

## Base de Datos SQL Server

---

```
create database CRUD;
use CRUD;

create table EMPLEADOS
(
NOMBRE varchar(20),
USUARIO varchar(15) not null,
CONTRASEÑA varchar(6) not null,
ROL varchar (15) not null,
primary key (USUARIO)
);

insert into EMPLEADOS (USUARIO, CONTRASEÑA, ROL) values
('edwin', '4567', 'Administrador');

create procedure sp_EMPLEADOS
@NOMBRE varchar (20),
@USUARIO varchar (15),
@CONTRASEÑA varchar (6),
@ROL varchar (15)
as
insert into EMPLEADOS values (@NOMBRE, @USUARIO, @CONTRASEÑA, @ROL)
```

## CLASE CONEXIÓN

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Windows.Forms;

namespace Matriz_CRUD
{
    class Conexion
    {
        private string con = @"server=.\SQLEXPRESS;Initial
Catalog=CRUD;Integrated Security=true";
        //encapsular la variable privada (con)//
        public string Cnn
        {
            get { return con; }
            set { value = con; }
        }

        //crear la variable para la conexion con sql server
        public SqlConnection conec;

        //crear el metodo para la conexion
        public SqlConnection conecte()
        {
            try
            {
                //instancia de la clase sqlconnection
                conec = new SqlConnection(Cnn);
                conec.Open();
                return conec;
            }
            catch (SqlException ex)
            {
                MessageBox.Show("Error al conectar " + ex);
                conec.Close();
                return conec;
            }
        }
    }
}
```

CLASE LOGIN

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Data;
using System.Windows.Forms;

namespace Matriz_CRUD
{
    class login : Conexion
    {
        public SqlDataReader lector;
        public SqlCommand comando;

        public String Muestra_Usuario(string usuario, string contraseña)
        {
            try
            {
                comando = new SqlCommand("select ROL from EMPLEADOS where
USUARIO='" + usuario + "' and CONTRASEÑA='" + contraseña + "'",
conecte());
                lector = comando.ExecuteReader();
                if (lector.Read())
                {
                    return lector[0].ToString();
                }
                else
                {
                    MessageBox.Show("Usuario o contraseña Incorrectos");
                    return "";
                }
            }
            catch (SqlException ex)
            {
                return "";
            }
        }
    }
}
```

CLASE EMPLEADO

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using System.Data.SqlClient;
using System.Windows.Forms;

namespace Matriz_CRUD
{
    class Empleado : Conexion
    {
        public SqlCommand comandoEmpleado;
        public DataTable TEmpleado;
        public SqlDataAdapter Clt;

        public void Insertar_Empleado(String Nombre, String Usuario,
String Contraseña, String Rol)
        {
            try
            {
                comandoEmpleado = new SqlCommand("sp_EMPLEADOS",
conecte());
                comandoEmpleado.CommandType =
CommandType.StoredProcedure;
                comandoEmpleado.Parameters.Add("@NOMBRE",
SqlDbType.VarChar, 20).Value = Nombre;
                comandoEmpleado.Parameters.Add("@USUARIO",
SqlDbType.VarChar, 15).Value = Usuario;
                comandoEmpleado.Parameters.Add("@CONTRASEÑA",
SqlDbType.VarChar, 6).Value = Contraseña;
                comandoEmpleado.Parameters.Add("@ROL", SqlDbType.VarChar,
15).Value = Rol;
                comandoEmpleado.ExecuteNonQuery();
            }
            catch (SqlException ex)
            {
                MessageBox.Show("Error al Insertar Datos" + ex);
            }
        }

        public DataTable Consultar_Todos_Empleados()
        {
            try
            {
                Clt = new SqlDataAdapter("select * from EMPLEADOS",
conecte());
                TEmpleado = new DataTable();
                Clt.Fill(TEmpleado);
                return TEmpleado;
            }
            catch (SqlException ex)
            {
                MessageBox.Show("Error al Consultar Todos Empleados" + ex);
            }
        }
    }
}
```

```

        {
            MessageBox.Show("Error al cargar datos" + ex);
            return TEmpleado;
        }
    }

    public DataTable Consultar_Empleado(string Usuario)
    {
        try
        {
            Clt = new SqlDataAdapter("select * from EMPLEADOS where
USUARIO='"+ Usuario + "'", conecte());
            TEmpleado = new DataTable();
            Clt.Fill(TEmpleado);
            return TEmpleado;
        }
        catch (SqlException ex)
        {
            MessageBox.Show("Error al cargar datos" + ex);
            return TEmpleado;
        }
    }

    public void Actualizar_Empleado(String Nombre, String Usuario,
String Contraseña, String Rol)
    {
        try
        {
            comandoEmpleado = new SqlCommand("update EMPLEADOS set
NOMBRE=@NOMBRE, CONTRASEÑA=@CONTRASEÑA, ROL=@ROL where USUARIO=@USUARIO",
conecte());
            comandoEmpleado.Parameters.Add("NOMBRE",
SqlDbType.VarChar, 20).Value = Nombre;
            comandoEmpleado.Parameters.Add("@USUARIO",
SqlDbType.VarChar, 15).Value = Usuario;
            comandoEmpleado.Parameters.Add("CONTRASEÑA",
SqlDbType.VarChar, 6).Value = Contraseña;
            comandoEmpleado.Parameters.Add("ROL", SqlDbType.VarChar,
15).Value = Rol;
            comandoEmpleado.ExecuteNonQuery();
            MessageBox.Show("Datos actualizados correctamente");
        }
        catch (SqlException ex)
        {
            MessageBox.Show("Error al actualizar datos " + ex);
        }
    }

    public void Eliminar_Empleado(string Usuario)
    {
        try
        {
            comandoEmpleado = new SqlCommand("delete from EMPLEADOS
where USUARIO ='"+ Usuario + "'", conecte());
            comandoEmpleado.ExecuteNonQuery();

```

