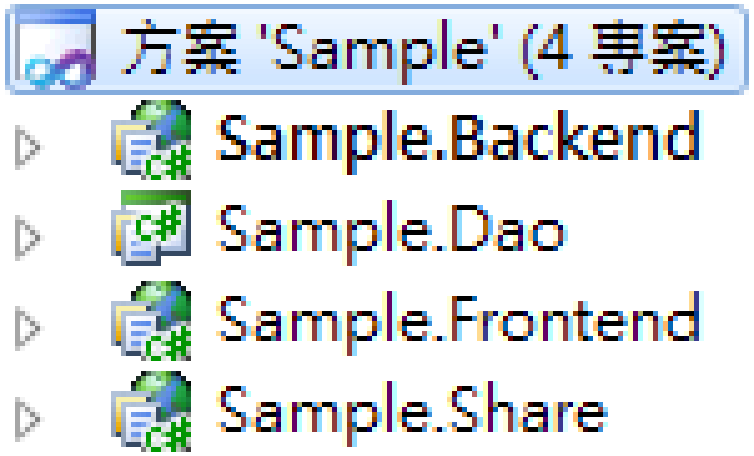
動作

```
//ASP.NET 的寫法，每個Method都是事件處理常式
public class ShoppingCartPage : Page
{
    protected void Page_Load(object sender, EventArgs e) { }
    protected void AddItemButton_Click(object sender, EventArgs e) { }
    protected void RemoveItemButton_Click(object sender, EventArgs e) { }
    protected void BillButton_Click(object sender, EventArgs e) { }
}

//ASP.NET MVC 的寫法，比較偏向物件導向的設計
public class ShoppingCartController : Controller
{
    public ActionResult List() { return View(); }
    public ActionResult AddItem(int itemId) { return View(); }
    public ActionResult RemoveItem(int itemId) { return View(); }
    public ActionResult Bill() { return View(); }
}
```

ASP.NET MVC 對物件導向的潛移默化

封裝



ASP.NET MVC 對物件導向的潛移默化

繼承與多型

```
public ActionResult Test(SampleModel sample)
{
    return new ViewResult();
}

public ActionResult Test(SampleModel sample)
{
    return new JsonResult();
}

public ActionResult Test(SampleModel sample)
{
    return new MyResult();
}
```

ASP.NET MVC 也用到了很多設計模式

如:

- MVC
- Factory
- Template
- Decorator(裝飾者)
- …等等

ValueProviderFactory	System.Web.Mvc
ChildActionValueProviderFactory	System.Web.Mvc
MvcWebRazorHostFactory	System.Web.Mvc
JsonValueProviderFactory	System.Web.Mvc
FormValueProviderFactory	System.Web.Mvc
HttpFileCollectionValueProviderFactory	System.Web.Mvc
QueryStringValueProviderFactory	System.Web.Mvc
RouteDataValueProviderFactory	System.Web.Mvc
ValueProviderFactoryCollection	System.Web.Mvc
IControllerFactory	System.Web.Mvc
DefaultControllerFactory	System.Web.Mvc

HttpRequestBase	HttpResponseBase
HttpRequestWrapper	HttpResponseWrapper
HttpRequest	HttpResponse

Web form VS MVC

- 有太多人拿這兩個來做比較，但是我們並不想比較，因為這兩種框架出發點就是不同的，拿來比較只是兄弟鬩牆而已，因此我們 tw MVC 並不希望各位一直想拿這兩種框架來比較。

休息時間五分鐘

Take a break

View

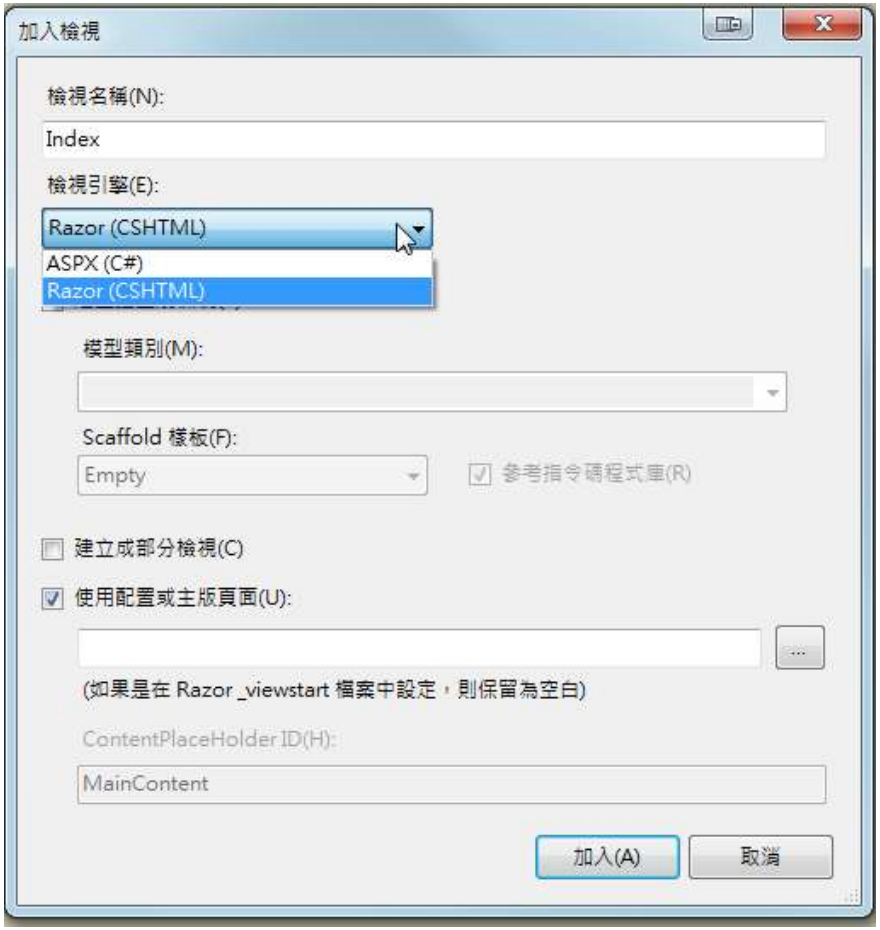
回歸初衷的HTML

fw MVC
Coding Road
MVP Microsoft Most Valuable Professional

Razor 簡介

- 發展者：Microsoft
- 發展於2010年6月
- Release 於2011年1月
- 主要利用 C# 或 VB.NET 來產生動態網頁的語法

MVC 3 內建的View Engine



ASPX 標記語法

- ASPX的標記語法是使用<%= %>

```
<ul id="product">
    <%foreach (var item in Model)
      { %>
        <li><%=item.Name %> ($<%=item.Price %>)</li>
    <%} %>
</ul>
```

Razor 標記語法

- Razor的標記語法是使用@字元表示

```
<ul id="product">
    @foreach (var item in Model)
    {
        <li>@item.Name ($@item.Price)</li>
    }
</ul>
```

Razor優勢

- 檔案更小、開發更直覺順暢
- 容易學習
- 不需要再學新的語言
- 能在任何編輯器上使用
- 有工具支援IntelliSense

MVC View V.S. ASP

- 重蹈義大利麵寫法的覆轍?
- 可讀性低?程式與顯示綁在一起難以維護?

