

Mục tiêu

- Sử dụng thành thạo các điều khiển thuộc nhóm Data
- Tạo trang web có dạng Master-Details

Nội dung

1. Tổng quan
2. Các điều khiển Danh sách
3. Điều khiển GridView
4. Các điều khiển DataSource
5. Điều khiển DataList
6. DetailsView và FormView

1. Tổng quan

- Phân loại
 - Điều khiển Danh sách - List Controls
 - Điều khiển liên kết dữ liệu – Data-Bound controls
 - Điều khiển nguồn dữ liệu – Data-Source controls
- Các điều khiển này có khả năng liên kết với một nguồn dữ liệu
- Nguồn dữ liệu dùng để liên kết: ArrayList, SortedList, HashTable, DataSet, DataTable, DataView, DataReader, ...
- Thao tác liên kết dữ liệu của điều khiển List và Data-Bound: 2 cách
 - Lúc thực thi: thường sử dụng thuộc tính DataSource của điều khiển
 - Lúc thiết kế: sử dụng các điều khiển Data source (AccessDataSource, SqlDataSource, ...) kết hợp với thuộc tính DataSourceID

2. Các điều khiển Danh sách

- ListBox và DropDownList
 - Dùng để hiển thị danh sách
 - Items: chứa tập hợp các mục chọn, có thể thêm vào lúc thiết kế (Design) hoặc lúc thực thi (Run-time)
 - Rows: số dòng hiển thị của ListBox
 - SelectionMode: cách thức chọn các mục trong ListBox (Single/Multi)
 - DataTextField: tên field có giá trị thể hiện trên điều khiển
 - DataValueField: tên field có giá trị lưu trữ
 - AutoPostBack: có tự độngPostBack về Server khi mục chọn của điều khiển bị thay đổi (True/False)
 - Một số thuộc tính lúc thực thi:
 - SelectedIndex: chỉ số của mục được chọn
 - SelectedItem: mục được chọn (kiểu ListItem)
 - SelectedValue: giá trị của mục được chọn
 - Sự kiện SelectedIndexChanged: xảy ra khi thay đổi dòng được chọn
 - Một số thao tác:
 - Thêm mới một mục vào cuối danh sách Items.Add(<String>) hoặc Items.Add(<ListItem>)
 - Thêm mới một mục tại một vị trí xác định Items.Insert(<Index>,<ListItem>) hoặc Items.Insert(<Index>,<String>)
 - Xóa một mục ra khỏi danh sách Items.Remove(<ListItem>) hoặc Items.Remove(<String>)

- Xóa tất cả các mục trong danh sách Items.clear
- Kiểm tra một mục đã có trong danh sách chưa Items.Contains(<ListItem>) As Boolean hoặc Items.IndexOf(<ListItem>) As Integer
- Lấy giá trị tổng số mục có trong danh sách Items.Count
- CheckBoxList và RadioButtonList
 - RepeatColumns: qui định số cột hiển thị
 - RepeatDirection: qui định hình thức hiển thị (Vertical/Horizontal)
 - Các thao tác thêm, xóa các mục chọn: giống ListBox
- BulletedList
 - BulletStyle: style dùng làm Bullet
 - BulletImageUrl: URL của tập tin hình ảnh, phải có BulletStyle=CustomImage
 - DisplayMode: hình thức thể hiện (Text, HyperLink, LinkButton)
 - Các thao tác thêm, xóa các mục chọn: giống ListBox
- Thao tác liên kết dữ liệu

```

1- ArrayList <nguồn> = new ArrayList(); //Khởi tạo nguồn
<nguồn>.Add(Object);                    //Thêm dữ liệu
2- <đ.k>.DataSource=<nguồn>;            //Khai báo nguồn dữ liệu
3- <đ.k>.DataTextField="Field";         //Khai báo field thể hiện
4- <đ.k>.DataValueField="Field"; //Khai báo field lưu trữ
5- <đ.k>.DataBind();                    //Liên kết
//Dòng 3, 4 có thể bỏ qua tùy vào nguồn dữ liệu

```

3. Điều khiển GridView

- Giới thiệu
 - Dùng hiển thị dữ liệu dạng dòng-cột (Tabular)
 - Thể hiện trên trình duyệt dưới dạng HTML Table
 - Đặc điểm:
 - Hỗ trợ cơ chế sắp xếp dữ liệu
 - Hỗ trợ cơ chế phân trang dữ liệu
 - Có thể cập nhật dữ liệu trực tiếp trên lưới

