

Lập trình Cơ sở dữ liệu Local SQL với C#

1. Thiết kế cơ sở dữ liệu đặt tên **QLBanHang**, trong đó có bảng:

tblMatHang(MaSP nchar(5), TenSP nvarchar(30), NgaySX Date, NgayHH Date, DonVi nvarchar(10), DonGia Float, GhiChu nvarchar(200))

2. Thiết kế giao diện:

- Các phần cửa sổ cơ giản hợp lý khi của sổ của Form phóng to hoặc thu nhỏ, ví dụ cửa sổ (3) luôn giữ cố định ở bên trái, phần cửa sổ (4) và (5) co giãn khi Form thay đổi kích thước, (1) và (2) luôn nằm trên đầu và phía dưới cùng của Form.
- Các nút có gắn các phím tắt ví dụ như **Alt + T** cho nút **Tìm kiếm**, **Alt + H** cho nút **Thêm**, v.v.

a) Chức năng Tìm kiếm

Hỗ trợ các tiêu chí: MaSP, TenSP, NgaySX, NgayHetHan, DonGia

Tiêu đề: TÌM KIẾM MẶT HÀNG

Khi ấn nút Tìm:

- **dgvDanhSach** sẽ hiển thị tất cả các thông tin về các sản phẩm thỏa mãn các tiêu chí lựa chọn.
- Click vào các nút **Sửa**, **Xóa**
- Khi Click vào từng trường, các thông tin sẽ hiển thị chi tiết lên **grbChiTiet**, nhưng không cho phép sửa đổi dữ liệu. Không hiển thị các nút **Lưu**, **Hủy** trong **grbChiTiet**.

b) Chức năng Thêm

Tiêu đề: THÊM MẶT HÀNG

Khi ấn nút Thêm:

- **grbChiTiet** sẽ xoá trắng và cho phép sửa đổi dữ liệu
- Cấm Click vào các nút **Sửa, Xoá**.
- Trong **grbChiTiet** hiển thị 2 nút lệnh **Lưu, Huỷ**. Khi ấn nút **Huỷ**, sẽ thoát khỏi chức năng thêm (Xoá trắng **grbChiTiet**, không cho nhập dữ liệu)
- Nếu Thêm thành công, cập nhật lại **dgvDanhSach**

c) Chức năng Sửa

Tiêu đề: CẬP NHẬT MẶT HÀNG

Nút **Sửa** chỉ cho *Enable = True* khi người dùng chọn một bản ghi trong **dgvDanhSach**.

Khi ấn nút Sửa:

- Cấm Click vào các nút **Thêm, Xoá**.
- **grbChiTiet** cho phép sửa dữ liệu. Trong **grbChiTiet** cho hiển thị nút lệnh **Lưu, Huỷ** **bỏ**. Khi ấn nút **Huỷ**, sẽ thoát khỏi chức năng **Sửa** (cấm Click vào nút **Sửa**, cho phép Click vào nút **Thêm**)
- Nếu Cập nhật thành công, cập nhật lại **dgvDanhSach**

d) Chức năng Xoá

Tiêu đề: XOÁ MẶT HÀNG

Nút **Xoá** chỉ cho phép Click khi người dùng chọn một bản ghi trong **dgvDanhSach**

Khi ấn nút Xoá:

- Cấm Click vào các nút **Thêm, Sửa**.
- **grbChiTiet** không cho phép sửa dữ liệu. Trong **grbChiTiet** cho hiển thị nút lệnh **Lưu, Huỷ** **bỏ**. Khi ấn nút **Huỷ** sẽ thoát khỏi chức năng **Xoá** (cấm Click vào nút **Xoá**, cho phép Click vào nút **Thêm**)

Xây dựng chương trình

1. Tạo mới Project

- Kích vào menu **Start > All Programs > Microsoft Visual Studio 2019** (tùy theo phiên bản bạn cài)
- Rồi trên cửa sổ VS 2010 kích vào menu **File > New > Project ...**
- Chọn ngôn ngữ **C#** và kiểu ứng dụng bạn lập trình. Trong trường hợp này bạn chọn **Windows Form Application**

2. Tạo cơ sở dữ liệu

Lập trình CSDL với .NET có một chuẩn chung với các nguồn cơ sở dữ liệu khác nhau, nó chỉ khác nhau về **.NET Data Provider**. Trong bài này chúng ta sẽ sử dụng hệ quản trị CSDL SQL Server được cài mặc định khi cài Visual Studio

- Kích chuột phải vào tên Project trong cửa sổ **Solution Explorer** > **Add** > **New Item** > **Service-based Database**