ASP 寫法

```
<%
'-------------------------------------------------
set rs_a=Server.CreateObject("adodb.recordset")
rs_a.CursorLocation=3
SQL_a="select * from [T_EV_2008Q4_Category] order by Code_TXT"
call Show_Debug(SQL_a)
%>
<table width="95%" border="0" cellpadding="2" cellspacing="2" class="table_border_
  <tr>
    <td width="10%" height="30" align="center" bgcolor="#CCCCCC">編碼</td>
    <td width="15%" height="30" align="center" bgcolor="#CCCCCC">分類名稱</td>
    <td width="25%" height="30" align="center" bgcolor="#CCCCCC">備註</td>
    <td width="17%" height="30" align="center" bgcolor="#CCCCCC">建立日期</td>
    <td width="8%" height="30" align="center" bgcolor="#CCCCCC">建立者</td>
    <td width="10%" height="30" align="center" bgcolor="#CCCCCC">功能</td>
  </tr>
    <td width="15%" align="center" bgcolor="#CCCCCC">菜名</td>
<%
rs_a.Open SQL_a,ConnDB,3,3
if not rs_a.EOF then
    i=1
    rs_a.MoveFirst
    do while not rs_a.EOF
      if (i mod 2 = 1) then bgcolor = "#F0F0F0" else bgcolor = "#F1F6FF" End if
%>
  <tr bgcolor="<%=bgcolor%>" onMouseOver="this.style.backgroundColor='#FFF0D0';" o
    <td width="10%" height="30" align="center"><%=Trim(rs_a("Code_TXT"))%></td>
    <td width="15%" height="30" align="center"><%=Trim(rs_a("Category_TXT"))%></td
    <td width="15%" height="30" align="center"><%=Trim(rs_a("Name_TXT"))%></td>
    <td width="25%" height="30"><%=Trim(rs_a("Memo_TXT"))%></td>
    <td width="17%" height="30" align="center"><%=Trim(rs_a("Create_TS"))%></td>
    <td width="8%" height="30" align="center"><%=Trim(rs_a("Create_Admin"))%></td>
```

MVC View 寫法

```
@model IEnumerable<MvcApplication1.Models.Product> 16

<!-- 資料來源從Controller傳遞過來，頁面上不包含任何資料存取及商業邏輯 -->
<!-- 只著重在顯示的邏輯上 -->
<ul id="product">
    @foreach (var item in Model)
    {
        <li>@item.Name ($@item.Price)</li>
    }
</ul>
```

HTML Helper

- 何謂Helper
  - 只是幫助程式開發的類別或方法
- 何謂HTML Helper
  - HTML Helper用來幫助View開發的方法
  - 用來簡化及包裝View上的瑣碎工作
  - 增加重用性

HTML Helper

- Html.ActionLink()
- Html.BeginForm()
- Html.CheckBox()
- Html.DropDownList()
- Html.EndForm()
- Html.Hidden()
- Html.ListBox()
- Html.Password()
- Html.RadioButton()
- Html.TextArea()
- Html.TextBox()

ActionLink

ActionLINk



MVC 3 新增的 Helper

名稱	描述
WebGrid	快速建立表格型態的資料展現方式，包含分頁及排序
Crypto	產生雜湊及字串加密
WebImage	處理圖片，包含取得、縮放、浮水印、輸出等…
Chart	將資料轉為圖表的呈現
ServerInfo	查看Server Varibles的屬性
WebMail	包裝System.Net.Mail的發信
WebCache	利用.NET 4.0的MemoryCache做快取
Json	將資料序列化成 JSON 格式

Layout

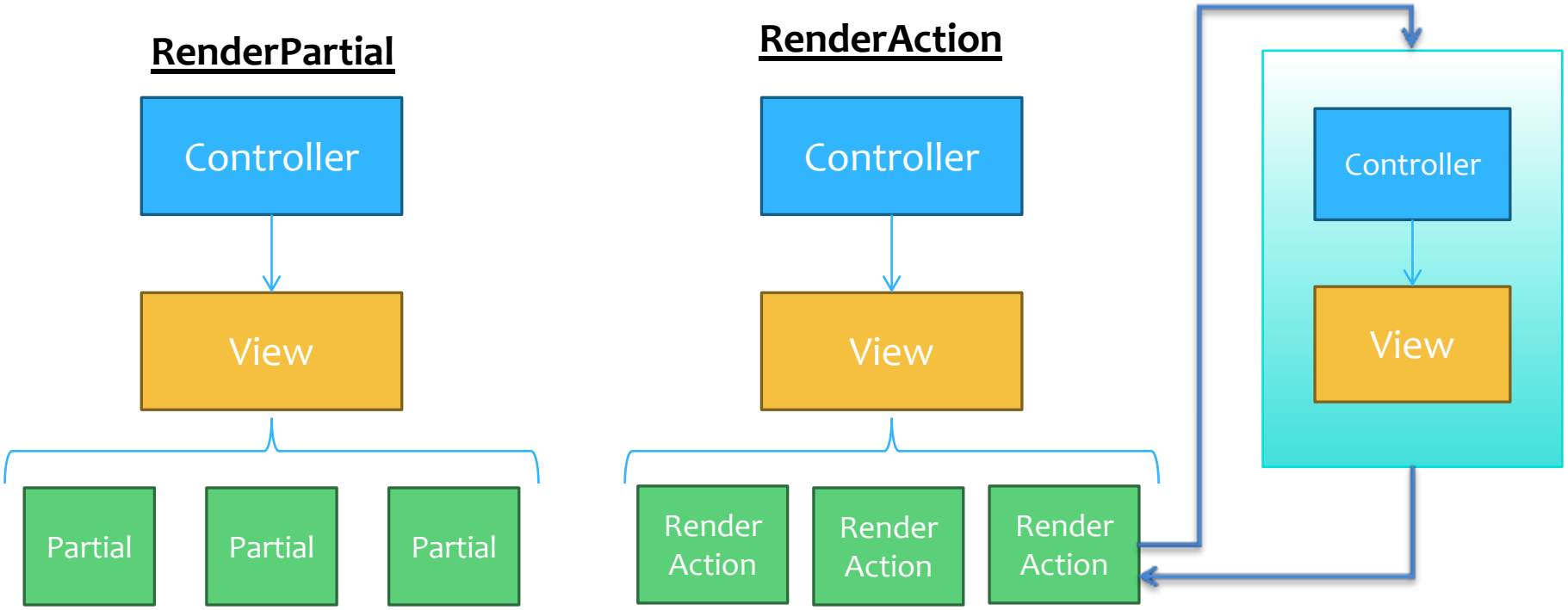
- 等同於Master Page
  - 用來放網站的共用元素。如Title、Footer、Menu等…
  - 在Razor中，副檔名依舊是cshtml
- 利用RenderBody()、@RenderSection() 保留在子頁面可變動的區塊

Partial View

- 當一個頁面每個區塊經常的被使用，或是頁面過於複雜，這時可以建立Partial View，將View切分成數個部分方便維護及重複使用。

RenderPartial & RenderAction

- RenderPartial 與 RenderAction 的差異?



