

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases				
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx		

<b>OBJETIVO:</b> El estudiante elaborará diagramas de clases en UML que apliquen relaciones de composición entre clases
<b>MATERIAL Y EQUIPO NECESARIO:</b> <ul style="list-style-type: none"> <li>Se recomienda la utilización de software para elaborar diagramas de clases de UML como <b>NClass</b>, el cual puede descargarse de manera gratuita del sitio web <a href="http://nclass.sourceforge.net/index.html">http://nclass.sourceforge.net/index.html</a></li> <li>Elaborar programas de los ejercicios en C#</li> </ul>

*Elabore el diagrama de clases en UML y la codificación de un programa para resolver los siguientes problemas:*

1. Una empresa tiene el siguiente conjunto de datos:

- Registro Federal de Contribuyentes (RFC)
- Razón social (nombre)
- Nombre del gerente

Y se compone de un conjunto de departamentos, cuyos datos son:

- Número
- Nombre
- Nombre del jefe

Diseñe una clase para cada entidad descrita anteriormente que tenga atributos privados con los datos mencionados, propiedades públicas para acceder dichos datos y un constructor default para inicializarlas.

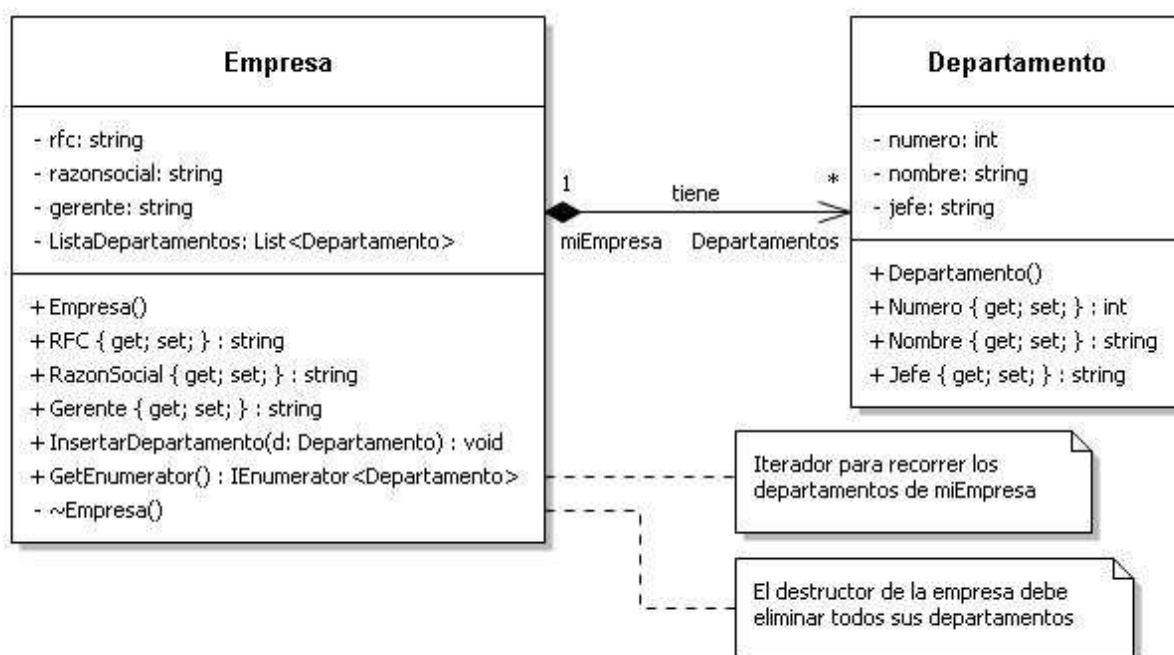
En virtud de que una empresa se compone de un conjunto de departamentos, es necesario establecer una relación de composición entre sus clases; para lograrlo, se requiere que el objeto de la empresa tenga una colección genérica de objetos en una lista simple con los datos de cada uno de sus departamentos. Además, la clase de la empresa debe tener lo siguiente:

- Método que reciba como parámetro un objeto con los datos de un departamento para insertarlo en la empresa.

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases				
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx		

- Iterador para recorrer los departamentos de la empresa (implementado a través del método `GetEnumerator()`).

