



ASP.NET 程式設計 資料存取

講師：林賢達

Peter.lin@imestech.com



課程大綱

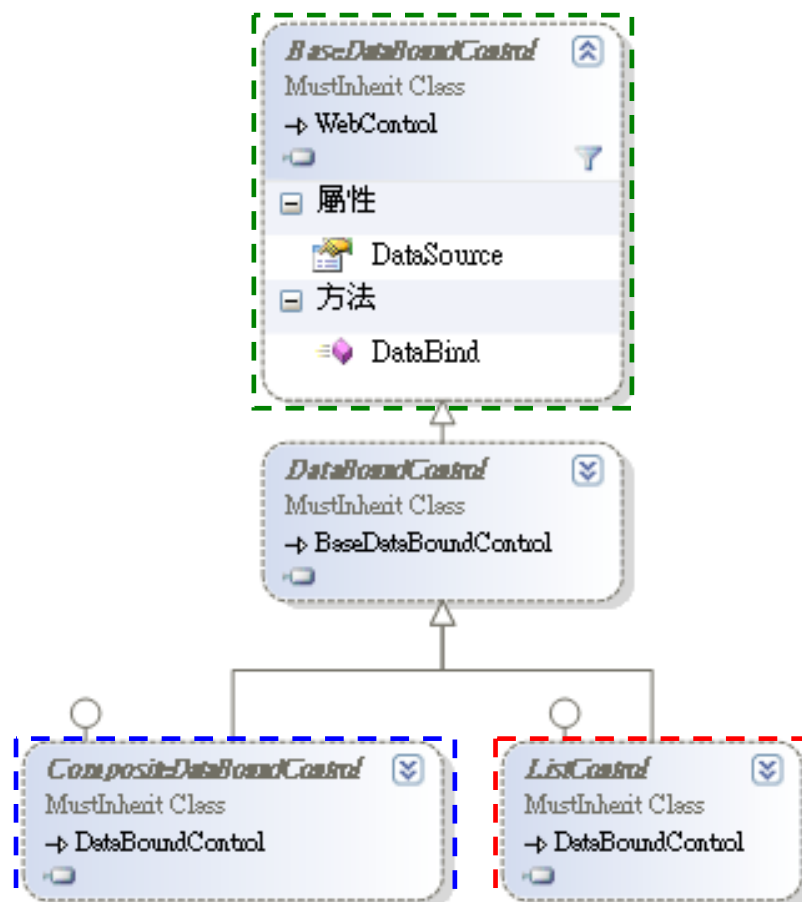
- 資料繫結簡介
- 資料來源控制項
- 資料繫結控制項
 - GridView控制項
 - DetailsView控制項

資料繫結簡介

- 將資料來源的資料顯示在控制項的過程
- 支援資料繫結的控制項
 - DataSource屬性
 - DataBind方法
- 資料繫結控制項
 - 資料顯示：DropDownList, ListBox, CheckBoxList, RadioButtonList, BulleteList等
 - 資料顯示和異動：GridView, DetailView, FormView等

