

### Bài 15: Thực hiện chức năng nhập hàng

Mục tiêu: Phần bài tập này sẽ hướng dẫn nhập hàng và kiểm soát hàng tồn của sản phẩm.

Dữ liệu sẽ được cập nhật vào 2 bảng, sanPham và ChiTietNhapHang.

Cấu trúc của mảng dữ liệu

<table><tr><th colspan="4">SanPham</th></tr><tr><th></th><th>Column Name</th><th>Data Type</th><th>Allow Nulls</th></tr><tr><td>🔑</td><td>MaSP</td><td>int</td><td><input type="checkbox"/></td></tr><tr><td></td><td>TenSP</td><td>nvarchar(100)</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td>DVT</td><td>nvarchar(20)</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td>DonGia</td><td>int</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td>SoLuong</td><td>int</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td>Hinh</td><td>image</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td></td><td></td><td><input type="checkbox"/></td></tr></table>	SanPham					Column Name	Data Type	Allow Nulls	🔑	MaSP	int	<input type="checkbox"/>		TenSP	nvarchar(100)	<input checked="" type="checkbox"/>		DVT	nvarchar(20)	<input checked="" type="checkbox"/>		DonGia	int	<input checked="" type="checkbox"/>		SoLuong	int	<input checked="" type="checkbox"/>		Hinh	image	<input checked="" type="checkbox"/>				<input type="checkbox"/>	<ul style="list-style-type: none"><li>- DonGia: lưu đơn giá nhập bình quân. Dùng trong trường hợp tính lợi nhuận theo bình quân gia quyền.</li><li>- SoLuong (Số lượng): số lượng tồn hiện tại của sản phẩm.</li></ul>				
SanPham																																									
	Column Name	Data Type	Allow Nulls																																						
🔑	MaSP	int	<input type="checkbox"/>																																						
	TenSP	nvarchar(100)	<input checked="" type="checkbox"/>																																						
	DVT	nvarchar(20)	<input checked="" type="checkbox"/>																																						
	DonGia	int	<input checked="" type="checkbox"/>																																						
	SoLuong	int	<input checked="" type="checkbox"/>																																						
	Hinh	image	<input checked="" type="checkbox"/>																																						
			<input type="checkbox"/>																																						
<table><tr><th colspan="4">ChiTietNhapHang</th></tr><tr><th></th><th>Column Name</th><th>Data Type</th><th>Allow Nulls</th></tr><tr><td>🔑</td><td>ID</td><td>int</td><td><input type="checkbox"/></td></tr><tr><td></td><td>MaSP</td><td>int</td><td><input type="checkbox"/></td></tr><tr><td></td><td>MaNV</td><td>int</td><td><input type="checkbox"/></td></tr><tr><td></td><td>NgayNhap</td><td>date</td><td><input type="checkbox"/></td></tr><tr><td></td><td>SoLuongNhap</td><td>int</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td>DonGiaNhap</td><td>float</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td>SoLuongNhapTon</td><td>int</td><td><input checked="" type="checkbox"/></td></tr><tr><td></td><td></td><td></td><td><input type="checkbox"/></td></tr></table>	ChiTietNhapHang					Column Name	Data Type	Allow Nulls	🔑	ID	int	<input type="checkbox"/>		MaSP	int	<input type="checkbox"/>		MaNV	int	<input type="checkbox"/>		NgayNhap	date	<input type="checkbox"/>		SoLuongNhap	int	<input checked="" type="checkbox"/>		DonGiaNhap	float	<input checked="" type="checkbox"/>		SoLuongNhapTon	int	<input checked="" type="checkbox"/>				<input type="checkbox"/>	<ul style="list-style-type: none"><li>- SoLuongNhap(Số lượng Nhập): Lưu số lượng nhập của từng lần nhập</li><li>- DonGiaNhap(Đơn giá nhập): Lưu đơn giá nhập của từng lần nhập.</li><li>- SoLuongNhapTon: (Số lượng nhập tồn) lưu số lượng tồn của từng lần nhập. thuộc tính này được dùng để tính lợi nhuận theo FIFO. Có ràng buộc liên thuộc tính liên quan hệ với thuộc tính SoLuong trong bảng sản phẩm.</li></ul>
ChiTietNhapHang																																									
	Column Name	Data Type	Allow Nulls																																						
🔑	ID	int	<input type="checkbox"/>																																						
	MaSP	int	<input type="checkbox"/>																																						
	MaNV	int	<input type="checkbox"/>																																						
	NgayNhap	date	<input type="checkbox"/>																																						
	SoLuongNhap	int	<input checked="" type="checkbox"/>																																						
	DonGiaNhap	float	<input checked="" type="checkbox"/>																																						
	SoLuongNhapTon	int	<input checked="" type="checkbox"/>																																						
			<input type="checkbox"/>																																						