- Bạn ấn nút **Next** và **Finish** để kết thúc tạo file. Sau đó bạn nhìn thấy có một file **QLBanHang.mdf** xuất hiện trong cửa sổ **Solution Explorer**
- Kích đúp vào file này, bạn nhìn thấy nó xuất hiện cửa sổ **Server Explorer**, chọn tên Database **QLBanHang** > **Tables** > kích chuột phải chọn **Add New Table**, rồi thực hiện tạo bảng với cấu trúc như yêu cầu

3. Giao diện thiết kế

-
- Dặt các thuộc tính**

Phần giao diện	Loại control	Các thuộc tính
Cửa sổ chính	Form	Name = frmMatHang Text = Quản lý sản phẩm
	ErrorProvider	Name = errChiTiet

(1)	Panel	Dock = Top
	Label	Name = lblTieuDe
(2)	Panel	Dock = Bottom
	Button	Name = btnTimKiem Text = &Tìm kiếm
	Button	Name = btnThem Text = T&hêm
	Button	Name = btnSua Text = &Sửa
	Button	Name = btnXoa Text = &Xoá
	Button	Name = btnThoat Text = Th&oát
	Button	Name = btnTimKiem Text = &Tìm kiếm
(3)	GroupBox	Dock = Right Text = Chi tiết
	Splitter	Dock = Left
	Label	Text = Mã SP:
	TextBox	Name = txtMaSP
	Label	Text = Tên SP:
	TextBox	Name = txtTenSP
	Label	Text = Ngày SX:
	DateTimePicker	Name = dtpNgaySX Format=Short
	Label	Text = Ngày HH:
	DateTimePicker	Name = dtpNgayHH Format=Short
	Label	Text = Đơn vị:
	TextBox	Name = txtDonVi
	Label	Text = Đơn giá:
	TextBox	Name = txtDonGia TextAlign = Right
	Label	Text = Ghi chú:

	TextBox	Name = txtGhiChu Multiline = True
	Button	Name = btnLuu Text = &Luu
	Button	Name = btnHuy Text = &Huy
(4)	GroupBox	Text = Tìm kiếm Dock = Top
	Splitter	Dock = Top
	Label	Text = Mã SP:
	TextBox	Name = txtTKMaSP
	Label	Text = Tên SP:
	TextBox	Name = txtTKTenSP
(5)	GroupBox	Text = Kết quả Dock = Fill
	DataGridView	Name = dgvKetQua

- **Chú ý** để thiết kế được giao diện thân thiện với người dùng và các control tự co giãn khi Form thay đổi kích thước cần thiết kế, chú ý kéo các control đúng thứ tự như sau:
 - Giao diện phần (1): là phần nhãn tiêu đề, kéo control **Panel** vào Form và đặt thuộc tính **Dock = Top** để cho **Panel** luôn nằm ở trên đầu của Form, sau đó kéo một **Label** nằm trên **Panel** này.
 - Giao diện phần (2): là phần các nút Tìm kiếm, Thêm ... cũng làm tương tự như phần (1), kéo control **Panel** và đặt thuộc tính **Dock = Bottom** để Panel này luôn nằm cuối của Form, sau đó mới đặt các **Button** nằm trên Panel này.
 - Giao diện phần (3): là phần chi tiết, kéo control **GroupBox** vào Form và đặt thuộc tính **Dock = Left**, sau đó thiết kế các control cho nhập Mã SP, Tên SP ... Sau đó ta kéo thêm control **Splitter** và đặt thuộc tính **Dock = Left** để cho phép người dùng thay đổi độ rộng giữa phần (3) với phần (4) và (5).
 - Giao diện phần (4): là phần tìm kiếm, kéo control **GroupBox** vào Form và đặt thuộc tính **Dock = Top**, lúc này nó sẽ nằm ở trên phần còn lại. Sau đó đặt các control cho phép nhập tìm kiếm như Mã SP, Tên SP ... Trong phần này ta cũng kéo control **Splitter**, đặt thuộc tính **Dock = Top** để cho phép người sử dụng thay đổi độ rộng giữa phần (4) và phần (5)
 - Giao diện phần (5): là phần hiển thị dữ liệu, kéo control **GroupBox** và đặt thuộc tính **Dock = Fill**, sau đó kéo control **DataGridView** nằm trong GroupBox này nó sẽ chiếm toàn bộ phần không gian còn lại.

