```
using Microsoft.Office.Core;
using MySql.Data.MySqlClient;
using sigal.Clases;
using System;
using System.Windows.Forms;

namespace sigal
{
    public partial class Det_Entradas : Form
    {
        int r = 0;
        float usd = 0;
        Int64 ID_ENT;
#pragma warning disable CS0169 // El campo 'Det_Entradas.PRECIO' nunca se usa
        Double PRECIO, TOTAL;
#pragma warning restore CS0169 // El campo 'Det_Entradas.PRECIO' nunca se usa
#pragma warning disable CS0414 // El campo 'Det_Entradas.filtro2' está asignado pero su valor nunca se usa
        string filtro2 = "";
#pragma warning restore CS0414 // El campo 'Det_Entradas.filtro2' está asignado pero su valor nunca se usa
        conexion conn = new conexion();
        conexion conn1 = new conexion();
        conexion conn2 = new conexion();
        conexion conn3 = new conexion();
        conexion conn4 = new conexion();
        conexion conn5 = new conexion();


        MySqlDataReader R;
        public Det_Entradas()
        {
            InitializeComponent();
        }

        private void bt_agregar_Click(object sender, EventArgs e)
        {
            if (validar() == 0) { return; }
            usd = float.Parse(txtusd.Text.ToString());
            panel2.Enabled = false;
            Grid_Det_Ent.Rows.Insert(r, txtid.Text, txtnombre.Text, txtcantidad.Text, txtprecio.Text, txtcosto.Text, txtmarca.Text, txtlote.Text, 
txtrequisicion.Text, txtcaducidad.Text,checkcanje.Checked,usd);
            r = r + 1;
            btnGrabar.Enabled = true;
            limpiar();
            //cargar_det();
            txtid.Focus();
        }

        internal static void show()
        {
            throw new NotImplementedException();
        }

        private void txtid_Leave(object sender, EventArgs e)
        {
            if (txtid.Text.ToString() != "")
            {
                R = conn.GetData("SELECT nombre, precio FROM cat_articulos where id=" + txtid.Text + ";");
                if (R.HasRows)
                {
                    R.Read();
                    txtnombre.Text = R[0].ToString().ToUpper();
                }
                else
                {
                    MessageBox.Show("No existe ese numero de articulo¡");
                    txtid.Focus();
                }
                R.Close();
            }
        }

        private void Det_Entradas_Load(object sender, EventArgs e)
        {
            //mensajes.SetToolTip(txtBusqueda, "Registro a buscar...");
            mensajes.SetToolTip(btnNuevo, "Clic para agregar un nuevo registro...");
            //mensajes.SetToolTip(btnEditar, "Mostrar Detalle del Registro...");
            mensajes.SetToolTip(btnSalir, "Regresar a Pantalla Anterior");
            mensajes.SetToolTip(btnGrabar, "Guardar Registro...");
            txtcaducidad.Value = DateTime.Now;
            cargar();
            conn.AbrirBD();
            conn1.AbrirBD();
            R = conn.GetData("SELECT usd FROM configuracion;");
            if (R.HasRows)
            {
                R.Read();
                usd = float.Parse(R[0].ToString());
            }
            else
            {
                MessageBox.Show("Error al Cargar eL valor del USD, favor de revisar la configuracion del sistema");
            }
            R.Close();
            cbtipo.Focus();
        }
        void cargar_det()
        {
            Grid_Det_Ent.DataSource = conn.GetTable("select d.id,d.idarticulo as Id_Articulo,a.nombre as Descripcion,cantidad as Cantidad," +
                "d.precio as Precio,totalprecio as Total,estado as Estado,e.nombre as Auditor " +
                "from det_entrada d, empleados e, cat_articulos a where d.idauditor=e.id and d.idarticulo=a.id and d.idinventario=" +
                ID_ENT + " order by d.id");
            if ((conn.PropertyDataSet.Tables[0].Rows.Count != 0))
            {
                Grid_Det_Ent.Columns["id"].Visible = false;
                Grid_Det_Ent.AutoResizeColumns();
            }
        }
        void cargar()
        {
            txtfecha_factura.Value = DateTime.Now;
            txtfecha.Value = DateTime.Now;
            txtcaducidad.Value = DateTime.Now;
            conn2.AbrirBD();
            cbproveedor.DisplayMember = "nombre";
            cbproveedor.ValueMember = "id";
