

# Unidad 3

## Conectividad de Base de Datos

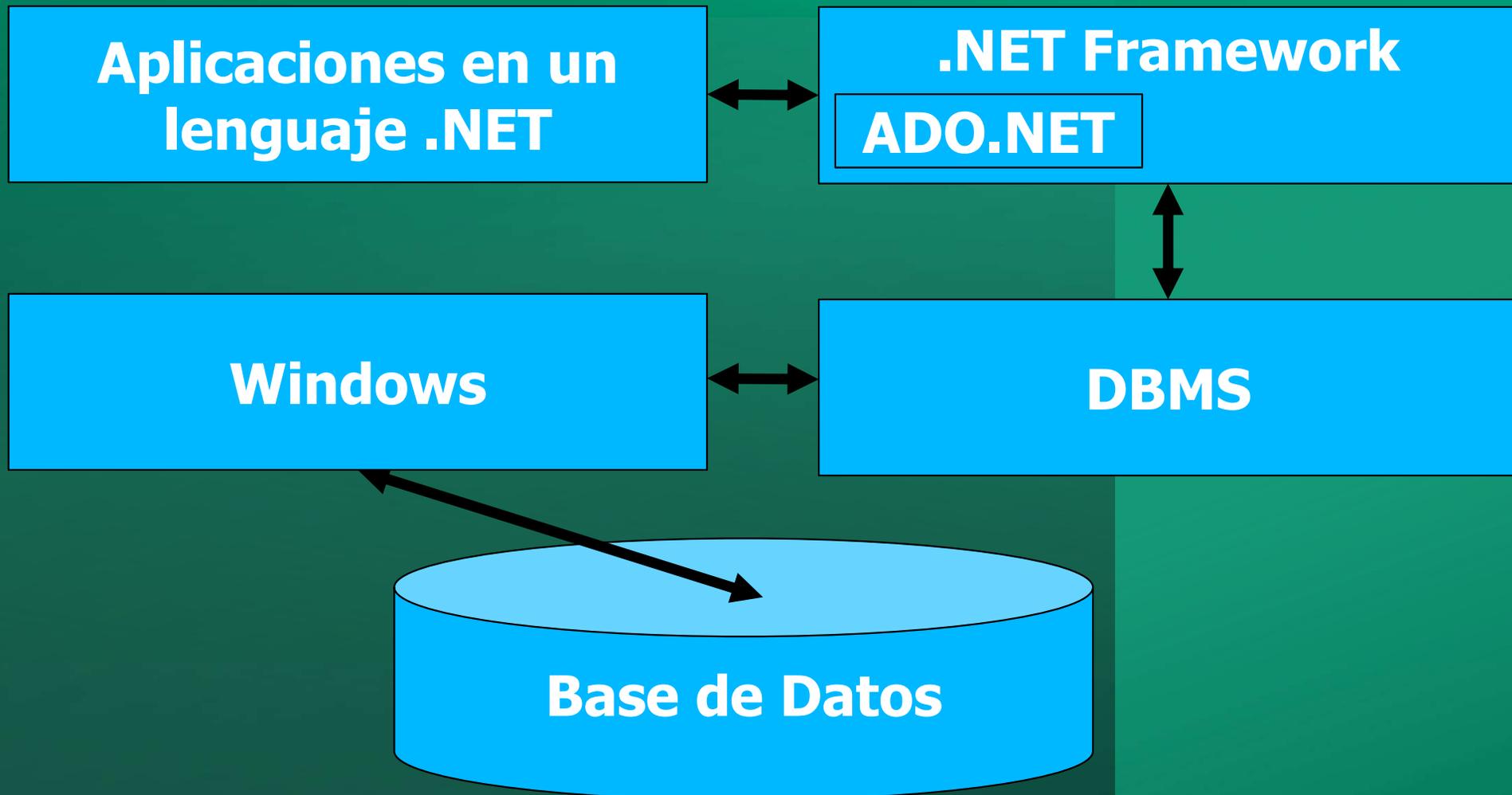
3.1 ADO.NET

3.2 Conectividad entre DML  
(huésped) y C# (anfitrión).

# 3.1 ADO.NET

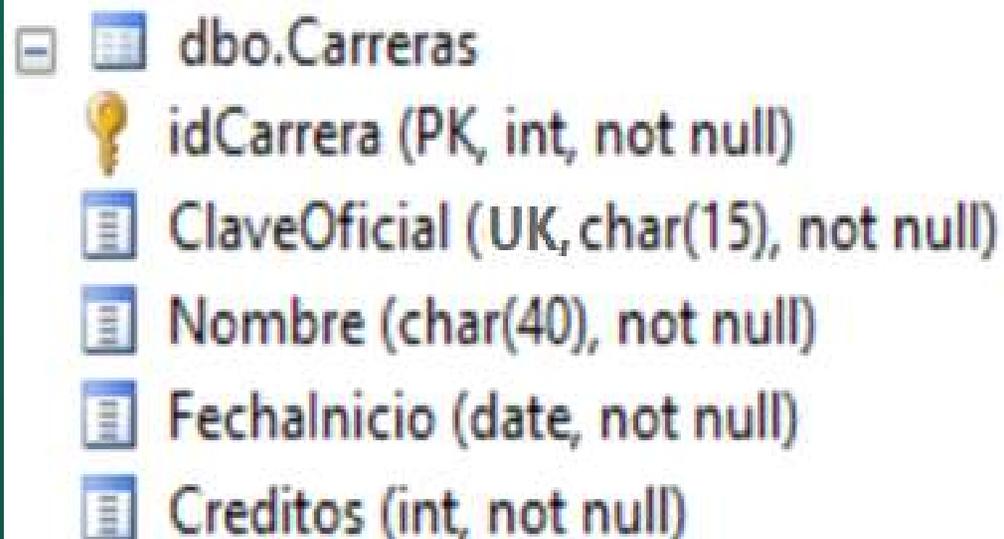
- ADO.NET es un conjunto de clases que provee la funcionalidad para el acceso y manipulación de bases de datos relacionales y otras fuentes de datos.
- Forma parte de .NET Framework de Microsoft.
  - .NET Framework es una plataforma de desarrollo de aplicaciones en Windows.
  - Se conforma de una biblioteca de clases para facilitar el desarrollo de aplicaciones de escritorio, web y servicios web.

## 3.1 ADO.NET



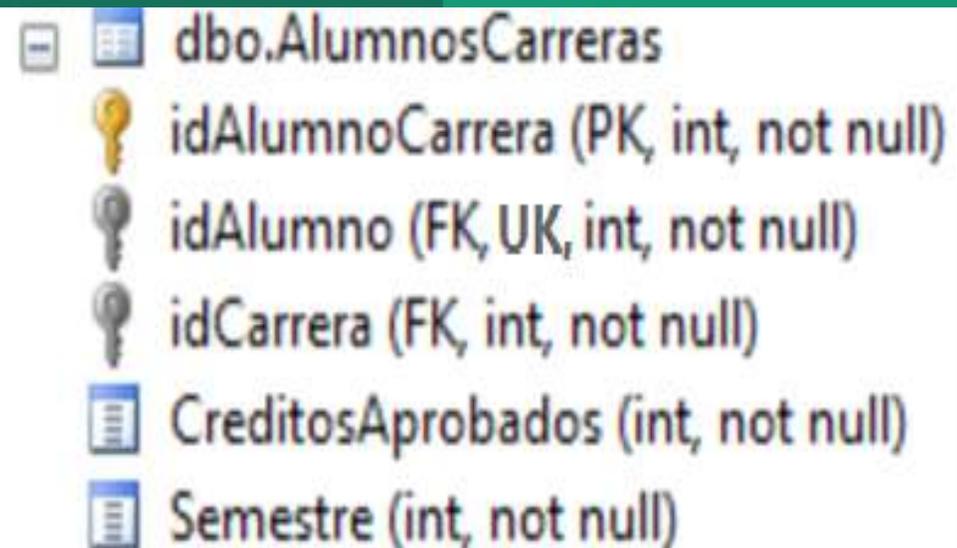
## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

- Crear las tablas **Carreras** y **AlumnosCarreras** en la Base de Datos **ITD**, con el esquema que se muestra.
  - No añade tuplas a ninguna de las tablas.



dbo.Carreras

	idCarrera (PK, int, not null)
	ClaveOficial (UK, char(15), not null)
	Nombre (char(40), not null)
	FechaInicio (date, not null)
	Creditos (int, not null)



dbo.AlumnosCarreras

	idAlumnoCarrera (PK, int, not null)
	idAlumno (FK, UK, int, not null)
	idCarrera (FK, int, not null)
	CreditosAprobados (int, not null)
	Semestre (int, not null)

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

Descargar Visual Studio Community 2022 de la URL siguiente:

<https://visualstudio.microsoft.com/es/vs/>

Durante la instalación elija al menos el paquete de desarrollo de aplicaciones **.NET para Escritorio**.

