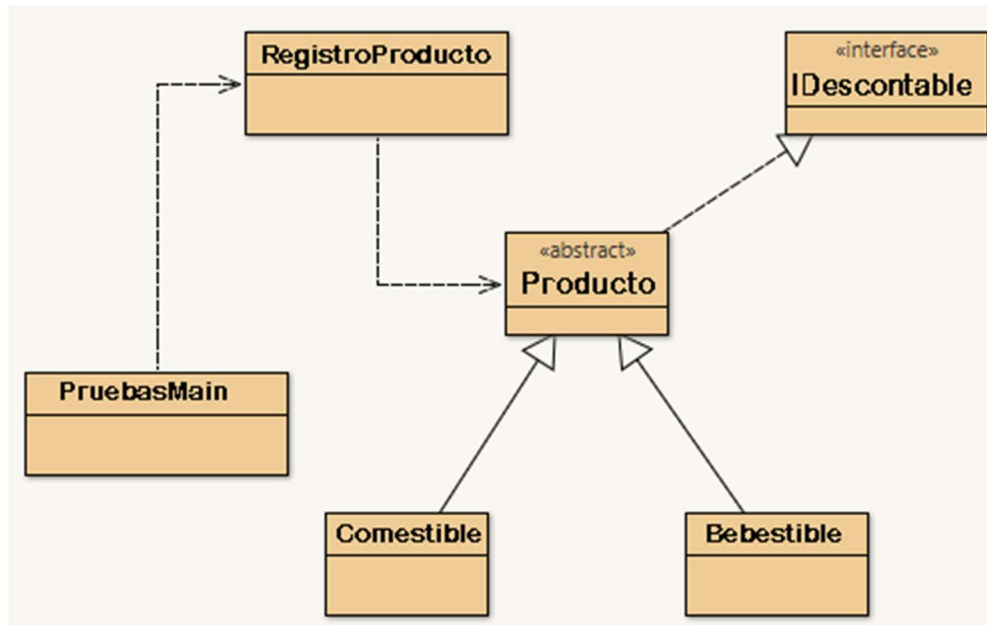


Enunciado

Se desea construir una aplicación que permita registrar los productos que a “FOOD TRUCK” para poder mostrar a sus posibles comensales los productos disponibles. Para esto se solicita su participación dentro del equipo de desarrollo y así permitir dar solución a este requerimiento. El jefe de proyecto ha generado el diagrama de clases que le corresponde implementar en Java.



A continuación, se describen los atributos a definir en las clases Producto, Comestible y Bebestible, donde correspondan:

Se sabe que los productos tienen código, descripción, precio. Los comestibles indican si incluye en porción de papas fritas de regalo (true/false) y los bebestibles los grados de alcohol).

La interface Descontable tiene:

- descuentoEfectivo: 20%
- descuentoComestible: 10%
- descuentoBebestible: 5%

Los métodos de la interfaz son:

- a) obtenerCompra: calcula el precio que deberá pagar el cliente por la compra del producto, que es el precio por la cantidad de unidades a comprar. Esta cantidad se ingresa por el usuario del sistema.
- b) descontar: calcula el descuento que se aplica sobre el valor calculado en el método anterior. Se debe calcular el descuento solicitando forma de pago y las unidades a comprar.

- c) `calcularTotal`: calcula el total a pagar considerando los descuentos y la propina del 10% sobre el total calculado, debiendo recibir la forma de pago y las unidades a comprar

La clase `Producto` debe tener un método abstracto `imprimirBoleta` que reciba la forma de pago y las unidades a comprar para que retorne el resumen del producto en cuestión (`COMESTIBLE` o `BEBESTIBLE`), descripción, precio y si es un comestible debe agregar si incluye porción de papas fritas. Si es bebestible, se debe agregar la cantidad de grados de alcohol. A modo de ejemplo el método debe retornar:

“COMESTIBLE: Sandwich Italiano, \$2900, incluye porción de papas”

O

“BEBESTIBLE: Lemon Stone, \$1200, 2,5% alcohol”

La clase `Comestible` no tiene subclases.

Además, deben existir los siguientes métodos en las clases:

- constructores, sobrecargando el método
- accesadores
- mutadores considerando las reglas de negocio
- `toString` que retorne los atributos de todas las clases sobrescribiendo el método.

Debe existir una clase `RegistroProducto` que permita:

- Guardar en una colección todos los `Productos` (`Comestible` o `Bebestible`) validando que no exista el código.
- Mostrar en pantalla una lista con todos los productos y sus atributos.
- Eliminar un `Producto` por su código.
- Obtener la cantidad de `Productos` que son económicos, que su precio sea inferior a \$2000

Desde la clase que contiene el main debe:

- Agregar un `Producto Comestible` y un `Bebestible` a la colección.
- Debe mostrar el total final para `Bebestible`.
- Obtener la cantidad de productos económicos.
- Eliminar un producto