            cbproveedor.DataSource = conn2.GetTable("select id,nombre from cat_proveedores");
            conn2.cerrarBd();

            conn3.AbrirBD();
            cbrecibio.DisplayMember = "nombre";
            cbrecibio.ValueMember = "id";
            cbrecibio.DataSource = conn3.GetTable("select id,nombre from empleados");
            conn3.cerrarBd();

            conn4.AbrirBD();
            cbalmacen.DisplayMember = "nombre";
            cbalmacen.ValueMember = "id";
            if (configuracion.ID_ALMACEN != 0)
            {
                cbalmacen.DataSource = conn4.GetTable("select id,nombre from cat_almacen where id=" + configuracion.ID_ALMACEN);
                cbalmacen.Enabled = false;
            }
            else { cbalmacen.DataSource = conn4.GetTable("select id,nombre from cat_almacen where id>0"); }
            conn4.cerrarBd();

            conn5.AbrirBD();
            cbarea.DisplayMember = "clue";
            cbarea.ValueMember = "id";
            cbarea.DataSource = conn5.GetTable("select id,clue from cat_clues");
            conn5.cerrarBd();

        }
        private void Det_Entradas_FormClosing(object sender, FormClosingEventArgs e)
        {
            conn.cerrarBd();
            conn1.cerrarBd();
        }

        private void txtid_TextChanged(object sender, EventArgs e)
        {
            if (txtid.Text == "")
            { txtnombre.Enabled = true; }
            else { txtnombre.Enabled = false; }
        }
        private void txtcantidad_KeyPress(object sender, KeyPressEventArgs e)
        {
            if (Char.IsDigit(e.KeyChar))
            { e.Handled = false; }
            else if (Char.IsControl(e.KeyChar))
            { e.Handled = false; }
            else if (Char.IsSeparator(e.KeyChar))
            { e.Handled = false; }
            else if (e.KeyChar == 46)
            { e.Handled = false; }
            else { e.Handled = true; }
        }
        void limpiar()
        {
            txtid.Text = "";
            txtnombre.Text = "";
            txtcantidad.Text = "0";
            txtprecio.Text = "0";
            txtcosto.Text = "0";
            txtmarca.Text = "";
            txtrequisicion.Text = "";
            txtlote.Text = "";

        }

        private void txtnombre_TextChanged(object sender, EventArgs e)
```

```
{
    if (txtnombre.TextLength > 3 && txtid.Text == "")
    {
        Grid_Articulos.Visible = true;

        //grid1.DataSource = conn.GetTable("select id,nombre_corto,nombre_largo from catarticulos");

        Grid_Articulos.DataSource = conn1.GetTable("select id as Id_Producto,Nombre," +
            "  Descripcion from cat_articulos where nombre like '%" + txtnombre.Text + "%' order by nombre limit 20");
        if (Grid_Articulos.RowCount < 2)
        {
            Grid_Articulos.Visible = false;
            return;
        }
        Grid_Articulos.AutoResizeColumns();
    }
}

private void Grid_Articulos_DoubleClick(object sender, EventArgs e)
{
    txtid.Text = Grid_Articulos.CurrentRow.Cells["Id_Producto"].Value.ToString();
    txtnombre.Text = Grid_Articulos.CurrentRow.Cells["Nombre"].Value.ToString();
    Grid_Articulos.Visible = false;
    txtcantidad.Focus();
    txtnombre.Enabled = false;
}

private void txtid_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
```

```
            { txtcantidad.Focus(); }
            else if (e.KeyCode == Keys.Down) { txtnombre.Focus(); }
        }
        int validar()
        {

            if (txtid.Text != "" && txtcantidad.Text != "" && txtprecio.Text != "" && txtmarca.Text != "" && txtlote.Text != "")
            { return 1; }
            return 0;
        }
        int validar_ent()
        {
            string P = suma_costos();
            if (txtfecha.Text != "" && cbtipo.Text != "" && cbalmacen.Text != "" && cbrecibio.Text != "" && cbproveedor.Text != "" && txtimporte.Text
== P.ToString())
            { return 1; }
            return 0;
        }

        private void bt_grabar_Click(object sender, EventArgs e)
        {
            try
            {
                if (validar_ent() == 0) { return; }
                Boolean resp;
                conn.AbrirBD();
                conn.inicio();
                resp = conn.Executa("insert into entradas (tipo,fechaentrada,numfactura,fechafactura,costofactura," +
