**METODO 1 LINQSP (necesario crear en SQL SERVER la BASE y los procedimientos almacenados)**

*LIBRERIAS:*

*GENERAL:*

DataClasses1DataContext ClienteLinq = new DataClasses1DataContext();

*ALTA:*

ClienteLinq.insertarcliente(TxtNombre.Text, TxtApellido1.Text, TxtApellido2.Text, TxtTelefono.Text);

*BAJA:*

ClienteLinq.EliminarCliente(TxtNombre.Text);

*MODI:*

ClienteLinq.Modificarcliente(TxtNombre.Text, TxtApellido1.Text, TxtApellido2.Text, TxtTelefono.Text);

*BUSQUEDA:*

GridDato.DataSource= ClienteLinq.Buscarcliente(TxtBuscar.Text);

*ACTUALIZAR GRID:*

GridDato.DataSource = ClienteLinq.ListarClientes();

\*\* Actualizar siempre el grid en cada operación es necesario

**METODO 2 INYECCION SQL**

*LIBRERIAS:*

using System.Data.SqlClient;

GENERAL:

private SqlConnection conexion = new SqlConnection("server=ASATA405\\SQLEXPRESSMARCOS;Initial Catalog=EJERCICIOSCLASE;Persist Security info=True;User Id=sa;PassWord=1234");

ALTA:

conexion.Open();

string iden = txtIdInsertar.Text;

string nombre = txtNombreInsertar.Text;

string edad = txtEdadInsertar.Text;

string cadena = "insert into Personas(id, nombre, edad)" + "values (" + iden + ",'" + nombre + "'," + edad + ")";

SqlCommand comando = new SqlCommand(cadena, conexion);

try

{

comando.ExecuteNonQuery();

MessageBox.Show("Los datos se guardaron correctamente");

txtEdadInsertar.Text = "";

txtNombreInsertar.Text = "";

txtIdInsertar.Text = "";

}

catch

{

MessageBox.Show("Error grabar datos");

}

conexion.Close();

BAJA:

conexion.Open();

string iden = txtIdEliminar.Text;

string cadena = "delete from Personas where id=" + iden;

SqlCommand comando = new SqlCommand(cadena, conexion);

int cant;

cant = comando.ExecuteNonQuery();

if (cant == 1) {

LabNombre.Content = "";

LabEdad.Content = "";

MessageBox.Show("Articulo Borrado");

} else

{

MessageBox.Show("No existe un articulo con ese codigo");

}

conexion.Close();

txtIdEliminar.Text = "";

MODI:

conexion.Open();

string id = txtIdModificar.Text;

string nombre = txtNombreModificar.Text;

string edad = txtEdadModificar.Text;

string cadena = "update Personas set nombre='" + nombre + "',edad=" + edad + " where id=" + id;

//MessageBox.Show(cadena);

SqlCommand comando = new SqlCommand(cadena, conexion);

int cant;

cant = comando.ExecuteNonQuery();

//MessageBox.Show(cant.ToString());

if (cant == 1)

{

txtEdadModificar.Text = "";

txtNombreModificar.Text = "";

txtIdModificar.Text = "";

MessageBox.Show("Articulo Modificado");

}

else

{

MessageBox.Show("No existe un articulo con ese codigo");

}

conexion.Close();

BtnModificar.IsEnabled = false;

BUSQUEDA:

conexion.Open();

string id = txtIdModificar.Text;

string cadena = "select id, nombre, edad from Personas where id=" + id;

SqlCommand comando = new SqlCommand(cadena, conexion);

try

{

SqlDataReader registro = comando.ExecuteReader();

if (registro.Read())

{

txtNombreModificar.Text= registro["nombre"].ToString();

txtEdadModificar.Text= registro["edad"].ToString();

BtnModificar.IsEnabled = true;

}

else

{

MessageBox.Show("No existe articulo con el codigo ingresado");

}

} // del try

catch

{

MessageBox.Show("Error producido al acceder base de datos");

}

conexion.Close();

ACTUALIZAR GRID:????