RenderPartial & RenderAction

- RenderPartial使用時機
  - 資料由主View的Model提供
  - 沒有其他的流程控制
  - 為了顯示上的重複使用
- RenderAction使用時機
  - 資料會因參數或是其他變因而改變
  - 有其餘的流程需回Controller去處理
  - 為了邏輯上的重複使用

jQuery

- jQuery是一套跨瀏覽器的JavaScript函式庫，強化HTML與JavaScript之間的操作

- .NET MVC 內建許多功能與jQuery整合
  - 驗證與jQuery.Validate整合
  - Ajax

ViewModel

- ViewData
- TempData
- ViewBag
- 自訂類別

Razor Helper

Organizing Razor, Make It More Powerful

Dino Wang

http://dinowang.blogspot.com



直接就來個範例吧

```
@helper ExternalAnchor(string href, string text) {
      <a href="@href" class="btn" rel="external">
      @text
      </a>
}

@ExternalAnchor("http://www.asp.net", "ASP.NET")
@ExternalAnchor("http://mvc.tw", "TW MVC")
```

使用 Razor Helper

- 兩種形式
  - Razor Helper (*.cshtml)
  - Razor Helper Package (~\App_Code*.cshtml)

Razor Helper Package 一詞出現於 O'Reilly – Programming Razor 中

Razor Helper

- 定義與使用 (同一支 *.cshtml 內)

```
@helper ExternalAnchor(string href, string text) {
    <a href="@href" class="btn" rel="external">
    @text
    </a>
}

@ExternalAnchor("http://www.asp.net", "ASP.NET")
@ExternalAnchor("http://mvc.tw", "TW MVC")
```

Razor Helper Package

- 定義 (~\App_Code\My.cshtml)

```
@helper ExternalAnchor(string href, string text) {
    <a href="@href" class="btn" rel="external">
    @text
    </a>
}
```

- 使用

```
@My.ExternalAnchor("http://www.asp.net", "ASP.NET")
@My.ExternalAnchor("http://mvc.tw", "TW MVC")
```

很多技術能做到一樣的效果！？

- Partial View
  - 沒錯！但是 Razor Helper 的參數化方式讓使用上更簡單
- HtmlHelper (Extension Method)
  - 沒錯！但是你在 code 裡面組 HTML 語法是很痛苦的…

    我改用 TagBuilder 組 HTML 也很簡單！

    嗯… 那遇到巢狀 tag 呢

Razor Helper Package 實例

RazorHelperDemo

休息時間五分鐘

Take a break

Controller

只有流程沒有其它

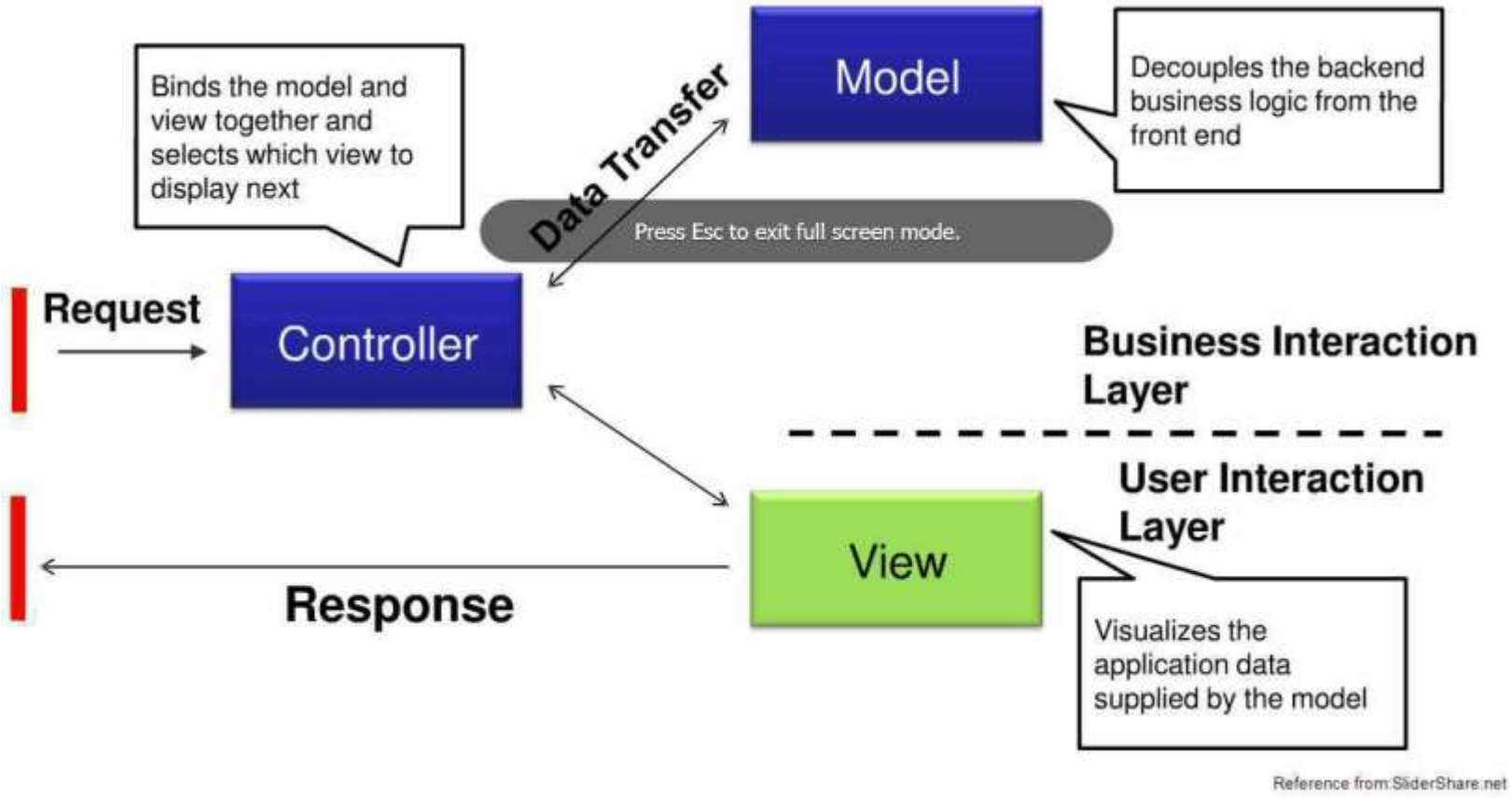
Bibby
http://bibby.be

fw MVC

會講的內容

- Controller 概述
- Controller - Action Result
- Controller - Action Filter
- 其它相關

ASP.NET MVC Web Request 流程



Controller 的特質

- 責任就是控制流程
- 簡短輕薄
- Web Request 永遠會通過 Controller
- 決定哪些 Data 是需要的
- 決定要丟出哪個 View

ActionResult

Controller - Action Result

主要功能就是

溝通 View 和 Controller

兩部份

ASP.NET MVC 3 Action Results

- ViewResultBase *** – ViewResult 、PartialViewResult
- RedirectResult **
- ContentResult **
- JsonResult **
- JavaScriptResult
- FileResult – FilePathResult、FileContentResult、FileStreamResult
- HttpUnauthorized
- RedirectToRouteResult
- EmptyResult

如何使用 Action Result

```
public class HomeController : Controller
{
    public ActionResult Index()
    {
        ViewBag.Message = "Welcome to ASP.NET MVC!";

        return View();
    }
    public ActionResult TextContent()
    {
        return Content("I am Bibby.");
    }
    public ActionResult JsonContent()
    {
        var obj = new { Name = "Bibby", Age = 25, Blog = "http://bibby.be" };
        return Json(obj);
    }
}
```