DANH SÁCH SÁCH

Mã	Tên sách	Đơn giá	Ngày cập nhật
1	Giáo trình Tin học cơ bản	26,000	17/07/2004
2	Giáo trình Tin học văn phòng	12,000	05/06/2004
3	Lập Trình Cơ Sở Dữ Liệu Với Visual Basic 2005 Và ADO.NET 2.0	11,500	12/07/2004
4	Visual Basic 2005 Tập 3, Quyển 2: Lập Trình Web Với Cơ Sở Dữ Liệu	20,000	18/07/2004
5	Giáo trình Tin học quản lý 1	21,000	18/06/2004
6	Giáo trình Tin học quản lý 2	21,000	15/05/2004
7	Giáo trình SQL Server	18,000	16/08/2004
8	Giáo trình Phân tích hệ thống	61,000	15/07/2003
9	Giáo trình VB.Net 1	25,000	11/02/2003
10	Giáo trình VB.Net 2	23,000	12/04/2002








1 2 3 4 5

- Liên kết dữ liệu và định dạng
 - Thao tác liên kết dữ liệu lúc thực thi:
 - <GridView>.DataSource = <Nguồn dữ liệu>;
 - <GridView>.DataBinding();
 - Thuộc tính định dạng:
 - HtmlEncoded(True/False): cho phép mã hóa dưới dạng HTML không
 - DataFormatString: định dạng cho giá trị của field liên kết
- Các xử lý sắp xếp trên GridView
 - Khai báo t.t AllowSorting = True
 - Khai báo tên cột cần sắp xếp trong ô Sort expression
 - Sử dụng sự kiện Sorting để xử lý:
 - Liên kết dữ liệu và sắp xếp theo cột cần sắp xếp
 - Giá trị e.SortExpression cho biết thông tin của cột được chọn
- Các xử lý phân trang trên GridView
 - Khai báo t.t AllowPaging = True
 - Khai báo t.t PageSize = <Số dòng hiển thị trên một trang>
 - Sử dụng sự kiện PageIndexChanged để xử lý: <GridView>.PageIndex = e.NewPageIndex ; //Liên kết dữ liệu
 - Giá trị e.NewPageIndex cho biết chỉ số trang được chọn
 - Thuộc tính PageIndex: chỉ số trang hiện hành trên lưới

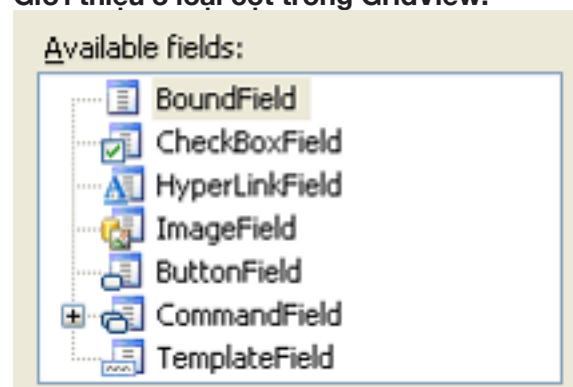
DANH SÁCH KHÁCH HÀNG

<i>Họ khách hàng</i>	<i>Tên đăng nhập</i>	<i>Địa chỉ</i>	<i>Ngày sinh</i>
Phạm Văn Khoa	pvkhoa	Trần Huy Liệu	08/07/1962
Nguyễn Tiến Luân	ntlun	Quận 6	15/07/1974
Đặng Quốc Hòa	dqhoa	Sư Vạn Hạnh	24/07/1965
Ngô Ngọc Ngân	nnngan	Khu chung cư	15/08/1946
Đỗ Quỳnh Hương	dqhuong	Cổng Quỳnh	24/03/1982
1 2 3			

