

Programación II / Laboratorio II

Área personal / Mis cursos / TSP039 / Clases / Clase 8

Clase 8

Teoría

[PPT]

[Introduccion-a-la-programacion-con-C-sharp]

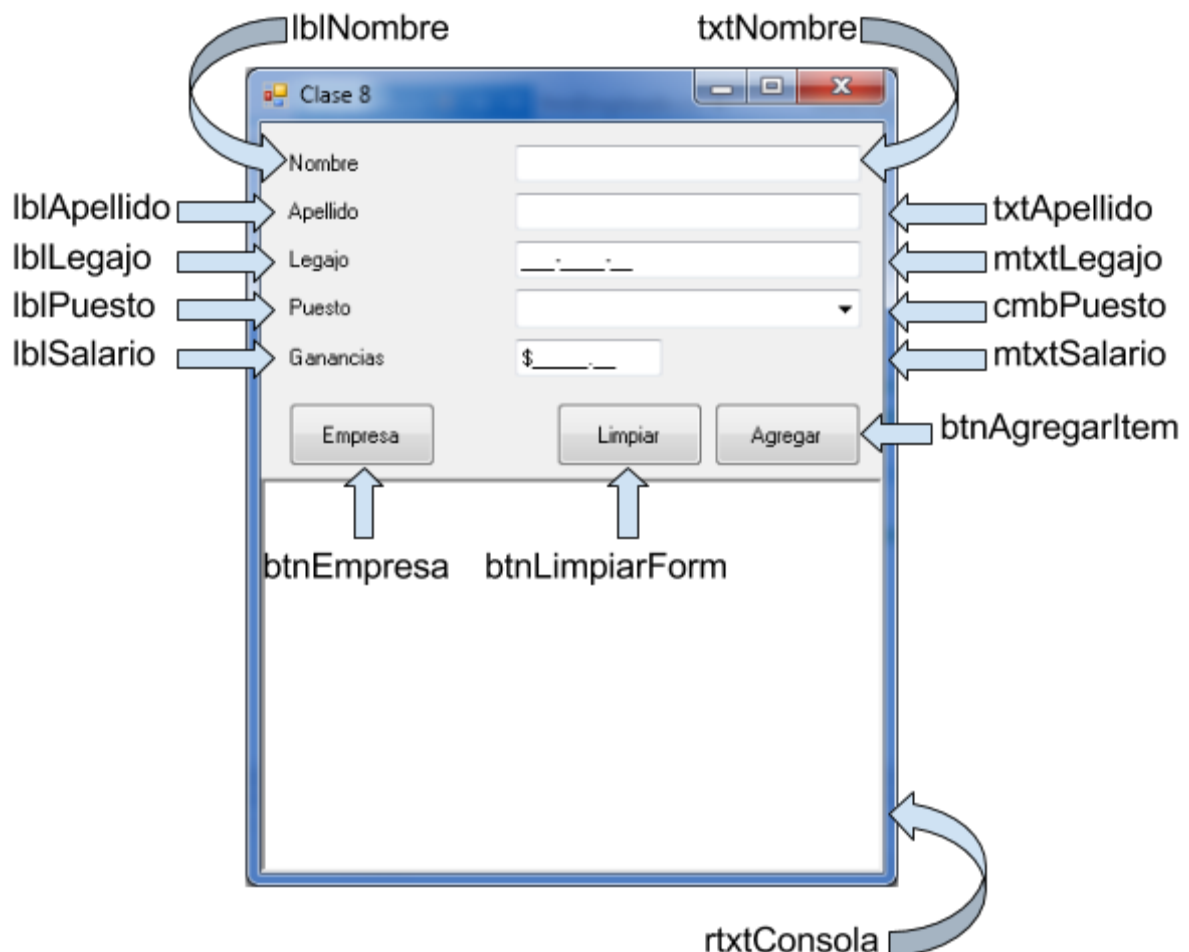
Cuestionario Teórico

[Guía de Estudio - Abstracción, Encapsulación y Ocultamiento de la información]

[Guía de Estudio - Propiedades, Indexadores y Enumerados]

