

## Agustín Nicolas Fernández 2.C

### TP 3 Documento.

El programa realiza la carga de 2 tipos de dispositivos (Celulares y notebooks) en una fábrica, también puede quitar los dispositivos y ver su stock. Además, tiene 2 funcionalidades de guardado de archivos en formato .txt y .XML.

#### Clase 15 - Excepciones:

Las Excepciones las use en la clase "Excepciones" y cree 2 tipos, una para archivos (ArchivosException) y otra para Dispositivos (DispositivoRepetidoException).

**ArchivosException:** Es Lanzada en caso de que haya algún problema en la lectura/escritura de algún archivo sea el formato que sea.

**DispositivoRepetidoException:** Esta excepción es lanzada en la sobrecarga de operadores "+" cuando trata de agregar un dispositivo a la lista. Si el dispositivo esta repetido se lanza la excepción.

#### Clase 16 – Test Unitario:

Realice 2 test unitarios en el proyecto TestUnitarios.

**TestProductoRepetidoException():** En este test verifico que se lance la excepción "**DispositivoRepetidoException**" al agregar un dispositivo repetido.

**TestVerificarListaNula():** En este test verifico que la lista de dispositivos de la fábrica sea diferente a nula.

### Clase 17 – Tipos Genéricos:

Se utiliza en la interfaz **IArchivo** para permitir serializar objetos de cualquier tipo, y así crear métodos genéricos de lectura/escritura.

### Clase 18 – Interfaces:

Las interfaces las use para crear 2 métodos de lectura/escritura de archivos, implementada por las clases **XML** y **ArchivoTxt**.

### Clase 19 – Archivos y Serialización:

En el proyecto Archivos, están las 2 clases encargadas de leer/escribir archivos de texto y en XML.

En la clase fabrica, cree un método estático "Guardar" que recibe una fábrica por parámetro y devuelve true si se pudo guardar el archivo en XML.

En cambio, para crear un archivo del tipo texto tengo que instanciar la clase porque no le genere otro método a la fábrica para los archivos de texto.