Una vez instalado, ejecútelo y cree un proyecto con las indicaciones que se muestran a partir de la página siguiente

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

Visual Studio 2022

Abrir recientes

Tareas iniciales



**Clonar un repositorio**

Obtiene código desde un repositorio en línea, como GitHub o Azure DevOps.



**Abrir un proyecto o una solución**

Abre un archivo .sln o proyecto de Visual Studio local.



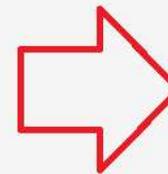
**Abrir una carpeta local**

Navegar y editar el código en cualquier carpeta



**Crear un proyecto**

Elija una plantilla de proyecto con la técnica scaffolding de código para comenzar.



[Continuar sin código →](#)

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

Crear un proyecto

Buscar plantillas (Alt+S)  [Borrar todo](#)

C# Windows Escritorio

 Aplicación de Windows Forms  
Plantilla de proyecto para crear una aplicación de Windows Forms (WinForms) de .NET.  
C# Windows Escritorio

 Aplicación de Windows Forms (.NET Framework)  
Proyecto para crear una aplicación con una interfaz de usuario de Windows Forms (WinForms)  
C# Windows Escritorio

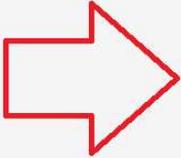
 Aplicación WPF  
Proyecto para crear una aplicación WPF de .NET.  
C# Windows Escritorio

 Biblioteca de clases de WPF  
Proyecto para crear una biblioteca de clases destinada a una aplicación WPF de .NET.  
C# Windows Escritorio Biblioteca

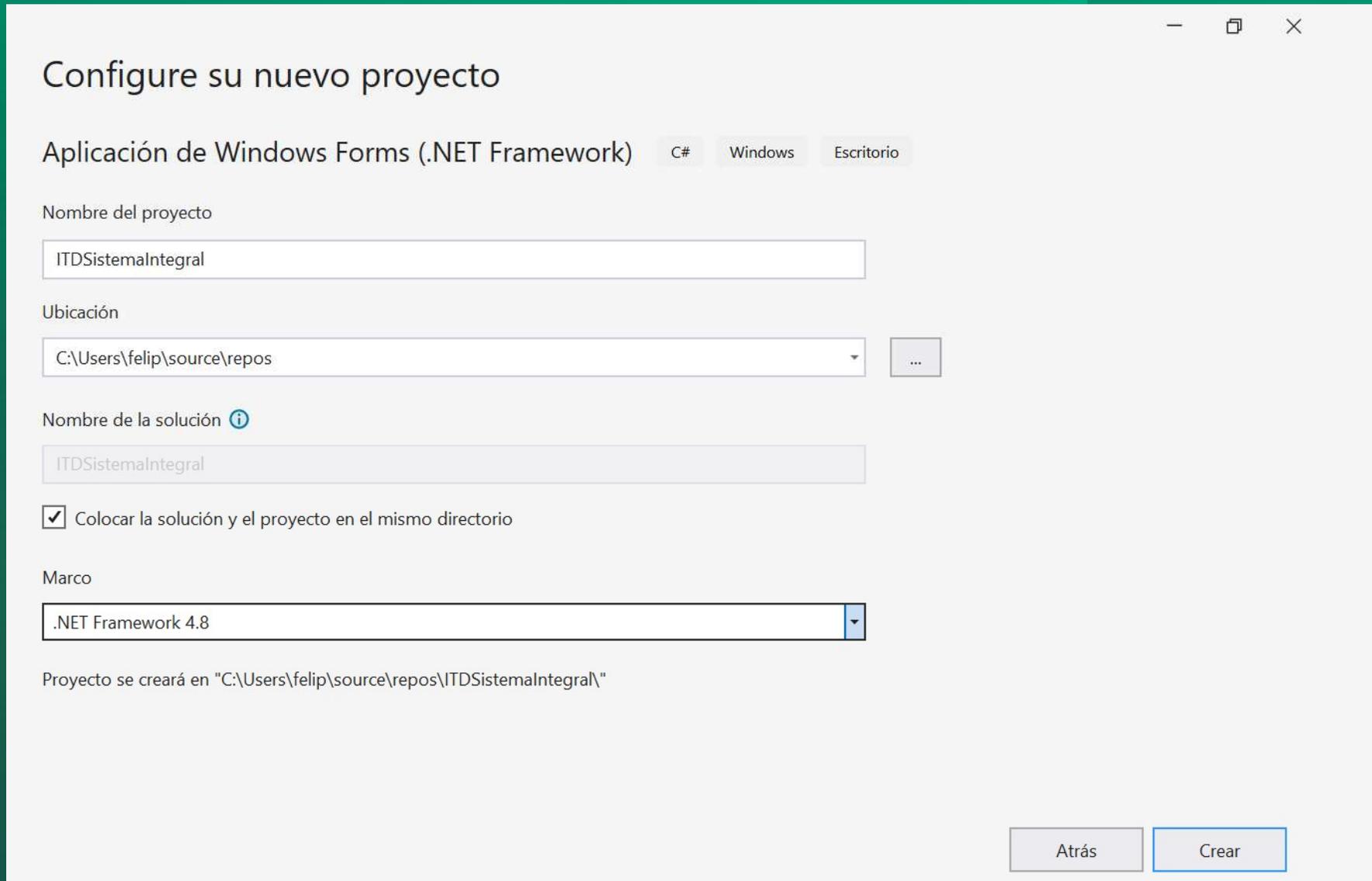
 Biblioteca de control personalizada de WPF  
Proyecto para crear una biblioteca de control personalizada para aplicaciones WPF de .NET.  
C# Windows Escritorio Biblioteca