Modele un sistema tomando como referencia el siguiente diagrama de clases en UML:

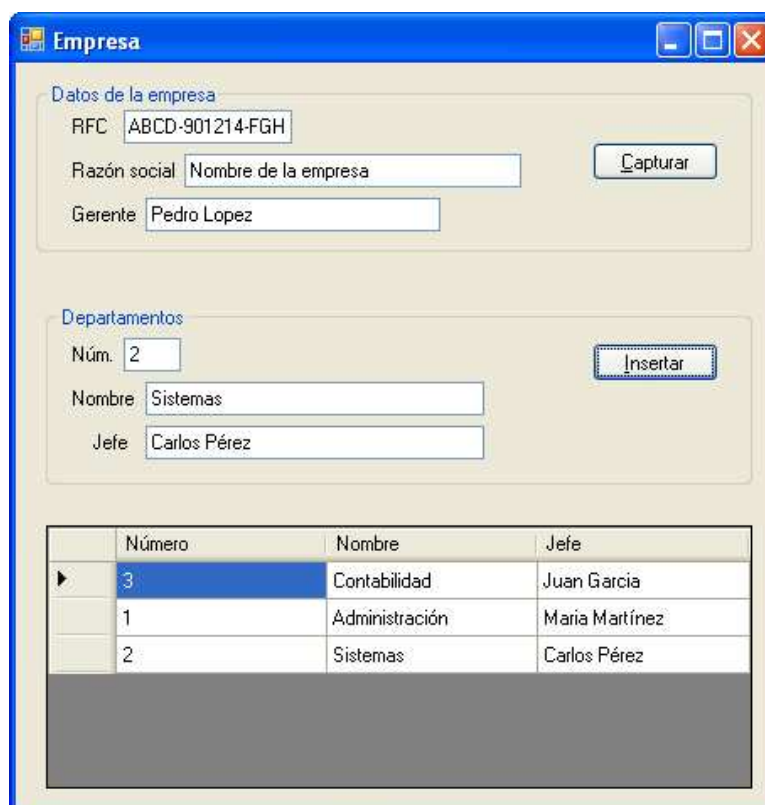


Diseñe una forma de Windows que contenga dos `groupBoxes`: uno para los datos de la empresa y otro con los datos de cada departamento.

Para insertar un departamento a la empresa, deben capturarse sus datos, oprimir el botón Insertar para crear un objeto y enviarlo al método `InsertarDepartamento()` de la clase `Empresa`; el cual agrega el nuevo departamento a la colección genérica de departamentos de la empresa (`ListaDepartamentos`).

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases				
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx		

Una vez insertado el departamento, se deben actualizar los datos de la colección genérica y mostrarlos en el `dataGridView1` utilizando el iterador para recorrer la lista de departamentos por medio del método `GetEnumerator()`.



The screenshot shows a Windows application titled "Empresa". It has two main sections: "Datos de la empresa" and "Departamentos".

**Datos de la empresa:**

- RFC: ABCD-901214-FGH
- Razón social: Nombre de la empresa
- Gerente: Pedro Lopez
- Buttons: Capturar

**Departamentos:**

- Núm.: 2
- Nombre: Sistemas
- Jefe: Carlos Pérez
- Buttons: Insertar

**Table of Departments:**

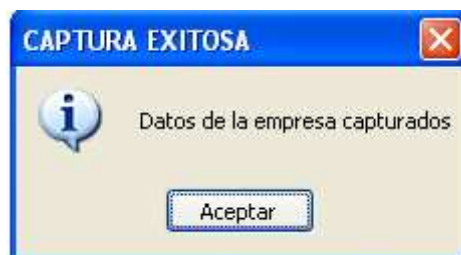
	Número	Nombre	Jefe
▶	3	Contabilidad	Juan Garcia
	1	Administración	Maria Martínez
	2	Sistemas	Carlos Pérez

Consideraciones para utilizar los objetos:

- Declare un objeto **global** llamado `miEmpresa` para administrar los datos de la empresa.
- Declare un objeto **local** para administrar los departamentos.

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases				
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx		

Confirme cada operación realizada con un aviso mediante un MessageBox:



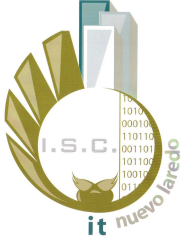
Limpie el contenido de los cuadros de texto (textBox) después de utilizarlos. Puede utilizar el siguiente fragmento de código:

```
// Limpia los datos de los textBoxes
foreach (Control x in groupBox2.Controls)
    if (x is TextBox)
        x.Text = "";
```

2. Agregue lo que considere necesario al ejercicio anterior para eliminar tanto la empresa como sus departamentos. Para eliminar un departamento de la empresa, selecciónelo del *dataGridView1* y cuando se oprima el botón correspondiente, entonces debe borrarse dicho departamento de la lista. Utilice como referencia el siguiente fragmento de código:

