-- Ôn ASP.NET--

1. Hello World

+ Biên dịch từ <asp: -> html control.

+Thêm Runat="server" khi thêm chữ này thì html control -> html server(Vì sử lí ở server) và gửi dữ liệu lên server xử lí.

gọi là postback event.<> Ngược lại non-postback.

VD: khi ấn vào button login thì IsPostBack =true; và dùng kiểm tra trong PageLoad để ẩn form>div đi.

+ AutoPostBack : true thì sẽ postback(Gửi lại) dữ liệu

CodeBebind

protected void Page\_Load(object sender, EventArgs e)

{

if (IsPostBack)

{

lblResult.Visible = false;

}

}

Code aspx.cs

<div id="lblResult" runat="server">

UserName: <asp:TextBox ID="txtUser" runat="server" />

<br />

PassWord: <asp:TextBox ID="txtPass" runat="server"

TextMode="Password" />

<br />

<asp:Button ID="btnLogin" Text="Login" runat="server" OnClick="ClickBtnLogin"/>

</div>

protected void ClickBtnLogin(object sender, EventArgs e)

{

Response.Write("UserName:" + txtUser.Text.ToString() + "<br/> PassWord:" + txtPass.Text.ToString());

}

liệu.

2. Control cơ bản,

+ Button

+ Response.Redirect("");

<asp:Button ID="btnLogin" Text="Login" runat="server" OnClick="ClickBtnLogin"/>

+ TextBox

<asp:TextBox ID="txtUser" runat="server" />

+HyperLink

<asp:HyperLink ID="txtBT2" runat="server" NavigateUrl="basic/btBuoi2.aspx">Bài Tập 2</asp:HyperLink>

3. Sự Kiện cơ bản

+ OnClick

+ OnTextChanged

+ OnSelectedIndexChanged

+ OnCheckedChanged

3. Bài tập

3.1 Tạo Form login gồm:

+ 2 TextBox : user và pass

+ 1 Button : login

* Khi nhấn vào Button thì in ra dữ liệu nhập và ẩn form login

3.2 Tạo MasterPage

+Tạo ContentPlaceHolder header

+ Tạo ContentPlaceHolder Content

+ Tạo ContentPlaceHolder Footer

-> Liên kết với masterpage

4. StyleProgramming

4.1 NameSpace

Kí tự đầu hoa

4.2 Class

Kí tự đầu hoa.

4.3 Function

Kí tự đầu hoa.

4.4 Properties

\_thường

5. MasterPage

Trang .master

<div>

<h1>Đây Là MasterPage</h1>

<asp:ContentPlaceHolder ID="cph\_header" runat="server">

</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="cph\_cọntent" runat="server">

</asp:ContentPlaceHolder>

<asp:ContentPlaceHolder ID="cph\_footer" runat="server">

</asp:ContentPlaceHolder>

</div>

Trang Kế thừa

<asp:Content ID="Content1" ContentPlaceHolderID="cph\_header" runat="server">

<h2>Đây là Header</h2>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="cph\_cọntent" runat="server">

<h2>Đây là content</h2>

</asp:Content>

<asp:Content ID="Content3" ContentPlaceHolderID="cph\_footer" runat="server">

<h2>Đây là Footer</h2>

</asp:Content>

6. Kết nối với ADO.NET

ADO.NET là 1 lớp đối tượng cầu nối giữa giao diện và CSDL.

public SqlConnection conn = new SqlConnection(@"Data Source=.\sqlexpress;Initial Catalog=Test;Integrated Security=True");

public const string status = " WHERE status =0";

public void Connect()

{

if (conn.State == ConnectionState.Closed || conn.State == ConnectionState.Broken)

{

conn.Open();

}

}

public void DisConnect()

{

if (conn.State != ConnectionState.Closed)

{

conn.Close();

}

}

6.1 System.Data;

public String SelectUseExeReader()

{

Connect();

SqlCommand cmd = new SqlCommand("SELECT \* FROM tbUser"+ status, conn);

SqlDataReader reader = cmd.ExecuteReader();

DataTable dtb = new DataTable();

//đổ tất cả

dtb.Load(reader);

string dsUser = string.Empty;

//Lập qua từng object lấy ra row

foreach(DataRow row in dtb.Rows)

{

dsUser += row["username"].ToString() + "<br/>";

}

DisConnect();

return dsUser;

}

Kết nối, Truy Vấn , Parameter,Reader,Adapter

SqlConnection, SqlCommand, SqlParameter, SqlDataReader, SqlDataAdapter

6.2 System.Data.SqlClient;

Bảng,Cột tạm

DataSet, DataTable, DataRow, DataColumn

SqlDataAdapter adapter = new SqlDataAdapter("SELECT \* FROM tbUser"+ status, conn);

DataTable dtb = new DataTable();

//đổ tất cả

adapter.Fill(dtb);

6.3. Ảnh hưởng

6.3.1 DataSet

Tương tác 1 trong 2 sẽ không ảnh hưởng.

