

**Universidad Tecnológica Nacional  
Facultad Regional Avellaneda**



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

**Materia: Laboratorio de Programación II**

Apellido:		Fecha:	25/10/2018
Nombre:		Docente <sup>(2)</sup> :	F. Dávila/H.Dillon
División:	2°C	Nota <sup>(2)</sup> :	
Legajo:		Firma <sup>(2)</sup> :	
Instancia <sup>(1)</sup> :	PP	RPP	X
		SP	
		RSP	
		FIN	

(1) Las instancias validas son: 1<sup>er</sup> Parcial (PP), Recuperatorio 1<sup>er</sup> Parcial (RPP), 2<sup>do</sup> Parcial (SP), Recuperatorio 2<sup>do</sup> Parcial (RSP), Final (FIN). Marque con una cruz.

(2) Campos a ser completados por el docente.

**IMPORTANTE:**

- **2 (dos) errores en el mismo tema anulan su puntaje.**
- Colocar nombre de la clase (en estáticos), **this** o **base** en todos los casos que corresponda.
- **La correcta documentación y reglas de estilo de la cátedra serán evaluadas.**
- Colocar sus datos personales en el nombre del proyecto principal, colocando: Apellido.Nombre.Departamento. Ej: Pérez.Juan.2D. No se corregirán proyectos que no sea identificable su autor.
- De no aclararse otra cosa, **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.
- De ser proporcionado un proyecto de Biblioteca de Clases, este es un esquema; controlar y modificar lo necesario.

*Libro*

1. El constructor privado será el único encargado de inicializar la lista.
  2. La propiedad TipolImpresion será abstracta.
  3. El método Mostrar será Virtual y retornará el título del Libro en cuestión.
  4. Sólo en la propiedad TamanoLetra, al asignar un nuevo tamaño, se deberá validar que este sea menor al 10% de la suma del ancho y alto de la hoja. Caso contrario, se asignará un tamaño de letra 12.
  5. Siempre se deberá validar el tamaño de la letra antes de asignarlo (la forma de validación es la del punto 4).
  6. El indexador retornará la página actual, siempre y cuando esta exista; sino retornará una vacío ("").
- En su set, agregará una nueva página a la lista de páginas.

*Novela*

7. El TipolImpresion será BlancoNegro.
8. Mostrar retornará el título, la cantidad de páginas y el tipo de impresión. Utilizar String.Format.
9. Dos novelas serán iguales si comparten todos los datos retornados por Mostrar.

*Comic*

10. El TipolImpresion será Color.
11. Mostrar retornará el título, la cantidad de cuadros y el tipo de impresión. Utilizar String.Format.

*Biblioteca*

12. El constructor estático inicializará la cantidad en 10.
13. El constructor privado será el único encargado de inicializar la lista.

14. El operador + agregará un nuevo libro a la lista, siempre y cuando haya lugar y este no se encuentre cargado previamente.
15. El operador explicito retornará la lista de libros de la biblioteca.

### FrmLector

16. Al final del evento FrmLector\_Load, agregar el código necesario para cargar el ComboBox con la información de los libros (utilizar su método Mostrar).
17. En el método SetPagina, agregar la línea de código para que la página se muestre en el RichTextBox llamado rtbPagina donde se encuentra el comentario que dice " // Muestro la página actual".

### Console

18. Agregar una Vista de Consola que tenga 1 libro instanciado (copiarlo de la clase Mock) y un menú que pregunte que página se desea abrir. Elegir una página y mostrarla.

### Diagrama de clases

19. Las clases Libro y sus derivadas pertenecerán al Namespace EntidadesLibro y la clase Biblioteca irá en el Namespace EntidadesBiblioteca.