**METODO 3 LINQ**

LIBRERIAS:

using System.Linq;

GENERAL:

DataClasses1DataContext nortwind2 = new DataClasses1DataContext();

ALTA:

Products MyProduct = new Products();

MyProduct.ProductName = TxtProd.Text;

MyProduct.UnitPrice = int.Parse(TxtPrec.Text);

MyProduct.UnitsInStock = short.Parse(TxtStock.Text);

MyProduct.CategoryID = int.Parse(TxtCat.Text);

nortwind2.Products.InsertOnSubmit(MyProduct);

nortwind2.SubmitChanges();

cargarGrid();

BAJA:

Products MyProduct = nortwind2.Products.Single(x =>

x.ProductName == CbNombreProd.SelectedItem.ToString());

nortwind2.Products.DeleteOnSubmit(MyProduct);

nortwind2.SubmitChanges();

cargarCombo();

cargarGrid();

}

MODI:

Products MyProducto = nortwind2.Products.Single(p =>

p.ProductName == CbNombreProd.SelectedItem.ToString());

MyProducto.ProductName = CbNombreProd.SelectedItem.ToString();

MyProducto.UnitPrice = int.Parse(TxtPrec.Text);

MyProducto.UnitsInStock = short.Parse(TxtStock.Text);

MyProducto.CategoryID = int.Parse(TxtCat.Text);

nortwind2.SubmitChanges();

cargarGrid();

CARGARCOMBO:

var cargaCombo = from p in nortwind2.Products

select p.ProductName;

CbNombreProd.DataSource = cargaCombo;

BUSQUEDA: (ES OTRO)

// busqueda por nombre y busco y muestro en grid

var x = from p in bsa.Empleados where txtbuscar.Text == p.Nombre select p;

dataGridView1.DataSource = x;

int cuenta = x.Count();

labbus.Text = "Se han encontrado "+cuenta.ToString()+" registros";

ACTUALIZAR GRID:

var cargarGrid = from p in nortwind2.Products select p;

dataGridView1.DataSource = cargarGrid;

EJEMPL GRABAR DATEPICKER Y RADIOS

RADIO ES CON IF Y DATEPICKR VALUE

**METODO 4 ENTITY**

**LIBRERIAS**

using System.Linq;

**GENERAL**

BDEntity2Entities ClientesEntity = new BDEntity2Entities();

**ALTA**

Clientes Misclientes = new Clientes()

{

DNI = txtDNI.Text,

Nombre = txtNombres.Text,

Apellidos = txtApellidos.Text,

EstadoCivil = cbEstadocivil.Text,

Email = txtEmail.Text,

Telefono = txtTelefono.Text,

FechaNacimiento = dateTimePicker1.Value

};

// llamamos contexto entidadesy añadimos al objeto clientes

ClientesEntity.Clientes.Add(Misclientes);

// guardar nuevo registro base de dato

ClientesEntity.SaveChanges();

// refrescar dataGrid

Cargargrid();

**BAJA**

// poner try para evitar no exista dni

// variable guardar DNI q es clave

string TextoDNI = txtDNI.Text;

// comprobar existe registro

bool existe = ClientesEntity.Clientes.Any(x => x.DNI == TextoDNI);

LabAviso.Text = "";

if (existe)

{

try {

// metodo lamba consigue un clinete por su DNI o se puede usar el de abajo Entity

var deleteCliente = ClientesEntity.Clientes.Where(c => c.DNI == TextoDNI).Single();

// metodo con linq para conseguir un cliente por DNI

//Clientes MyCliente =(From c in ClientesEntity.Clientes where c.DNI == TextoDNI select c).Single();

// contains en linkq seria como like en sql

// borramos el cliente indicado

ClientesEntity.Clientes.Remove(deleteCliente);

ClientesEntity.SaveChanges();