- Các xử lý tùy biến cột trên Gridview
 - Bước 1: khi thiết kế, cột thể hiện có loại là TemplateField
 - Bước 2: dành cho xử lý đơn giản
 - Chèn một điều khiển vào cột TemplateField, liên kết DL với điều khiển này
 - Khai báo biểu thức trong ô Custom binding => Code Expression
 - Ví dụ: hiển thị cột Giới_tính là “Nam” hay “Nữ” ((bool) Eval("Giới_tính") == false?"Nữ":"Nam")
 - Bước 2: dành cho xử lý phức tạp, viết code trong sự kiện RowDataBound
 - RowDataBound: xảy ra khi lưới được liên kết dữ liệu (DataBinding). Ứng với mỗi dòng, xảy ra sự kiện này
 - e.Row.RowIndex: chỉ số của dòng hiện hành (Header /Footer= -1)
 - e.Row.RowType: loại của mỗi dòng (Dòng chuẩn, lề, ...)
 - Bước 2: dành cho xử lý phức tạp, viết code trong sự kiện RowDataBound
 - e.Row.FindControl("Tên điều khiển"): tìm kiếm điều khiển trên dòng hiện hành
 - DataBinder.Eval(e.Row.DataItem, "Tên cột"): truy xuất dữ liệu của cột trên dòng hiện hành

DANH SÁCH KHÁCH HÀNG				
Họ tên	Ngày sinh	Giới tính		Địa chỉ
Phạm Văn Khoa	08/07/1962	Nam		Trần Huy Liệu
Nguyễn Tiến Luân	15/07/1974	Nam		Quận 6
Đặng Quốc Hòa	24/07/1965	Nam		Sư Vạn Hạnh
Ngô Ngọc Ngân	15/08/1946	Nam		Khu chung cư
Đỗ Quỳnh Hương	24/03/1982	Nữ		Cổng Quỳnh
Trần Thị Thu Trang	05/08/1962	Nữ		Nơ Trang Long
Nguyễn Thiên Thanh	04/04/1979	Nữ		Hai Bà Trưng
1 2				

Giới thiệu 3 loại cột trong GridView:



- ImageField
 - DataImageUrlField: chứa Image URL liên kết với field
 - DataImageUrlFormatString: định dạng cho giá trị liên kết với thuộc tính DataImageUrlField

```
//Ví dụ
DataImageUrlField: Hình_minh_hoa
DataImageUrlFormatString: ~/Hinh_du_lieu/{0}.jpg
```

- HyperLinkField

- DataNavigateUrlFields: tên field liên kết với thuộc tính NavigateUrl của hyperlink
 - DataNavigateUrlFormatString: định dạng cho giá trị liên kết với thuộc tính DataNavigateUrlFields
 - DataTextField: field có nội dung được hiển thị
- ButtonField
 - ButtonType: kiểu nút lệnh (Button/Link/Image)
 - CommandName: tên kết hợp với nút lệnh
 - DataTextField: field có nội dung được hiển thị
 - DataTextFormatString: định dạng chuỗi thể hiện trên nút
 - Sự kiện RowCommand: xảy ra khi nhấn nút lệnh

CẬP NHẬT THÔNG TIN NHÀ XUẤT BẢN					
Tên nhà xuất bản	Địa chỉ	Điện thoại			
NXB Thống kê	Đồng Nai	19001511	Chọn	X	Sửa
Kim đồng	Tp.HCM	19001570	Chọn	X	Sửa
Đại học quốc gia	Tp.HCM	0908419981	Chọn	X	Sửa
Văn hóa nghệ thuật	Đà Nẵng	0903118833	Chọn	X	Sửa
Văn hóa	Bình Dương	0913336677			Ghi Không
Lao động - Xã hội	Tp.HCM	0989888888	Chọn	X	Sửa
Khoa Học & Kỹ Thuật	Hà Nội	8201552	Chọn	X	Sửa
NXB Tài Chính	Huế		Chọn	X	Sửa
NXB Phụ Nữ	Tp.HCM	8351056	Chọn	X	Sửa

Cập nhật dữ liệu:

Yêu cầu đầu tiên: khai báo các field dùng làm khóa cho lưới <GridView>.DataKeyNames = new string[] { "tên field1", ... };

