



# ASP.NET程式設計 常用的ASP.NET控制項(二)

講師：林賢達

Peter.lin@imestech.com



# 課程大綱

- Adrotator控制項
- Calendar控制項
- ImageMap控制項
- MultiView控制項
- Wizard控制項
- FileUpload控制項
- 動態建立控制項
- Web使用者控制項

# Adrotator控制項

- 在網頁上產生廣告輪番的效果
- 使用XML檔案作為廣告資料來源，控制廣告圖片的出現機率以及點選後重新導向的網址
- XML檔案格式

```
<?xml version="1.0" encoding="big5" ?>
```

```
<Advertisements><Ad>
```

```
  <ImageUrl>廣告圖片來源</ImageUrl>
```

```
  <NavigateUrl>點選後，開啟的連結網頁</NavigateUrl>
```

```
  <AlternateText>說明文字</AlternateText>
```

```
  <Keyword>廣告分類 </Keyword>
```

```
  <Impressions>出現機率</Impressions>
```

```
</Ad></Advertisements>
```

# Adrotator控制項

## ■ XML檔案格式(續)

- <Advertisements>為根元素，所有的廣告設定必須包含在<Advertisements>元素內
- 使用<ad>元素設定每個廣告的圖形、連結網頁、說明文字和顯示機率等資訊
- 每次重整網頁時，顯示機率的計算  
假設三個廣告顯示的機率分別為3次、2次及1次，則Impressions值依序為 $3/(3+2+1)$ 約50%， $2/(3+2+1)$ 約33%，以及 $1/(3+2+1)$ 約17%

# Adrotator控制項

## ■ 重要成員

- Target屬性：設定開啟網頁的方式
- AdvertisementFile屬性：指定XML檔案來源
- OnAdCreated事件：當建立一個廣告後，便會觸發此事件

使用AdCreatedEventArgs參數取出AlternateText, ImageUrl, NavigateUrl屬性

或者利用AdProperties("Caption")取出自訂標籤<Caption>公益廣告</Caption>的內容

# Calendar控制項

- 在網頁上顯示一個月曆資訊，讓使用者選取某個有效日期
- 重要成員
  - SelectionMode屬性：設定選取日期的模式
    - Day，只能選取某一天
    - DayWeek，選取某一天或某一週
    - DayWeekMonth，選取某一天、某一週或某個月
    - None，單純顯示日期

# Calendar控制項

## ■ 重要成員(續)

- SelectedDate屬性：設定預設日期或取出選取的日期
- SelectionChanged事件：當使用者選取某一天，某一週或某個月時，便會觸發此事件，常用於配合 SelectedDate屬性取得選取的日期
- DayRender事件：當展現每個日期時，便會觸發此事件，常用於在月曆中自訂節日

# ImageMap控制項

- 在網頁上顯示一張圖形地圖
- 當使用者按下事先定義的作用點 (Hot Spot) 區域時，將產生回傳值到伺服器端或連結到指定的網頁
- 重要成員
  - ImageUrl屬性：指定欲顯示的圖形來源
  - HotSpotMode屬性：指定作用點的行為模式，執行回傳還是連結網頁
  - HotSpots屬性：設定圓形、矩形或多邊形作用點區域



# ImageMap控制項

## ■ 重要成員(續)

- NavigateUrl屬性：欲連結網頁的網址
- PostBackValue屬性：回傳給伺服器的值
- Target屬性：指定連結網頁的開啟方式
- Click事件：當使用者按下圖形地圖時，便會觸發此事件，用於取出回傳值

# MultiView控制項

- 定義View控制項的設計工具，View控制項是Web控制項的容器
- 在網頁上建立多個View控制項，並放置所需的控制項，然後使用MultiView來控制顯示哪一個View，可用於在網頁上製作頁籤的效果
- 重要屬性
  - ActiveViewIndex：使用索引值切換不同的View

# Wizard控制項

- 建立逐步操作的頁面，例如意見調查
- 使用 WizardStep 建立精靈中的每個步驟操作
- 重要成員
  - WizardSteps 屬性：WizardStep 控制項集合
  - StartNextButtonText, StepNextButtonText, StepPreviousButton 屬性：設定按鈕的說明文字
  - DisplaySideBar：是否顯示提示欄位
  - FinishDestinationPageUrl 屬性：指定按下[完成]按鈕後，重新導向的目標網頁Url

# Wizard控制項

## ■ 重要成員(續)

- PreviousButtonClick事件，當按下[上一頁]按鈕後，便會觸發此事件
- NextButtonClick事件，當按下[下一頁]按鈕後，便會觸發此事件
- FinishButtonClick事件，當按下[完成]按鈕後，便會觸發此事件

# FileUpload控制項

- 上傳檔案到Web伺服器
- 重要成員
  - HasFile：是否指定上傳檔案
  - FileName：取出檔案名稱
  - SaveAs：儲存上傳的檔案到指定路徑

# 動態建立控制項

- 使用 Panel 或 Placeholder 控制項作為動態建立 Web 控制項的容器

```
Dim Label1 As New Label
```

```
Label1.Text="Test 123"
```

```
Panel1.Controls.Add(Label1)
```

```
Dim Control1 As Control
```

```
Control1=Page.ParseControl("<asp:Label id='Label1'  
Text='Test 123' />")
```

```
Panel1.Controls.Add(Control1)
```

# 動態建立控制項

- 使用 `Page.FindControl("ControlID")` 取出動態建立的 control 項
- 建立與事件處理程序的關聯
- 使用 `Table` control 項來定位 control 項

# 開發自訂控制項

- 具有與ASP.NET網頁相同的生命週期，可使用Page\_Load事件初始化自訂控制項
- VS 2005提供開發自訂控制項的視覺化設計工具
- 使用者自訂控制項的好處
  - 重覆使用
  - 降低開發與維護成本



# 開發自訂控制項

- 使用者自訂控制項的分類
  - 使用者控制項：使用標準控制項整合其功能成為一個新的控制項
  - 組合控制項：使用標準控制項組合成一個新的控制項
  - 自訂控制項：直接衍生自Control覆寫Render方法
- 使用者控制項與組合控制項之間主要的差別在於使用者控制項是封裝在ascx檔案，而組合控制項則是封裝在dll組件

# 使用者控制項

## ■ 開發步驟

- 使用[Web使用者控制項]範本建立ascx檔案，不可包含<Html>、<Body>及<Form>HTML標籤
- 使用標準控制項建立使用者介面
- 撰寫公開的屬性和方法，透過這些屬性和方法在網頁和使用者控制項之間傳遞訊息
- 編譯

# 使用者控制項

## ■ 測試步驟

- 使用Registry指令在網頁上註冊使用者控制項

```
<%@Registry Src="MyUserControl.ascx"  
TagName="MyUserControl" TagPrefix="uc1" %>
```

- 在aspx檔案中宣告或使用LoadControl 動態載入使用者控制項

```
<uc1:MyUserControl ID="MyUserControl1"  
RunAt="Server" />
```

或者

```
Page.Form.Controls.Add(Page.LoadControl("MyUserControl1.ascx"))
```