ActionFilter

Controller – Action Filter

主要功能就是

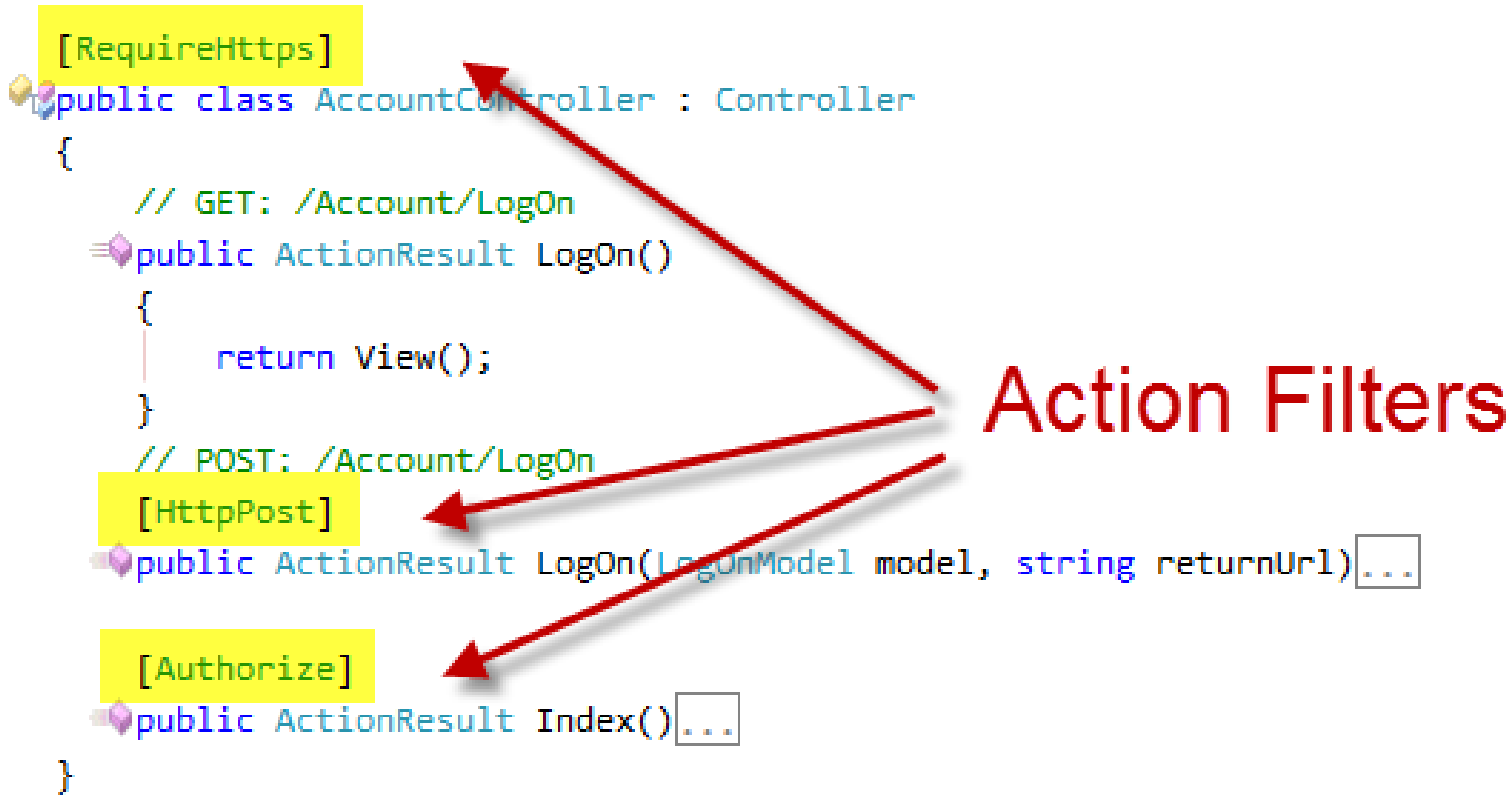
延伸加強 Controller 的

能力

Action Filters 的特質

- Action filters 本身就是 attribute
- 它可以放在任何的 Action 上
- 它也可以放在任何的 Controller 上
- Action 執行的時候就是 Action Filter 執行的時候
- 可以很快的客製化自己想要的 Action Filter