Ví dụ xóa 1 dòng (row) trong dataSet sẽ không ảnh hưởng đến CSDL . nếu không thực hiện cập nhật.

1. Đặt tên
2. Truy Cập 1 dòng

6.3.2 DataTable

Tương tác 1 trong 2 sẽ không ảnh hưởng.

1. Fill

* Load tất cả

SqlDataAdapter

adapter.Fill(dtb);

1. Load

* Load tất cả từ trên xuống dưới

SqlDataReader

dtb.Load(reader);

1. Truy Cập 1 dòng

foreach(DataRow row in dtb.Rows)

{

dsUser += row["username"].ToString() + "<br/>";

}

6.4 Thao Tác

6.4 SqlCommand.

SqlCommand cmd = new SqlCommand("INSERT INTO tbUser(username,password) VALUES('"+ username + "','"+ password + "')", conn);

SqlCommand cmd = new SqlCommand("SELECT count(\*) FROM tbUser"+status,conn);

6.5 SqlParameter

SqlCommand cmd = new SqlCommand();

cmd.Connection = conn;

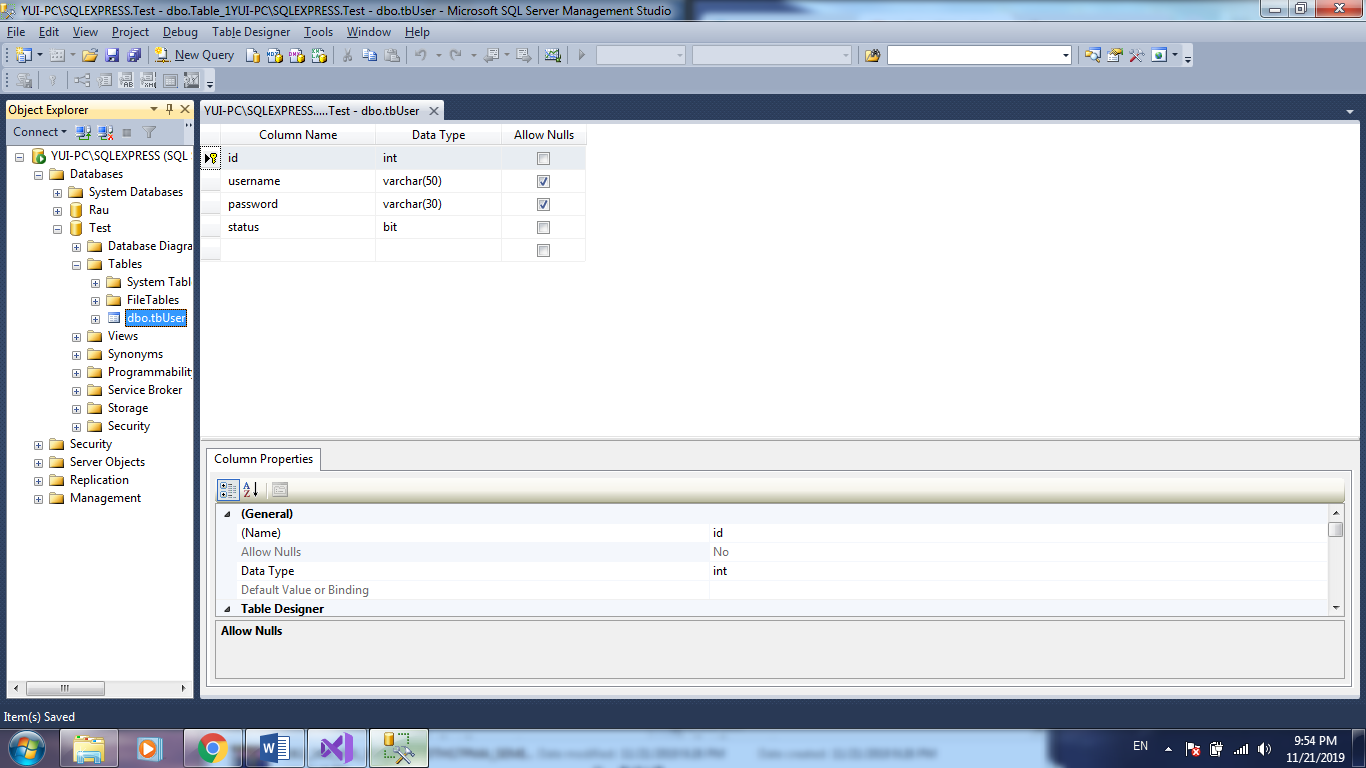
cmd.CommandText = "UPDATE tbUser SET status =1 WHERE id=@id";

