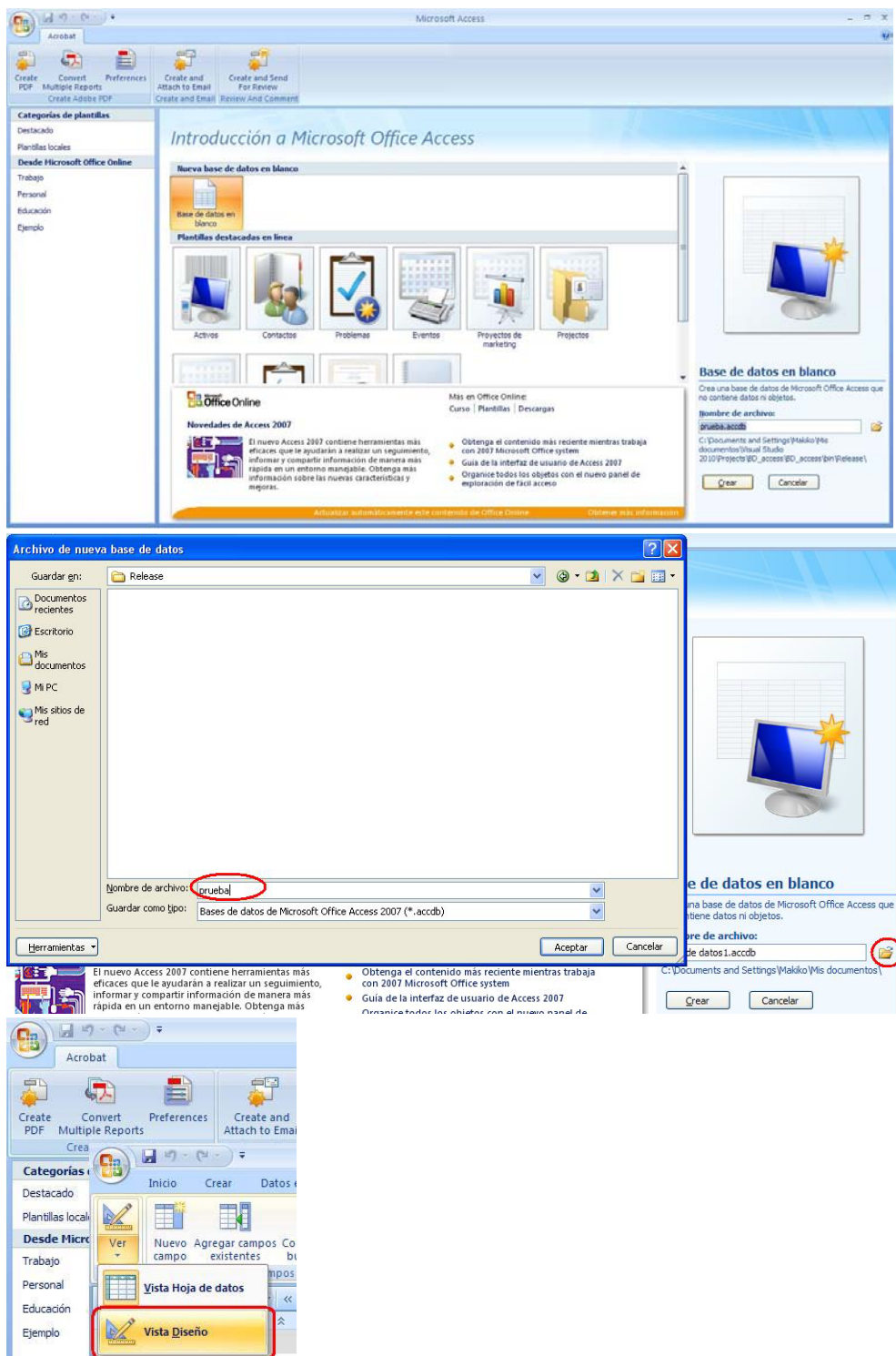


Bases de Datos Access y C#. Insert, select, update, delete

1. Crea una base de Datos en Access



Guardar como

Nombre de la tabla:

alumnos

Aceptar Cancelar

Inicio Crear Datos externos Herramientas de base de datos Acrobat

Ver Clave principal Generador de reglas de validación Eliminar filas Insertar filas Hoja de propiedades Índices

Todas las tablas alumnos

alumnos : Tabla

Campos

Nombre del campo	Tipo de datos
ncuenta	Texto
nombre	Texto
apellidop	Texto
apellidom	Texto
edad	Número
semestre	Número