- Giai đoạn 1: thiết kế, chọn loại cột thể hiện là CommandField. Thuộc tính CommandName của các nút sau có giá trị mặc định theo chức năng tương ứng:
 - Nút Delete => CommandName = "Delete"
 - Nút Edit => CommandName = "Edit"
 - Nút Update => CommandName = "Update"
 - Nút Cancel, Select
- Giai đoạn 2: xử lý. Sử dụng các sự kiện sau:
 - RowDeleting: xảy ra khi nhấn nút Delete, trước khi GridView xóa dòng
 - RowEditing: xảy ra khi nhấn nút Edit, trước khi GridView chuyển sang chế độ edit
 - RowUpadating: xảy ra khi nhấn nút Update, trước khi GridView cập nhật
 - RowCanceling: xảy ra khi nhấn nút Cancel, trước khi GridView thoát khỏi chế độ edit
 - SelectedIndexChanged: khi nhấn nút Select
- Giai đoạn 2: xử lý. Sử dụng các thuộc tính thường dùng sau:
 - <GridView>.EditIndex: chỉ số dòng được sửa ở trạng thái edit
 - e.NewEditIndex: chỉ số của dòng đang sửa (sự kiện RowEditing)
 - e.RowIndex: chỉ số của dòng hiện hành e.Rows[n1].Cells[n2].Controls[n3]: nhận điều khiển
 - <GridView>.DataKeys[n]: truy xuất giá trị khóa tại dòng có chỉ số là n

- Giai đoạn 2: xử lý. Có các chức năng sau:
 - Chọn dòng
 - Sửa: chuyển GridView sang trạng thái edit
 - Không: khôi phục dữ liệu ban đầu và trả về trạng thái ReadOnly
 - Ghi: cập nhật dữ liệu vào CSDL và trả về trạng thái ReadOnly

4. Các điều khiển DataSource

Giới thiệu

- Là các điều khiển dùng để chứa nguồn dữ liệu được rút trích từ các loại CSDL (Access, Sql Server, ...)
- Dùng để liên kết với các điều khiển Data-Bound và khi liên kết thì chúng sẽ hỗ trợ các tính năng sau:
 - Hỗ trợ phân trang
 - Hỗ trợ sắp xếp tăng/giảm tự động
 - Cập nhật dữ liệu
 - Tất cả các xử lý trên gần như không cần viết code

Các điều khiển

- AccessDataSource: kết nối đến tập tin CSDL Access
- SqlDataSource : kết nối đến CSDL SQL Server, Oracle, ...
- XmlDataSource : kết nối đến tập tin XML
- ObjectDataSource : kết nối đến một object (class, ...) hoặc Typed-DataSet đặt trong thư mục Bin | App_Code
- SitemapDataSource : liên kết đến tập tin .sitemap

Các thuộc tính cơ bản

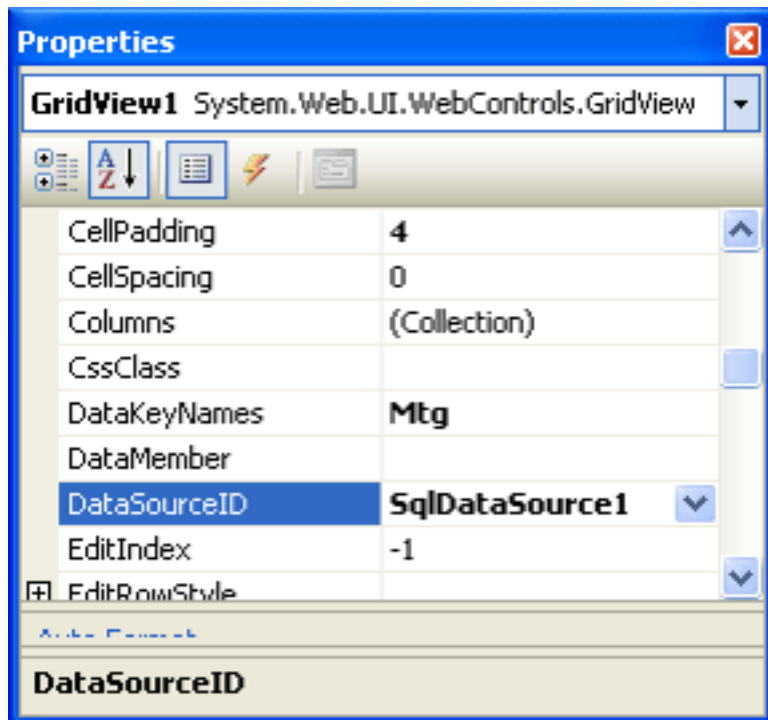
- DataFile: tập tin CSDL Access (đường dẫn đầy đủ)
- ConnectionString: chuỗi liên kết với các CSDL(SQL Server, Oracle, ...)
- SelectQuery: truy vấn chọn lựa
- InsertQuery: truy vấn thêm mẫu tin
- UpdateQuery: truy vấn cập nhật
- DeleteQuery: truy vấn xóa