cmd.Parameters.AddWithValue("@id", id);

6.5 Bài Tập.

6.5.1 Thực hiện các câu lệnh truy vấn đơn giản. Tạo from login và khi đang nhập vào hiện ra danh sách.với các chức năng như bên dưới.

+ Database



+ Select (Trả về)

- ExecuteReader()

- SqlDataAdapter()

- Tính tổng bằng ExcuteScalar

+ Insert,Update,Delete(Cập nhật status) (Không trả về)

* Không tham số
* Có tham số

7. Mô Hình 3 Lớp.

7.1 DAO

7.1 DataProvider

public class DataProvider

{

public static SqlConnection conn = new SqlConnection(@"Data Source=.\sqlexpress;Initial Catalog=WebBanHang;Integrated Security=True");

public static SqlDataAdapter adapter = new SqlDataAdapter();

public DataProvider() { }

private static SqlConnection OpenConnection()

{

if(conn.State == ConnectionState.Closed || conn.State == ConnectionState.Broken)

{

conn.Open();

}

return conn;

}

private static void Disconnection()

{

if (conn.State != ConnectionState.Closed || conn.State != ConnectionState.Broken)

{

conn.Close();

}

}

public static DataTable ExecuteSelectQuery(string query,SqlParameter[] param)

{

SqlCommand cmd = new SqlCommand();

DataTable dtbKetQua = new DataTable();

try

{

cmd.Connection = OpenConnection();

cmd.CommandText = query;

cmd.Parameters.AddRange(param);

adapter.SelectCommand = cmd;

adapter.Fill(dtbKetQua);

Disconnection();

}

catch (Exception)

{

return null;

}

return dtbKetQua;

}

public static int ExecuteInsertQuery(string query, SqlParameter[] param)

{

SqlCommand cmd = new SqlCommand();

int rowsAffected;

try

{

cmd.Connection = OpenConnection();

cmd.CommandText = query;

cmd.Parameters.AddRange(param);

adapter.InsertCommand = cmd;

rowsAffected = cmd.ExecuteNonQuery();

Disconnection();

}

catch (Exception)

{

return 0;

}

return rowsAffected;

}

public static int ExecuteUpdateQuery(string query, SqlParameter[] param)

{

SqlCommand cmd = new SqlCommand();

int rowsAffected;

try

{

cmd.Connection = OpenConnection();

cmd.CommandText = query;

cmd.Parameters.AddRange(param);

adapter.InsertCommand = cmd;

rowsAffected = cmd.ExecuteNonQuery();

Disconnection();

}

catch (Exception)

{

return 0;

}

return rowsAffected;

}

public static int ExecuteDeleteQuery(string query, SqlParameter[] param)

{

SqlCommand cmd = new SqlCommand();

int rowsAffected;

try

{

cmd.Connection = OpenConnection();

cmd.CommandText = query;

cmd.Parameters.AddRange(param);

adapter.DeleteCommand = cmd;

rowsAffected = cmd.ExecuteNonQuery();

Disconnection();

}

catch (Exception)

{

return 0;

}

return rowsAffected;

}

}

7.2 DangNhapDAO

public static bool KTTKTonTai(string tenTK)

{

string query = "SELECT COUNT(\*) FROM TaiKhoan WHERE TenTaiKhoan =@TenTaiKhoan ";

SqlParameter[] param = new SqlParameter[1];

param[0] = new SqlParameter("@TenTaiKhoan", tenTK);

return Convert.ToInt32(DataProvider.ExecuteSelectQuery(query, param).Rows[0][0]) == 1;

}

public static string LayMatKhau(string tenTK)

{

string query = "SELECT MatKhau FROM TaiKhoan WHERE TenTaiKhoan =@TenTaiKhoan";