Propiedades

General	Búsqueda
Tamaño del campo	10
Formato	
Máscara de entrada	
Título	
Valor predeterminado	
Regla de validación	
Texto de validación	
Requerido	Sí
Permitir longitud cero	No
Indexado	Sí (Sin duplicados)
Compresión Unicode	No
Modo IME	Sin Controles
Modo de oraciones IME	Nada

Inicio Crear Datos externos Herramientas de base de datos Acrobat

Tabla Plantillas de tabla Listas de SharePoint Diseño de tabla Gráfico dinámico Formulario en blanco Más formularios

Formulario dividido Formularios

alumnos Tabla1

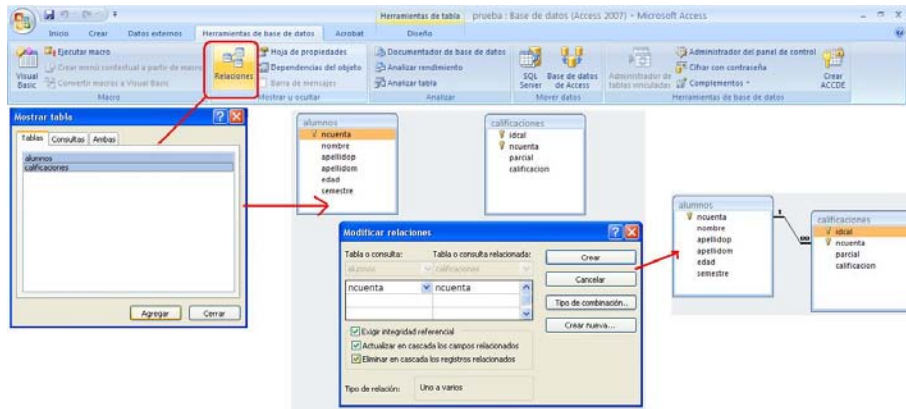
Nombre del campo	Tipo de datos
idcal	Autonumérico
ncuenta	Texto
parcial	Número
calificacion	Número

Guardar como

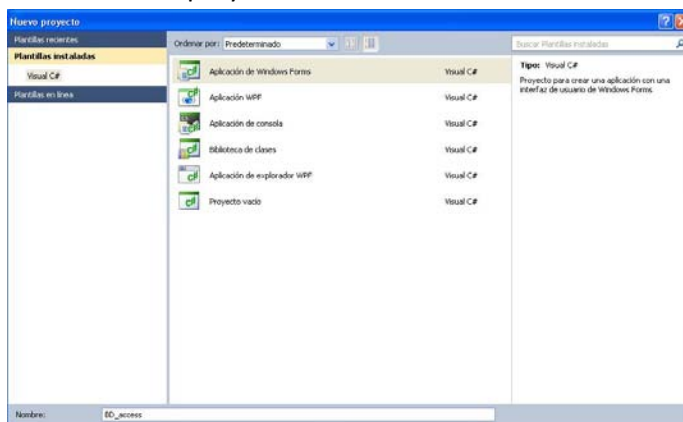
Nombre de la tabla:

calificaciones

Aceptar Cancelar



2. Crea un Nuevo proyecto en C#



3. Agrega una nueva clase llamada **Database** y a continuación escribe el siguiente código:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using System.Data.OleDb;

namespace bd_access2
{
    class Database
    {
        private string StrConexion;

        private OleDbConnection Conexion;
        private OleDbDataAdapter Adapter;
        private DataSet miDataSet = new DataSet();

        public void IniciarConexion(string DataBase)
        {
            //Creo la cadena de conexion para Office 2007
            StrConexion = "Provider=Microsoft.ACE.OLEDB.12.0; Data Source = " + DataBase;

            //Objeto conexion
            Conexion = new OleDbConnection(StrConexion);
        }
    }
}
```

```

public int ejecutar_sql(string sql)
{
    //inserto en la BD
    int i = 0;
    try
    {
        Conexion.Open();
        OleDbCommand cmd = new OleDbCommand(sql, Conexion);
        i = cmd.ExecuteNonQuery();
    }
    catch
    {
        i = -1;
    }
    return i;
}

public DataTable consultar(string sql, string tabla)
{
    Adapter = new OleDbDataAdapter(sql, Conexion);

    //Creo el DataTable
    DataTable dt = new DataTable();