Cargargrid();

}

catch {

LabAviso.Text = "ERROR DE BASE DATOS";

}

}else

{

// No existe DNI

LabAviso.Text = "ERROR NO EXISTE DNI";

**MODIFICAR**

string TextoDNI = txtDNI.Text;

bool existe = ClientesEntity.Clientes.Any(x => x.DNI == TextoDNI);

LabAviso.Text = "";

if (existe)

{

try

{

// Mediante LINQ OBTENEMOS ESE CLIENTE

Clientes MyCliente = (from c in ClientesEntity.Clientes where c.DNI == TextoDNI select c).Single();

// atributos de mycliente objeto

MyCliente.Nombre = txtNombres.Text;

MyCliente.Apellidos = txtApellidos.Text;

MyCliente.Email = txtEmail.Text;

MyCliente.EstadoCivil = cbEstadocivil.Text;

MyCliente.FechaNacimiento = dateTimePicker1.Value;

MyCliente.Telefono = txtTelefono.Text;

// añadirlo base de datos

ClientesEntity.SaveChanges();

Cargargrid();

}

catch

{

LabAviso.Text = "ERROR DE BASE DATOS";

}

}

else

{

LabAviso.Text = "ERROR NO EXISTE DNI";

}

**ELIMINAR**

// poner try para evitar no exista dni

// variable guardar DNI q es clave

string TextoDNI = txtDNI.Text;

// comprobar existe registro

bool existe = ClientesEntity.Clientes.Any(x => x.DNI == TextoDNI);

LabAviso.Text = "";

if (existe)

{

try {

// metodo lamba consigue un clinete por su DNI o se puede usar el de abajo Entity

var deleteCliente = ClientesEntity.Clientes.Where(c => c.DNI == TextoDNI).Single();

// metodo con linq para conseguir un cliente por DNI

//Clientes MyCliente =(From c in ClientesEntity.Clientes where c.DNI == TextoDNI select c).Single();

// contains en linkq seria como like en sql

// borramos el cliente indicado

ClientesEntity.Clientes.Remove(deleteCliente);

ClientesEntity.SaveChanges();

Cargargrid();

}

catch {

LabAviso.Text = "ERROR DE BASE DATOS";

}

}else

{

// No existe DNI

LabAviso.Text = "ERROR NO EXISTE DNI";

}

}

BUSCAR:

private void btBuscar\_Click(object sender, EventArgs e)

{

// buscar por el dni y rellenar a los campos de texto

string TextoDNI = txtDNI.Text;

// con lambsa tengo qu estudiar hacerlo tipo funcionamiento like de sql

var BuscaCliente = ClientesEntity.Clientes.Where(x => x.DNI == TextoDNI).Single();

txtNombres.Text = BuscaCliente.Nombre;

txtApellidos.Text = BuscaCliente.Apellidos;

txtTelefono.Text = BuscaCliente.Telefono;

txtEmail.Text = BuscaCliente.Email;

cbEstadocivil.Text = BuscaCliente.EstadoCivil;

dateTimePicker1.Text = BuscaCliente.FechaNacimiento.ToString();

}

CARGAR GRID:

dataGridView1.DataSource = ClientesEntity.Clientes.ToList();

*BOTONES PRIMERO*

// posicionarse primero

// con lambsa tengo qu estudiar hacerlo tipo funcionamiento like de sql

var BuscaCliente = ClientesEntity.Clientes.First();

txtDNI.Text = BuscaCliente.DNI;

txtNombres.Text = BuscaCliente.Nombre;

txtApellidos.Text = BuscaCliente.Apellidos;

txtTelefono.Text = BuscaCliente.Telefono;

txtEmail.Text = BuscaCliente.Email;

cbEstadocivil.Text = BuscaCliente.EstadoCivil;

dateTimePicker1.Text = BuscaCliente.FechaNacimiento.ToString();

dataGridView1.CurrentCell = dataGridView1.Rows[0].Cells[0];