SqlParameter[] param = new SqlParameter[1];

param[0] = new SqlParameter("@TenTaiKhoan", tenTK);

return DataProvider.ExecuteSelectQuery(query, param).Rows[0][0].ToString();

}

public static bool Them(string tenTK, string mk, string email, string sdt,string diaChi, string hoTen)

{

string query = "INSERT INTO [dbo].[TaiKhoan] ([TenTaiKhoan] ,[MatKhau] ,[Email] ,[SDT] ,[DiaChi] ,[HoTen]) VALUES (@TenTaiKhoan ,@MatKhau,@Email,@SDT,@DiaChi,@HoTen)";

SqlParameter[] param = new SqlParameter[6];

param[0] = new SqlParameter("@TenTaiKhoan", tenTK);

param[1] = new SqlParameter("@MatKhau", mk);

param[2] = new SqlParameter("@Email", email);

param[3] = new SqlParameter("@SDT", sdt);

param[4] = new SqlParameter("@DiaChi", diaChi);

param[5] = new SqlParameter("@HoTen", hoTen);

return DataProvider.ExecuteInsertQuery(query, param) == 1;

}

7.2 BUS

public static bool KTDangNhap(string tenTK,string mk)

{

if (!TaiKhoanDAO.KTTKTonTai(tenTK))

{

return false;

}

else

{

return mk == TaiKhoanDAO.LayMatKhau(tenTK);

}

}

public static bool KTDangKi(string tenTK, string mk,string email,string sdt,string diaChi,string hoTen)

{

if (TaiKhoanDAO.KTTKTonTai(tenTK))

{

return false;

}

else

{

return TaiKhoanDAO.Them(tenTK,mk,email,sdt,diaChi,hoTen);

}

}

7.3 GUI

protected void ClickBtnDangNhap(object sender,EventArgs e)

{

string tenTK = txtTenTaiKhoan.Text;

string mk = txtMatKhau.Text;

if (TaiKhoanBUS.KTDangNhap(tenTK, mk))

{

Response.Write("<script>alert('Đăng Nhập Thành Công');</script>");

}

else

{

Response.Write("<script>alert('Đăng Nhập Thất Bại');</script>");

}

}

protected void ClickBtnDangKi(object sender, EventArgs e)

{

string tenTaiKhoan = txtTenTaiKhoan.Text.ToString();

string matKhau = txtMatKhau.Text.ToString();

string email = txtEmail.Text.ToString();

string sdt = txtSDT.Text.ToString();

string diaChi = txtDiaChi.Text.ToString();

string hoTen = txtHoTen.Text.ToString();

if (TaiKhoanBUS.KTDangKi(tenTaiKhoan, matKhau,email,sdt,diaChi,hoTen))

{

Response.Write("<script> alert('Thêm Thành Công'); </script>");

}

else

{

Response.Write("<script> alert('Thêm Thất Bại'); </script>");

}

}

7.4 Bài Tập

Viết Đăng nhập và đăng kí sử dụng mô hình 3 lớp.

+ Đăng nhập

* Gồm 2 textBox: username, password và 1 button signin
* Validate

+ Đăng kí

* Gồm 6 TextBox: usermame,password ,... và 1 button signup.

