

Universidad Tecnológica Nacional
Facultad Regional Avellaneda



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

Materia: Laboratorio de Programación II

Apellido:		Fecha:	10-10-2019
Nombre:		Docente ⁽²⁾ :	
División:	2°C	Nota ⁽²⁾ :	
Legajo:		Firma ⁽²⁾ :	
Instancia ⁽¹⁾ :	<div style="display: flex; justify-content: space-around;"> PP X RPP </div>	<div style="display: flex; justify-content: space-around;"> SP RSP FIN </div>	

(1) Las instancias validas son: 1^{er} Parcial (**PP**), Recuperatorio 1^{er} Parcial (**RPP**), 2^{do} Parcial (**SP**), Recuperatorio 2^{do} Parcial (**RSP**), Final (**FIN**). Marque con una cruz.

(2) Campos a ser completados por el docente.

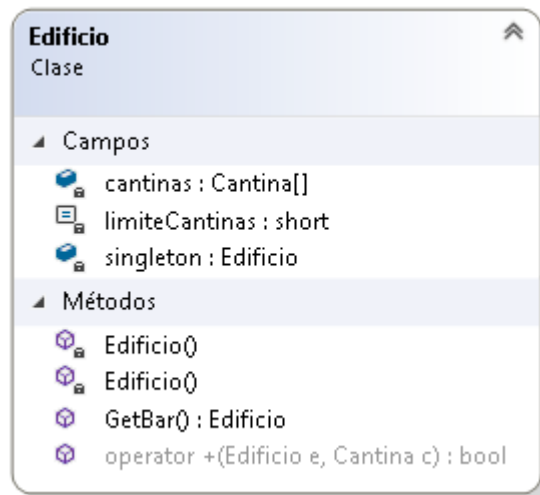
IMPORTANTE:

- Guardar el proyecto en el **disco D:**. Ante un corte de energía o problema con el archivo de corrección, el alumno será responsable de que el proyecto sea recuperable.
- **2 (dos) errores en el mismo tema anulan su puntaje.**
- **Errores de conceptos de POO anulan el punto.**
- **Cada tema vale 1 (un) punto (Herencia, Generics, Test Unitarios, etc.). La correcta documentación también será evaluada.**
- **Se deberán tener al menos el 60% bien de los temas a evaluar según la instancia para lograr la aprobación.**
- Colocar sus datos personales en el nombre del proyecto principal, colocando: Apellido.Nombre.AñoCursada. Ej: Pérez.Juan.2019. No se corregirán proyectos que no sea identificable su autor.
- **Salvo que se indique lo contrario, TODAS** las clases deberán ir en una Biblioteca de Clases llamada Entidades.
- No se corregirán exámenes que no compilen.
- **Reutilizar** tanto código como crean necesario.

Al finalizar, colocar la carpeta de la Solución completa en un archivo ZIP que deberá tener como nombre Apellido.Nombre.AñoCursada.zip y dejar este último en el Escritorio de la máquina. Luego presionar el botón de la barra superior, **colocar un mensaje** y presionar *Aceptar*. **Aguardar a que el profesor indique que el examen fue copiado de forma correcta.** Luego retirarse del aula.

TIEMPO MÁXIMO PARA RESOLVER EL EXAMEN 110 MINUTOS.

1. Agregar dentro de la biblioteca la siguiente clase:



2. Clase **Edificio**:

- No se deberá poder heredar de ella.
- `limiteCantinas` será una constante con valor 2.
- Tendrá un constructor privado que instanciará el array `cantinas` con un tamaño de `limiteCantinas`.
- Tendrá un constructor de clase que inicializará la variable `singleton`.
- `GetBar` retornará `singleton`.
- El operador `+` buscará una posición libre en el array:
 - Si la encuentra, cargará la cantina en ese lugar y retornará `true`.
 - Si se encuentra completo, retornará `false`.

3. Clase **Cantina**:

- Quitar patrón singleton y hacer público su constructor.
- Agregar un operador `==` entre `Cantina` y `Botella`. Dos botellas serán iguales según lo que retorne el `==` entre ellas.

4. Clase **Botella**:

- Quitar sobrescritura de `ToString`. Hacer la misma operación dentro de un conversor implícito.
- Dos botellas serán iguales (`==`) si y sólo si comparten la misma Marca.
- Dentro del método `GenerarInforme` se deberá utilizar `StringBuilder` y `string.Format`.

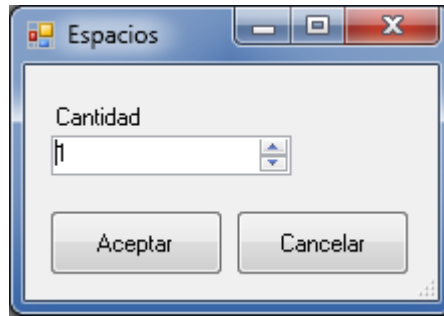
5. En ningún caso se podrá utilizar concatenaciones de texto. O sea, queda prohibido el uso de "**Texto**" + **variable** o similares.