 Biblioteca de controles de usuario de WPF  
Proyecto para crear una biblioteca de controles de usuario para aplicaciones WPF de .NET.

[Atrás](#) [Siguiente](#)



## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)



Configure su nuevo proyecto

Aplicación de Windows Forms (.NET Framework) C# Windows Escritorio

Nombre del proyecto

ITDSistemaIntegral

Ubicación

C:\Users\felip\source\repos

Nombre de la solución ⓘ

ITDSistemaIntegral

Colocar la solución y el proyecto en el mismo directorio

Marco

.NET Framework 4.8

Proyecto se creará en "C:\Users\felip\source\repos\ITDSistemaIntegral\"

Atrás Crear

# 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

The image shows a screenshot of the Visual Studio IDE. The main window displays a Windows Form titled "Form1" with the following fields and controls:

- Clave Oficial:** A text box, highlighted with a red rectangle.
- Nombre:** A text box.
- Fecha de Inicio:** A date picker set to 01/10/2024.
- Total de Créditos:** A text box.
- Buttons:** "Guardar" and "Cancelar".

The Properties window on the right shows the selected control, **tbClaveOficial**, with its properties listed. The (Name) property is highlighted with a red box.

The bottom status bar shows the compilation status: 0 Errores, 0 Advertencias, and 0 de 2 Mensajes. The "Lista de errores de búsqueda" table is empty.

Códi...	Descripción	Proyecto	Archivo	Lí...	Estado suprimido
---------	-------------	----------	---------	-------	------------------

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

```
string conexionString = "Data Source=ING-ALANIS-LAP\\SQLEXPRESS;" +
    "Initial Catalog=ITD;Integrated Security=True";
try
{
    string sFechaInicio = dtpFechaInicio.Value.ToString("yyyy-MM-dd");
    String comandoString = "insert into Carreras "+
        "(ClaveOficial, Nombre, FechaInicio, Creditos) values ('" +
        tbClaveOficial.Text + "','" + tbNombre.Text + "','" +
        sFechaInicio + "','" + tbCreditos.Text + "')";
    using (SqlConnection conexion = new SqlConnection(conexionString))
    {
        SqlCommand comando = new SqlCommand(comandoString, conexion);
        comando.Connection.Open();
        int numTuplas = comando.ExecuteNonQuery();
        MessageBox.Show("; " + numTuplas + " Alta(s) Efectuada(s)!");
        comando.Connection.Close();
        this.Close();
    }
}
catch (Exception ex)
{
    if (ex.Message.ToUpper().Contains("UNIQUE"))
        MessageBox.Show(";CLAVE DE CARRERA YA EXISTE! ");
    else
        MessageBox.Show(";Error DESCONOCIDO al dar el alta! " +
            ex.Message);
}
```