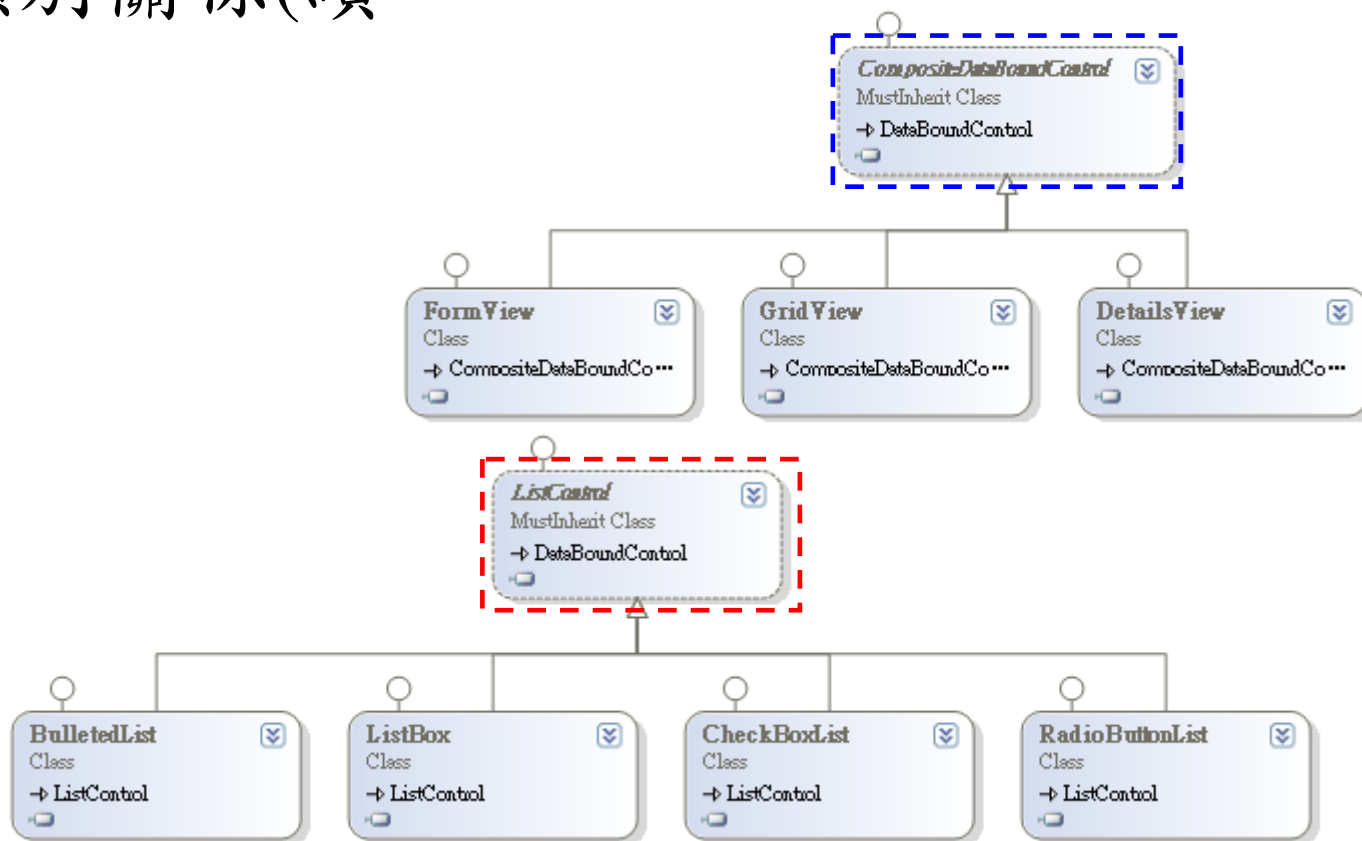
資料繫結控制項

■ 類別關係



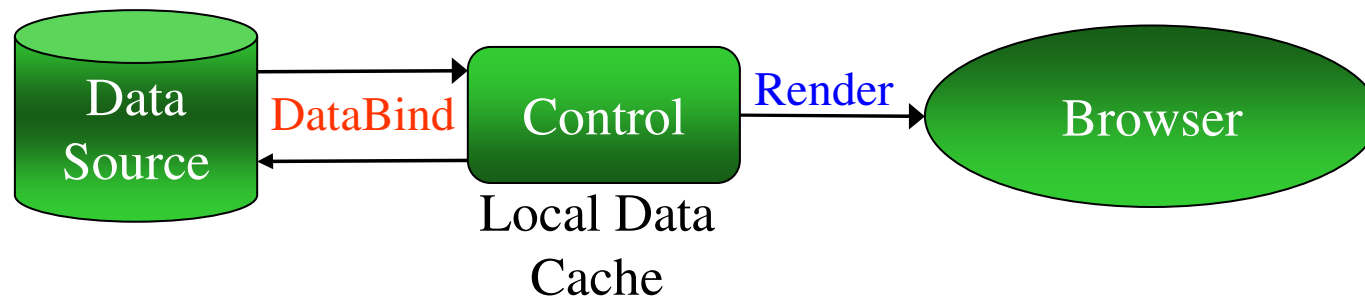
資料繫結控制項

■ 類別關係(續)



資料繫結簡介

- 初始化資料來源，將資料來源指定給 DataSource 屬性，然後呼叫 DataBind 方法進行資料繫結



- 支援的資料來源，例如陣列, 物件, DataSet, XML, 資料庫和資料來源控制項等

資料來源控制項

- 提供從資料來源取出資料，顯示在資料繫結控制項，提供新增、修改、刪除和排序功能
- [資料來源]與[資料繫結控制項]之間的橋梁
- 將[資料來源控制項]指定給[資料繫結控制項]的DataSourceID屬性即可顯示資料