                    "idproveedor,idarea,idalmacen,idrecibio,estatus) values('" +
                cbtipo.Text.ToUpper() + "','" + txtfecha.Value.ToString("yyyy-MM-dd") + "','" +
txtnumero_factura.Text.ToUpper() + "','" + txtfecha_factura.Value.ToString("yyyy-MM-dd") + "'," +
txtimporte.Text.ToUpper() + "," + cbproveedor.SelectedValue + "," + cbarea.SelectedValue + "," +
cbalmacen.SelectedValue + "," + cbrecibio.SelectedValue + ",'A')");
//OBTENER ID INVENTARIO
if (!resp) { conn.fallo(); return; }
R = conn.GetData("SELECT id FROM entradas where tipo='" + cbtipo.Text.ToUpper()
        + "' and fechaentrada='" + txtfecha.Value.ToString("yyyy-MM-dd")
        + "' and fechafactura='" + txtfecha_factura.Value.ToString("yyyy-MM-dd")
        + "' and numfactura='" + txtnumero_factura.Text.ToString()
        + "' and idproveedor=" + cbproveedor.SelectedValue + ";");
if (R.HasRows)
{
    R.Read();
    ID_ENT = Int32.Parse(R[0].ToString());
    label16.Visible = true;
    label16.Text = "E N T R A D A   #" + ID_ENT.ToString();
}
R.Close();
resp = grabar_grid();
if (!resp) { return; }
resp = conn.Executa("insert into bitacora (fechasis,usuario,motivo,tabla,idtabla) values('"
       + DateTime.Now.ToString("yyyy-MM-dd hh:mm:ss") + "','" + configuracion.USER +
       "','ALTA','ENTRADAS'," + ID_ENT + ")");
if (!resp) { conn.fallo(); return; }
conn.exito();
conn.cerrarBd();
MessageBox.Show("Alta de Entrada al Almacen " + ID_ENT + " Realizada con Exito¡");

MessageBox.Show("CREAR REPORTE DE ENTRADA Y MOSTRARLO AQUI¡");
```

```
                this.Close();
            }
            catch (Exception ex)
            {
                MessageBox.Show("Fallo al guardar en la BD:" + ex.Message, Application.ProductName, MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
            }

        }

        private void Grid_Det_Ent_DoubleClick(object sender, EventArgs e)
        {
            if (Grid_Det_Ent.CurrentRow.Index != -1)
            {
                Grid_Det_Ent.Rows.RemoveAt(Grid_Det_Ent.CurrentRow.Index);
            }
        }

        private void btsalir_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        private void txtprecio_KeyDown(object sender, KeyEventArgs e)
        {
            if (e.KeyCode == Keys.Enter || e.KeyCode == Keys.Down)
            {
                txtmarca.Focus();
            }
            else if (e.KeyCode == Keys.Up)
```

```
    {
        txtcantidad.Focus();
    }

}

private void txtcantidad_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter || e.KeyCode == Keys.Down)
    {
        txtprecio.Focus();
    }
    else if (e.KeyCode == Keys.Up)
    {
        txtid.Focus();
    }
}


private void bt_agregar_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    { txtid.Focus(); }
}



private void txtid_KeyPress(object sender, KeyPressEventArgs e)
{
    if (Char.IsDigit(e.KeyChar))
    { e.Handled = false; }
    else if (Char.IsControl(e.KeyChar))
    { e.Handled = false; }
    else if (Char.IsSeparator(e.KeyChar))
    { e.Handled = false; }
    else { e.Handled = true; }
}

private void txtprecio_Leave(object sender, EventArgs e)
{
    if (modulo1.IsNumeric(txtcantidad.Text.ToString()) && modulo1.IsNumeric(txtprecio.Text.ToString()))
    {
        costo();
    }
}

private void txtprecio_KeyPress(object sender, KeyPressEventArgs e)
{
    if (Char.IsDigit(e.KeyChar))
    { e.Handled = false; }
    else if (Char.IsControl(e.KeyChar))
    { e.Handled = false; }
    else if (Char.IsSeparator(e.KeyChar))
    { e.Handled = false; }
    else if (e.KeyChar == 46)
    { e.Handled = false; }
    else { e.Handled = true; }
}
```

```
private void txtcantidad_Leave(object sender, EventArgs e)
{
    if (modulo1.IsNumeric(txtcantidad.Text.ToString()) && modulo1.IsNumeric(txtprecio.Text.ToString()))
    {
        costo();
    }

}

void costo()
{
    if (txtcantidad.Text != "" && txtprecio.Text != "")
    {
        if (cbxcambio.Text.ToString() == "MXN")
        {
            usd = 1;
        }
        else {
            usd = float.Parse(txtusd.Text.ToString());
        }

        float costo = float.Parse(txtcantidad.Text) * float.Parse(txtprecio.Text) * usd;
        float iva1 = float.Parse(costo.ToString()) * configuracion.IVA;