Điều khiển SqlDataSource

- Thao tác cấu hình:
- Chọn điều khiển, nhấn phím phải => configure Data Source
- Khai báo Connection Chọn Data Source
- Chọn Microsoft SQL Server
- Khai báo Server name và CSDL
- Cấu hình câu lệnh Select và phát sinh các câu lệnh cập nhật
- Kiểm tra và kết thúc

Sử dụng SqlDataSource với GridView

Khai báo thuộc tính DataSourceID của GridView



Sử dụng SqlDataSource với GridView Lọc dữ liệu

- FilterExpression: biểu thức lọc, chỉ áp dụng khi thuộc tính DataSourceMode là DataSet
- FilterParameters: các tham số tham gia trong thuộc tính
- FilterExpression SelectCommandType (Text/StoredProcedure): kiểu tạo truy vấn chọn lựa (dùng câu lệnh SQL hay thủ tục nội)

Sử dụng SqlDataSource với GridView Cập nhật dữ liệu

- Hủy: viết câu lệnh truy vấn xóa có tham số tại thuộc tính DeleteQuery. Có thể bổ sung phần kiểm tra trước khi hủy trong sự kiện RowDeleting
- Cập nhật: viết câu lệnh truy vấn cập nhật có tham số tại thuộc tính UpdateQuery
 - Có thể bổ sung phần kiểm tra dữ liệu trước khi cập nhật trong sự kiện RowUpdating
 - Sử dụng thuộc tính e.NewValues["tên field"] để kiểm tra

```
protected void GridView4_RowUpdating(object sender, GridViewUpdateEventArgs e)
{
    //Kiểm tra Tên chủ đề khác ""
    if (string.IsNullOrEmpty((string)e.NewValues["Ten_chu_de"]))
        e.Cancel = true;
}
```

5. Điều khiển DataList

Giới thiệu

- Dùng để hiển thị dữ liệu theo nhiều dạng
- Thể hiện trên trình duyệt dưới dạng HTML Table
- Đặc điểm:
 - Không hỗ trợ cơ chế sắp xếp dữ liệu, phân trang
 - Có thể cập nhật dữ liệu trực tiếp trên danh sách
 - Phải tự thiết kế hình thức

SÁCH MỚI CẬP NHẬT



Giáo trình Tin học cơ bản

Đặt mua



Giáo trình Tin học văn phòng

Đặt mua



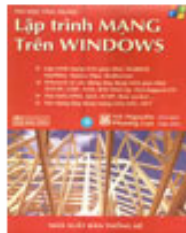
Lập Trình Cơ Sở Dữ Liệu Với Visual Basic 2005 Và ADO.NET 2.0

Đặt mua



Visual Basic 2005 Tập 3, Quyển 2: Lập Trình Web Với Cơ Sở Dữ Liệu

Đặt mua



Giáo trình Tin học quản lý 1

Đặt mua



Giáo trình Tin học quản lý 2

Đặt mua

Liên kết và định dạng

- Thao tác liên kết dữ liệu: giống GridView
- Thao tác thiết kế: giống như cột TemplateField của GridView
- Định dạng: giống GridView
 - RepeatDirection(Horizontal/Vertical): hướng hiển thị dữ liệu
 - RepeatColumns: số cột hiển thị

Cập nhật dữ liệu datalist

- Yêu cầu đầu tiên: khai báo field dùng làm khóa cho danh sách <DataList>.DataKeyField = "tên field";
- Bước 1: thiết kế, bổ sung các điều khiển trong thành phần EditItemTemplate. Nên khai báo thuộc tính CommandName của các nút chức năng:
 - Nút Edit => CommandName = "Edit"
 - Nút Update => CommandName = "Update"
 - Nút Cancel => CommandName = "Cancel"
- Bước 2: xử lý, có các sự kiện
 - EditCommand # RowEditing (GridView)
 - UpdateCommand # RowDeleting (GridView)
 - ItemCommand # RowCommand (GridView)
- Bước 2: xử lý, có các thuộc tính thường dùng
 - <DataList>.EditItemIndex: chỉ số dòng được sửa ở trạng thái edit
 - e.Item.ItemIndex: chỉ số của dòng hiện hành
 - e.Item.FindControl("tên điều khiển"): tìm kiếm điều khiển trên dòng hiện hành
 - <DataList>.DataKeys[n]: truy xuất giá trị khóa tại dòng có chỉ số là n