**Añadir este código al Botón  
*Guardar***

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

Form1

### Alta de Nueva Carrera

Clave Oficial

Nombre

Fecha de Incorporación  ▼

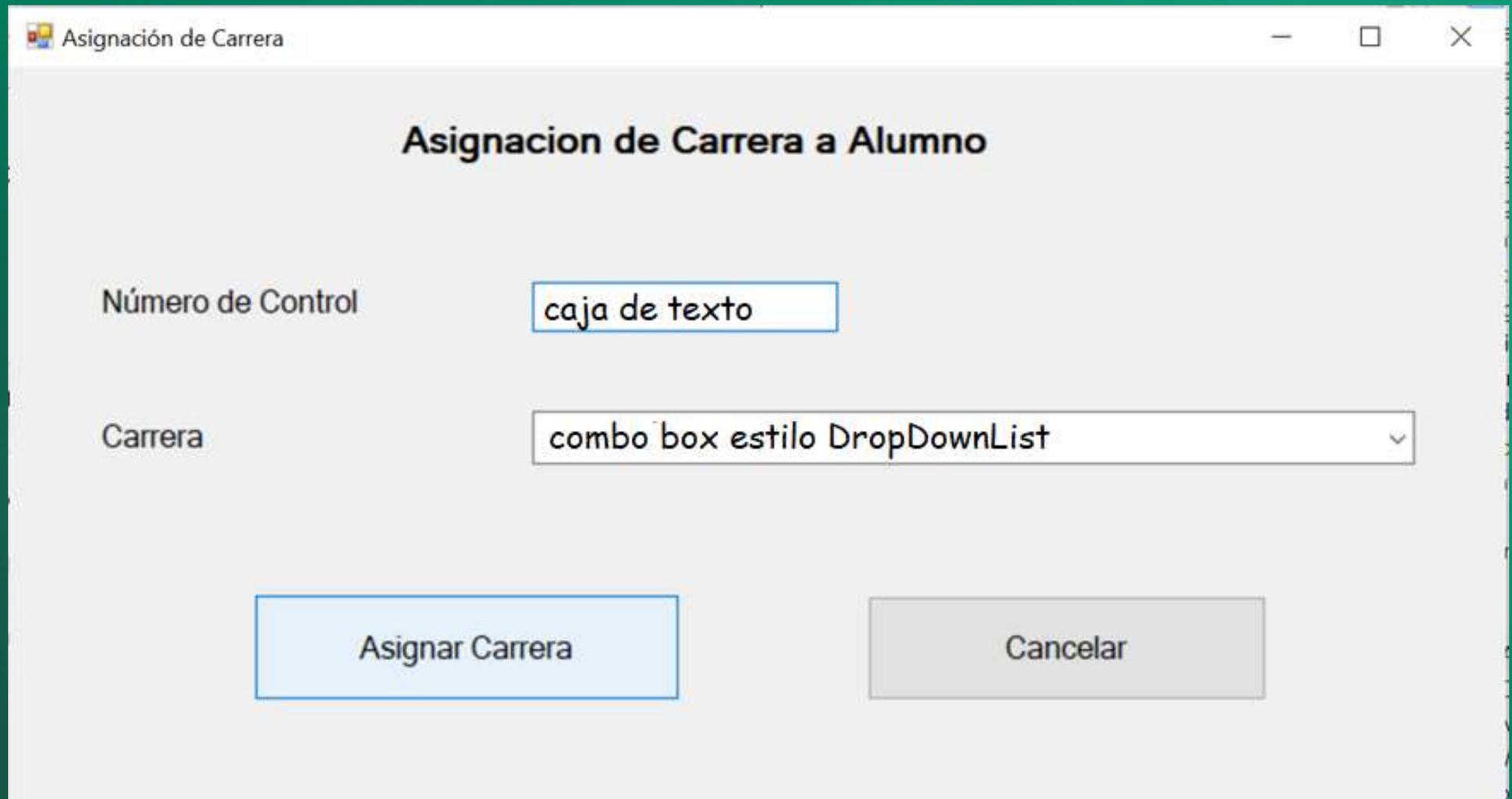
Total de Créditos

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

- Añada una restricción check a cada uno de los siguientes atributos:
  - *ClaveOficial*.
    - No se puede dejar vacía la caja de texto (si la dejamos en blanco no le asigna NULL por lo que la restricción de que no permita nulos no es efectiva en este caso).
  - *Nombre*.
    - *Idem*.
  - *Creditos*.
    - Debe encontrarse en el rango de 400 a 440.
- Atrape los mensajes de error en el código por las restricciones check y el mensaje de error de truncado para que muestre de una manera amigable a los usuarios.
- Haga las pruebas necesarias.

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

Crear este formulario para la asignación de la carrera a un estudiante



The image shows a screenshot of a Windows application window titled "Asignación de Carrera". The window contains a form with the following elements:

- Form Title:** "Asignacion de Carrera a Alumno"
- Form Fields:**
  - Número de Control:** A text box containing the placeholder text "caja de texto".
  - Carrera:** A dropdown menu (comboBox) containing the placeholder text "combo box estilo DropDownList".
- Buttons:**
  - Asignar Carrera:** A blue button with a white border.
  - Cancelar:** A gray button with a white border.

```

string conexionString = "Data Source=ING-ALANIS-LAP\\SQLEXPRESS;" +
                        "Initial Catalog=ITD;Integrated Security=True";
using (SqlConnection conexion = new SqlConnection(conexionString))
{
    //
    // Obtener idAlumno
    //
    string consulta = "select idAlumno from Alumnos where NumControl='" + tbNumControl.Text + "'";
    SqlCommand comando = new SqlCommand(consulta, conexion);
    try
    {
        comando.Connection.Open();
        if (comando.ExecuteScalar() == null)
            throw new ArgumentException("Número de Control no Existe");
