

# Universidad Tecnológica Nacional

## Facultad Regional Avellaneda



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

**Materia: Laboratorio de Programación II**

Apellido:		Fecha:	28/06/2018
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	SP X 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.**
- **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 explicitarse nada, 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.
- Colocar nombre de la clase (en estáticos), **this** o **base** en todos los casos que corresponda.

---

*TIEMPO MÁXIMO PARA RESOLVER EL EXAMEN 90 MINUTOS.*

---

Se intentará armar una base para un software de votación en la Cámara de Senadores.

1. Partir del proyecto entregado. Modificar el nombre de la carpeta y la solución con el siguiente formato: [APELLIDO].[NOMBRE].
2. Las excepciones propias irán en un nuevo proyecto llamado Excepciones.
3. El resto de las clases propuestas irán en un proyecto con nombre Entidades.
4. Dentro de la clase Votacion agregar las condiciones para que pueda:
  - 4.1. Ser serializada en XML.
  - 4.2. Invocar un evento (con nombre de evento EventoVotoEfectuado y de delegado Voto) que pueda ser manejado por el método ManejadorVoto de la clase FrmSenador.
5. Dentro del método Simular invocar el evento creado anteriormente bajo el comentario `// Invocar Evento.`
6. En el manejador del evento Click del botón btnSimular enlazar el evento de la votación con su manejador en el formulario, y lanzar el método Simular de votacion en un nuevo Thread.
7. Crear una Interfaz llamada IArchivos con los métodos:
  - 7.1. T Leer(string rutaArchivo)
  - 7.2. bool Guardar(string rutaArchivo, T objeto)
8. Implementar dicha interfaz en
  - 8.1. Una clase SerializarXML:
    - 8.1.1. Leer retornará un elemento del tipo T leído desde el archivo ubicado en rutaArchivo.
    - 8.1.2. Guardar serializará en XML el objeto dado.

- 8.1.3. Ante cualquier error en cualquiera de los dos métodos, lanzará la excepción propia `ErrorArchivoException` con el mensaje correspondiente.
- 8.2. Otra clase Dao:
  - 8.2.1. Leer lanzará la excepción del sistema `NotImplementedException`.
  - 8.2.2. Guardar guardará los resultados de la votación en la base de datos dada.
  - 8.2.3. Ante cualquier error, relanzar la excepción capturada.
9. Dentro de `ManejadorVoto`:
  - 9.1. Reemplazar donde dice `NOMBRE_EVENTO` con el nombre del delegado de Votacion.
  - 9.2. Debajo del comentario `// Guardar resultados` guardar los resultados de la votación tanto en la Base de Datos como en XML.
10. Test Unitarios:
  - 10.1. Realizar un test que compruebe que si hay un error al querer serializar un objeto del tipo `Votacion` lance la excepción `ErrorArchivoException`.
  - 10.2. Realizar un test que compruebe que el evento de la clase `Votacion` tantas veces como Senadores haya. O sea, si hay 2 senadores el evento será invocado 2 veces.

Al finalizar, colocar la carpeta de la Solución completa en un archivo ZIP que deberá tener como nombre `Apellido.Nombre.division.zip` y dejar este último en el Escritorio de la máquina.

Luego presionar el botón  de la barra superior, colocar un mensaje y apretar **Aceptar**.

Finalmente retirarse del aula y aguardar por la corrección.