

Mục tiêu

Hiển thị dữ liệu dạng Master-Detail có navigator bằng cách sử dụng **binding**.

Nội dung

Với database Northwind, trình bày dữ liệu Product theo Category như sau

The screenshot displays a Master-Detail application interface. At the top, a 'Categories' dropdown menu is set to 'Beverages'. Below it, a 'Products' table is shown with columns: ProductID, ProductName, SupplierID, CategoryID, QuantityPerUnit, and UnitPrice. The table contains 4 rows, with the 3rd row (ProductID 24, Guaraná Fantástica) selected. Below the table, a 'Navigations' section contains buttons for 'First', 'Prev', '2 / 12', 'Next', and 'Last'. At the bottom, a 'Product detail' section shows fields for Name, Price, Stocks, and Units, with values corresponding to the selected product.

	ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice
	1	Chai	1	1	10 boxes x 20 bags	18.0000
	2	Chang	1	1	24 - 12 oz bottles	19.0000
▶	24	Guaraná Fantástica	10	1	12 - 355 ml cans	4.5000
	34	Sasquatch Ale	16	1	24 - 12 oz bottles	14.0000

Navigations: First Prev 2 / 12 Next Last

Product detail

Name: Guaraná Fantástica Stocks: 20

Price: 4.5000 Units: 12 - 355 ml cans

Hướng dẫn

1. Khai báo lưu giữ dữ liệu

```
// sử dụng DataSet vì có 2 bảng dữ liệu
DataSet ds = new DataSet("Northwind");
```

2. Lấy dữ liệu

```
// lấy connection string từ file config
var conString = ConfigurationManager.ConnectionStrings["SQL_northwind"].ConnectionString;
var connect = new SqlConnection(conString);
// lấy Categories
var adapterC = new SqlDataAdapter(@"SELECT * FROM Categories", connect);
// map default tableName
adapterC.TableMappings.Add("Table", "Categories");
adapterC.Fill(ds);
// lấy Products
var adapterP = new SqlDataAdapter(@"SELECT * FROM Products", connect);
adapterP.TableMappings.Add("Table", "Products");
adapterP.Fill(ds);
```

3. Tạo relation thể hiện quan hệ giữa 2 bảng dữ liệu

```
// tạo relationship giữa Category và Product
var colMaster = ds.Tables["Categories"].Columns["CategoryID"];
var colDetail = ds.Tables["Products"].Columns["CategoryID"];
var relCatPro = new DataRelation("relCatPro", colMaster, colDetail);
// đưa relation vào dataset
ds.Relations.Add(relCatPro);
```

4. Thiết lập binding

```
cbCategories.DataSource = ds;
cbCategories.DisplayMember = "Categories.CategoryName";
cbCategories.ValueMember = "Categories.CategoryID";

dgvProducts.DataSource = ds;
dgvProducts.DataMember = "Categories.relCatPro";

txtName.DataBindings.Add("Text", ds, "Categories.relCatPro.ProductName");
txtPrice.DataBindings.Add("Text", ds, "Categories.relCatPro.UnitPrice");
txtQuantityPerUnit.DataBindings.Add("Text", ds, "Categories.relCatPro.QuantityPerUnit");
txtStock.DataBindings.Add("Text", ds, "Categories.relCatPro.UnitsInStock");
```

5. Thiết lập các event cần thiết

```
btnFirst.Click += BtnFirst_Click;
btnLast.Click += BtnLast_Click;
btnNext.Click += BtnNext_Click;
btnPrev.Click += BtnPrev_Click;
// thực hiện navigate trên data member nào thì lấy theo path tương ứng
BindingContext[ds, "Categories.relCatPro"].PositionChanged += relCatPro_PositionChanged;
```

6. Cấu trúc tổng thể của code

```
public partial class frmMain : Form
{
    // sử dụng DataSet vì có 2 bảng dữ liệu
    DataSet ds = new DataSet("Northwind");
    public frmMain()
    {
        InitializeComponent();

        // là các phần 2, 3, 4, 5

        btnFirst.Enabled = false;
        btnPrev.Enabled = false;
    }

    private void relCatPro_PositionChanged(object sender, EventArgs e)
    {
        lblCurent.Text = string.Format($"{BM_relCatPro.Position} / {BM_relCatPro.Count}");
        // cần xử lý chặt chẽ hơn để hiển thị chính xác hơn
    }

    // sử dụng property để đơn giản trong dòng lệnh
    BindingManagerBase BM_relCatPro
    {
        get { return BindingContext[ds, "Categories.relCatPro"]; }
    }
}
```

```
private void BtnPrev_Click(object sender, EventArgs e)
{
    // cần xử lý
}

private void BtnNext_Click(object sender, EventArgs e)
{
    // minh họa đơn giản, cần xử lý chặt chẽ hơn
    BM_relCatPro.Position++;
    btnFirst.Enabled = true;
    btnPrev.Enabled = true;
}

private void BtnLast_Click(object sender, EventArgs e)
{
    // cần xử lý
}

private void BtnFirst_Click(object sender, EventArgs e)
{
    // cần xử lý
}
}
```