    //Relleno el adaptador con los datos en memoria
    Adapter.Fill(dt);

    return dt;
}
}
}

```

4. Crea una nueva clase llamada **Alumnos** y escribe el siguiente código:

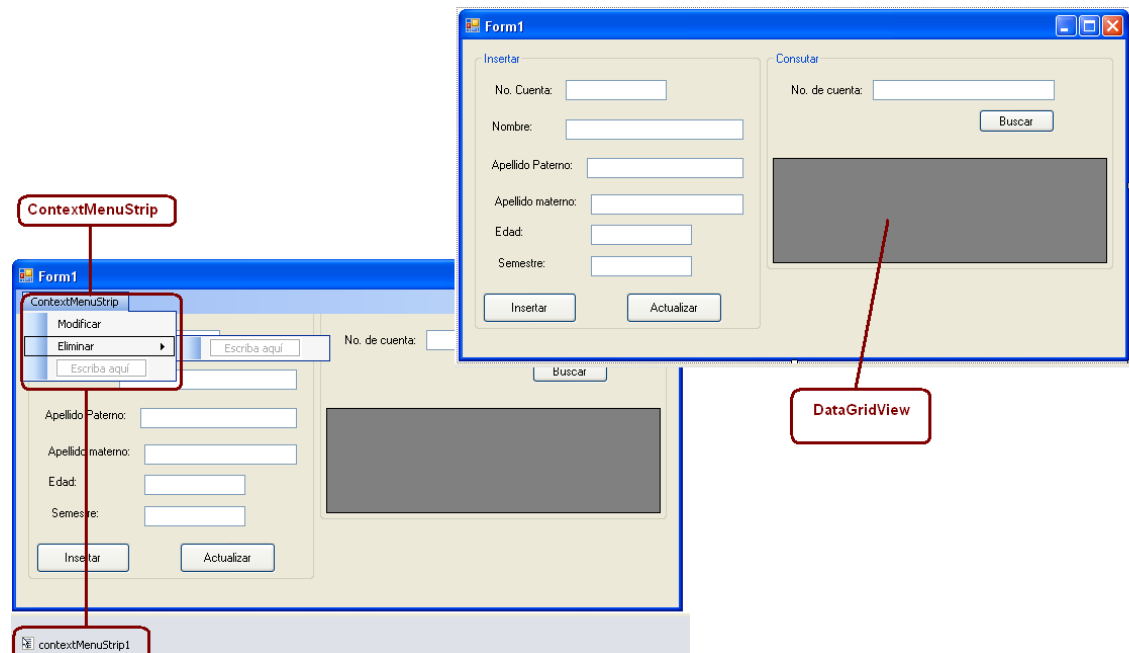
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace bd_access2
{
    class Alumnos
    {
        public string ncuenta;
        public string nombre;
        public string apellidop;
        public string apellidom;
        public int edad;
        public int semestre;

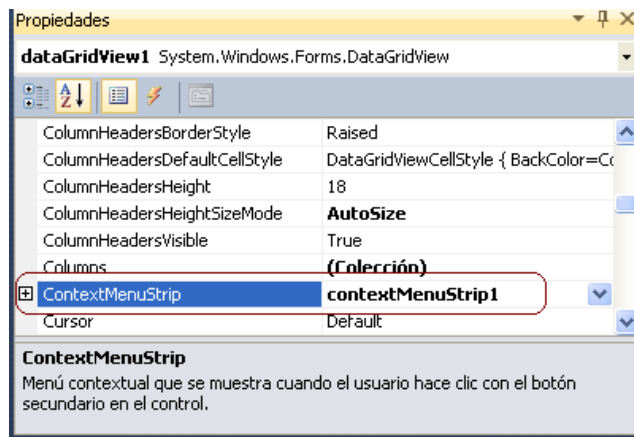
        public Alumnos()
        {
        }

        public Alumnos(string ncuenta, string nombre, string apellidop, string apellidom, int edad, int semestre)
        {
            this.ncuenta = ncuenta;
            this.nombre = nombre;
            this.apellidop = apellidop;
            this.apellidom = apellidom;
            this.edad = edad;
            this.semestre = semestre;
        }
    }
}
```

5. Edita el formulario agregando los siguientes componentes:



6. Edita la propiedad ContextMenuStrip del DataGridView1. (Le permitirá mostrar un menú flotante al dar clic con el botón derecho sobre el DataGridView).



