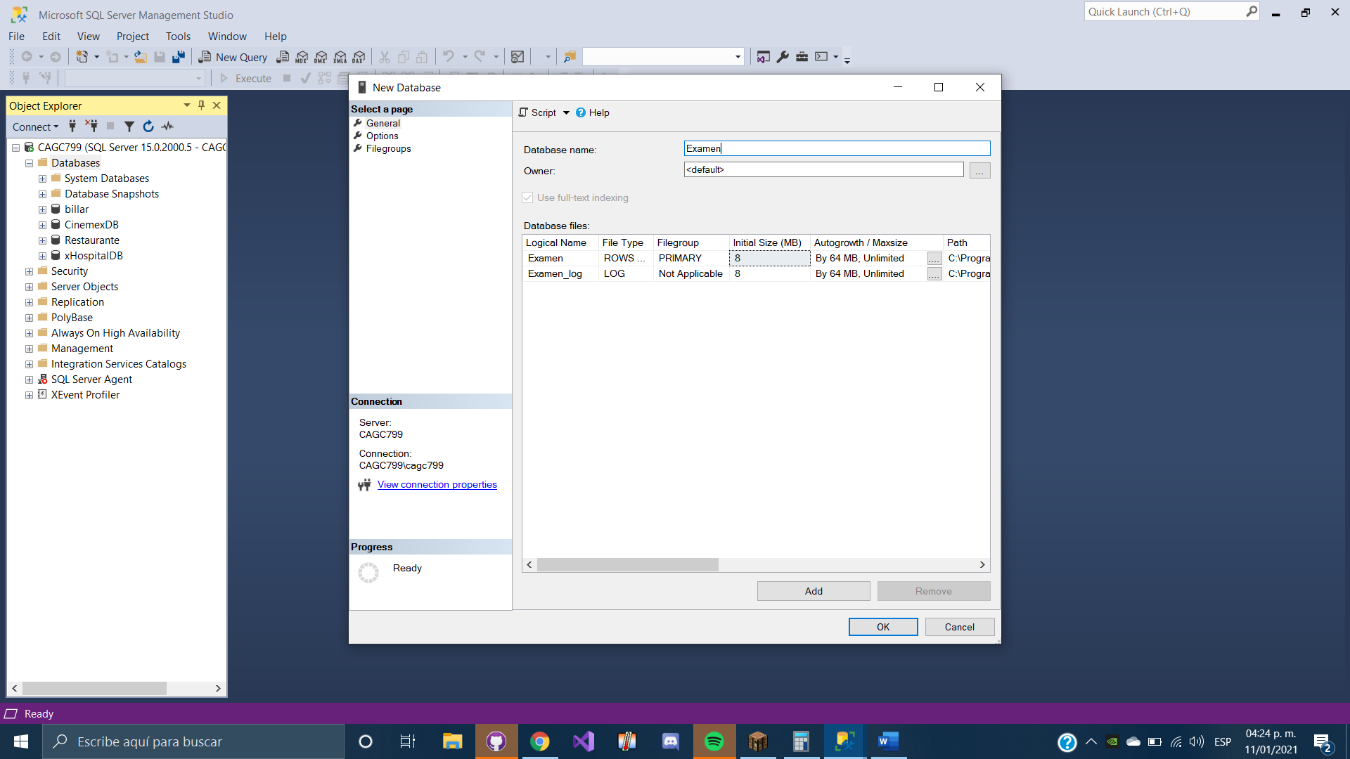
1.-



2,3 y 4

TABLA DE BASE DE DATOS

use examen

create table empleados

(IdDepto int,

apPaterno varchar(50),

apMaterno varchar (50),

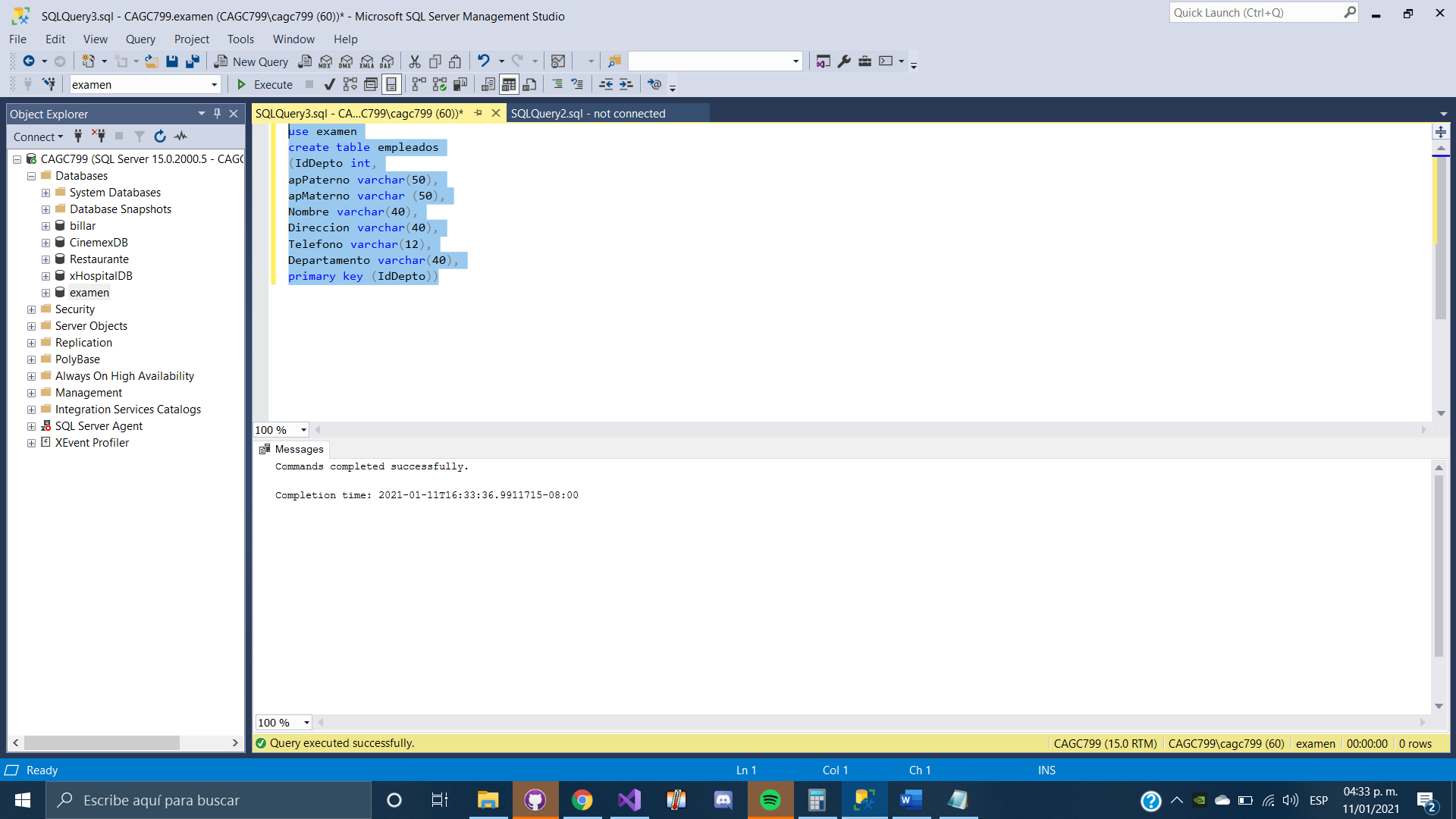
Nombre varchar(40),

Direccion varchar(40),

Telefono varchar(12),

Departamento varchar(40),

primary key (IdDepto))🡨--- esta es al llave primaria



6.-

MODIFICAR LA TABLA

use examen

alter table empleados

(IdDepto int,

apPaterno cadena,

apMaterno char (50),

Nombre varchar(40),

Direccion varchar(40),

Telefono varchar(12),

Departamento varchar(40),

CorreoE varchar(40),

primary key (IdDepto))

7.-

Como sugerencia, desde tu aplicacion (front-end) para obligar a que carguen siempre un correo electronico valido puedes usar una mascara de entrada o una expresion regular como esta

\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*

CONEXION

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using MaterialSkin;

using MaterialSkin.Animations;

using MaterialSkin.Controls;

using System.Data.SqlClient;

using System.Drawing.Printing;

namespace Billar

{

public partial class Cliente : MaterialForm

{

public Cliente()

{

InitializeComponent();

MaterialSkinManager m = MaterialSkinManager.Instance;

m.AddFormToManage(this);

m.Theme = MaterialSkinManager.Themes.LIGHT;

m.ColorScheme = new ColorScheme(Primary.BlueGrey900, Primary.BlueGrey900, Primary.Blue900, Accent.Cyan100, TextShade.WHITE);

}

SqlConnection conexion = new SqlConnection(@"server=CAGC799; Initial Catalog=examen; integrated security=true");

private void Cliente\_Load(object sender, EventArgs e)

{

}

METODO DE LAS ALTAS

private void MaterialFlatButton6\_Click(object sender, EventArgs e)