```
        MessageBox.Show("Datos eliminados");
    }
    catch (SqlException ex)
    {
        MessageBox.Show("Error al eliminar" + ex);
    }
}
}
```

## FORM 1

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

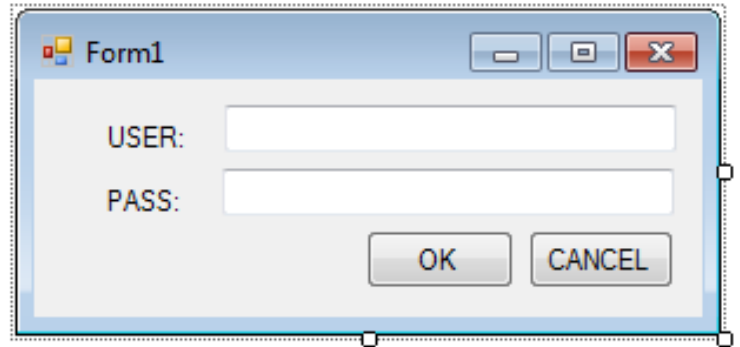
```
namespace Matriz_CRUD
{
    public partial class FrmInicioSesion : Form
    {
        public FrmInicioSesion()
        {
            InitializeComponent();
        }

        login sesion = new login();

        private void FrmInicioSesion_Load(object sender, EventArgs e)
        {
        }

        private void button1_Click_1(object sender, EventArgs e)
        {
            String retorno = sesion.Muestra_Usuario(textBox1.Text,
            textBox2.Text);
            switch (retorno)
            {
                case "Administrador":
                    FrmEmpleados Empleado = new FrmEmpleados();
                    Empleado.Show();

                    this.Hide();
                    break;
            }
        }
    }
}
```



## FORM 2

```
using System;
using
System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace Matriz_CRUD
{
    public partial class
    FrmEmpleados : Form
    {
        public FrmEmpleados()
        {
            InitializeComponent();
        }
```

```
Empleado E = new Empleado();
private void button1_Click(object sender, EventArgs e)
{
    E.Insertar_Empleado(TxtNombre.Text, TxtUsuario.Text,
    TxtContraseña.Text, CbxRol.Text);
    DGVEmpleado.DataSource = E.Consultar_Todos_Empleados();
}

private void button2_Click(object sender, EventArgs e)
{
    E.Actualizar_Empleado(TxtNombre.Text, TxtUsuario.Text, TxtContraseña.Text,
    CbxRol.Text);
    DGVEmpleado.DataSource = E.Consultar_Todos_Empleados();
}

private void button3_Click(object sender, EventArgs e)
{
    DGVEmpleado.DataSource = E.Consultar_Todos_Empleados();
}

private void button4_Click(object sender, EventArgs e)
{
    E.Eliminar_Empleado(TxtUsuario.Text);
    DGVEmpleado.DataSource = E.Consultar_Todos_Empleados();
}

private void DGVEmpleado_CellClick_1(object sender,
DataGridViewCellEventArgs e)
{
}
```

The screenshot shows a Windows Forms application window titled "Form2". The window contains a form with the following elements:

- Four text input fields labeled "NOMBRE", "USUARIO", "CONTRASEÑA", and "ROL".
- A dropdown menu for "ROL" currently displaying "Administrador".
- Buttons for "Buscar", "Agregar", "Actualizar", "Consultar", "Eliminar", and "Limpiar".
- A large grey rectangular area at the bottom, which is a DataGridView, currently empty.



```

        DataGridView u = new DataGridView();

        u.Pasarcampo(DGVEmpleado, TxtNombre, "NOMBRE");
        u.Pasarcampo(DGVEmpleado, TxtUsuario, "USUARIO");
        u.Pasarcampo(DGVEmpleado, TxtContraseña, "CONTRASEÑA");
        u.PasarcampoCB(DGVEmpleado, CbxRol, "ROL");
    }

    private void DGVEmpleado_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {

    }

    private void FrmEmpleados_Load(object sender, EventArgs e)
    {

    }

    private void button5_Click(object sender, EventArgs e)
    {
        DGVEmpleado.DataSource =
E.Consultar_Empleado(TxtUsuario.Text);
    }

    private void button6_Click(object sender, EventArgs e)
    {
        TxtNombre.Clear();
        TxtUsuario.Clear();
        TxtContraseña.Clear();
        CbxRol.Text = "";
    }

    }
}

```

```

class Datagrid

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Data;
using System.Windows.Forms;
namespace Matriz_CRUD
{
    class DataGridView
    {
        public void Pasarcampo(DataGridView midgv, TextBox txb, string
columna)
        {

```

```
        // especifico que campo de la fila que este seleccionada
        vamos a pasar al textbox

        txb.Text =
midgv.Rows[midgv.CurrentRow.Index].Cells[columna].Value.ToString();

    }
    public void PasarcampoCB(DataGridView midgv, ComboBox Cbx, string
columna)
    {
        // especifico que campo de la fila que este seleccionada
        vamos a pasar al textbox

        Cbx.Text =
midgv.Rows[midgv.CurrentRow.Index].Cells[columna].Value.ToString();

    }
}
```