

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

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

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 de la empresa)
- 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.

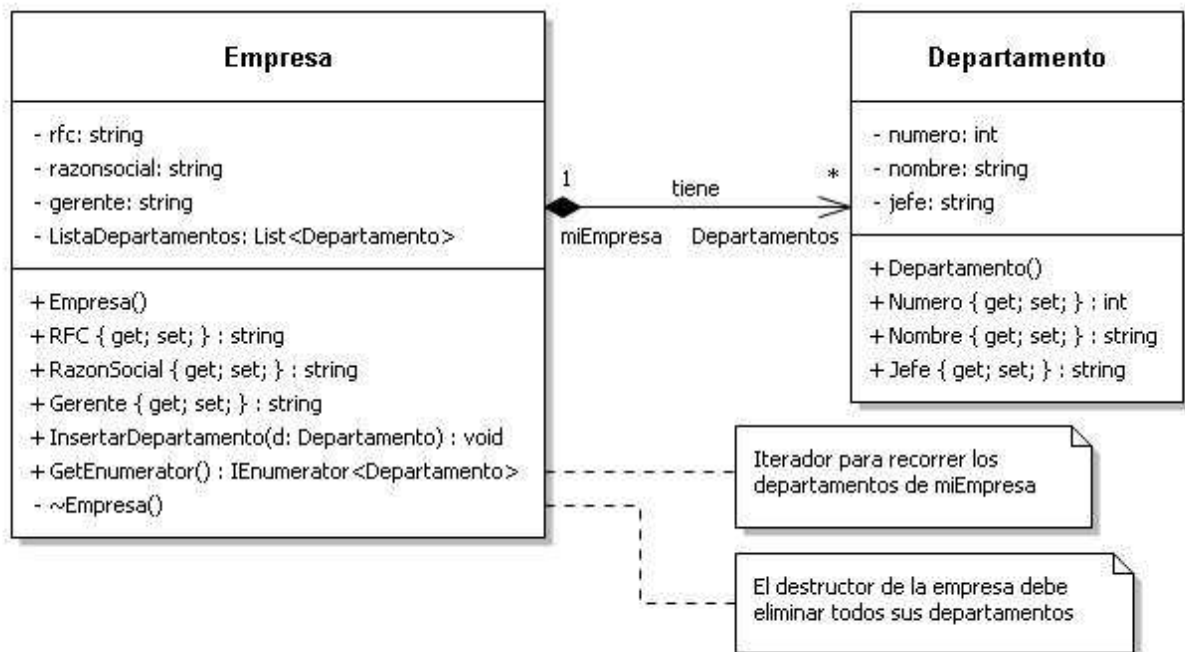
En virtud de que una empresa se compone de un conjunto de departamentos, es necesario establecer una relación de composición entre sus clases; para lograrlo, se requiere que el objeto de la empresa tenga una colección genérica de objetos en una lista simple con los datos de cada uno de sus departamentos.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

