

```

using MySql.Data.MySqlClient;
using sigal.Clases;
using System;
using System.Windows.Forms;

namespace sigal
{
    public partial class Det_Salidas : Form
    {
        int r = 0;
        Int64 ID_SAL;
#pragma warning disable CS0169 // El campo 'Det_Salidas.PRECIO' nunca se usa
        Double PRECIO, TOTAL = 0;
#pragma warning restore CS0169 // El campo 'Det_Salidas.PRECIO' nunca se usa
        Boolean resp;
        conexion conn = new conexion();
        conexion conn1 = new conexion();
        conexion conn2 = new conexion();
        conexion conn3 = new conexion();
        conexion conn4 = new conexion();
        conexion conn5 = new conexion();

        MySqlConnection R;
        public Det_Salidas()
        {
            InitializeComponent();
        }

        private void bt_agregar_Click(object sender, EventArgs e)

```

```

{
    if (validar() == 0 && validar_sal() == 0) { return; }
    panel2.Enabled = false;
    Grid_Det_Sal.Rows.Insert(r, txtid.Text, txtnombre.Text, txtcantidad.Text, txtprecio.Text, txtcosto.Text, txtmarca.Text, txtrequisicion.Text,
txtcaducidad.Text);
    r = r + 1;
    TOTAL = double.Parse(txtimporte.Text) + double.Parse(txtcosto.Text);
    txtimporte.Text = TOTAL.ToString();
    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_corto, preciopromedio FROM catarticulos where id=" + txtid.Text + ";");
        if (R.HasRows)
        {
            R.Read();
            txtnombre.Text = R[0].ToString().ToUpper();
            txtprecio.Text = R[1].ToString();
        }
        else

```

```

    {
        MessageBox.Show("No existe ese numero de articulo");
        txtid.Focus();
    }
    R.Close();
}
}

```

```

private void Det_Salidas_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...");

```

```

    cargar();
    conn.AbrirBD();
    conn1.AbrirBD();
    cbtipo.Focus();

```

```

}

```

```

void cargar_det()

```

```

{

```

```

    return;

```

```

#pragma warning disable CS0162 // Se detectó código inaccesible

```

```

    Grid_Det_Sal.DataSource = conn.GetTable("select d.id,d.idarticulo as Id_Articulo,a.nombre_corto as Descripcion,cantidad as Cantidad," +
        "d.precio as Precio,totalprecio as Total,estado as Estado,d.marbete as Marbete,e.nombre as Auditor " +
        "from det_inventarios d, empleados e, catarticulos a where d.idauditor=e.id and d.idarticulo=a.id and d.idinventario=" +
        ID_SAL + " order by d.id");

```

```

#pragma warning restore CS0162 // Se detectó código inaccesible

```

```

if ((conn.PropertyDataSet.Tables[0].Rows.Count != 0))
{
    Grid_Det_Sal.Columns["id"].Visible = false;
    Grid_Det_Sal.AutoResizeColumns();
}

}

void cargar()
{
    txtfecha.Value = DateTime.Now;

    conn2.AbrirBD();
    cbentrego.DisplayMember = "nombre";
    cbentrego.ValueMember = "id";
    cbentrego.DataSource = conn2.GetTable("select id,nombre from empleados");
    conn2.cerrarBd();

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

    conn5.AbrirBD();
    cbarea.DisplayMember = "nombre";

```

```

        cbarea.ValueMember = "id";
        cbarea.DataSource = conn5.GetTable("select id,nombre from catareas");
        conn5.cerrarBd();

    }
    private void Det_Salidas_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 = "";
    txtprecio.Text = "";
    txtcosto.Text = "";
    txtmarca.Text = "";
    txtrequisicion.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");
        llena_grid("select id as Id_Producto,nombre_corto as Nombre," +
            " nombre_largo as Descripcion from catarticulos where nombre_corto like '%" + txtnombre.Text + "%' order by nombre_corto limit
20");

    }
}

private void llena_grid(string sql)
{
    Grid_Articulos.DataSource = conn1.GetTable(sql);
    if (Grid_Articulos.RowCount < 2)
    {
        Grid_Articulos.Visible = false;
    }
}