*BOTONES SIGUIENTE*

// declaramos variable indicar la fila anteior

int siguiente = dataGridView1.CurrentRow.Index + 1;

int k = dataGridView1.RowCount;

if (siguiente < k)

{

dataGridView1.CurrentCell = dataGridView1.Rows[siguiente].Cells[dataGridView1.CurrentCell.ColumnIndex];

// cargamos el text box

txtDNI.Text = dataGridView1.Rows[siguiente].Cells[0].Value.ToString();

txtNombres.Text = dataGridView1.Rows[siguiente].Cells[1].Value.ToString();

txtApellidos.Text = dataGridView1.Rows[siguiente].Cells[2].Value.ToString();

cbEstadocivil.Text = dataGridView1.Rows[siguiente].Cells[5].Value.ToString();

txtTelefono.Text = dataGridView1.Rows[siguiente].Cells[3].Value.ToString();

txtEmail.Text = dataGridView1.Rows[siguiente].Cells[4].Value.ToString();

dateTimePicker1.Text = dataGridView1.Rows[siguiente].Cells[6].Value.ToString();

}

*BOTONES ANTERIOR*

// declaramos variable indicar la fila anteior

int anterior = dataGridView1.CurrentRow.Index - 1;

if (anterior >=0)

{

// desplazamos a la anterior

dataGridView1.CurrentCell = dataGridView1.Rows[anterior].Cells[dataGridView1.CurrentCell.ColumnIndex];

// cargamos el text box

txtDNI.Text = dataGridView1.Rows[anterior].Cells[0].Value.ToString();

txtNombres.Text = dataGridView1.Rows[anterior].Cells[1].Value.ToString();

txtApellidos.Text = dataGridView1.Rows[anterior].Cells[2].Value.ToString();

cbEstadocivil.Text = dataGridView1.Rows[anterior].Cells[5].Value.ToString();

txtTelefono.Text = dataGridView1.Rows[anterior].Cells[3].Value.ToString();

txtEmail.Text = dataGridView1.Rows[anterior].Cells[4].Value.ToString();

dateTimePicker1.Text = dataGridView1.Rows[anterior].Cells[6].Value.ToString();

}

BOTONES ULTIMO

// posicionarse ultimo

// con lambsa tengo qu estudiar hacerlo tipo funcionamiento like de sql

var BuscaCliente = ClientesEntity.Clientes.ToList().Last();

txtDNI.Text = BuscaCliente.DNI;

txtNombres.Text = BuscaCliente.Nombre;

txtApellidos.Text = BuscaCliente.Apellidos;

txtTelefono.Text = BuscaCliente.Telefono;

txtEmail.Text = BuscaCliente.Email;

cbEstadocivil.Text = BuscaCliente.EstadoCivil;

dateTimePicker1.Text = BuscaCliente.FechaNacimiento.ToString();

int xx = dataGridView1.RowCount;

dataGridView1.CurrentCell = dataGridView1.Rows[xx-1].Cells[0];

}

CARGAR TXTBOX CON SELECCIÓN GRID (SELECTION CHANGED)

// cargar datos alcambiar

int siguiente = dataGridView1.CurrentRow.Index;

txtDNI.Text = dataGridView1.Rows[siguiente].Cells[0].Value.ToString();

txtNombres.Text = dataGridView1.Rows[siguiente].Cells[1].Value.ToString();

txtApellidos.Text = dataGridView1.Rows[siguiente].Cells[2].Value.ToString();

cbEstadocivil.Text = dataGridView1.Rows[siguiente].Cells[5].Value.ToString();

txtTelefono.Text = dataGridView1.Rows[siguiente].Cells[3].Value.ToString();

txtEmail.Text = dataGridView1.Rows[siguiente].Cells[4].Value.ToString();

dateTimePicker1.Text = dataGridView1.Rows[siguiente].Cells[6].Value.ToString();

}