Nombres:

- Samuel Leonardo Acero Lozano
- Juan Diego Muñoz

Contexto:

La tienda de mascotas "Huellitas Felices" ha sido un referente en la comunidad por su amplio surtido de productos para perros y gatos. Sin embargo, con el tiempo, ha empezado a enfrentar serios problemas en la gestión de su inventario. Actualmente, el control de los productos se realiza de manera manual, utilizando libretas y hojas de cálculo que no están sincronizadas ni actualizadas en tiempo real. Esto ha llevado a múltiples inconvenientes, como la venta de productos vencidos, falta de stock de artículos esenciales y acumulación de mercancía innecesaria en la bodega.

Los empleados de la tienda dedican horas cada semana a revisar el inventario físicamente, lo que no solo consume mucho tiempo, sino que también aumenta la posibilidad de errores humanos. En varias ocasiones, clientes han adquirido comida para mascotas próxima a vencer sin ser advertidos, lo que ha generado quejas y afectado la reputación de la tienda. Además, la falta de un sistema estructurado impide a los administradores tomar decisiones informadas sobre qué productos deben reabastecerse con prioridad y cuáles necesitan una promoción para evitar su desperdicio.

Para solucionar estos problemas, "Huellitas Felices" busca implementar un sistema digital que automatice la gestión del inventario, asegurando un control eficiente y en tiempo real sobre todos los productos disponibles en la tienda.

Definición del problema:

El desorden en la gestión del inventario de "Huellitas Felices" ha generado múltiples problemas operatIVos y financieros. Entre los principales inconvenientes se encuentran:

1. Productos vencidos en venta: La tienda no cuenta con un sistema que notifique cuándo un producto está próximo a caducar, lo que ha ocasionado la venta involuntaria de artículos vencidos, afectando la salud de las mascotas y la confianza de los clientes.

2. Falta de control sobre el stock: No existe un registro actualizado en tiempo real de los productos disponibles. En varias ocasiones, los empleados han prometido productos a los clientes que en realidad ya estaban agotados.
3. Gestión manual ineficiente: El uso de libretas y hojas de cálculo ha demostrado ser una solución poco efectIVa, ya que requiere demasiado tiempo y está propensa a errores humanos.
4. Acumulación de productos innecesarios: La falta de un sistema de control ha generado la compra excesIVa de productos que no tienen alta demanda, provocando pérdidas económicas y desperdicio de mercancía.
5. Falta de alertas de productos próximos a vencer: No hay un mecanismo que ayude a identificar rápidamente qué productos deben venderse con prioridad o donarse antes de que se conviertan en pérdidas.
6. Decisiones de compra poco informadas: Sin un historial de ventas bien estructurado, los administradores no pueden tomar decisiones estratégicas sobre qué productos reabastecer con mayor prioridad.

Documentación del Sistema de Inventario para Tienda de Mascotas

---

1. Interfaces

1.1. Caducable

I. Nombre de la clase: Caducable

II. Descripción de su función en el sistema: Interfaz para productos que tienen fecha de caducidad.

III. Métodos:

- Date getFechaCaducidad(): Devuelve la fecha de caducidad del producto.
- boolean estaPorVencer(): Verifica si el producto está próximo a vencer.

1.2. NoCaducable

I. Nombre de la clase: NoCaducable

II. Descripción de su función en el sistema: Interfaz para productos que no tienen fecha de caducidad.

III. Método por defecto:

- void mostrarInformacionDeCaducidad(): Indica que el producto no caduca.

---

2. Clases

2.1. Producto

I. Nombre de la clase: Producto

II. Descripción de su función en el sistema: Clase abstracta base para todos los productos de la tienda.

III. Atributos:

- String nombre: Nombre del producto.
- double precio: Precio del producto.
- int cantidad: Cantidad disponible en inventario.

IV. Métodos:

- Constructores y métodos getter/setter.
- abstract void mostrarInformacion(): Debe ser implementado en subclases.

2.2. Juguete

I. Nombre de la clase: Juguete

II. Descripción de su función en el sistema: Representa los juguetes para mascotas.

III. Atributos:

- Hereda de Producto.

IV. Métodos:

- mostrarInformacion(): Muestra detalles del juguete e indica que no tiene caducidad.

2.3. Comida

I. Nombre de la clase: Comida

II. Descripción de su función en el sistema: Representa productos alimenticios con fecha de caducidad.

III. Atributos:

- Date fechaCaducidad: Fecha de vencimiento del producto.

IV. Métodos:

- getFechaCaducidad(): Devuelve la fecha de caducidad.
- estaPorVencer(): Retorna true si faltan 10 días o menos para caducar.
- mostrarInformacion(): Muestra detalles y advierte si está por vencer.

2.4. Medicamento

I. Nombre de la clase: Medicamento

II. Descripción de su función en el sistema: Representa medicamentos con fecha de vencimiento.

III. Atributos:

- String nombreProfesional: Nombre del componente actIVo.
- Date fechaCaducidad: Fecha de vencimiento.

IV. Métodos:

- getFechaCaducidad(): Devuelve la fecha de caducidad.
- estaPorVencer(): Retorna true si faltan 7 días o menos.
- mostrarInformacion(): Muestra detalles y advierte si está próximo a vencer.

2.5. Inventario

I. Nombre de la clase: Inventario

II. Descripción de su función en el sistema: Gestiona la lista de productos en la tienda.

III. Atributos:

- List<Producto> productos: Lista de productos en inventario.

IV. Métodos:

- agregarProducto(Producto nuevoProducto): Agrega un producto, sumando cantidades si ya existe.
- mostrarInventario(): Lista todos los productos en inventario.
- mostrarCantidadPorCategoria(): Muestra la cantidad de productos por tipo.
- mostrarProductosPorVencer(): Lista productos que están por caducar.

2.6. Main

I. Nombre de la clase: Main

II. Descripción de su función en el sistema: Punto de entrada del programa. Ejecuta la lógica principal de creación y gestión del inventario.

III. Atributos: (No tiene atributos, solo contiene el método main.)

IV. Métodos:

- main(String[] args): Ejecuta la inicialización del inventario y muestra la información de los productos.