Yêu cầu thiết kế giao diện

Frm\_NhapHang\_Main.cs

## Giáo trình thực hành lập trình ứng dụng CSDL với C#

STT	MaSP	Tên sản phẩm	DVT	Số tồn	Ngày nhập	SLNhap	SLNhapton	DonGiaNhap	NhanVien
-----	------	--------------	-----	--------	-----------	--------	-----------	------------	----------

Frm\_NhapHang\_Modifies.cs

Frm\_NhapSanPham\_Modifies

MaSP

TenSP

DVT

SoLuong

DonGia

NgayNhap

Code

**SQL: thủ tục để nhập mới và nhập thêm số lượng.**

```
Create proc [dbo].[PSP_SanPham_InsertUpdate]
@MaSP int,
@TenSP nvarchar(100),
@DVT nvarchar(20),
@DonGia int,
@SoLuong int,
@Hinh image,
```

```
@MaNV int ,
@NgayNhap date
as
if exists (select 1 from SanPham where MaSP=@MaSP)
begin --update
--khi thêm số lượng cho sản phẩm đã có
declare @DonGiaTB int
set @DonGiaTB=(select isnull(DonGia,0) from SanPham where MaSP=@MaSP)
update SanPham
set SoLuong+=@SoLuong,
DonGia=case @DonGiaTB when 0 then @DonGia else (@DonGiaTB+@Dongia)/2 end
where MaSP=@MaSP

insert into ChiTietNhapHang(MaSP, MaNV, NgayNhap, SoLuongNhap, DonGiaNhap,
SoLuongNhapTon)
values(@MaSP, @MaNV, @NgayNhap, @SoLuong, @DonGia, @SoLuong)

end--update
else
begin--insert
--thêm sản phẩm mới.
insert into SanPham(TenSP, DVT, DonGia, SoLuong, Hinh)
values( @TenSP, @DVT, @DonGia, @SoLuong, @Hinh)

Declare @maSPNew int
set @maSPNew=(select max(masp)from SanPham)

insert into ChiTietNhapHang(MaSP, MaNV, NgayNhap, SoLuongNhap, DonGiaNhap,
SoLuongNhapTon)
values(@maSPNew, @MaNV, @NgayNhap, @SoLuong, @DonGia, @SoLuong)
end--insert
```