6. DetailsView và FormView

Giới thiệu FormView

- Dùng hiển thị dữ liệu dạng cột (Columnar)
- Thể hiện trên trình duyệt dưới dạng HTML Table
- Đặc điểm:
 - Hỗ trợ cơ chế phân trang dữ liệu
 - Cho phép cập nhật dữ liệu trực tiếp
 - Cho phép kết hợp với GridView để tạo ra dạng Master-Detail

Thông tin khách hàng	
Họ tên	Nguyễn Thiên Thanh
Ngày sinh	04/04/1979
Phái	<input type="checkbox"/>
Điện thoại	0908320111
Địa chỉ	Hai Bà Trưng
Email	ntthanh@t3h.hcmuns.edu.vn

1 2 3 4 5 6 7 8 9 10 ...

Liên kết dữ liệu và định dạng

- Thao tác liên kết dữ liệu: giống GridView
- Thao tác định dạng

Các xử lý

- Phân trang: giống GridView
- Cập nhật dữ liệu:
 - Khai báo các thuộc tính phát sinh các nút cập nhật:
 - AutoGenerateDeleteButton
 - AutoGenerateEditButton
 - AutoGenerateInsertButton
 - Phần xử lý thì giống GridView, chỉ khác nhau về tên sự kiện và một số thuộc tính

Sự kiện trên GridView	Sự kiện trên DetailView
RowCommand	ItemCommand
RowInserting	ItemInserting
RowUpdating	ItemUpdating
RowDeleting	ItemDeleting
	ModeChanging: khi chuyển sang một trạng thái khác(edit, insert, read-only)

Tạo dạng Master-Detail

- Bước 1: tạo SqlDataSource1 dùng làm nguồn cho GridView
- Bước 2: tạo GridView dùng làm Master và kết nguồn vào (Nên tạo một nút lệnh xử lý chọn dòng)
- Bước 3: tạo SqlDataSource2 dùng làm nguồn cho DetailsView + FilterParameters: giá trị của tham số là <GridView>.SelectedValue + FilterExpression: biểu thức lọc tương ứng với tham số
- Bước 4: tạo DetailsView dùng làm Detail và kết nguồn vào

Danh sách sách		Chi tiết sách	
Mã sách	Tên sách	Tên sách	Giáo trình Tin học cơ bản
<u>1</u>	Giáo trình Tin học cơ bản	<u>Hình ảnh</u>	
<u>2</u>	Giáo trình Tin học văn phòng		
<u>3</u>	Lập Trình Cơ Sở Dữ Liệu Với Visual Basic 2005 Và ADO.NET 2.0	<u>Mô tả</u>	<p>Nội dung của cuốn: Tin Học Cơ Bản Windows XP gồm có 7 chương: Chương 1: Một số vấn đề cơ bản. Chương 2: Sử dụng nhanh thành công cụ và thanh thực đơn trong My Computer và Windows Explorer. Chương 3: Các thao tác trong windows XP. Chương 4: Các thiết lập trong Windows XP. Chương 5: Bảo trì máy tính. Chương 6: Các phím tắt Chương 7: Hỏi và đáp các thắc mắc. Xin trân trọng giới thiệu cuốn sách cùng bạn</p>
<u>4</u>	Visual Basic 2005 Tập 3, Quyển 2: Lập Trình Web Với Cơ Sở Dữ Liệu		
<u>5</u>	Giáo trình Tin học quản lý 1	<u>Đơn giá</u>	26,000
<u>6</u>	Giáo trình Tin học quản lý 2		
<u>7</u>	Giáo trình SQL Server		
<u>8</u>	Giáo trình Phân tích hệ thống		
<u>9</u>	Giáo trình VB.Net 1		
<u>10</u>	Giáo trình VB.Net 2		

Giới thiệu FormView

- Tương tự DetailsView nhưng phải tự thiết kế dạng hiển thị
- Thể hiện trên trình duyệt dưới dạng HTML Table
- Đặc điểm:
 - Hỗ trợ cơ chế phân trang dữ liệu
 - Cho phép cập nhật dữ liệu trực tiếp

Liên kết, định dạng và các xử lý:

- Giống DetailsView