```

```

        return;
    }
    Grid_Articulos.AutoSizeColumns();
}

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 != "")
    { return 1; }
    return 0;
}

int validar_sal()
{
    string P = suma_costos();

```

```

        if (txtfecha.Text != "" && cbtipo.Text != "" && cbalmacen.Text != "" && cbentrego.Text != "" && txtimporte.Text == P.ToString())
        { return 1; }
        return 0;
    }

    private void bt_grabar_Click(object sender, EventArgs e)
    {
        try
        {
            if (validar_sal() == 0) { return; }
            conn.AbrirBD();
            conn.inicio();
            resp = conn.Executa("insert into salidas (tipo,fechasalida,importe,idrequisicion," +
                "identrego,idarea,idalmacen,estatus) values('" + cbtipo.Text.ToUpper() +
                "','" + txtfecha.Value.ToString("yyyy-MM-dd") + "','" + txtimporte.Text.ToUpper() +
                "','" + txtrequisicion.Text.ToUpper() + "','" + cbentrego.SelectedValue + "','" +
                cbarea.SelectedValue + "','" + cbalmacen.SelectedValue + "','A')");
            if (!resp) { conn.fallo(); return; }
            //OBTENER ID INVENTARIO
            R = conn.GetData("SELECT id FROM salidas where tipo='" + cbtipo.Text.ToUpper()
                + "' and fechasalida='" + txtfecha.Value.ToString("yyyy-MM-dd")
                + "' and importe=" + txtimporte.Text.ToUpper()
                + " and idarea=" + cbarea.SelectedValue + "');");
            if (R.HasRows)
            {
                R.Read();
                ID_SAL = Int32.Parse(R[0].ToString());
                label16.Visible = true;
                label16.Text = "S A L I D A  #" + ID_SAL.ToString();
            }
        }
    }

```



```

        R.Close();
        resp = grabar_grid();
        if (!resp) { conn.fallo(); 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','SALIDAS,'" + ID_SAL + "')");
        if (!resp) { conn.fallo(); return; }
        conn.exito();
        MessageBox.Show("Alta de Entrada al Almacen " + ID_SAL + " Realizada con Exito!");
        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_Sal_DoubleClick(object sender, EventArgs e)
{
    if (Grid_Det_Sal.CurrentRow.Index != -1)
    {
        TOTAL = double.Parse(txtimporte.Text) - double.Parse(Grid_Det_Sal.CurrentRow.Cells[4].Value.ToString());
        txtimporte.Text = TOTAL.ToString();
        Grid_Det_Sal.Rows.RemoveAt(Grid_Det_Sal.CurrentRow.Index);
    }
}

```

```
private void btssalir_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)
    {
        btnNuevo.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();
        R = conn.GetData("SELECT sum(de.existencia) FROM det_entradas de, entradas e " +
            " where de.identrada=e.id and e.idalmacen=" + cbalmacen.SelectedValue + " and idarticulo=" + txtid.Text + ";");
        if (R.HasRows)
        {
            R.Read();
            if (Int64.Parse(txtcantidad.Text.ToString()) > Int64.Parse(R[0].ToString()))
            {
                MessageBox.Show("No se cuenta con esa existencia en el almacen de ese articulo");
                txtcantidad.Focus();
            }
        }
    }
    else
    {

```

```

        MessageBox.Show("No existe ese numero de articulo");
        txtid.Focus();
    }
    R.Close();
}
}

void costo()
{
    if (txtcantidad.Text != "" && txtprecio.Text != "")
    {
        float costo = float.Parse(txtcantidad.Text) * float.Parse(txtprecio.Text);
        float iva1 = float.Parse(costo.ToString()) / configuracion.IVA;
        txtcosto.Text = costo.ToString();
        txtiva.Text = iva1.ToString();
    }
}

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

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

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