- Viết chương trình chạy trên Windows thì cần tuân theo chuẩn của Windows, ví dụ khi ấn phím **Tab** trên bàn phím thì con trỏ sẽ chạy từ *bên trên* -> *sang trái* -> *xuống dưới* -> *sang phải*. Nếu thứ tự bạn kéo control không đúng thì khi ấn phím **Tab** sẽ nhảy lung tung. Bạn có thể đặt lại bằng cách vào menu **View > Tab Order** để đặt lại thứ tự Tab trên các control. Sau khi đặt xong quay lại chọn lại menu **View > Tab Order** để kết thúc.

4. Phần code lập trình các chức năng

- Khai báo **.Net Data Provider** là SQL Server

```
using System.Data.SqlClient;
```

- Khai báo các biến toàn cục nằm ngay dưới class

```
namespace QLBanHang
{
    public partial class frmMatHang : Form
    {
        //Khai báo các biến toàn cục
        SqlConnection con;//Khai báo đối tượng thực hiện kết nối đến cơ sở dữ liệu
        SqlCommand cmd;//Khai báo đối tượng thực hiện các câu lệnh truy vấn
        SqlDataAdapter dap;//Khai báo đối tượng gắn kết DataSource với DataSet
        DataSet ds;//Đối tượng chứa dữ liệu tại local
        public frmMatHang()
        {
            InitializeComponent();
        }
    }
}
```

- Viết sự kiện **Form_Load**, xảy ra khi form khởi động lên. Chúng ta thực hiện khởi tạo kết nối và nạp dữ liệu lên **DataGridView**

```
private void frmMatHang_Load(object sender, EventArgs e)
{
    //Tạo đối tượng Connection
    con = new SqlConnection();
    //Truyền vào chuỗi kết nối tới cơ sở dữ liệu
    //Gọi Application.StartupPath để lấy đường dẫn tới thư mục chứa file chạy chương trình
    con.ConnectionString = @"Data Source=.\SQLEXPRESS;AttachDbFilename=" + Application.StartupPath + @"\QLBanHang.mdf;Integrated Security=True;User Instance=True";
    //Gọi phương thức Load dữ liệu
    LoadDuLieu("Select * from tblMatHang");
    //Khi Form mới Load lên thì ấn các nút Sửa và Xóa
```

```

        btnSua.Enabled = false;
        btnXoa.Enabled = false;
        //An groupbox chi tiet
        HienChiTiet(false);
    }
    //Viết một hàm nạp dữ liệu lên DataGridView
    private void LoadDuLieu(String sql)
    {
        //tạo đối tượng DataSet
        ds = new DataSet();
        //Khởi tạo đối tượng DataAdapter và cung cấp vào câu lệnh SQL và đối tượng
        Connection
        dap = new SqlDataAdapter(sql,con);
        //Dùng phương thức Fill của DataAdapter để đổ dữ liệu từ DataSource tới DataSet
        dap.Fill(ds);
        //Gắn dữ liệu từ DataSet lên DataGridView
        dgvKetQua.DataSource = ds.Tables[0];
    }
    //Phương thức ẩn hiện các control ở groupbox chi tiết
    private void HienChiTiet(Boolean hien)
    {
        txtMaSP.Enabled = hien;
        txtTenSP.Enabled = hien;
        dtpNgayHH.Enabled = hien;
        dtpNgaySX.Enabled = hien;
        txtDonVi.Enabled = hien;
        txtDonGia.Enabled = hien;
        txtGhiChu.Enabled = hien;
        //Ẩn hiện 2 nút Lưu và Hủy
        btnLuu.Enabled = hien;
        btnHuy.Enabled = hien;
    }

```

- Chức năng tìm kiếm, sử dụng sự kiện **Click** của nút **Tìm kiếm**

```

//Chức năng tìm kiếm
private void btnTimKiem_Click(object sender, EventArgs e)
{
    //Cập nhật trên nhãn tiêu đề
    lblTieuDe.Text = "TÌM KIẾM MẶT HÀNG";
    //Cấm nút Sửa và Xóa
    btnSua.Enabled = false;
    btnXoa.Enabled = false;
    //Viết câu lệnh SQL cho tìm kiếm
    String sql = "SELECT * FROM tblMatHang";
    String dk = "";
    //Tìm theo MaSP khác rỗng

```

```

if (txtTKMaSP.Text.Trim() != "")
{
    dk += " MaSP like '%" + txtTKMaSP.Text + "%'";
}
//kiem tra TenSP va MaSP khác rỗng
if (txtTKTenSP.Text.Trim() != "" && dk != "")
{
    dk += " AND TenSP like N '%" + txtTKTenSP.Text + "%'";
}
//Tìm kiếm theo TenSP khi MaSP là rỗng
if (txtTKTenSP.Text.Trim() != "" && dk == "")
{
    dk += " TenSP like N '%" + txtTKTenSP.Text + "%'";
}
//Ket hoi dk
if (dk != "")
{
    sql += " WHERE" + dk;
}
//Goi phương thức Load dữ liệu kết hợp điều kiện tìm kiếm
LoadDuLieu(sql);
}