        iIdAlumno = (Int32)comando.ExecuteScalar();
        //MessageBox.Show("Alumno "+iIdAlumno.ToString());
    }
    catch (Exception ex)
    {
        string mensajeError = ex.Message;
        MessageBox.Show(ex.Message);
        this.Close();
    }
    comando.Connection.Close();
    //
    // Obtener idCarrera
    //
    consulta = "select idCarrera from Carreras where Nombre='" + cbCarrera.Text + "'";
    comando = new SqlCommand(consulta, conexion);
    try
    {
        comando.Connection.Open();
        if (comando.ExecuteScalar() == null)
            throw new ArgumentException("Carrera no Existe");
        iIdCarrera = (Int32)comando.ExecuteScalar();
    }
    catch (Exception ex)
    {
        string mensajeError = ex.Message;
        MessageBox.Show(ex.Message);
        this.Close();
    }
    comando.Connection.Close();
}

```

Añadir este código al Botón  
***Asignar Carrera***

```

//
// Añadir la tupla en AlumnosCarreras
//
try
{
    String comandoString = "insert into AlumnosCarreras values (" +
        iIdAlumno.ToString() + "," + iIdCarrera.ToString() + ",0,1)";
    using (SqlConnection conexion = new SqlConnection(conexionString))
    {
        SqlCommand comando = new SqlCommand(comandoString, conexion);
        comando.Connection.Open();
        int numTuplas = comando.ExecuteNonQuery();
        MessageBox.Show("¡Asignación Efectuada Correctamente!");
        tbNumControl.Text = "";
    }
}
catch (Exception ex)
{
    if (ex.Message.ToUpper().Contains("FOREIGN") && ex.Message.ToUpper().Contains("idAlumno"))
        MessageBox.Show("¡Número de Control no existe! ");
    else if (ex.Message.ToUpper().Contains("FOREIGN") && ex.Message.ToUpper().Contains("idCarrera"))
        MessageBox.Show("¡Carrera no existe! ");
    else if (ex.Message.ToUpper().Contains("UNIQUE"))
        MessageBox.Show("¡ERROR. Alumno ya tiene carrera asignada! ");
    else
        MessageBox.Show(ex.Message);
}
//
// Final de Añadir Tupla en AlumnosCarreras
//

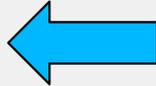
```

