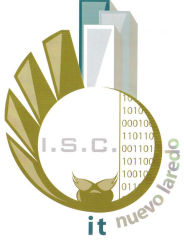
	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	

<b>OBJETIVO:</b> El estudiante elaborará un programa que elimine datos en una base de datos de MS Access a través de un programa en C#
<b>MATERIAL Y EQUIPO NECESARIO:</b> <ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• Microsoft Visual C#</li> </ul>

### **EJERCICIO: BAJAS A LA TABLA EMPLEADOS (MODO MIXTO – CONSULTA DESCONECTADA – BORRADO CONECTADO)**


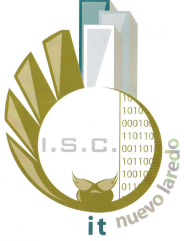
1. Para este ejercicio, se requiere tener el archivo Empleados.mdb en C:\

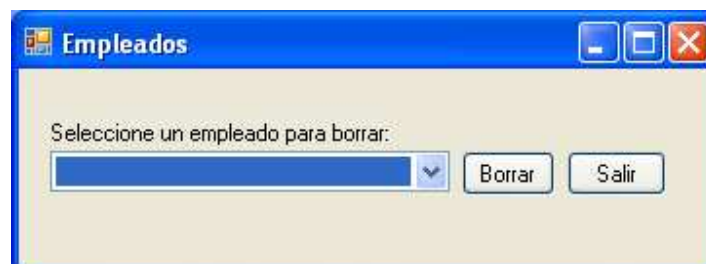
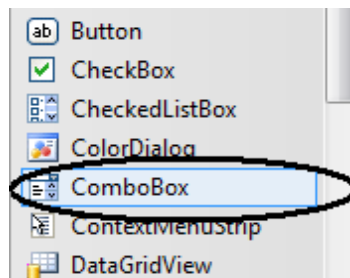
Esta es una base de datos creada en Microsoft Access, previamente en el ejercicio de Altas a la tabla de Empleados. Para fines de esta práctica, puede descargar este archivo desde el sitio web de la materia. Es necesario asegurarse que el archivo se llame `Empleados.mdb` y se encuentre ubicado en C:\ (Si desea ponerlo en otra ubicación, asegúrese de cambiar dentro del programa la cadena de conexión donde se menciona la ruta y el nombre del archivo).

Nota: Si usted utiliza Windows Vista, se recomienda utilizar la base de datos dentro de un Folder (Por ejemplo: `C:\BaseDeDatos\Empleados.mdb` en lugar de en la raíz), esto debido al sistema de permisos que se maneja en ese sistema operativo. En este caso, cambie también la cadena de conexión en este ejercicio para incorporar la ruta correcta del archivo.

2. Crear un nuevo proyecto Windows en C#.NET. En la Forma (Form1), colocar 2 botones de comando, un `comboBox`, y una etiqueta de la siguiente manera:

NOTA: El `comboBox` es un control de Windows que permite seleccionar un elemento de una lista desplegable. Para colocarlo, arrástrelo a la forma desde el `ToolBox`. Para ayudar a ubicarlo se muestra la siguiente figura:

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	



3. Cambie las siguientes propiedades:

En la Forma (Form1)

Name → Form1

Text → Bajas


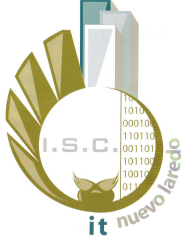
En el comboBox:

Name → cmbEmpleados

4. Dar doble click en el boton "SALIR" y teclear lo siguiente:

```
this.Close();
```

5. Buscar arriba, al principio del código, la sección donde se colocan las sentencias `using`. Solo busque la siguiente sección, no cambie ni borre nada:

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	

```

Form1.cs  Form1.cs [Design]  Start Page
WindowsApplication1.Form1
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

```

Inmediatamente después, agregue esta línea: (NOTA: Esta línea es la referencia a las clases necesarias para acceso a datos mediante ADO.NET en este programa.).

`using System.Data.OleDb;`

6. Ubicar la siguiente parte del código de la forma:

```

Object Browser  Start Page  Form1.cs*  Form1.cs [Design]*
WindowsFormsApplication1.Form1
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.OleDb;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {


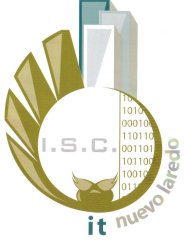
```

7. Inmediatamente abajo, teclear el siguiente método:

```

private void PonerDatosEnElCombo()
{
    string cc = "provider=Microsoft.Jet.OleDb.4.0;data
source=c:\\Empleados.mdb";
    string comando = "Select * from Empleados";

```

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	

```


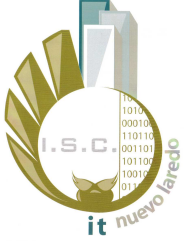
OleDbDataAdapter da = new OleDbDataAdapter(comando, cc);
DataTable dt = new DataTable();
da.Fill(dt);
cmbEmpleados.DataSource = dt;
cmbEmpleados.DisplayMember = "Nombre";
cmbEmpleados.ValueMember = "Clave";
cmbEmpleados.DropDownStyle = ComboBoxStyle.DropDownList;
cmbEmpleados.SelectedIndex = -1;
}

```

Observe que el uso de un `DataAdapter` nos indica que estamos utilizando el modo desconectado de ADO.NET al momento de leer los datos que se van a colocar en el `comboBox` llamado `cmbEmpleados`.