```

- Chức năng Thêm, sử dụng sự kiện **Click** của nút **Thêm**

```

//Sự kiện thêm mới mặt hàng
private void btnThem_Click(object sender, EventArgs e)
{
    lblTieuDe.Text = "THÊM MẶT HÀNG";
    //Xóa trang GroupBox chi tiết sản phẩm
    XoaTrangChiTiet();
    //Cam nút sửa xóa
    btnSua.Enabled = false;
    btnXoa.Enabled = false;
    //Hiện GroupBox Chi tiết
    HienChiTiet(true);
}

```

- Chức năng Sửa xảy ra khi chọn một bản ghi trên DataGridView. Kích đúp vào DataGridView, VS sẽ phát sinh tự động sự kiện **CellContentClick**. Chúng ta lấy giá trị hàng hiện tại thông qua tham số của sự kiện *DataGridViewCellEventArgs e* trả về qua thuộc tính *RowIndex*. Sau đó ta viết sự kiện **Click** của nút **Sửa**

```

//Sự kiện khi người dùng chọn một hàng trên DataGridView

```



```

private void dgvKetQua_CellContentClick(object sender,
DataGridViewCellEventArgs e)
{
    //Hiện thi nut sua
    btnSua.Enabled = true;
    btnXoa.Enabled = true;
    btnThem.Enabled = false;
    //Bắt lỗi khi người sử dụng kích linh tinh lên datagrid
    try
    {
        txtMaSP.Text = dgvKetQua[0, e.RowIndex].Value.ToString();
        txtTenSP.Text = dgvKetQua[1, e.RowIndex].Value.ToString();
        dtpNgaySX.Value = (DateTime)dgvKetQua[2, e.RowIndex].Value;
        dtpNgayHH.Value = (DateTime)dgvKetQua[3, e.RowIndex].Value;
        txtDonVi.Text = dgvKetQua[4, e.RowIndex].Value.ToString();
        txtDonGia.Text = dgvKetQua[5, e.RowIndex].Value.ToString();
        txtGhiChu.Text = dgvKetQua[6, e.RowIndex].Value.ToString();
    }
    catch (Exception ex)
    {
    }
}
//Sự kiện Click của nút sửa
private void btnSua_Click(object sender, EventArgs e)
{
    //Cập nhật tiêu đề
    lblTieuDe.Text = "CẬP NHẬT MÃ HÀNG";
    //Ẩn hai nút Thêm và Sửa
    btnThem.Enabled = false;
    btnXoa.Enabled = false;
    //Hiện gropbox chi tiết
    HienChiTiet(true);
}

```

- Chức năng Xóa xảy ra khi chọn một bản ghi trên DataGridView. Chúng ta vẫn sử dụng lại sự kiện ở **CellContentClick** của DataGridView ở phía trên. Sau đó ta viết sự kiện **Click** của nút **Xóa**

```

//Sự kiện Click của nút Xóa
private void btnXoa_Click(object sender, EventArgs e)
{
    //Bật Message Box cảnh báo người sử dụng
    if (MessageBox.Show("Bạn có chắc chắn xóa mã mặt hàng " + txtMaSP.Text + " không? Nếu có ấn nút Lưu, không thì ấn nút Hủy", "Xóa sản phẩm",
        MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
    }
}

```

```

        lblTieuDe.Text = "XÓA MẶT HÀNG";
        btnThem.Enabled = false;
        btnSua.Enabled = false;
        //Hiện gropbox chi tiết
        HienChiTiet(true);
    }
}

```

- Chức năng Lưu, chúng ta dựa vào trạng thái Enable của các nút Thêm, Sửa hoặc Xóa để xác nhận trạng thái thực thi câu lệnh SQL

```

//Sự kiện click vào button Lưu
private void btnLuu_Click(object sender, EventArgs e)
{