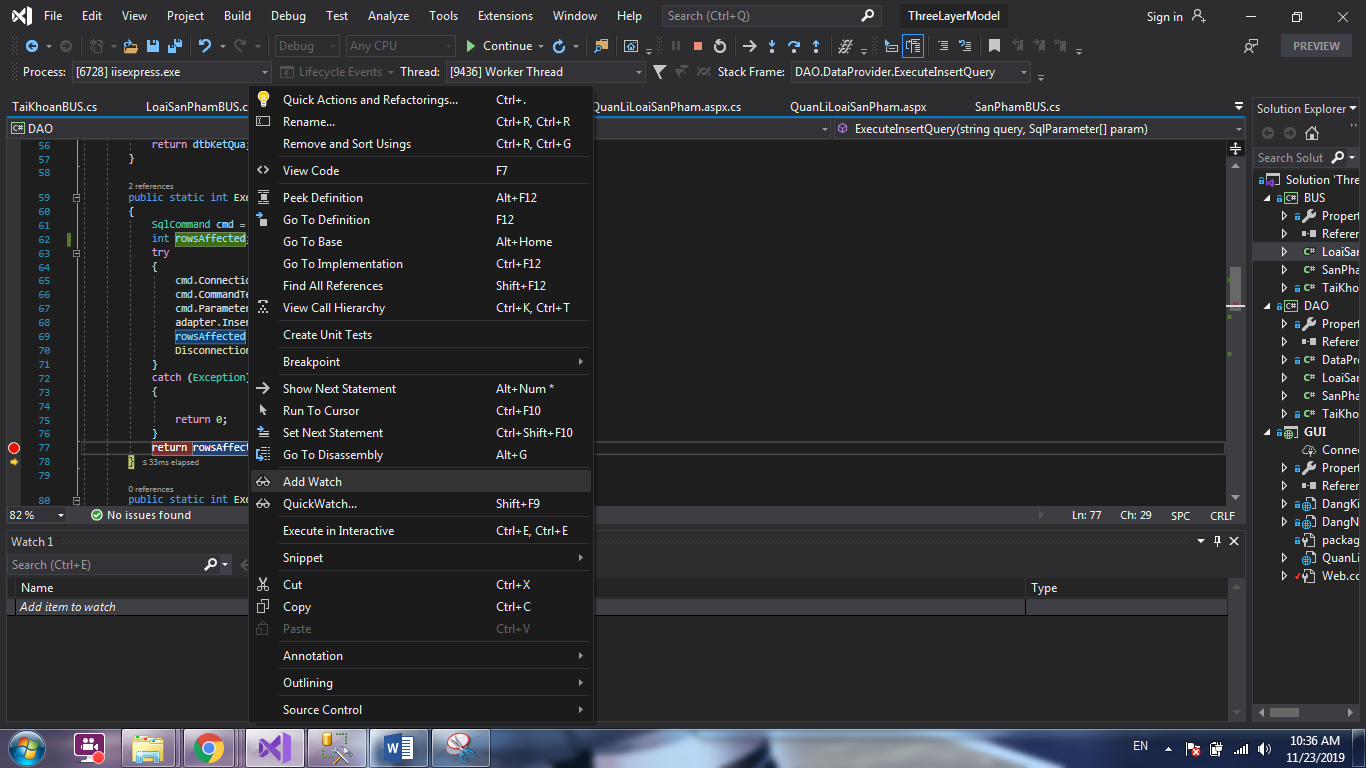
8. Debug

B1: Đặt Breakpoint

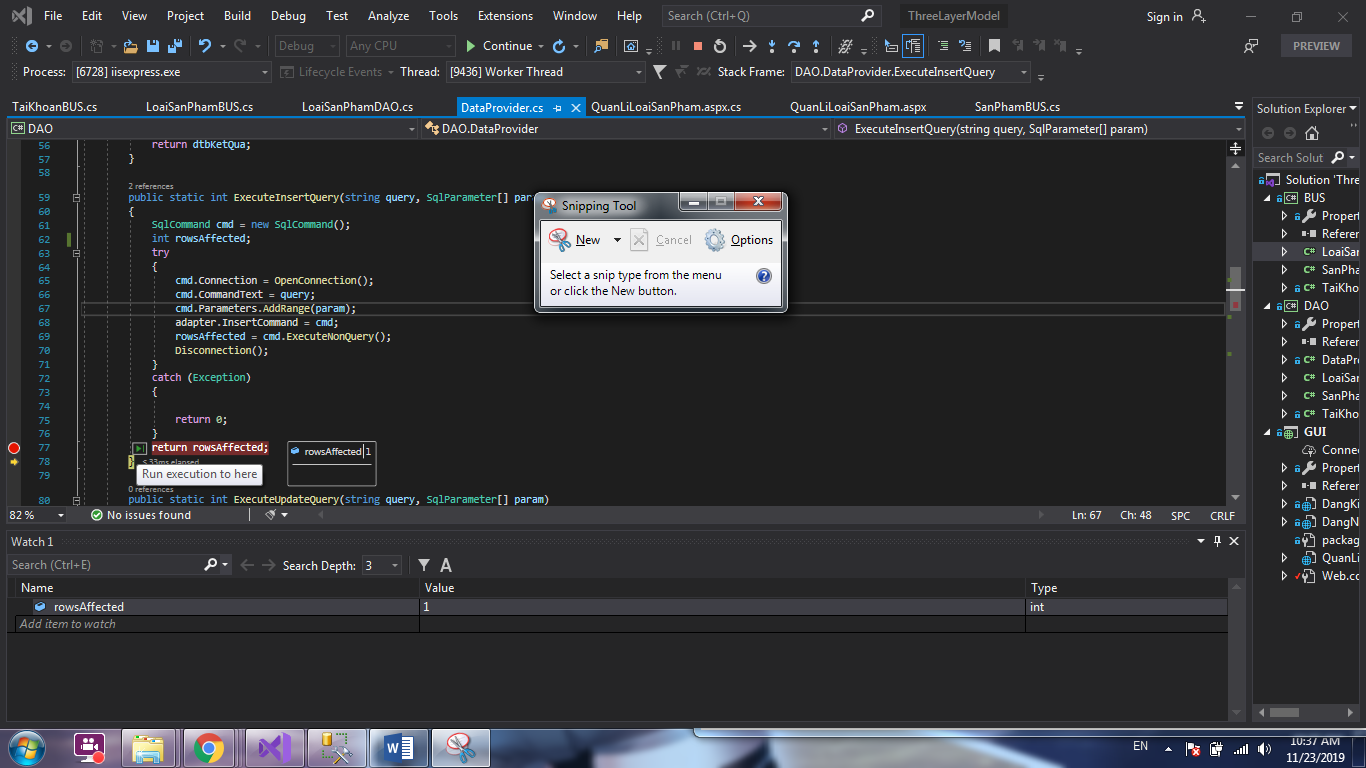
B2: Fn+F5: Bật debug

B3: đợi cho nó đến breakpoint

B4: add Watch



B5.Tiếp tục chương trình bỏ qua các breakpoint còn lại



9 GridView

9.1 GUI

protected void Page\_Load(object sender, EventArgs e)

{

grvLoaiSanPham.DataSource = LoaiSanPhamBUS.LayDSLoaiSanPham();

//cho nó gắn vào các column như cột bên dưới grid không có nó sẽ không hiện

grvLoaiSanPham.DataBind();

}

protected void BtnThem\_Click(object sender, EventArgs e)

{

string maLoaiSP = txtMaLoaiSP.Text;

string tenLoaiSP = txtTenLoaiSP.Text;

bool trangThai = cbTrangThai.Checked;

if (LoaiSanPhamBUS.ThemLoaiSanPham(maLoaiSP,tenLoaiSP,trangThai))

{

grvLoaiSanPham.DataSource = LoaiSanPhamBUS.LayDSLoaiSanPham();

grvLoaiSanPham.DataBind();

}

else

{

Response.Write("<script> alert('Thêm Thất Bại') </script>");

}

}

9.2 BUS

public static DataTable LayDSLoaiSanPham()

{

return LoaiSanPhamDAO.LayDSLoaiSanPham();

}

public static bool ThemLoaiSanPham(string maLoaiSP,string tenLoaiSP,bool trangThai)

{

if (LoaiSanPhamDAO.KTMaLoaiSanPhamTonTai(maLoaiSP))