#### Code frm\_NhapHang\_Modifies

```
using BussnessLayer;
using DTOLayer;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Project_WindowsFormPart02
{
    public partial class Frm_NhapSanPham_Modifies : Form
    {
        public Frm_NhapSanPham_Modifies()
        {
            InitializeComponent();
        }
        byte[] imagedata = null;
        public bool _addNew = false;
    }
}
```

```

public DTO_SanPham _dto_SanPham;
BLL_SanPham bd;
string err = string.Empty;
public static byte[] ImageToByte(Image img)
{
    ImageConverter converter = new ImageConverter();
    return (byte[])converter.ConvertTo(img, typeof(byte[]));
}
private byte[]
convertPictureBoxImageToByte(System.Windows.Forms.PictureBox pbImage)
{
    System.IO.MemoryStream ms = new System.IO.MemoryStream();
    pbImage.Image.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg);
    return ms.ToArray();
}
private void SetControlToDTO()
{
    _dto_SanPham = new DTO_SanPham();
    _dto_SanPham.MaSP = txtMaSP.Text;
    _dto_SanPham.TenSP = txtTenSP.Text;
    _dto_SanPham.DVT = txtDVT.Text;
    _dto_SanPham.SoLuong = Convert.ToInt32(txtSoLuong.Text);
    _dto_SanPham.DonGia = Convert.ToInt32(txtDonGia.Text);
    _dto_SanPham.Hinh = convertPictureBoxImageToByte(ptbHinh);
}
private void btnLuu_Click(object sender, EventArgs e)
{
    SetControlToDTO();
    if(bd.SaveSanPham(ref
err, _dto_SanPham, Cls_Main._NhanVien.MaNV, dtpNgayNhap.Value))
//Phương thức này được viết trong lớp BLL_SanPham.
    {
        MessageBox.Show("Thành công");
    }
    else
    {
        MessageBox.Show("Không thành công " + err);
    }
}
private void SetDTOToControl(DTO_SanPham _sanPham)
{
    txtMaSP.Text = _sanPham.MaSP;
    txtTenSP.Text = _sanPham.TenSP;
    txtDonGia.Text = _sanPham.DonGia.ToString(); ;
    txtDVT.Text = _sanPham.DVT;
    txtTenSP.Enabled = true;
}
private void Frm_NhapSanPham_Modifies_Load(object sender, EventArgs
e)

```

```
{
    bd = new BLL_SanPham(Cls_Main.pathconn, Cls_Main._ketNoi.WinNT);
    if(_addNew==true)
    {
        //thêm san pham mới.
        txtMaSP.Text = "0";
        ptbHinh.Image =
Image.FromFile(Cls_Main.PathImageSanPhamNull);
        txtTenSP.Focus();
    }
    else
    {
        if(_dT0_SanPham!=null)
        {
            ptbHinh.Image =
Image.FromFile(Cls_Main.PathImageSanPhamNull);
            SetDT0ToControl(_dT0_SanPham);
            txtSoLuong.Focus();
        }
        //Nhap thêm số lượng
    }
}

private void txtSoLuong_KeyPress(object sender, KeyPressEventArgs e)
{
    if(!char.IsDigit(e.KeyChar)&&!char.IsControl(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void txtDonGia_KeyPress(object sender, KeyPressEventArgs e)
{
    if (!char.IsDigit(e.KeyChar) && !char.IsControl(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void btnchonanh_Click(object sender, EventArgs e)
{
    OpenFileDialog _openimage = new OpenFileDialog();
    _openimage.Filter = "";
    if(_openimage.ShowDialog()==DialogResult.OK)
    {
        ptbHinh.Image = Image.FromFile(_openimage.FileName);
    }
}
}
```

#### Code frm\_NhapHang\_Main

```
using BussnessLayer;
using DTOLayer;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Project_WindowsFormPart02
{
    public partial class Frm_NhapSanPham_Main : Form
    {
        public Frm_NhapSanPham_Main()
        {
            InitializeComponent();
        }
        BLL_SanPham bd;
        DataTable dtDsSanPham;
        string err = string.Empty;

        public delegate void _deDongTab(string name);
        public _deDongTab deDongTab;

        DTO_SanPham _sanPham;
        private void HienThiDsSanPham()
        {
            dtDsSanPham = new DataTable();
            dtDsSanPham = bd.GetDsSanPham(ref err);
            dgvdssanpham.DataSource = dtDsSanPham.DefaultView;
        }
        private void Frm_NhapSanPham_Main_Load(object sender, EventArgs e)
        {
            bd = new BLL_SanPham(Cls_Main.pathconn, Cls_Main._ketNoi.WinNT);
            HienThiDsSanPham();
        }
        private void btnAddNew_Click(object sender, EventArgs e)
        {
            Frm_NhapSanPham_Modifies _NhapSanPhamMoi = new
            Frm_NhapSanPham_Modifies();
            _NhapSanPhamMoi._addNew = true;
            _NhapSanPhamMoi.ShowDialog();
            HienThiDsSanPham();
        }
        private void btnRefresh_Click(object sender, EventArgs e)
        {