資料來源控制項

■ 主要的資料來源控制項

- SqlDataSource：存取所有支援SQL語法的資料庫，例如SQL Server, Oracle和MS Access等
- ObjectDataSource：存取自訂物件
- XmlDataSource：存取XML檔案

SqlDataSource控制項

- 建立資料庫連接
- 提供取出、新增、修改和刪除資料的功能
 - SelectCommand
 - InsertCommand, UpdateCommand, DeleteCommand

```
<asp:SqlDataSource ID="SqlDataSource1" runat="server"  
    ConnectionString="<%= $ConnectionStrings:testdbConnectionString! %>"  
    SelectCommand="SELECT * FROM [employee]"></asp:SqlDataSource>
```

```
<connectionStrings>  
    <add name="testdbConnectionString"  
        connectionString="Data Source=D820;Initial Catalog=testdb;Integrated Security=True"  
        providerName="System.Data.SqlClient"/>  
</connectionStrings>
```

SqlDataSource控制項

■ 重要屬性

- ConnectionString：資料庫連接字串
- SelectCommand：欲執行的查詢命令
- InsertCommand：欲執行的新增命令
- UpdateCommand：欲執行的更新命令
- DeleteCommand：欲執行的刪除命令
- DataSourceMode：指定資料來源的類型是DataSet或者DataReader(唯讀)

SqlDataSource控制項

■ 支援SQL參數，從相關的控制項取出參數值

```
<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConflictDetection="None"
ConnectionString="<%= $ConnectionString:testdbConnectionString %>"
InsertCommand="INSERT INTO [employee] ([EmpNo], [Name], [Sex], [BirthDay], [Height], [DeptNo], [Salary], [Weight], [original_EmpNo], [original_Name], [original_Sex], [original_Birthday], [original_Height], [original_DeptNo], [original_Salary], [original_Weight]) VALUES (@EmpNo, @Name, @Sex, @BirthDay, @Height, @DeptNo, @Salary, @Weight, @original_EmpNo, @original_Name, @original_Sex, @original_Birthday, @original_Height, @original_DeptNo, @original_Salary, @original_Weight)"
OldValuesParameterFormatString="original_{0}"
SelectCommand="SELECT [EmpNo], [Name], [Sex], [BirthDay], [Height], [DeptNo], [Salary], [Weight], [original_EmpNo], [original_Name], [original_Sex], [original_Birthday], [original_Height], [original_DeptNo], [original_Salary], [original_Weight] FROM [employee]"
UpdateCommand="UPDATE [employee] SET [Name] = @Name, [Sex] = @Sex, [BirthDay] = @BirthDay, [Height] = @Height, [DeptNo] = @DeptNo, [Salary] = @Salary, [Weight] = @Weight, [original_EmpNo] = @original_EmpNo, [original_Name] = @original_Name, [original_Sex] = @original_Sex, [original_Birthday] = @original_Birthday, [original_Height] = @original_Height, [original_DeptNo] = @original_DeptNo, [original_Salary] = @original_Salary, [original_Weight] = @original_Weight"
<DeleteParameters>...</DeleteParameters>
<UpdateParameters>
  <asp:Parameter Name="Name" Type="String" />
  <asp:Parameter Name="Sex" Type="Boolean" />
  <asp:Parameter Name="BirthDay" Type="DateTime" />
  <asp:Parameter Name="Height" Type="Single" />
  <asp:Parameter Name="DeptNo" Type="String" />
  <asp:Parameter Name="Salary" Type="Decimal" />
  <asp:Parameter Name="Weight" Type="Single" />
  <asp:Parameter Name="original_EmpNo" Type="String" />
  <asp:Parameter Name="original_Name" Type="String" />
  <asp:Parameter Name="original_Sex" Type="Boolean" />
  <asp:Parameter Name="original_Birthday" Type="DateTime" />
  <asp:Parameter Name="original_Height" Type="Single" />
  <asp:Parameter Name="original_DeptNo" Type="String" />
  <asp:Parameter Name="original_Salary" Type="Decimal" />
  <asp:Parameter Name="original_Weight" Type="Single" />
</UpdateParameters>
<InsertParameters>...</InsertParameters>
</asp:SqlDataSource>
```

SqlDataSource控制項

■ 支援快取(Cache)查詢結果

- EnableCaching：是否啟用快取
- CacheDuration：指定快取過期期限(以秒為單位)
- CacheExpirationPolicy：指定快取的期限策略是絕對(absolute)或者相對(Sliding)
- CacheKeyDependency：快取相依的索引鍵，當鍵值改變時，快取便失效
- SqlCacheDependency：快取相依的資料表，當資料表的資料改變時，快取便失效