```

```

    string sql = "";
    //Kiểm tra nếu kết nối chưa mở thì thực hiện mở kết nối
    if (con.State != ConnectionState.Open)
        con.Open();
    //Chúng ta sử dụng control ErrorProvider để hiển thị lỗi
    //Kiểm tra tên sản phẩm có bị để trống không
    if (txtTenSP.Text.Trim() == "")
    {
        errChiTiet.SetError(txtTenSP, "Bạn không để trống tên sản phẩm!");
        return;
    }
    else
    {
        errChiTiet.Clear();
    }
    //Kiểm tra ngày sản xuất, lỗi nếu người sử dụng nhập vào ngày sản xuất lớn hơn
    ngày hiện tại
    if (dtpNgaySX.Value > DateTime.Now)
    {
        errChiTiet.SetError(dtpNgaySX, "Ngày sản xuất không hợp lệ!");
        return;
    }
    else
    {
        errChiTiet.Clear();
    }
    //Kiểm tra ngày hết hạn xem có lớn hơn ngày sản xuất không
    if (dtpNgayHH.Value < dtpNgaySX.Value) { errChiTiet.SetError(dtpNgayHH,
    "Ngày hết hạn nhỏ hơn ngày sản xuất!"); return; } else { errChiTiet.Clear(); } //Kiểm tra
    đơn vị xem có để trống không if (txtDonVi.Text.Trim() == "")
    { errChiTiet.SetError(txtDonVi, "Bạn không để trống đơn vị!"); return; } else

```

```

{ errChiTiet.Clear(); } //Kiểm tra đơn giá if (txtDonGia.Text.Trim() == "")
{ errChiTiet.SetError(txtDonGia, "Bạn không để trống đơn giá!"); return; } else
{ errChiTiet.Clear(); } //Nếu nút Thêm enable thì thực hiện thêm mới //Dùng ký tự N' trước
mỗi giá trị kiểu text để insert giá trị có dấu tiếng việt vào CSDL được đúng if
(btnThem.Enabled == true) { //Kiểm tra xem ô nhập MaSP có bị trống không if
(txtMaSP.Text.Trim() == "") { errChiTiet.SetError(txtMaSP, "Bạn không để trống mã sản
phẩm trường này!"); return; } else { //Kiểm tra xem mã sản phẩm đã tồn tại chưa để tránh
việc insert mới bị lỗi sql = "Select Count(*) From tblMatHang Where MaSP =" +
txtMaSP.Text + """; cmd = new SqlCommand(sql, con); int val = (int)cmd.ExecuteScalar();
if (val > 0)
{
errChiTiet.SetError(txtMaSP, "Mã sản phẩm trùng trong cơ sở dữ liệu");
return;
}
errChiTiet.Clear();
}
//Insert vào CSDL
sql = "INSERT INTO
tblMatHang(MaSP,TenSP,NgaySX,NgayHH,DonVi,DonGia,GhiChu)VALUES (";
sql += "N'" + txtMaSP.Text + "',N'" + txtTenSP.Text + "','" +
dtpNgaySX.Value.Date + "','" + dtpNgayHH.Value.Date + "',N'" + txtDonVi.Text + "',N'" +
txtDonGia.Text + "',N'" + txtGhiChu.Text + ")";
}
//Nếu nút Sửa enable thì thực hiện cập nhật dữ liệu
if (btnSua.Enabled == true)
{
sql = "Update tblMatHang SET ";
sql += "TenSP = N'" + txtTenSP.Text + "',";
sql += "NgaySX = '" + dtpNgaySX.Value.Date + "',";
sql += "NgayHH = '" + dtpNgayHH.Value.Date + "',";
sql += "DonVi = N'" + txtDonVi.Text + "',";
sql += "DonGia = '" + txtDonGia.Text + "',";
sql += "GhiChu = N'" + txtGhiChu.Text + "' ";
sql += "Where MaSP = N'" + txtMaSP.Text + """;
}
//Nếu nút Xóa enable thì thực hiện xóa dữ liệu
if (btnXoa.Enabled == true)
{
sql = "Delete From tblMatHang Where MaSP =N'" + txtMaSP.Text + """;
}
//Thực thi câu lệnh sql
cmd = new SqlCommand(sql, con);
cmd.ExecuteNonQuery();
//Cập nhật lại DataGridView
sql = "Select * from tblMatHang";
LoadDuLieu(sql);
//dong ket noi
con.Close();

```

```
//Ẩn hiện các nút phù hợp chức năng
HienChiTiet(false);
btnSua.Enabled = false;
btnXoa.Enabled = false;
}
```

- Kích vào nút Huỷ để huỷ bỏ công việc đang làm

```
//Click vào nút Huỷ
private void btnHuy_Click(object sender, EventArgs e)
{
    //Thiết lập lại các nút như ban đầu
    btnXoa.Enabled = false;
    btnSua.Enabled = false;
    btnThem.Enabled = true;
    //xoa trang
    XoaTrangChiTiet();
    //Cam nhap
    HienChiTiet(false);
}
```

- Kích vào nút Thoát

```
private void btnThoat_Click(object sender, EventArgs e)
{
    //Đóng form
    this.Close();
}
```