```

```

        HienThiDsSanPham();
    }
    private void btnThoat_Click(object sender, EventArgs e)
    {
        deDongTab(this.Name);
    }
    private void dgvdssanpham_Click(object sender, EventArgs e)
    {
        if (dgvdssanpham.RowCount > 0)
        {
            try
            {
                _sanPham = new DTO_SanPham();
                _sanPham.MaSP =
dgvdssanpham.CurrentRow.Cells["colmasp"].Value.ToString();
                _sanPham.TenSP =
dgvdssanpham.CurrentRow.Cells["coltensp"].Value.ToString();
                _sanPham.DVT =
dgvdssanpham.CurrentRow.Cells["coldvt"].Value.ToString();
                _sanPham.DonGia =
Convert.ToInt32(dgvdssanpham.CurrentRow.Cells["coldongianhap"].Value.ToString
());
            }
            catch (Exception ex)
            {
                err = ex.Message;
            }
        }
    }

    private void btnModifies_Click(object sender, EventArgs e)
    {
        if(_sanPham!=null)
        {
            Frm_NhapSanPham_Modifies _NhapThemSLSanPham = new
Frm_NhapSanPham_Modifies();
            _NhapThemSLSanPham._addNew = false;
            _NhapThemSLSanPham._dto_SanPham = _sanPham;
            _NhapThemSLSanPham.ShowDialog();
            _sanPham = null;
            HienThiDsSanPham();
        }
    }
}

```

BLL\_SanPham

```

using DataLayer;
using DTOLayer;
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace BussnessLayer
{
    public class BLL_SanPham: IDisposable
    {
        Cls_Database data;
        public void Dispose()
        {
            if (data != null)
            {
                data.Dispose();
                data = null;
            }
        }
        public BLL_SanPham(string path, bool winNT)
        {
            data = new Cls_Database(path, winNT);
        }
        public DataTable GetDsSanPham(ref string err)
        {
            return data.GetDataTable("SP_SanPham_Select",
            CommandType.StoredProcedure, ref err, null);
        }
        public bool SaveSanPham(ref string err, DTO_SanPham _dto_SanPham, string
maNV, DateTime ngayNhap)
        {
            return data.MyExcuteNonQuery("PSP_SanPham_InsertUpdate",
            CommandType.StoredProcedure, ref err,
                new SqlParameter("@MaSP", _dto_SanPham.MaSP),
                new SqlParameter("@TenSP", _dto_SanPham.TenSP),
                new SqlParameter("@DVT", _dto_SanPham.DVT),
                new SqlParameter("@SoLuong", _dto_SanPham.SoLuong),
                new SqlParameter("@DonGia", _dto_SanPham.DonGia),
                new SqlParameter("@Hinh", _dto_SanPham.Hinh),
                new SqlParameter("@MaNV", maNV),
                new SqlParameter("@NgayNhap", ngayNhap));
        }
    }
}

```

**DTO\_SanPham**

```

using System;

```



```
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace DTOLayer
{
    public class DTO_SanPham
    {
        string maSP, tenSP, dVT;
        public string DVT
        {
            get { return dVT; }
            set { dVT = value; }
        }
        public string TenSP
        {
            get { return tenSP; }
            set { tenSP = value; }
        }
        public string MaSP
        {
            get { return maSP; }
            set { maSP = value; }
        }
        int donGia, soLuong;
        public int SoLuong
        {
            get { return soLuong; }
            set { soLuong = value; }
        }
        public int DonGia
        {
            get { return donGia; }
            set { donGia = value; }
        }
        Byte[] hinh;
        public Byte[] Hinh
        {
            get { return hinh; }
            set { hinh = value; }
        }
    }
}
```