Ejercicios

Realizar un formulario de carga de empleados con los siguientes datos:



En un proyecto del tipo Class Library codificar la clase Empleado para que reciba estos datos:

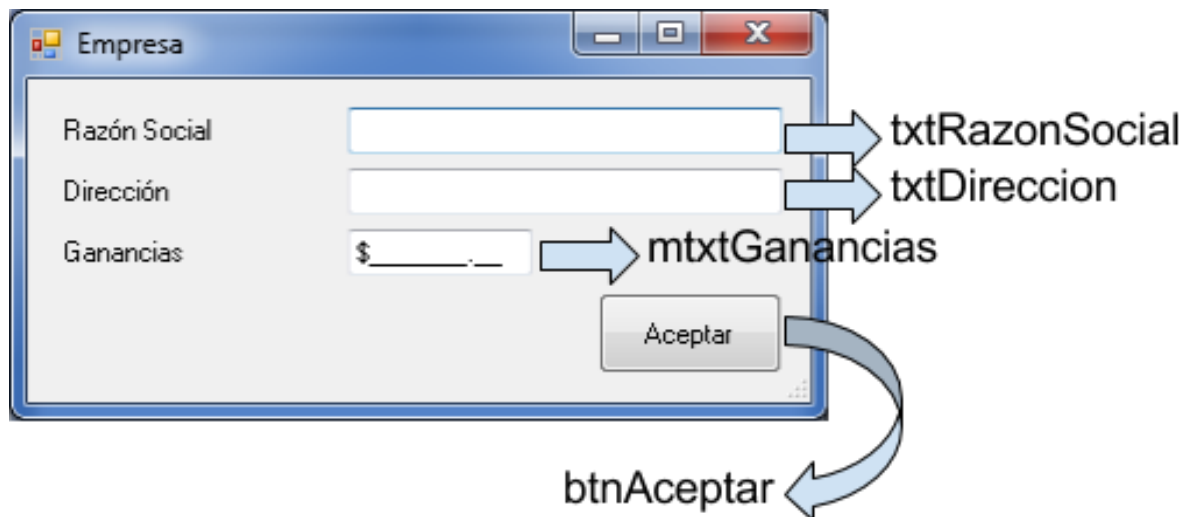
- `_nombre` del tipo string, privado y de instancia
- `_apellido` del tipo string, privado y de instancia
- `_legajo` del tipo string, privado y de instancia
- `_puesto` del tipo `EPuestoJerarquico`, privado y de instancia
- `_salario` del tipo entero, privado y de instancia
- Generar un constructor de instancia para cargar todos estos atributos.
- Sobrecarga del operador `==`. Dos empleados son iguales si comparten el mismo legajo.
- Método `Mostrar()` deberá retornar un string con los datos del empleado.

El enumerado `EPuestoJerarquico` contendrá las opciones:

1. Administración
2. Gerencia
3. Sistemas
4. Accionista

Botones:

- Al pulsar sobre el botón `btnLimpiarForm` se deberá limpiar el contenido de todos los campos del formulario.
- Al presionar el botón `btnAgregarItem` se deberá generar una nueva instancia de la clase `Empleado` y agregarlo a la Empresa. Luego se deberá mostrar la nómina en el campo `rtxtConsola`.
- Al pulsar `btnEmpresa` se deberá abrir el formulario `frmEmpresa` y actualizar los datos de la empresa.



Clase Empresa que tendrá:

- `_nominaEmpleados` del tipo lista de empleados, privado y de instancia.
- `_razonSocial` del tipo string, privado y de instancia. Generar Property.
- `_direccion` del tipo string, privado y de instancia. Generar Property.
- `_ganancias` del tipo float, privado y de instancia. Generar Property.
- Codificar un constructor privado por defecto. Este será el único lugar donde se instanciará la nómina de empleados.
- Codificar otro constructor que reciba como parámetro la razón social, dirección y ganancias de la Empresa.
- Sobrecarga del operador `+` (`Empresa, Empleado`). Si el empleado no existe, lo agrega a la nómina.