{

return false;

}

else

{

return LoaiSanPhamDAO.ThemLoaiSanPham(maLoaiSP, tenLoaiSP, trangThai);

}

}

9.3 DAO

public static DataTable LayDSLoaiSanPham()

{

string query = "SELECT \* FROM LoaiSanPham";

SqlParameter[] param = new SqlParameter[0];

return DataProvider.ExecuteSelectQuery(query, param);

}

public static bool ThemLoaiSanPham(string maLoaiSP,string tenLoaiSP,bool trangThai)

{

string query = "INSERT INTO LoaiSanPham(MaloaiSP,TenLoaiSP,TrangThai) VALUES(@MaLoaiSP,@TenLoaiSP,@TrangThai)";

SqlParameter[] param = new SqlParameter[3];

param[0] = new SqlParameter("@MaLoaiSP", maLoaiSP);

param[1] = new SqlParameter("@TenLoaiSP", tenLoaiSP);

param[2] = new SqlParameter("@TrangThai", trangThai);

return DataProvider.ExecuteInsertQuery(query, param) == 1;

}

public static bool KTMaLoaiSanPhamTonTai(string maLoaiSP)

{

string query = "SELECT COUNT(\*) FROM LoaiSanPham WHERE MaLoaiSP = @MaloaiSP";

SqlParameter[] param = new SqlParameter[1];

param[0] = new SqlParameter("@MaLoaiSP", maLoaiSP);

return Convert.ToInt32(DataProvider.ExecuteSelectQuery(query, param).Rows[0][0]) == 1;

}