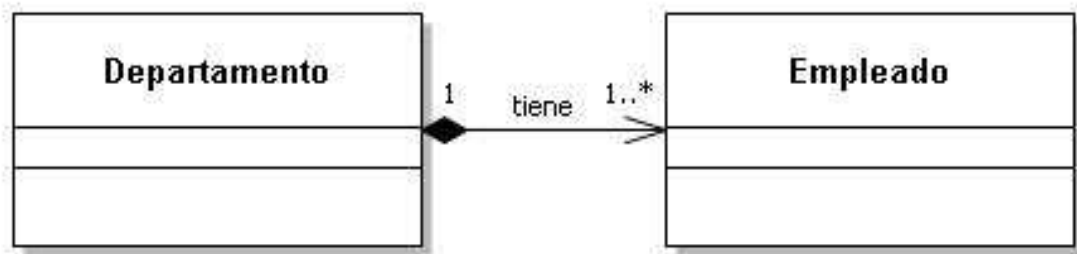
```
DataGridViewRow RenglonSeleccionado = dataGridView1.CurrentRow;

// Verifica si se seleccionó un renglón del dataGridView1
if (RenglonSeleccionado == null)
{
    MessageBox.Show("Seleccione un renglón del dataGridView");
    return;
}
else
{
    // Obtiene un dato del renglón seleccionado del dataGridView1
    Dato = int.Parse(RenglonSeleccionado.Cells[0].Value.ToString());
}
```

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases				
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx		

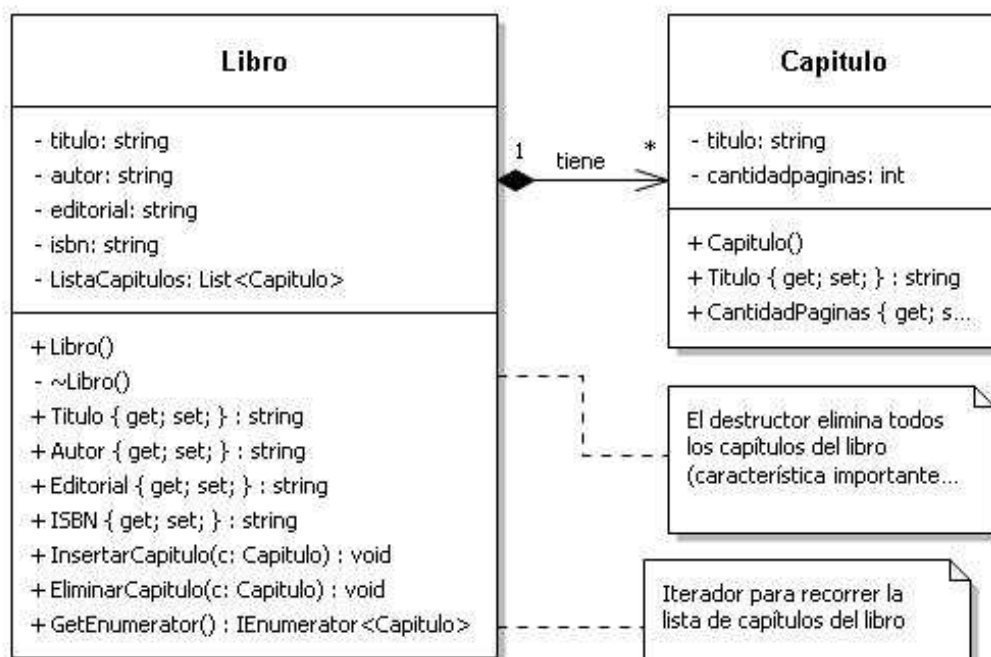
Tome en cuenta que al tener una relación de composición, si se elimina la empresa, también deben eliminarse todos sus departamentos (se recomienda el uso de un destructor).

- De manera semejante al ejercicio anterior, modele un sistema en el que se establezca una relación de composición entre un departamento y sus empleados. Agregue a cada clase los componentes que considere necesarios.




	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>			
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2	
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases				
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx		

4. Un libro se compone de un conjunto de capítulos. Modele un sistema donde se establezca una relación de composición de la siguiente forma:



Diseñe una forma que incluya los controles visuales que considere necesarios para capturar y desplegar los datos.

5. Agregue lo que considere necesario al ejercicio anterior para eliminar tanto el libro como sus capítulos. Para eliminar un capítulo, debe seleccionarse del *dataGridView*. Tome en cuenta que al tener una relación de composición, si se elimina el libro, también deben eliminarse todos sus capítulos (se recomienda el uso de un destructor).

	<b>INSTITUTO TECNOLÓGICO DE NUEVO LAREDO</b> <b>ING. EN SISTEMAS COMPUTACIONALES</b>				
	<b>MATERIA:</b> Programación Orientada a Objetos (C#)	<b>UNIDAD:</b> 3	<b>PRÁCTICA:</b> 2		
<b>NOMBRE DE LA PRÁCTICA:</b> Ejercicios aplicando relaciones de composición entre clases					
<b>MAESTRO:</b> Ing. Bruno López Takeyas, M.C.		<b>EMAIL:</b> takeyas@itnuevolaredo.edu.mx			

- Analice un sistema de la vida cotidiana que pueda modelarse mediante relaciones de composición entre clases. Realice el diagrama de clases de UML, diseñe la forma y codifique el sistema correspondiente.