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

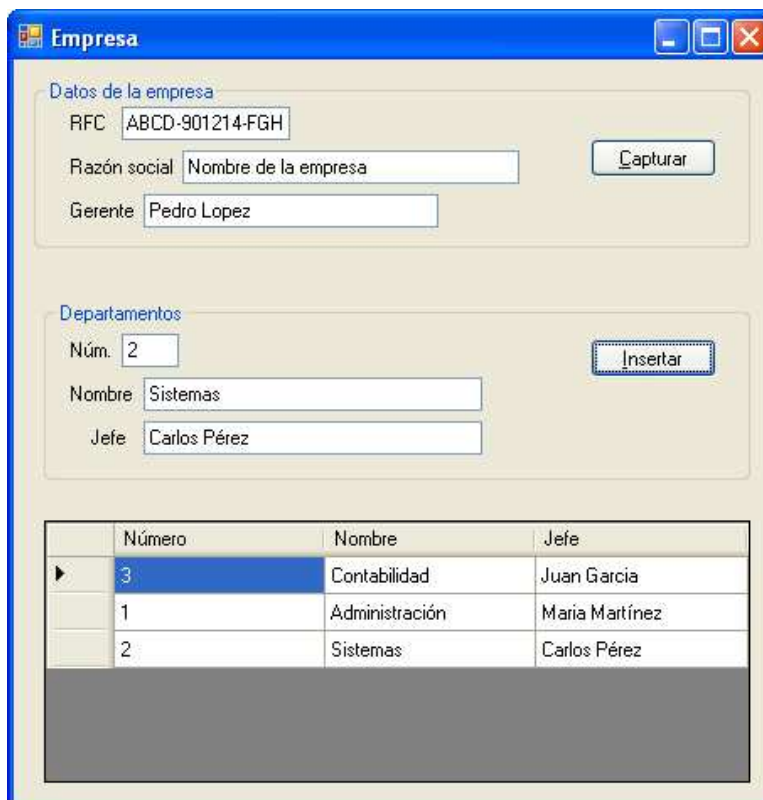


Diseñe una forma de Windows que contenga dos `groupBoxes`: uno para los datos de la empresa y otro con los datos de cada departamento.

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`).

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

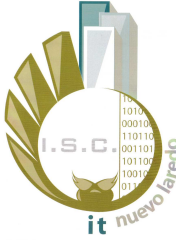
Una vez insertado el departamento, se deben actualizar los datos de la colección genérica y mostrarlos en el `dataGridView1` utilizando el iterador para recorrer la lista de departamentos por medio del método `GetEnumerator()`.



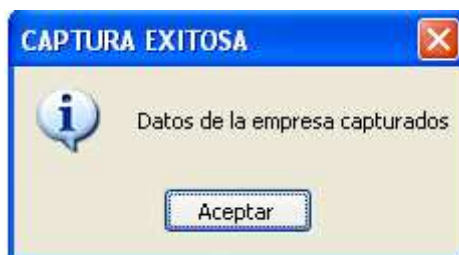
	Número	Nombre	Jefe
▶	3	Contabilidad	Juan Garcia
	1	Administración	Maria Martínez
	2	Sistemas	Carlos Pérez

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.

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

Confirme cada operación realizada con un aviso mediante un MessageBox:




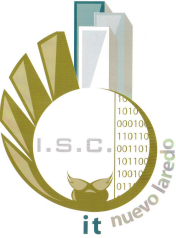
Limpie el contenido de los cuadros de texto (textBox) después de utilizarlos. Puede utilizar el siguiente fragmento de código:

```
// 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:

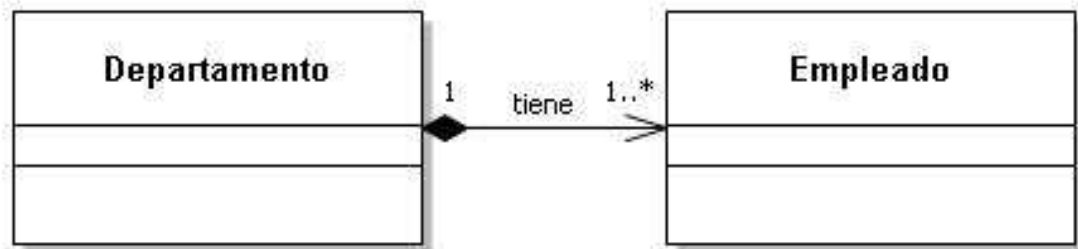
```
DataGridViewRow RenglonSeleccionado = dataGridView1.CurrentRow;