{

SqlCommand altas = new SqlCommand

("insert into empleados (IdDepto,apPaterno,apMaterno,Nombre,Direccion,Telefono,Departamento) values(@IdDepto,@apPaterno,@apMaterno,@Nombre,@Direccion,@Telefono,@Departamento)", conexion);

// se pasan los valores de los text box a las variables temporales

altas.Parameters.AddWithValue("IdDepto", textBox1.Text);

altas.Parameters.AddWithValue("apPaterno", textBox2.Text);

altas.Parameters.AddWithValue("apMaterno", textBox3.Text);

altas.Parameters.AddWithValue("Nombre", textBox4.Text);

altas.Parameters.AddWithValue("Direccion", textBox5.Text);

altas.Parameters.AddWithValue("Telefono", textBox6.Text);

altas.Parameters.AddWithValue("Departamento", textBox7.Text);

conexion.Open();// se abre la conexion

altas.ExecuteNonQuery();

conexion.Close();// se cierra la conexion

MessageBox.Show("Ok");

// limpiar los textbox

textBox1.Clear();

textBox2.Clear();

textBox3.Clear();

textBox4.Clear();

textBox4.Clear();

textBox5.Clear();

textBox6.Clear();

textBox7.Clear();

}

METODO DE BORRAR

[Obsolete]

private void MaterialFlatButton9\_Click(object sender, EventArgs e)

{

string baja = "DELETE FROM empleadosWHERE IdDepto=@IdDepto";

conexion.Open();

SqlCommand comando = new SqlCommand(baja, conexion);

comando.Parameters.Add("IdDepto", textBox1.Text);

comando.ExecuteNonQuery();

comando.Dispose();

comando = null;

textBox1.Clear();

conexion.Close();

MessageBox.Show("Eliminado");

}

private void GroupBox1\_Enter(object sender, EventArgs e)

{

}

private void MaterialFlatButton14\_Click(object sender, EventArgs e)

{

string baja = "truncate table empleados";

conexion.Open();

SqlCommand cmdIns = new SqlCommand(baja, conexion);

cmdIns.ExecuteNonQuery();

cmdIns.Dispose();

cmdIns = null;

textBox2.Clear();

conexion.Close();

MessageBox.Show("Se ha eliminado toda la tabla");

}

METODO PARA CONSULTAR

private void MaterialFlatButton7\_Click(object sender, EventArgs e)

{

conexion.Open();

string cadena = "select\*from empleados";

SqlCommand comando = new SqlCommand(cadena, conexion);

SqlDataAdapter adaptador = new SqlDataAdapter(comando);

DataTable datatable1 = new DataTable();

adaptador.Fill(datatable1);

dataGridView1.DataSource = datatable1;

conexion.Close();

}

IMPRIMIR EN PDF

private void MaterialFlatButton1\_Click(object sender, EventArgs e)

{

PrintDocument doc = new PrintDocument();

doc.DefaultPageSettings.Landscape = true;

doc.PrinterSettings.PrinterName = "Microsoft Print to PDF";

PrintPreviewDialog ppd = new PrintPreviewDialog { Document = doc };

((Form)ppd).WindowState = FormWindowState.Maximized;

doc.PrintPage += delegate (object ev, PrintPageEventArgs ep)

{

const int DGV\_ALTO = 35;

int left = ep.MarginBounds.Left, top = ep.MarginBounds.Top;

foreach (DataGridViewColumn col in dataGridView1.Columns)

{

ep.Graphics.DrawString(col.HeaderText, new Font("Segoe UI", 12, FontStyle.Bold), Brushes.Black, left, top);

left += col.Width;

if (col.Index < dataGridView1.ColumnCount - 1)

ep.Graphics.DrawLine(Pens.Gray, left - 5, top, left - 5, top + 43 + (dataGridView1.RowCount - 1) \* DGV\_ALTO);

}

left = ep.MarginBounds.Left;

ep.Graphics.FillRectangle(Brushes.Black, left, top + 40, ep.MarginBounds.Right - left, 3);

top += 43;

foreach (DataGridViewRow row in dataGridView1.Rows)

{

if (row.Index == dataGridView1.RowCount - 1) break;

left = ep.MarginBounds.Left;

foreach (DataGridViewCell cell in row.Cells)

{

ep.Graphics.DrawString(Convert.ToString(cell.Value), new Font("Segoe UI", 10), Brushes.Black, left, top + 4);

left += cell.OwningColumn.Width;

}

top += DGV\_ALTO;

ep.Graphics.DrawLine(Pens.Gray, ep.MarginBounds.Left, top, ep.MarginBounds.Right, top);

}

};

ppd.ShowDialog();

}

}

}

VISTA