Posteriormente, se cambian algunas propiedades de `cmbEmpleados`. Por ejemplo, el origen de datos (`DataSource`) se asigna al `DataTable` (`dt`) que es donde se encuentran los datos. Al hacer esto, debemos decirle al `cmbEmpleados` cual es la columna que deseamos mostrar (Recordemos que hay 3 columnas: Clave, Nombre, Sueldo), por lo que indicamos en la propiedad `DisplayMember` que deseamos mostrar el Nombre del empleado. Por otra parte, también indicamos en la propiedad `ValueMember` cuál es la columna de la que nos interesa tomar su valor cuando el usuario seleccione un elemento del `comboBox`. En este caso, deseamos conocer la clave del empleado pues es la columna que lo identifica de manera única en la base de datos.

También se selecciona el estilo del combo (propiedad `DropDownStyle`) y se indica que deseamos que inicialmente ningún elemento se encuentre seleccionado (Esto se logra colocando la propiedad `SelectedIndex` en -1).

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	

8. Regresar al modo de diseño de la Forma, y dar doble click en cualquier área gris de la Forma (Cuidando no hacerlo sobre alguno de los controles). Esto nos permite teclear código en el evento "LOAD" de esta Forma.


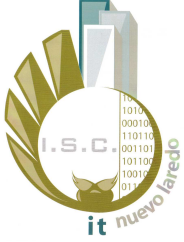
Esto quiere decir que el código introducido ahí, se va a ejecutar cuando la forma se está cargando en memoria por primera vez (justo cuando el programa comienza su ejecución). La idea de hacer esto es llenar el `comboBox` con los nombres de los empleados que están registrados en la base de datos, para que el usuario pueda seleccionar cual de ellos es el que quiere borrar. El siguiente código es el que se va a teclear ahí:

**PonerDatosEnElCombo();**

9. Ahora, justo **después** del método llamado **private void PonerDatosEnElCombo()** teclear el siguiente método:

```
private void BorrarEmpleadoSeleccionado()
{
    int clave = (int)cmbEmpleados.SelectedValue;
    string cc = "provider=Microsoft.Jet.OleDb.4.0; data
source=c:\\Empleados.mdb";
    OleDbConnection cn = new OleDbConnection(cc);
    cn.Open();
    string comando = "Delete from Empleados where clave = " + clave;
    OleDbCommand cm = new OleDbCommand(comando, cn);
    cm.ExecuteNonQuery();
    cn.Close();
    MessageBox.Show(" Eliminado correctamente! ");
    PonerDatosEnElCombo();
}
```

10. Ahora, en el botón BORRAR, teclear el siguiente código:

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	

```


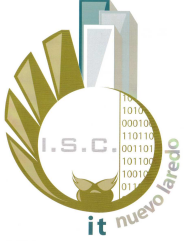
if (cmbEmpleados.SelectedIndex < 0)
{
    MessageBox.Show(" Error! No hay ningun empleado seleccionado ");
}
else
{
    DialogResult r;
    r = MessageBox.Show(" Esta seguro de borrarlo? ", "Confirmar",
    MessageBoxButtons.YesNo);
    if (r == DialogResult.Yes)
    {
        BorrarEmpleadoSeleccionado();
    }
    else
    {
        MessageBox.Show(" Operacion cancelada. No se ha eliminado nada
        ");
    }
}
}

```

11. Ejecutar el programa. Seleccionar un registro del `comboBox` y presionar el botón de Borrar. Observar los resultados. Verificar en Microsoft Access que el registro que se ha borrado, efectivamente se haya removido de la base de datos.

**[ Fin de la práctica ]**

NOTA: Observe que para cargar los datos al `comboBox` se utiliza una manera “desconectada”, y para borrar el registro deseado se utiliza una manera “conectada”.

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 6	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Bajas en bases de datos de MS Access con aplicaciones en C#				
<b>MAESTRO:</b> Dr. Ramón Roque Hernández, M.C. Bruno López Takeyas			<b>EMAIL:</b> bruno.lt@nlaredo.tecnm.mx	

### EXPLICACIÓN DE LA ESTRATEGIA UTILIZADA EN EL BOTÓN “BORRAR”

1. Si ningún elemento está seleccionado en el `comboBox`, indicar error (pues el usuario debe indicar cuál empleado desea borrar).
2. Si hay un elemento seleccionado en el `comboBox`:
  - a. Preguntar al usuario si está seguro de borrar ese registro.
  - b. Si el usuario confirmó, borrarlo de manera “conectada”.
  - c. Si el usuario NO confirmó, no borrar nada y salir.