GridView控制項

- 以HTML表格的形式來顯示資料
- GridView的組成
 - 標題資料列(Header)
 - 資料列(DataRow)
 - 顯示頁面按鈕的資料列(EmptyDataRow)
 - 頁尾資料列(Footer)
 - 分頁(Pager)，顯示頁面巡覽控制項
 - 資料列的分隔符號(Separator)

GridView控制項

■ 欄位類型

- BoundField：顯示欄位資料
- CheckBoxField：顯示Boolean型別的欄位資料
- HyperLinkField：顯示超連結
- ButtonField：顯示命令按鈕
- CommandField：顯示[編輯、更新、取消][選取]或者[刪除]等命令按鈕
- ImageField：顯示圖片
- TemplateField：顯示自訂的HTML或ASP.NET控制項模板

GridView控制項

■ 重要事件

- RowDataBound：與資料列進行資料繫結時，便會觸發此事件，常用於當滿足某個條件時，改變資料列的顯式樣式或執行欄位運算式
- RowCreated：當建立資料列時，便會觸發此事件，常用於設定每個資料列的樣式，例如當游標在GridView上移動時，改變資料列的背景顏色
- RowCommand：當按下[命令欄位]時，便會觸發此事件。使用e.CommandName可知是由哪一個命令按鈕觸發此事件

GridView控制項

■ 格式化輸出

- 自動格式化

- 自訂樣式：ControlStyle, FooterStyle, HeaderStyle, ItemStyle...屬性

- 使用面板(skin)

- 在RowDataBound事件中，自行處理輸出格式

■ DataFormatString屬性，根據資料類型設定套用的輸出格式，例如{0:c}，其中c表示金額

GridView控制項

- 分頁，將大量資料分割成多個頁面來顯示，每頁顯示固定的記錄筆數
- 相關成員
 - AllowPaging：是否啟用GridView的分頁功能
 - PageSize：每頁顯示的記錄筆數
 - PagerSettings和PagerStyle：分頁UI設定和樣式
 - PageIndexChanged事件

GridView控制項

■ 排序

- 使用某個欄位的資料來排序資料，以便更快地找到所需的資料，例如以員工編號來排序員工基本資料

■ 相關屬性

- AllowSorting：是否使用欄位標題來排序資料
- SortExpression：與欄位關聯的排序運算式

GridView控制項

- 顯示圖片，在GridView中加入ImageFiled欄位，並指定其繫結的物件
- 使用模板欄位(TemplateField)作為HTML或者ASP.NET Server Control的容器
 - 編輯ItemTemplate, AlternatingItemTemplate, EditItemTemplate, HeaderTemplate和FooterTemplate
- 使用DataBinder.Eval方法取出DataRow中的欄位值

GridView控制項

■ 顯示Master Detail資料

- 在資料庫中存在一對多的資料關聯，例如一筆訂單包含多筆的訂單項次
- 將 Detail資料來源控制項的參數設定為Master資料繫結控制項

加入 WHERE 子句

在陳述式的 WHERE 子句加入一或多項條件。您可以對每一項條件指定常值或參數型的值。參數型的值會在執行階段時依據它們的屬性取得它們的值。

資料行 (C):
EmpNo

運算子 (O):
=

來源 (S):
Control

SQL 運算式:
[EmpNo] = @EmpNo

WHERE 子句 (W):

SQL 運算式	值
---------	---

參數屬性

控制項 ID (ID):
DropDownList1

預設值 (V):

值:
DropDownList1.Selected.Value

加入 (A)

刪除 (D)

確定 取消

GridView控制項

- 新增、修改和刪除資料
- 資料來源控制項
 - 必須顯示主索引欄位
 - 產生Insert,Update,Delete SQL語法
 - DataSourceMode必須是DataSet
- GridView控制項
 - 設定DataKeyNames屬性為主索引欄位
 - 啟用編輯和刪除功能即可

Details View控制項

- 每次只顯示一筆記錄
- 支援新增, 修改和刪除資料
- 建立新增資料網頁
 - 建立SqlDataSource，指定資料庫並產生新增命令
 - 建立DetailView，與SqlDataSource關聯，並啟動插入功能
 - 設定DetailView的DefaultMode屬性為Insert
 - 在DetailsView的ItemInserted事件程序，返回前個網頁