Este código es complementario al de la diapositiva anterior

## 3.2 Conectividad entre DML (huésped) y C# (anfitrión)

Asignación de Carrera

**Asignacion de Carrera a Alumno**

Número de Control   Asignar carrera a un estudiante

Carrera

Asignación de Carrera

**Asignacion de Carrera a Alumno**

Número de Control   Intentar asignar otra carrera al mismo estudiante

Carrera

¡ERROR. Alumno ya tiene carrera asignada!

# Unidad 3

## Conectividad de Base de Datos

### **Ejercicio:**

Cree un formulario para registrar los datos de un alumno en forma completa (tanto los que pertenecen a la tabla Alumnos como Personas) y guardarlos en la Base de Datos en las tablas correspondientes.

# Unidad 3

## Conectividad de Base de Datos

Considere que debe hacer en primer lugar el insert en la tabla *Personas*.

Tabla PERSONAS

<u>IdPersona</u>	Nombre	Apellidos	Calle	<u>NumExt</u>	<u>Poblacion</u>	<u>Pais</u>	<u>FechaNac</u>	Sexo	CURP
1	Parejita	López	Zarco	123	Cd de México	México	07-02-1981	Hombre	L1
2	<u>Johanness</u>	Gutenberg	Carlomagno	1	Mainz	Alemania	12-01-1398	Hombre	G2
3	Benito	Juárez García	Monte Albán	100	Cd de México	México	21-03-1806	Hombre	J4
4	Luis	Pasteur	Campos Elíseos	234	París	Francia	20-03-1850	Hombre	P1
5	Abraham		Calle del	347	Jerusalén	Israel	11-04-1890	Hombre	A0
6	José	Revueltas	Negrete	1002	Durango	México	24-03-1982	Hombre	R7
7	Lorena	Ochoa	Fresno	1410	Guadalajara	México	23-06-1981	Mujer	O1
8	Aristóteles		Templo Atenea	542	Atenas	Grecia	23-07-1905	Hombre	A1
9		Tchaikovski	Plaza Roja	471	Moscú	Rusia	13-08-1920	Hombre	T4
10		Botticelli	Filipo <u>Lippi</u>	2	Florenia	Italia	07-09-1919	Hombre	B9
11	José Luis	López	Francisco Zarco	123	Cd de México	México	09-03-1961	Hombre	JLL01
12	<u>Friele</u>	<u>Gensfleisch</u>	Carlomagno	1	Mainz	Alemania	09-03-1370	Hombre	GF7
13	Antonio	Maza	Guelaguetza	201	Oaxaca	México	15-05-1780	Hombre	AM01
14	Margarita	Maza	Monte Albán	100	Cd de México	México	19-08-1820	Mujer	MM01

# Unidad 3

## Conectividad de Base de Datos

Una vez ejecutado el *insert* en la tabla ***Personas***, se obtiene el valor de la *idPersona* y se podrá completar el *insert* en la tabla ***Alumnos***.

Tabla **ALUMNOS**

<b>IdAlumno</b>	<b>NumControl</b>	<b>EscuelaProcede</b>	<b>IdPersona</b>
1	98040151	Prepa PUMAS	1
2	97040587	Palacio Nacional	3
3	97040014	Colegio Vizcaya	6
4	96040121	LPGA	7
5	98040150	Colegio Alemán	2

# Unidad 3

## Conectividad de Base de Datos

### Observaciones:

- Como en la tabla *Alumnos* hay integridad referencial hacia *Personas*, primero hay que añadir la tupla en *personas* para obtener el valor de la *idPersona* y poder insertar la tupla en *Alumnos*.
- El control de las excepciones con *try-catch* debe considerar las llaves únicas en ambas tablas.
- Considere, que en la tabla *Personas*, los valores válidos para el atributo *Sexo* no deben asignarse como constantes sino tomarse de la Base de Datos:
  - De una tabla de dominio o del Diccionario de Datos (tabla *sys.check\_constraints*).