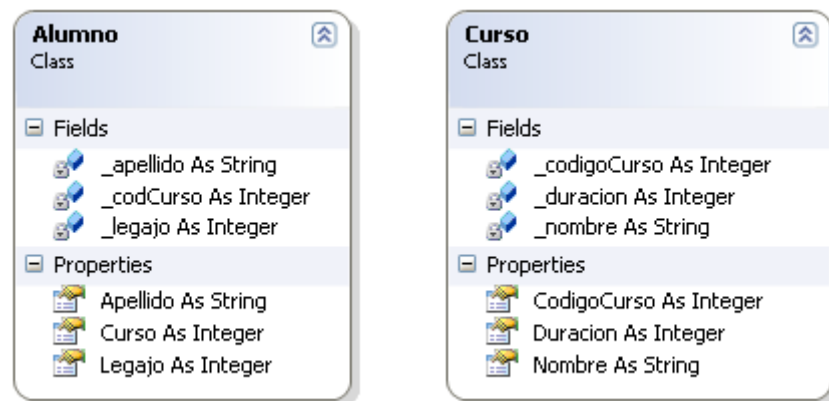


**1- Formularios y Clases:** Realizar una solución llamada con su apellido punto (.) su nombre. Ej: Maradona.Diego.

A esta solución agregarle un proyecto de tipo **Biblioteca de Clases** llamada **Entidades** en la cual se deberá crear las siguientes clases con sus respectivos constructores:



Además se deberá agregar en la solución un proyecto de tipo **WindowsForm** llamado **Modelo.SegundoParcial.LabIII** con los siguientes formularios:

The screenshot shows a Windows Form titled 'Cambiar el texto o 1 pt. menos'. It has a tab labeled 'Datos del Alumno'. Inside the tab, there are three input fields: 'Apellido:' with a text box, 'Legajo:' with a text box, and 'Curso:' with a dropdown menu. At the bottom of the dialog, there are two buttons: 'Aceptar' and 'Cancelar'.

**frmAlumno:** Este formulario nos permitirá realizar las tareas de alta, baja y modificación de los alumnos.

# En el manejador de evento click del botón **Aceptar**, se creará una instancia de tipo **Alumno** que se expondrá en la **propiedad de sólo lectura** de este formulario. Además se colocará en la propiedad **DialogResult** del formulario el valor DialogResult.OK.

# En el manejador de evento click del botón **Cancelar** se colocará en la propiedad **DialogResult** del formulario el valor DialogResult.Cancel.

**frmPrincipal:**

The screenshot shows a Windows Form titled 'Cambiar el nombre o 1 pt. menos' with a menu bar containing 'Alumno', 'Mostrar', and 'Guardar'. To the right of the form is the 'Document Outline - frmPrincipal' pane. It shows a tree view of the form's components: 'frmPrincipal Form' is the root, followed by 'mnuMenu MenuStrip'. Under 'mnuMenu' are several 'ToolStripMenuItem' items: 'AlumnoToolStripMenuItem', 'AltaToolStripMenuItem', 'BajaToolStripMenuItem', 'ModificacionToolStripMenuItem', 'MostrarToolStripMenuItem', 'AlumnosToolStripMenuItem', 'CursosToolStripMenuItem', 'AlumnosConElNombreDelCursoToolStripMenuItem', 'MostrarAlumnosdeASPNETToolStripMenuItem1', and 'GuardarToolStripMenuItem'.

## frmPrincipal (Atributos)

\_dataSetAlumnos\_Cursos As **DataSet**

\_dataAdapterAlumnos As **XXXDataAdapter**

## frmPrincipal (Métodos)

# CrearDataTableCursos() As **DataTable**: Será invocado sólo si el archivo “CursosDatos.XML” y “CursosEsquema.XML” NO existen, este método creará el DataTable **dtCursos** con los siguientes campos: **Código** (entero), **Duración** (entero) y **Nombre** (cadena).

**Código** será auto-incremental, comenzando de 1000 y aumentando de 5 en 5.

Cargar en el DataTable al menos **3 cursos** con distinta carga horaria y luego guardar el esquema y los datos en dos archivos distintos (**CursosDatos.XML** y **CursosEsquema.XML**)

# ConfigurarDataAdapter (): Este método configurará el DataAdapter con su conexión, comandos y parámetros para interactuar con la tabla “**Alumnos**” de la base de datos.

**Nota:** *Se pide que la instrucción T-SQL del SelectCommand retorne: Legajo, Apellido y Curso (utilizar alias).*

# TraerDatos(): Carga los datos desde la base de datos en el DataSet (en el DataTable **dtAlumno** con un **dataReader**) y los datos del DataTable **dtCursos** desde los archivos .XML o desde el método **CrearDataTableCurso()**, según corresponda.

**Nota:** *Se pide que la instrucción T-SQL del Command retorne: Legajo, Apellido y Curso (utilizar alias).*

# EstablecerRelacion(): Este método relacionará los DataTables **Alumnos** y **Cursos** que se encuentran dentro del DataSet.

## 2- Alta de Alumnos:

Se instanciará un objeto de tipo frmAlumno (**frmAltaAlumno**) y se mostrará en forma Modal. Con los atributos que posee el objeto de tipo **Entidades.Alumno** en este formulario, se creará una nueva fila del DataTable **dtAlumno** que se agregará a la colección **Rows** del mismo.

## 3- Baja de Alumnos:

Se pedirá el **Legajo** (a través de un **InputBox**) y se buscará en el DataTable el alumno con ese número de legajo. Si se encuentra, se creará un objeto de tipo **Entidades.Alumno** que se le pasará como parámetro al constructor de **frmAlumno**, cuya instancia se llamará **frmBajaAlumno**.

En dicha instancia se mostrarán los datos del alumno y si se elige **Aceptar**, el elemento se marcará como borrado. Si no se encuentra el número de legajo se deberá informar de lo acontecido.

## 4- Modificación de Alumnos:

Se pedirá el **Legajo** (a través de un **InputBox**) y se buscará en el DataTable el alumno con ese número de legajo. Si se encuentra, se creará un objeto de tipo **Entidades.Alumno** que se le pasará como parámetro al constructor de **frmAlumno**, cuya instancia se llamará **frmModificarAlumno**.

En dicha instancia se mostrarán los datos del alumno y se permitirán modificar todos a excepción del número de legajo. Si se elige **Aceptar**, se modificarán los datos en la fila que corresponda del DataTable **dtAlumno**. Si no se encuentra el número de legajo se deberá informar de lo acontecido.

## 5- Mostrar Alumnos y Cursos:

Al elegir esta opción del menú, se creará una instancia del formulario **frmMostrar** (que usted deberá crear) para mostrar los datos del DataTable que corresponda.

## 6- Mostrar los datos de los Cursos junto con el nombre de los alumnos que lo cursan(frmMostrar)

Utilizar: **GetParentRow** o **GetChildRows**, según corresponda.

## 7- Mostar los datos de los Alumnos junto con el nombre del Curso, filtrados para los que están en el curso “1005” (FrmMostrar).

Utilizar: **Select** y **GetParentRow** o **GetChildRows**, según corresponda.