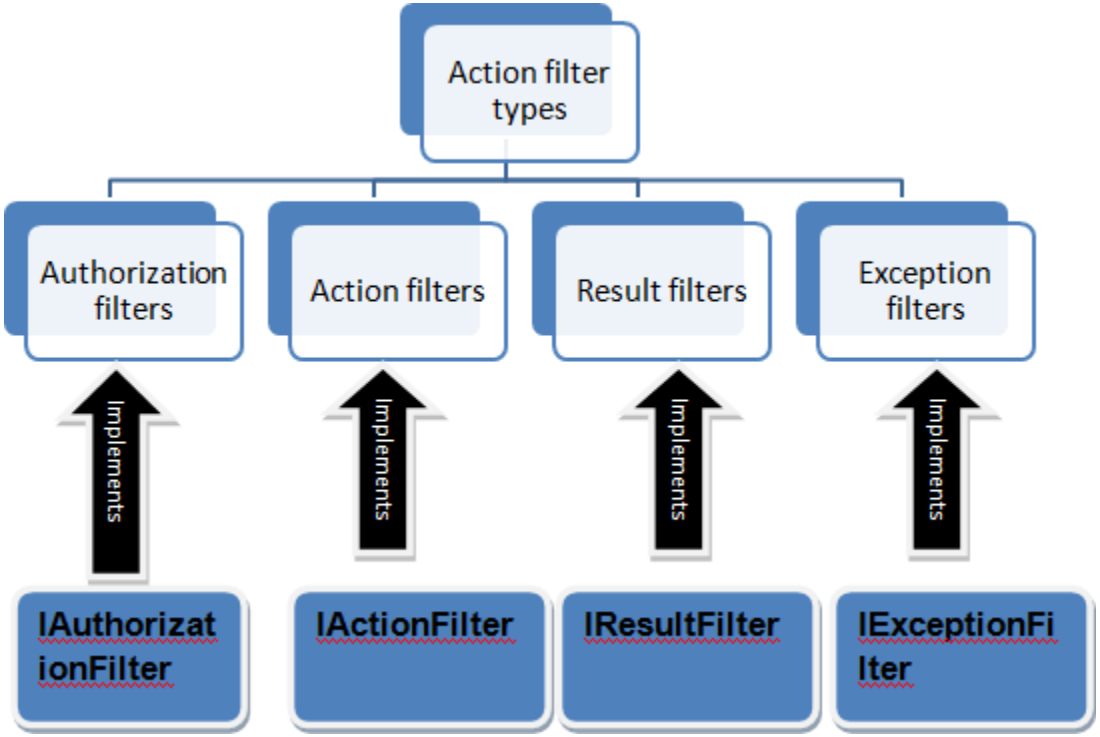
如何使用 Action Filter



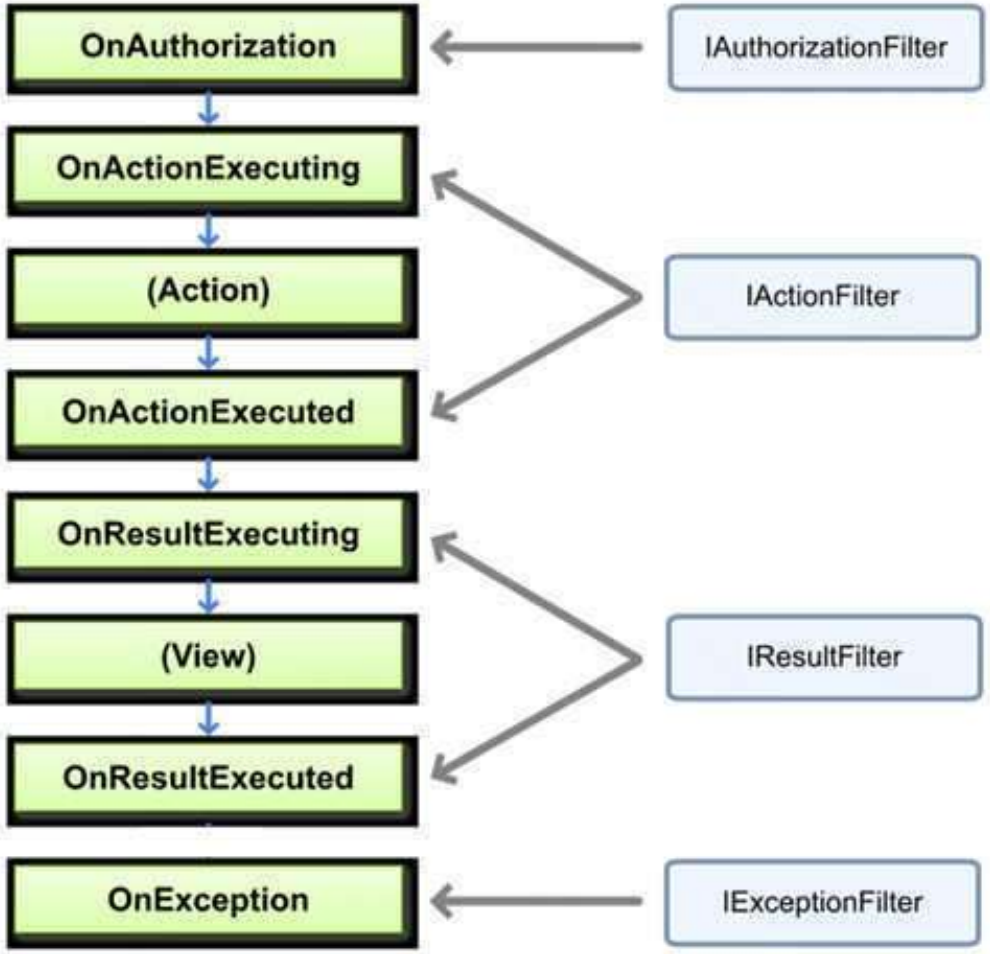
Action Filter 的分類

- Authorization Filters
- Action Filters
- Result Filters
- Exception Filters

Action Filter 型別



Action Filter 的執行流程



Authorization Filters

- Authorize
- ChildActionOnly
- ValidateInput
- RequireHttps
- ValidateAntiForgerytoken

Action Filters

- AsyncTimeout
- NoAsyncTimeout

=>範例

Result Filters

- OutputCache

```
[OutputCache(Duration=10)]//millisecond
```

  - VaryByContentEncoding
  - VaryByCustom
  - VaryByHeader
  - VaryByParam

Exception Filters

- HandleError

  = >範例

客製化 Action Filters

繼承 ActionFilterAttribute 來實做

```
public abstract class ActionFilterAttribute
                          : FilterAttribute, IActionFilter, IResultFilter
{
  public virtual void OnActionExecuted(ActionExecutedContext filterContext);
  public virtual void OnActionExecuting(ActionExecutingContext filterContext);
  public virtual void OnResultExecuted(ResultExecutedContext filterContext);
  public virtual void OnResultExecuting(ResultExecutingContext filterContext);
}
```

=>範例

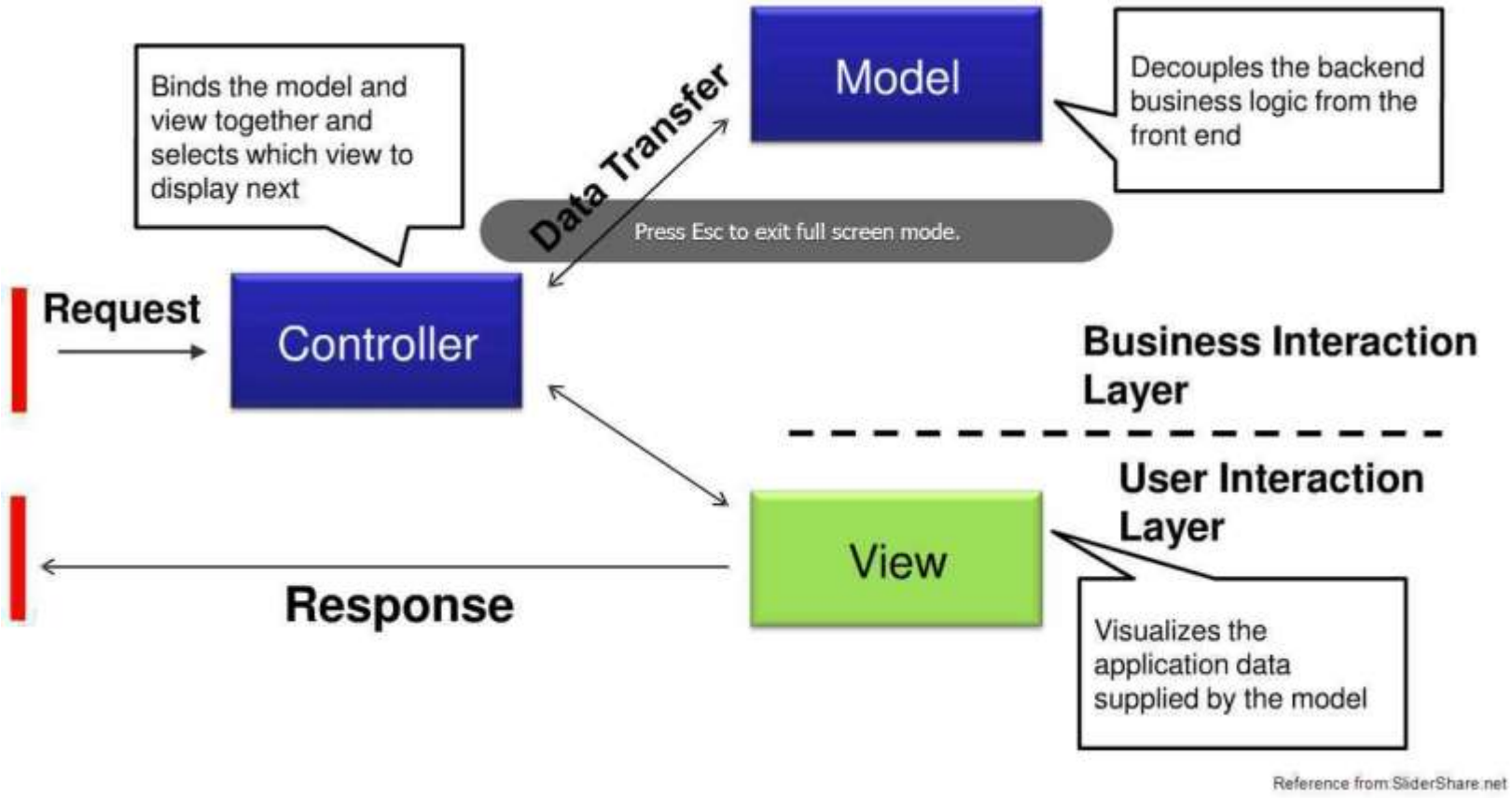
Controller之

其它

Controller – 其他

- HttpGet, HttpPost, HttpDelete, Http Put
- ViewData, TempData
- Model Binding

Controller - 最後



休息時間五分鐘

Take a break

Model



MR.KT

的程式學習筆記

fw MVC

Agenda

- 簡介
- Metadata
- 可用哪些東西來當做Model
- ASP.NET MVC專案的Model
- ViewModel

簡介

- 在ASP.NET MVC中Model負責所有與「資料」有關的任務
- 不處理所有與資料處理無關的事
- 不直接處理來自瀏覽器的輸入，也不向瀏覽器產生 HTML 輸出。
- ASP.NET MVC 架構對可以建置的各種模型物件沒有任何限制。

