

Tema

Actividad #2: Sistema de Gestión de
Inventario Avanzado

Asignatura

Programación Estructurada Y Fundamental

Presenta

José Calle, Mateo Calvopiña

Fecha

27/11/2025

Practica de Programación

Sistema de Gestión de Inventario Avanzado

Objetivo General

Desarrollar un programa capaz de gestionar el registro, cálculo de impuestos, control de categorías, validación de datos, manejo de riesgo tecnológico y generación de reportes avanzados. Los estudiantes trabajarán en grupos de tres personas y competirán para lograr el sistema más completo, funcional y robusto.

Enunciado del Sistema

La empresa TecnoLogix Global necesita un sistema avanzado para administrar los productos tecnológicos que ingresan diariamente a su inventario. Existen impuestos diferenciados según categoría, control de riesgo tecnológico y tasas especiales dependiendo del día de operación. El sistema debe registrar productos en base a un límite diario y generar un reporte detallado del cierre de operaciones.

Descripción del código

```
import java.util.Scanner;
public class class271125 {
    public static void main(String[] args) {

        Scanner sc= new Scanner(System.in);

        System.out.println("Cuantos productos vas a pedir hoy:");
        int limite = sc.nextInt();

        while (limite <= 0) {
            System.out.println("Ese numero no sirve, pon uno mayor a cero:");
            limite = sc.nextInt();
        }
    }
}
```

En esta parte creo el programa principal, preparo el Scanner para leer datos del teclado y pido cuántos productos voy a registrar. También valido que el número sea mayor que cero, repitiendo la pregunta hasta que el usuario ingrese un valor correcto.

```
System.out.println("Ingresa el dia (1 lunes 2 martes 3 miercoles 4 jueves 5 viernes 6 sabado 7 domingo):");
int dia = sc.nextInt();

while (dia < 1 || dia > 7) {
    System.out.println("Ese dia no existe, solo del 1 al 7 pon:");
    dia = sc.nextInt();
}
```

```

int c = 0;

float sumaiva = 0;
float sumaimc = 0;
float sumairt = 0;
float sumatce = 0;
float sumatotal = 0;

int cat1 = 0;
int cat2 = 0;
int cat3 = 0;
int cat4 = 0;

String nombreMayor = "";
float precioMayor = 0;

```

Aquí pido el día de la semana y verifico que esté entre 1 y 7. Después inicializo el contador de productos, los acumuladores de impuestos y ventas, los contadores de cada categoría y las variables que guardan el nombre y el precio del producto más caro del día.

```

while (c < limite) {

    System.out.println("escribe el nombre del producto:");
    String nombre = sc.nextInt();

    System.out.println("pon el precio del producto:");
    float precio = sc.nextFloat();

    while (precio <= 0) {
        System.out.println("no puedes poner cero o negativo, pon un precio bien:");
        precio = sc.nextFloat();
    }
}

```

En este while recorro todos los productos que se van a ingresar. Para cada uno pido el nombre y el precio, y verifico que el precio no sea cero ni negativo, obligando al usuario a escribir un valor válido.

```

    precio = sc.nextFloat();
}

System.out.println("elige la categoria (1 electronica 2 componentes 3 accesorios 4 redes):");
int categoria = sc.nextInt();

while (categoria < 1 || categoria > 4) {
    System.out.println("esa categoria no hay, pon 1 2 3 o 4:");
    categoria = sc.nextInt();
}

System.out.println("tiene riesgo? (1 si 0 no):");
int riesgo = sc.nextInt();

```

```

while (_riesgo != 0 && _riesgo != 1) {
    System.out.println("solo va 1 o 0:");
    _riesgo = sc.nextInt();
}

System.out.println("cuanto stock tiene (minimo 1):");
int _stock = sc.nextInt();

while (_stock < 1) {
    System.out.println("no se puede menos de 1, pon de nuevo:");
    _stock = sc.nextInt();
}

```

En esta parte pido la categoría, si el producto tiene riesgo y el stock disponible. En cada caso uso bucles while para asegurarme de que los datos estén dentro de los rangos permitidos, evitando valores inválidos.

```

float _iva = 0;
if (_categoria != 3) {
    _iva = _precio * 0.12f;
}

float _imc = _precio * 0.015f;

float _irt = 0;
if (_riesgo == 1) {
    _irt = _precio * 0.037f;
}