        txtcosto.Text = costo.ToString();
        txtiva.Text = iva1.ToString();
    }
}


private void txtlote_KeyDown(object sender, KeyEventArgs e)
```

```
{
    if (e.KeyCode == Keys.Enter || e.KeyCode == Keys.Down)
    {
        txtrequisicion.Focus();
    }
    else if (e.KeyCode == Keys.Up)
    {
        txtmarca.Focus();
    }
}


private void txtmarca_KeyDown_1(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter || e.KeyCode == Keys.Down)
    {
        txtlote.Focus();
    }
    else if (e.KeyCode == Keys.Up)
    {
        txtprecio.Focus();
    }
}


private void txtrequisicion_KeyDown_1(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter || e.KeyCode == Keys.Down)
    {
        btnNuevo.Focus();
    }
    else if (e.KeyCode == Keys.Up)
    {
        txtlote.Focus();
    }
}


public string suma_costos()
{
    string resp = "";
    TOTAL = 0;
    for (int fila = 0; fila < Grid_Det_Ent.Rows.Count - 1; fila++)
    {
        TOTAL = TOTAL + double.Parse(Grid_Det_Ent.Rows[fila].Cells[4].Value.ToString());
    }
    resp = TOTAL.ToString();
    return resp;
}

private void txtprecio_TextChanged(object sender, EventArgs e)
{

}

private void txtfecha_ValueChanged(object sender, EventArgs e)
{

}
```

```
private void txtcantidad_TextChanged(object sender, EventArgs e)
{

}

private void cbxcambio_SelectedIndexChanged(object sender, EventArgs e)
{

}

private void cbxcambio_Leave(object sender, EventArgs e)
{
    if (cbxcambio.Text == "USD")
    {
        txtusd.Text = usd.ToString();
        txtusd.Visible = true;
    }
    else { txtusd.Visible= false;
    }
}

Boolean grabar_grid()
{

    DateTime fec = DateTime.Now;
    int canj = 0;
    double nuevo_costo = 0, nueva_exis = 0;
    for (int fila = 0; fila < Grid_Det_Ent.Rows.Count - 1; fila++)
    {
```

```
if (Grid_Det_Ent.Rows[fila].Cells[9].Value.ToString()=="true")
{
   canj = 1;
}

fec = DateTime.Parse(Grid_Det_Ent.Rows[fila].Cells[8].Value.ToString());
nueva_exis = double.Parse(Grid_Det_Ent.Rows[fila].Cells[2].Value.ToString());
nuevo_costo = double.Parse(Grid_Det_Ent.Rows[fila].Cells[4].Value.ToString());
string str = "insert into det_entradas (identrada,idarticulo," +
   "cantidad,existencia,precio,totalprecio,marca,requisicion,lote,caducidad,canje,usd) " +
   "values(" + ID_ENT + "," +
   Grid_Det_Ent.Rows[fila].Cells[0].Value.ToString() + "," +
   Grid_Det_Ent.Rows[fila].Cells[2].Value.ToString() + "," +
   Grid_Det_Ent.Rows[fila].Cells[2].Value.ToString() + "," +
   Grid_Det_Ent.Rows[fila].Cells[3].Value.ToString() + "," +
   Grid_Det_Ent.Rows[fila].Cells[4].Value.ToString() + ",'" +
   Grid_Det_Ent.Rows[fila].Cells[5].Value.ToString() + "','" +
   Grid_Det_Ent.Rows[fila].Cells[6].Value.ToString() + "','" +
   Grid_Det_Ent.Rows[fila].Cells[7].Value.ToString() + "','" +
   fec.ToString("yyyy-MM-dd") + "'," +
   canj + "," +
   Grid_Det_Ent.Rows[fila].Cells[10].Value.ToString() + ")";
//MessageBox.Show(str);
if (!conn.Executa(str))
{
   conn.fallo();
   return false;
}
//GUARDAR EN LA TABLA KARDEX EL MOV DE ENTRADA
str = "insert into kardex (idtabla,idarticulo,entrada,salida,precio,costo,fecha)" +
                    "values(" + ID_ENT + "," +
                    Grid_Det_Ent.Rows[fila].Cells[0].Value.ToString() + "," +
                    Grid_Det_Ent.Rows[fila].Cells[2].Value.ToString() + ",0," +
                    Grid_Det_Ent.Rows[fila].Cells[3].Value.ToString() + "," +
                    Grid_Det_Ent.Rows[fila].Cells[4].Value.ToString() + ",'" +
                    txtfecha.Value.ToString("yyyy-MM-dd") + "')";
                //MessageBox.Show(str);
                if (!conn.Executa(str))
                {
                    conn.fallo();
                    return false;
                }
                //CALCULAR PRECIO
                str = "update cat_articulos " +
                        "set precio=((existencia*precio)+" + nuevo_costo + ")/(existencia+" + nueva_exis +
                        "), existencia=existencia+" + nueva_exis + " where id=" + Grid_Det_Ent.Rows[fila].Cells[0].Value.ToString();
                if (!conn.Executa(str))
                {
                    conn.fallo();
                    return false;
                }

            }
            //Inventario.ActiveForm=true;
            return true;
        }

    }

}
```