Metadata

- Metadata是用來定義資料模型的相關屬性，例如：
  - 是否為必填資料
  - 資料長度
  - 資料格式驗證 … etc
- **System.ComponentModel.DataAnnotations** 命名空間 (Namespace) 提供屬性類別，這些類別可用來定義 ASP.NET MVC 和 Dynamic Data 控制項的中繼資料 (Metadata)。

Metadata

```
[MetadataType(typeof(MemberMetadata))]
public partial class Member
{
    private class MemberMetadata
    {
        [DisplayName("會員編號")]
        public Guid MemberNo { get; set; }

        [DisplayName("真實姓名")]
        [Required(ErrorMessage = "請輸入真實姓名")]
        [StringLength(51, ErrorMessage = "請勿輸入超過 50 個字")]
        public string Name { get; set; }

        [DisplayName("手機號碼")]
        [Required(ErrorMessage = "請輸入手機號碼，手機號碼是你的登入帳號")]
        [StringLength(11, ErrorMessage = "請勿輸入超過 10 個字")]
        public string MobileNo { get; set; }

        [DisplayName("身分證後五碼")]
        [DataType(DataType.Password)]
        [Required(ErrorMessage = "請輸入身分證後五碼")]
        [StringLength(6, ErrorMessage = "請勿輸入超過 5 個字")]
        public string IdNo { get; set; }
```

可用哪些來當做Model

- LINQ to Entities
- ADO.NET Entity Framework
- NHibernate, SubSonic, LLBLGen Pro … etc

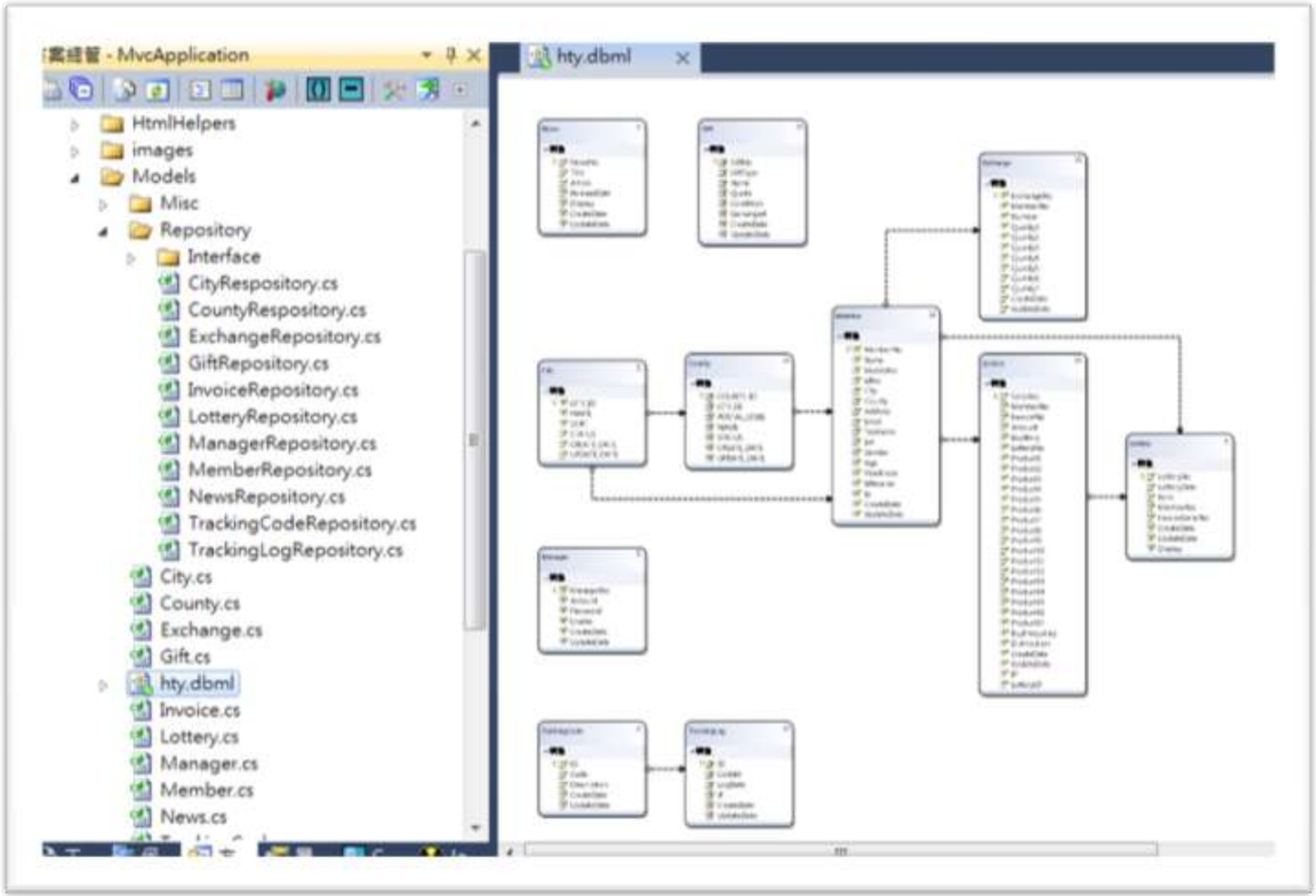
可用哪些來當做Model

- ADO.NET Entity Framework
- 已經支援並可以使用的資料庫有：
  - MS SQL Server
  - Oracle
  - MySQL
  - PostgreSQL
  - SQLite
  - DB2
  - Sybase … etc