```

```

float _tce = 0;
if (_dia == 6 || _dia == 7) {
    _tce = _precio * 0.02f;
}

float _preciofinal = _precio + _iva + _imc + _irt + _tce;

```

Aquí calculo los distintos recargos del producto: el IVA solo si no es categoría 3, el IMC siempre, el IRT cuando el producto es de riesgo y el TCE si estamos en fin de semana. Al final sumo todo para obtener el precio final del producto.

```

String nombrecategoría = "";
if (categoria == 1) {
    nombrecategoría = "electronica";
    cat1 = cat1 + 1;
} else if (categoria == 2) {
    nombrecategoría = "componentes";
    cat2 = cat2 + 1;
} else if (categoria == 3) {
    nombrecategoría = "accesorios";
    cat3 = cat3 + 1;
} else if (categoria == 4) {
    nombrecategoría = "redes";
    cat4 = cat4 + 1;
}

```

En este bloque traduzco el número de categoría a un nombre en texto y aumento el contador correspondiente. De esta forma voy registrando cuántos productos se ingresan en cada tipo.

```

System.out.println("precio: " + precio);
System.out.println("iva: " + iva);
System.out.println("imc: " + imc);
System.out.println("irt: " + irt);
System.out.println("tce: " + tce);
System.out.println("precio final: " + preciofinal);
System.out.println("categoria: " + nombrecategoría);
System.out.println("stock: " + stock);

```

Aquí muestro en pantalla todos los detalles del producto actual: su precio base, cada impuesto por separado, el precio final calculado, la categoría en texto y la cantidad de stock.

```

float ivaTotalProd = iva * stock;
float imcTotalProd = imc * stock;
float irtTotalProd = irt * stock;
float tceTotalProd = tce * stock;
float totalProd = preciofinal * stock;

sumaiva = sumaiva + ivaTotalProd;
sumaimc = sumaimc + imcTotalProd;
sumairt = sumairt + irtTotalProd;
sumatce = sumatce + tceTotalProd;
sumatotal = sumatotal + totalProd;

```

En esta parte multiplico los valores por el stock para saber cuánto se recauda en total por ese producto. Luego voy sumando esos valores a los acumuladores generales del día, tanto para los impuestos como para el total de ventas.

```
    if (preciofinal > precioMayor || c == 0) {
        precioMayor = preciofinal;
        nombreMayor = nombre;
    }

    c = c + 1;
}
```

Aquí comparo el precio final del producto con el más alto registrado hasta ahora. Si es el primero o es más caro, actualizo el producto más caro. Al final incremento el contador c para pasar al siguiente producto en el bucle.

```
int maxcat = cat1;
int catMayor = 1;

if (cat2 > maxcat) {
    maxcat = cat2;
    catMayor = 2;
}
if (cat3 > maxcat) {
    maxcat = cat3;
    catMayor = 3;
}
if (cat4 > maxcat) {
    maxcat = cat4;
    catMayor = 4;
}
```

Después de ingresar todos los productos, en este bloque comparo los contadores de cada categoría para encontrar cuál tiene más registros. Guardo tanto la cantidad máxima como el número de la categoría que ganó.

```
String nombreCatMayor = "";
if (catMayor == 1) {
    nombreCatMayor = "electronica";
} else if (catMayor == 2) {
    nombreCatMayor = "componentes";
} else if (catMayor == 3) {
    nombreCatMayor = "accesorios";
} else if (catMayor == 4) {
    nombreCatMayor = "redes";
}
```

Aquí convierto el número de la categoría ganadora en un nombre en texto. Esto me permite mostrar al final de forma clara cuál fue la categoría que más se registró durante el día.

```
System.out.println("resumen del dia:");
System.out.println("total productos: " + c);
System.out.println("total iva: " + sumaiva);
System.out.println("total imc: " + sumaimc);
System.out.println("total irt: " + sumairt);
System.out.println("total tce: " + sumatce);
System.out.println("total ventas: " + sumatotal);
System.out.println("categoria que mas se registro: " + nombreCatMayor);
System.out.println("producto mas caro al final: " + nombreMayor + " con " + precioMayor);
}
}
```

En la última parte muestro un resumen general del día: el total de productos ingresados, lo recaudado por cada impuesto, el total de ventas, la categoría con más registros y el nombre del producto más caro junto con su precio final.