7. Edita el código del Form1.cs

```
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 bd_access2
{
    public partial class Form1 : Form
    {
        Database DB = new Database();
        String sql;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            DB.IniciarConexion("../prueba.accdb");
            //Creo el miembro de datos del DataGridView
            dataGridview1.DataMember = "alumnos";
        }

        private Alumnos asig_alum_text()
        {
            string ncta, nombre, apellidop, apellidom;
            int edad = 0, semestre;
            ncta = txtnCta.Text;
            nombre = txtNombre.Text;
            apellidop = txtApellidop.Text;
            apellidom = txtApellidom.Text;
            try
```

```

    {
        edad = Convert.ToInt32(txtEdad.Text);
    }
    catch
    {
        //Ejemplo de validacion
        MessageBox.Show("Ingrese un valor numerico");
        txtEdad.Focus();
        txtEdad.SelectionStart = 0;
        txtEdad.SelectionLength = txtEdad.TextLength;
    }
    semestre = Convert.ToInt32(txtSemestre.Text);
    Alumnos A = new Alumnos(ncuenta, nombre, apellidop, apellidom, edad, semestre);
    return A;
}

private void btnInsertar_Click(object sender, EventArgs e)
{
    Alumnos a = asig_alum_text();
    sql = "INSERT INTO alumnos(ncuenta,nombre,apellidop,apellidom,edad,semestre)";
    sql += "VALUES('" + a.ncuenta + "','" + a.nombre + "','" + a.apellidop + "','" + a.apellidom + "','" + a.edad + "','" + a.semestre + "')";
    int insert = DB.ejecutar_sql(sql);
    if (insert == 1) //Si se logro la insercion limpio el formulario
    {
        MessageBox.Show("Se insertaron correctamente sus datos");
        foreach (Control txt in this.Controls)
        {
            if (txt.GetType() == typeof(TextBox))
                txt.Text = "";
        }
    }
    else
        MessageBox.Show("Hubo un error al insertar los datos");
}

private void btnBuscar_Click(object sender, EventArgs e)
{
    sql = "SELECT * FROM alumnos WHERE ncuenta='" + txtCriterio.Text + "'";

    //Vuelco los datos al DataGridView
    dataGridView1.DataSource = DB.consultar(sql, "alumnos");
}

private void modificarToolStripMenuItem_Click(object sender, EventArgs e)
{
    //Para modificar los datos del alumnos
    int renglon = dataGridView1.CurrentCell.RowIndex;
    int ncuenta = Convert.ToInt32(dataGridView1[0, renglon].Value.ToString());
    if (ncuenta != -1)
    {
        txtnCta.Text = dataGridView1[0, renglon].Value.ToString();
        txtNombre.Text = dataGridView1[1, renglon].Value.ToString();
        txtApellidop.Text = dataGridView1[2, renglon].Value.ToString();
        txtApellidom.Text = dataGridView1[3, renglon].Value.ToString();
        txtEdad.Text = dataGridView1[4, renglon].Value.ToString();
        txtSemestre.Text = dataGridView1[5, renglon].Value.ToString();
        btnActualizar.Enabled = true;
    }
}

```

```

private void eliminarToolStripMenuItem_Click(object sender, EventArgs e)
{
    //Para eliminar un alumno
    int renglon = dataGridView1.CurrentRow.RowIndex;
    int ncuenta = Convert.ToInt32(dataGridView1[0, renglon].Value.ToString());
    if (ncuenta != -1)
    {
        if (MessageBox.Show("Esta seguro que desea eliminar el alumno " + dataGridView1[1,
            renglon].Value.ToString(), "Eliminar registro", MessageBoxButtons.YesNo) ==
        DialogResult.Yes)
        {
            DB.ejecutar_sql("DELETE FROM alumnos WHERE Cuenta=" + ncuenta + "");
            dataGridView1.Rows.RemoveAt(renglon);
        }
    }
}

private void btnActualizar_Click(object sender, EventArgs e)
{
    Alumnos a = asig_alum_text();
    sql = "UPDATE alumnos SET nombre=" + a.nombre + ",";
    sql += " apellidop=" + a.apellidop + ",apellidom=" + a.apellidom + ",";
    sql += " edad=" + a.edad + ", semestre=" + a.semestre + ",";
    sql += " WHERE ncuenta=" + a.ncuenta + "";

    int update = DB.ejecutar_sql(sql);
    if (update == 1) //Si se logro la insercion limpio el formulario
    {
        MessageBox.Show("Se actualizaron correctamente sus datos");
        foreach (Control txt in this.Controls)
        {
            if (txt.GetType() == typeof(TextBox))
            {
                txt.Text = "";
            }
        }
    }
    else
    {
        MessageBox.Show("Hubo un error al actualizar los datos");
    }
}
}
}

```

NOTAS:

- No olviden generar todo el código de los eventos.
- Verifiquen bien el nombre de sus controles.
- Integren un módulo de validación de datos.
- Agreguen control de errores en las partes que lo requieran.