// 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());
}
```

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

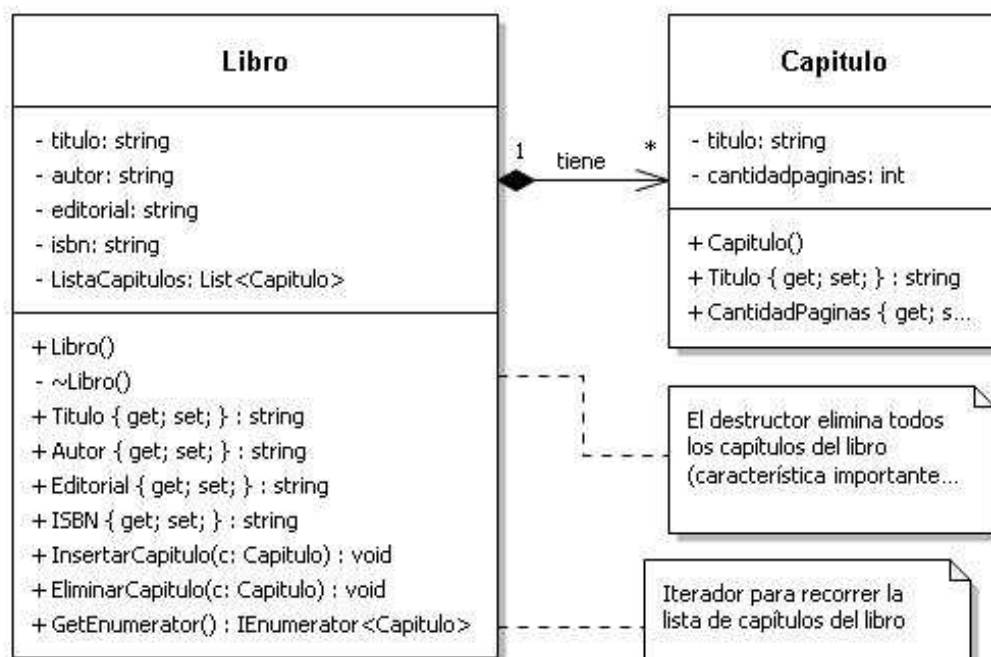
Tome en cuenta que al tener una relación de composición, si se elimina la empresa, también deben eliminarse todos sus departamentos (se recomienda el uso de un destructor).

- 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.



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

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

6. Una empresa tiene muchos empleados y requiere un sistema para modelar esta relación. Los datos de la empresa son:

- Razón social (nombre de la empresa)

Mientras que los datos de cada empleado son:

- Número
- Nombre
- Sueldo

Diseñe un diagrama de clases que modele esta relación y agregue los siguientes métodos a la clase de la empresa:

- Devolver la cantidad de empleados de la empresa
- Devolver la suma de los sueldos de todos los empleados

Considere lo siguiente:

- No se permite que un empleado labore en varias empresas a la vez
- No se permite que varios empleados tengan el mismo número

7. Un equipo de baloncesto se forma de varios jugadores y requiere un sistema para modelar esta relación.

Los datos del equipo son:

- Nombre del equipo

Mientras que los datos de cada jugador son:


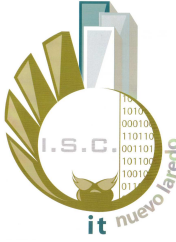
- Número de uniforme
- Nombre
- Puntos anotados

Diseñe un diagrama de clases que modele esta relación y agregue los siguientes métodos a la clase del equipo:

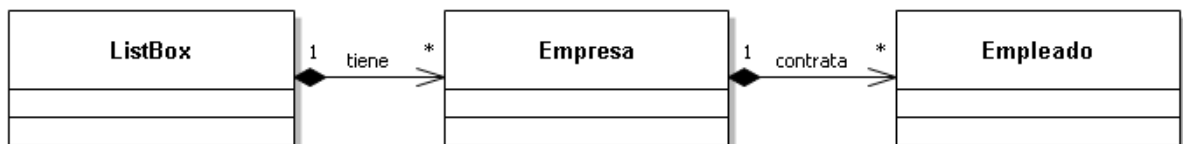
- Devolver la cantidad de jugadores registrados en el equipo
- Devolver la suma de los puntos anotados por los jugadores

Considere lo siguiente:

- No se permite que un jugador participe en varios equipos a la vez
- No se permite que varios jugadores tengan el mismo número de uniforme

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

8. Modifique el ejercicio 6 y agregue un `listBox` a la forma para que almacene muchos objetos de tipo `Empresa` con sus respectivos empleados. Al seleccionar una empresa del `listBox`:
- Automáticamente deben mostrarse sus empleados en un `dataGridView`.
 - Al capturar un nuevo empleado, éste debe agregarse a la empresa seleccionada en el `listBox` y mostrarse en el `dataGridView`.



