

## PROGRAMACION ORIENTADA A OBJETOS

### ARCHIVOS Y EXCEPCIONES

#### Tareas a cumplir:

- Se proporcionan archivos de productos por categorías, y un arreglo de categorías en la clase principal. Debe crear una lista de tipo Productos que sea llenada con cada producto de los archivos de cada categoría.
- Los archivos tienen líneas invalidas que no deberán ser tomadas en cuenta. Adicional si el stock es negativo debe lanzarse una excepción personalizada Stock Negativo. **○ NO PUEDE MODIFICAR LOS ARCHIVOS**

La estructura del archivo tiene código, nombre separado por comas (,) precio y stock separado por (;)

`codigo,nombre,precio;stock`

- Debe permitir al usuario ingresar ventas hasta que el usuario no desee continuar se proporciona un archivo base. Se proporciona un método base ingresarVenta
- Debe crear una clase venta y almacenar la información de las ventas en una lista de tipo Venta. Esta lista debe ser serializada una vez que el usuario decida no crear mas ventas. Crear métodos adecuados.
- Por cada venta el stock del producto debe reducirse de acuerdo a la cantidad
- La lista de ventas de ser deserializada cada vez que se inicie el programa. • La clase venta debe tener como versión el valor de 1

#### Ejemplo de ejecución del programa

1era. Ejecución (No existe aun las ventas serializadas)

Total de Ventas Acumuladas: 0

=====

Codigo Producto: 30

Nombre Producto: Refrigeradora Mabe

Precio: 400

Categoria: electrodomesticos

Cantidad: 50

No hay stock suficiente

Cantidad: 5

Total a Pagar: 2000

Desea ingresar otra venta? S/N

N

2da. Ejecución (Una vez serializada la lista se presenta como van las ventas):

Total de Ventas Acumuladas: 2000

=====

Codigo Producto: 22

Nombre Producto: laptop asus n1100

Precio: 1600

Categoria: computadoras

Cantidad: 1

Total a Pagar: 1600

Desea ingresar otra venta? S/N

N

**NOTA: TODO LO MARCADO CON AZUL ES LA INFORMACION QUE EL SISTEMA DEBE IMPRIMIR, LO NEGRO ES LAS INTERACCIONES CON EL USUARIO**