```
{
    txtfecha.Focus();
}

}

private void txtfecha_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter || e.KeyCode == Keys.Down)
    {
        txtrequisicion.Focus();
    }
    else if (e.KeyCode == Keys.Up)
    {
        cbtipo.Focus();
    }
}

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

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

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

```
Boolean grabar_grid()
{
    DateTime fec = DateTime.Now;
    double nuevo_costo = 0, nueva_exis = 0, sal = 0, precio = 0;
    Int64 ID_ART = 0;
    DateTime caducidad;
```



```
string f = " order by de.entrada asc", marca = "";  
if (configuracion.METODO == "PE")  
{ f = " order by de.entrada desc"; }
```

```
for (int fila = 0; fila < Grid_Det_Sal.Rows.Count - 1; fila++)  
{  
    fec = DateTime.Parse(txtfecha.Value.ToString());  
    ID_ART = Int64.Parse(Grid_Det_Sal.Rows[fila].Cells[0].Value.ToString());  
    nueva_exis = Int32.Parse(Grid_Det_Sal.Rows[fila].Cells[2].Value.ToString());  
    precio = double.Parse(Grid_Det_Sal.Rows[fila].Cells[3].Value.ToString());  
    nuevo_costo = double.Parse(Grid_Det_Sal.Rows[fila].Cells[4].Value.ToString());
```

```
    llena_grid("select de.existencia as existencia, de.precio as precio, de.caducidad as caducidad, de.marca, de.id as id from det_entradas  
de, entradas e " +  
        " where de.entrada=e.id and de.existencia>0 and de.idarticulo=" + ID_ART + " and e.idalmacen=" + cbalmacen.SelectedValue +  
f.ToString());
```

```
for (int fila2 = 0; (fila2 < Grid_Articulos.Rows.Count - 1 && nueva_exis > 0); fila2++)  
{  
    if (Int64.Parse(Grid_Articulos.Rows[fila2].Cells["existencia"].Value.ToString()) >= nueva_exis)  
    { sal = nueva_exis; }  
    else  
    { sal = Int64.Parse(Grid_Articulos.Rows[fila2].Cells["existencia"].Value.ToString()); }
```

```
    caducidad = DateTime.Parse(Grid_Articulos.Rows[fila2].Cells["caducidad"].Value.ToString());  
    marca = Grid_Articulos.Rows[fila2].Cells["marca"].Value.ToString();  
    nueva_exis = nueva_exis - sal;
```

```
if (configuracion.METODO != "PP")  
{
```

```

//cambiar precio y costo
precio = double.Parse(Grid_Articulos.Rows[fila2].Cells["precio"].Value.ToString());
nuevo_costo = precio * sal;
//MessageBox.Show("cambiar precio y costo");
}

```

```

string str = "insert into det_salidas (idsalida,idarticulo," +
"cantidad,precio,totalprecio,marca,caducidad) " +
"values(" + ID_SAL + "," + ID_ART + "," + sal + "," + precio + "," + nuevo_costo + "," +
marca.ToString() + "," + caducidad.ToString("yyyy-MM-dd") + ")";

```

```

resp = conn.Executa(str);
if (!resp) { 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_SAL + "," + ID_ART + ",0," + sal + "," + precio + "," +
nuevo_costo + "," + caducidad.ToString("yyyy-MM-dd") + ")";
//MessageBox.Show(str);
resp = conn.Executa(str);
if (!resp) { conn.fallo(); return false; }
//CALCULAR PRECIO PROMEDIO
str = "update catarticulos set existencia=existencia-" + sal + " where id=" + ID_ART;
resp = conn.Executa(str);
if (!resp) { conn.fallo(); return false; }

```

```

str = "update det_entradas set existencia=existencia-"
+ sal + " where id=" + Grid_Articulos.Rows[fila2].Cells["id"].Value.ToString();
resp = conn.Executa(str);
if (!resp) { conn.fallo(); return false; }

```

```
        } //segundo for  
  
    } //primer for  
    //Inventario.ActiveForm=true;  
    return true;  
}  
  
}  
  
}
```