Preguntas:

- ¿Este modelo permite que los empleados laboren en varias empresas?
- ¿Cómo se diseña un modelo para permitir que las empresas compartan sus empleados?
- ¿Cómo se contaría la cantidad de empleados de todas las empresas?
- ¿Cómo se sumarían los sueldos de los empleados de todas las empresas?

9. Una florería vende muchos tipos arreglos florales y cada arreglo se compone de varias flores.

Los datos de cada flor son:

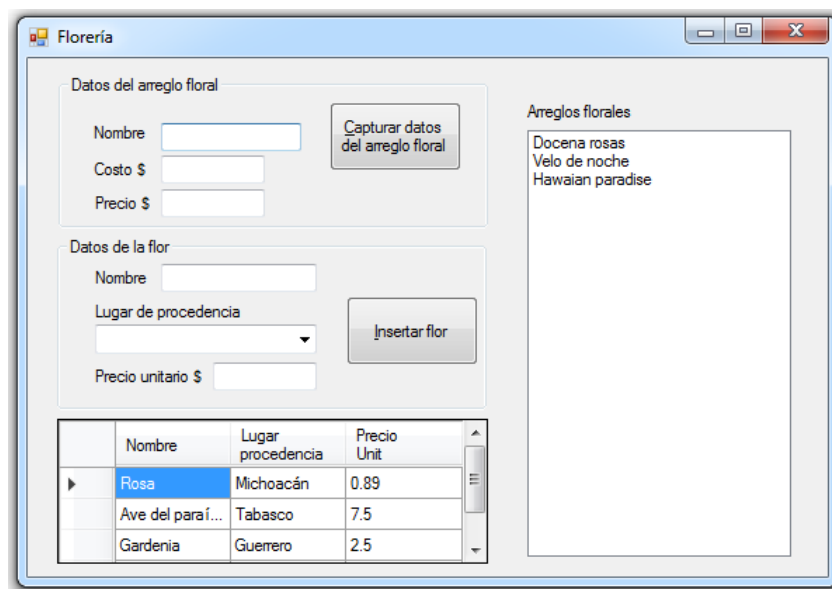
- *Nombre*
- *Lugar de procedencia*
- *Precio unitario*

Los datos de cada arreglo floral son:

- *Nombre*
- *Costo de elaboración*
- *Precio de venta al público*

Modele un sistema orientado a objetos que tenga la siguiente forma:

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



Nombre	Lugar procedencia	Precio Unit
Rosa	Michoacán	0.89
Ave del paraí...	Tabasco	7.5
Gardenia	Guerrero	2.5

Agregue botones a la forma para que resuelvan los siguientes problemas:

- ¿Cuántos arreglos florales tiene la florería?
- ¿Cuántos tipos de flores tiene un arreglo floral seleccionado del `ListBox`?
- ¿Cuál es la suma de los costos de elaboración de todos los arreglos florales?
- ¿Cuál es la suma de los precios unitarios de las flores de un arreglo floral seleccionado del `ListBox`?
- ¿Cuál es la suma de los precios unitarios de las flores de todos los arreglos florales?
- ¿Cuál es el arreglo floral con el mayor costo de elaboración?
- De un arreglo floral seleccionado de un `ListBox`, ¿Cuál es el lugar de procedencia con mayor suministro de flores?
- De todos los arreglos florales, ¿Cuál es el lugar de procedencia con mayor suministro de flores?
- De un arreglo floral seleccionado del `ListBox`, ¿Cuál es la flor con el menor precio unitario?
- De todos los arreglos florales, ¿Cuál es la flor con el menor precio unitario?

NOTA: Diseñe un método para resolver cada uno de los incisos y determine su ubicación, es decir, si dicho método lo coloca dentro de la clase del “todo”, de la clase “parte” o se trata de un método independiente implementado en la forma.

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

10. 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.