6. Clase **FrmCantina**:

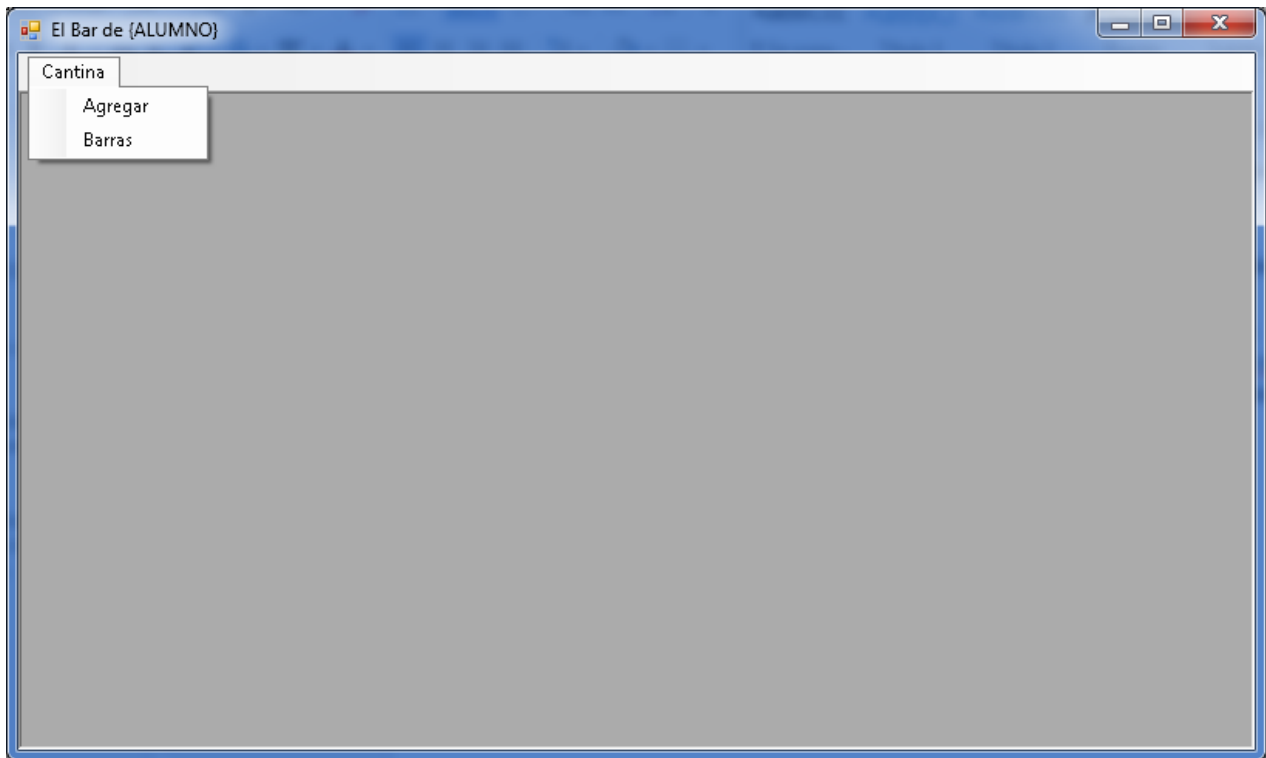
- Crear la propiedad de sólo lectura `GetInforme` que retornará la información de todas las botellas.
- Agregar el evento `FormClosing`. Cancelar el cierre sólo si se cumple la condición `e.CloseReason == CloseReason.UserClosing`.

7. Agregar el formulario `FrmCantidadEspaciosCantina`:

- Deberá iniciar centrado en la pantalla.
- Al presionar `btnCancelar` deberá retornar `DialogResult.Cancel` y al presionar `btnAceptar` `DialogResult.OK`.
- Agregar la propiedad de solo lectura `CantidadEspacios` que retornará el **short** el contenido del `nudCantidad`.



8. Agregar el formulario *FrmBar*:



- a. El formulario será del tipo MDI.
- b. Colocar su nombre en el título.
- c. Implementar el StripMenu como se muestra en la imagen.
- d. Menú Agregar abrirá el *FrmCantidadEspaciosCantina*. Si retorna OK como respuesta:
 - i. Consultar la cantidad de espacios que se cargó en *FrmCantidadEspaciosCantina*.
 - ii. Instanciará un nuevo *FrmCantina*.
 - iii. Si la operación `this.edificio + frmCantina.GetCantina` retorna true:


```
frmCantina.MdiParent = this;
frmCantina.Show();
```
- e. Menú Ventas utilizará la propiedad `this.MdiChildren` para llamar al método *GetInforme* de cada *FrmCantina* instanciado.