- Método `MostrarEmpresa()` : string. Retornará la lista de empleados con el siguiente formato:

La empresa [razón social] sita en la calle [dirección] cuenta con ganancias por [ganancias] y con [cantidad de empleados] empleados:

NOMBRE: [nombre]

APELLIDO: [apellido]

LEGAJO: [legajo]

SALARIO: \$[salario]

¡IMPORTANTE! reemplazar los contenidos entre [] por los datos correspondientes. Reutilizar código.

AYUDA:

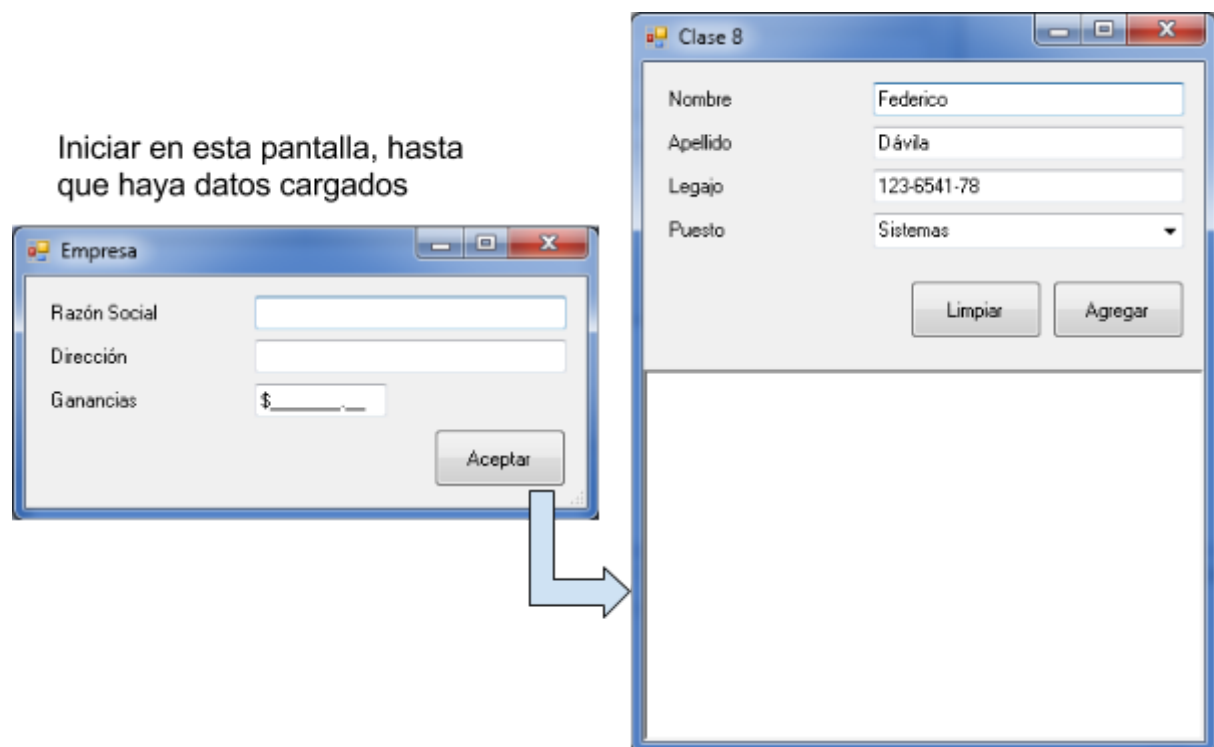
Cargar el combo con un enumerado:

```
cmbPuesto.DataSource = Enum.GetValues(typeof(EPuestoJerarquico));
```

Leer el valor del combo y recuperar un valor del enumerado:

```
Enum.TryParse<EPuestoJerarquico>(cmbPuesto.SelectedValue.ToString(), out valor);
```

Flujo del sistema:



Ejercicios Resueltos

Resolución del Ejercicio [\[DESCARGAR\]](#)

Última modificación: viernes, 12 de abril de 2019, 10:25

Español - Internacional (es)

English (en)

Español - Internacional (es)