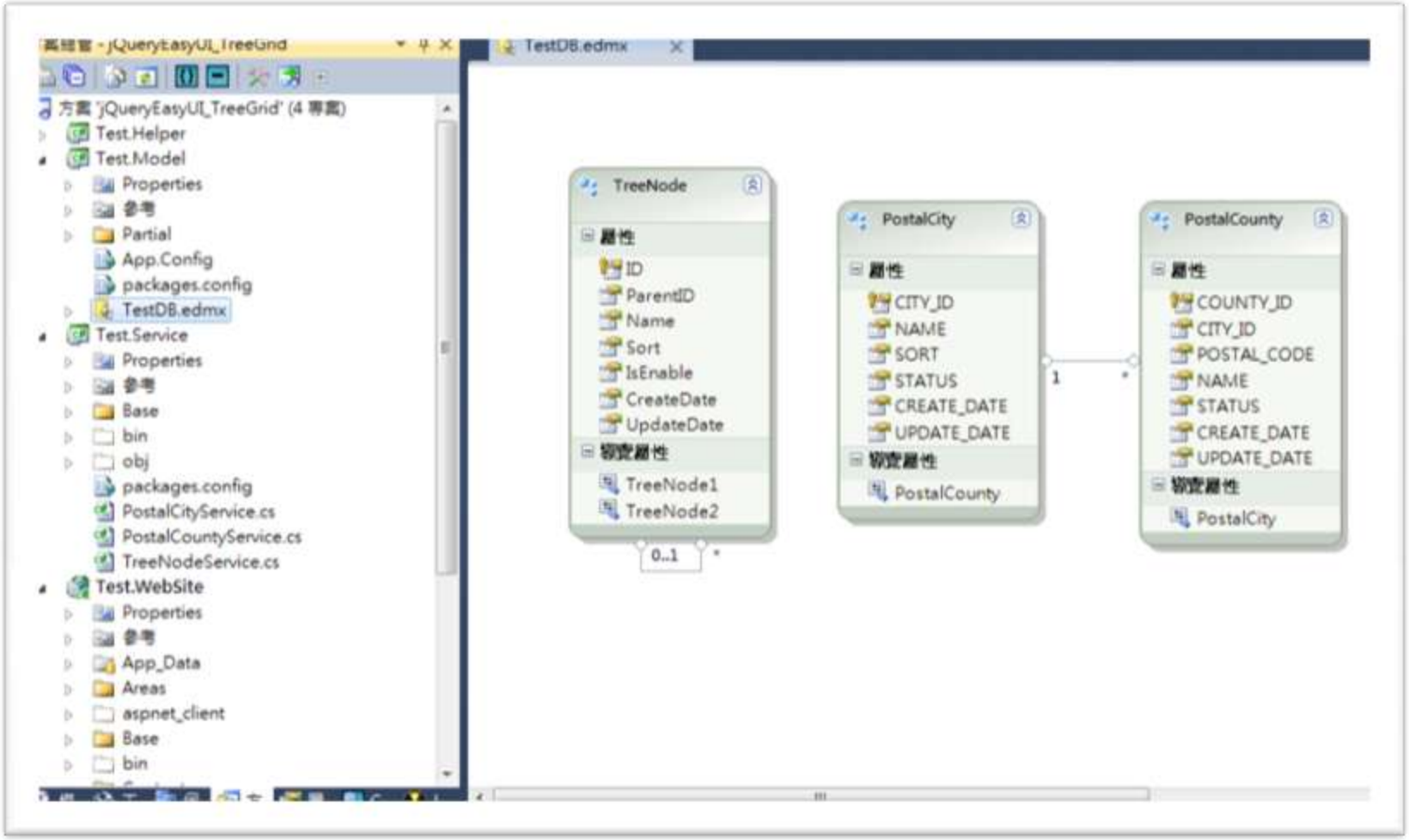
可用哪些東西來當做Model

- NHibernate是Hibernate的.NET版。
- 目前可以支援使用的Database有：
  - MS SQL Server
  - Oracle
  - MySQL
  - DB2
  - Sybase
  - SQLite … etc

ASP.NET MVC專案的Model – 同專案中



ASP.NET MVC專案的Model – 不同專案



ViewModel

- 針對某些頁面的需求而產生的Model
- 專給View所使用
- 類別，其資料成員大多來自底層Model類別的資料成員
- 在MVC網站專案中建立「ViewModels」目錄來存放與管理

ViewModel

```
public class MemberDetailViewModel
{
    public Member Member { get; set; }

    public List<Invoice> InvoiceList { get; set; }

    public Exchange Exchange { get; set; }

    public List<Lottery> Lotteries { get; set; }

    public int TotalAmount { get; set; }

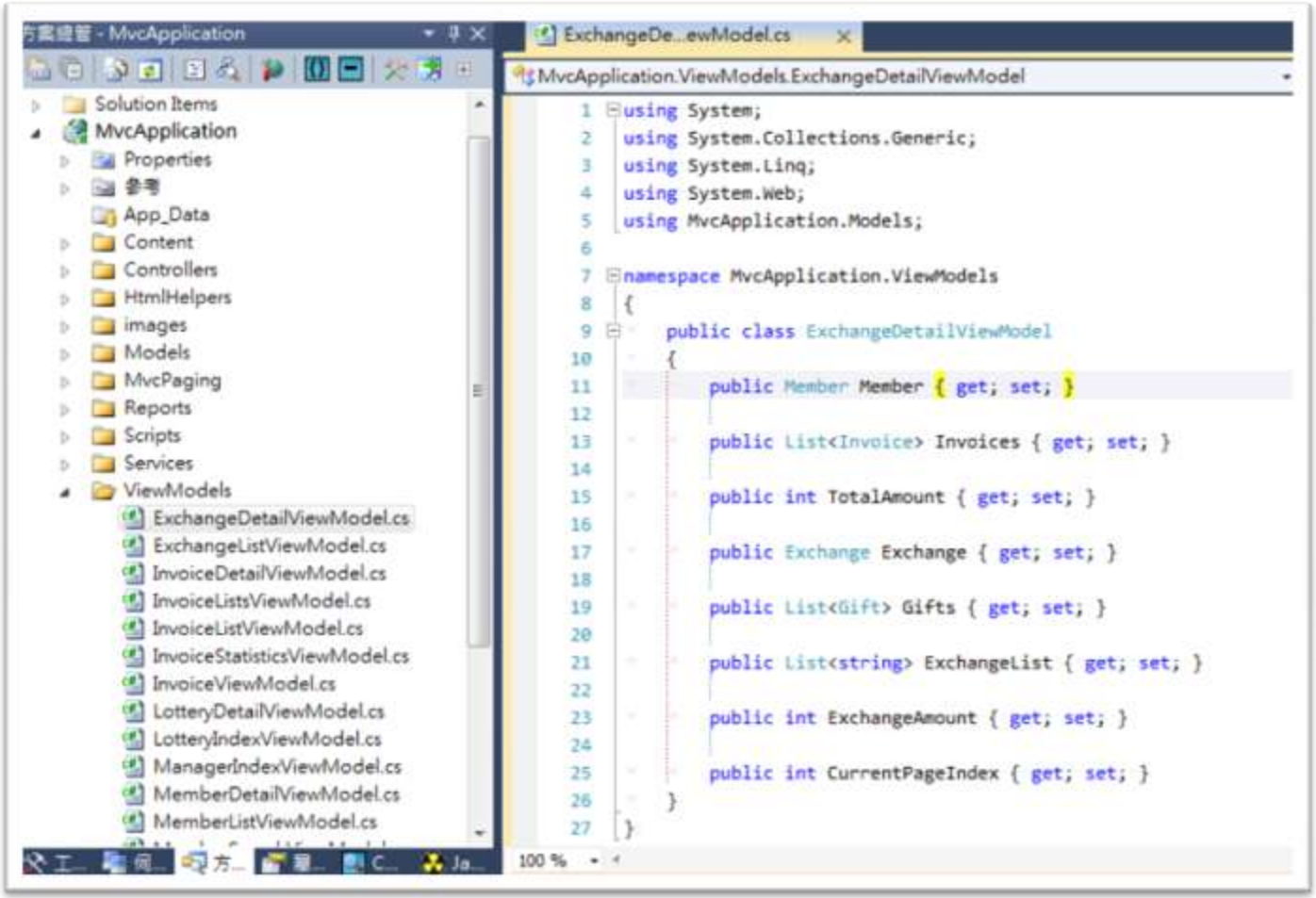
    public int ExchangeAmount { get; set; }

    public string CityName { get; set; }

    public string CountyName { get; set; }

    public int CurrentPageIndex { get; set; }
}
```

ViewModel


方案總管 - MvcApplication
Solution Items
MvcApplication
Properties
參考
App_Data
Content
Controllers
HtmlHelpers
images
Models
MvcPaging
Reports
Scripts
Services
ViewModels
ExchangeDetailViewModel.cs
ExchangeListViewModel.cs
InvoiceDetailViewModel.cs
InvoiceListsViewModel.cs
InvoiceListViewModel.cs
InvoiceStatisticsViewModel.cs
InvoiceViewModel.cs
LotteryDetailViewModel.cs
LotteryIndexViewModel.cs
ManagerIndexViewModel.cs
MemberDetailViewModel.cs
MemberListViewModel.cs
ExchangeDe...ewModel.cs
MvcApplication.ViewModels.ExchangeDetailViewModel
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using MvcApplication.Models;
6
7 namespace MvcApplication.ViewModels
8 {
9 public class ExchangeDetailViewModel
10 {
11 public Member Member { get; set; }
12
13 public List<Invoice> Invoices { get; set; }
14
15 public int TotalAmount { get; set; }
16
17 public Exchange Exchange { get; set; }
18
19 public List<Gift> Gifts { get; set; }
20
21 public List<string> ExchangeList { get; set; }
22
23 public int ExchangeAmount { get; set; }
24
25 public int CurrentPageIndex { get; set; }
26 }
27 }
100 %

ASP.NET Routing







ASP.NET Routing

- ASP.NET Routing是做什麼的
- 套用ASP.NET Routing有什麼好處

各位覺得那一個比較友善?

- Http://Server/Products/ProductView.aspx?Id=123
- Http://Server/Product/View/123

ASP.NET Routing的基本觀念

- Url Pattern
  - 常數
  - /區段
  - {variable} 變數預留位置
  - {*variable} 變數預留位置(catch-all)
- Ignore
- RouteData
- RouteHandler

設定 Routing

- 通常會在Global.asax.cs中設定

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    routes.MapRoute(
        "Default", // Route name
        "{controller}/{action}/{id}", // URL with parameters
        new { controller = "Home", action = "Index", id = UrlParameter.Optional } // Parameter defaults
    );

}
```

常見的配置

Route definition	Example of matching URL
`{controller}/{action}/{id}`	`/Products/show/beverages`
`blog/{entry}`	`/blog/123`
`{reporttype}/{year}/{month}/{day}`	`/sales/2008/1/5`
`{locale}/{action}`	`/US/show`
`{language}-{country}/{action}`	`/en-US/show`
`{version}/{controller}/{action}`	`/v2/Product/Update`

ASP.NET MVC 與 ASP.NET Routing

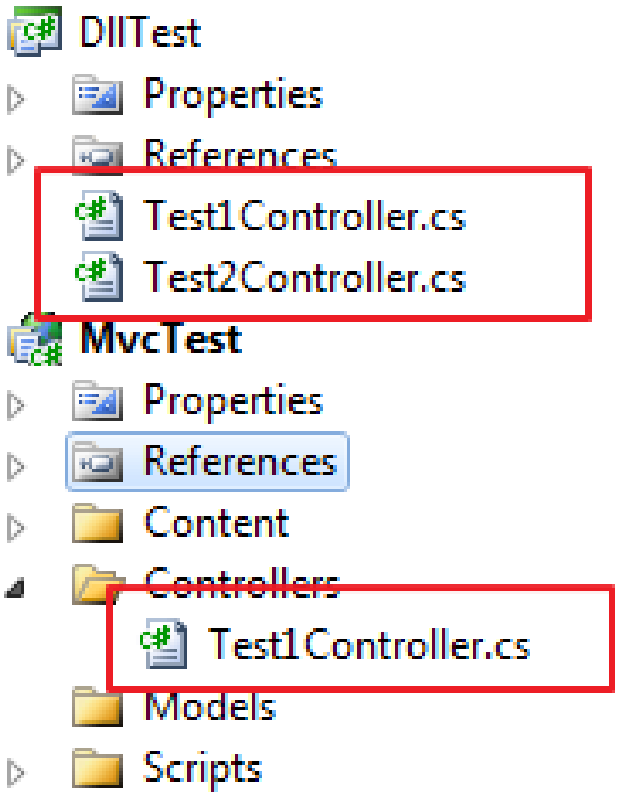
- 它們的關係是??

秤不離錘、錘不離秤

- Url對應執行的Controller/Action
- 由Controller/Action產生Url
  - UrlHelper.Action(Action,Controller)
  - RedirectToActionResult(Action,Controller)等等

ASP.NET Routing的進階觀念

- Controller與Routing
- 衝突



ASP.NET Routing的進階觀念

- Debug
  - RouteDebugger 2.0

ASP.NET Routing的進階觀念

- Areas
  - {Area}/{Controller}/{Action}/{Id}
- 約束

```
routes.MapPageRoute("",
    "Category/{action}/{categoryName}",
    "~/categoriespage.aspx",
    true,
    new RouteValueDictionary
        {{"categoryName", "food"}, {"action", "show"}},
    new RouteValueDictionary
        {{"locale", "[a-z]{2}-[a-z]{2}"},{"year", @"\d{4}"}}
   );
```

ASP.NET Routing的進階觀念

- Areas
  - {Area}/{Controller}/{Action}/{Id}
- 約束
- 新增的順序

```
routes.MapRoute(
    "Default", // Route name
    "{controller}/{action}/{id}", // URL with parameters
    new { controller = "Home", action = "Index", id = UrlParameter.Optional } // Param
);

//如果新增在Default之後永遠都不會執行到，因為Default的Pattern太大了幾乎通吃掉了
routes.MapRoute(
    "Sales清單",
    "sales/{type}", // URL with parameters
    new { controller = "Sale", action = "List", type = "Top" } // Parameter defaults
);
```

休息時間五分鐘

Take a break

ASP.NET MVC 快速開發

輪子先做好







快速開發？

- 沒有任何一種 Web 開發比 ASP.NET Web Form 快
- 那我們要 ASP.NET MVC 幹泌？
  - 因為程式開發中唯一不變的就是「變」

ASP.NET Web Form
就好像F1 它一直都是那麼的快，但是就只有那麼快

ASP.NET MVC
就好像飛機，一開始只是跑再來就會飛了

ASP.NET MVC 如何飛起來

- HTML Helper
- Page Method
- jQuery Plug in

ASP.NET MVC 如何飛起來

- DisplayTemplates
- EditorTemplates

```
[UIHint("Date")]
public DateTime Birthday { get; set; }

@Html.DisplayFor(model => model.Birthday)
```

ASP.NET MVC 如何飛起來

- T4
  - Code Templeat
  - "X:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE\ItemTemplates\CSharp\Web\MVC 3\CodeTemplates"

ASP.NET MVC 如何飛起來

- T4
  - 後台產生器

ASP.NET MVC 如何飛起來

- http://www.microsoft.com/downloads/en/details.aspx?FamilyID=82cbd599-d29a-43e3-b78b-0f863d22811a




一切都是幻覺

學習資源

官方網站

- http://www.asp.net/mvc

官方影片

- http://www.asp.net/mvc/videos

MSDN(CN)影片

- http://goo.gl/FUIh

Scott gu

- http://weblogs.asp.net/scottgu/archive/tags/MVC/default.aspx

Haacked

- http://haacked.com/Tags/aspnetmvc/default.aspx

學習資源

KKBruce

- http://blog.kkbruce.net/p/net-framework.html#allmvc

阿源哥哥

- http://mvc.keigen.net/

小朱

- http://www.dotblogs.com.tw/regionbbs/Tags/ASP.NET MVC

天空的垃圾場

- http://blog.sanc.idv.tw/search/label/ASP.NET%20MVC

Will

- http://blog.miniasp.com/category/ASPNET-MVC.aspx

學習資源



Bibby http://bibby.be



Demo http://demo.tc



Dino http://dinowang.blogspot.com



Jerry http://www.dotblogs.com.tw/lastsecret/



Kevin http://kevintsengtw.blogspot.com



Wade http://www.dotblogs.com.tw/wadehuang36/

twMVC 網站



twMVC 固定聚會

- 每週小型聚會，不特定主題
  - 固定於週四 19:30-21:30，不改時間，逢國定假日暫停一次
  - 議程不會提前排定，以當天參加者最近的心得分享為主



twMVC

- 手上的問券煩請填寫，讓我們下一次會更好。
- 有任何問題歡迎到每週四的定期聚會詢問。

謝謝各位今天的參與。



twMVC

