Prueba técnica - Arturo Morcillo

Detalle del ticket

Arturo Morcillo Penares amorcillop@gmail.com

Prueba técnica
Madrid, España
Entregado el 7 de enero de 2022

Índice

1. Introducción

Tenemos una prueba que consiste en implementar una funcionalidad para tickets. Tenemos que poder diferenciar entre los diferentes objetos, la cantidad y los impuestos que se les aplican.

1.1. Consideraciones

Las consideraciones son:

- Tenemos productos normales. A estos productos se les aplica un impuesto del 10
- Tenemos productos *especiales* que pueden ser comida, medicamentos o libros. A estos productos no se le aplica el impuesto del 10
- Todos los productos importados tienen un impuesto extra del 5 %.

2. Análisis

Para llevar a cabo este problema se han creado tres clases. Una clase Ticket, una clase Product (con su clase especializada SpecialProduct) y una clase LineSolver.

2.1. Ticket

La clase Ticket posee un Map con el producto y las veces que este aparece, ya que si un producto aparece dos veces se cobrará dos veces. Este se implementa con un LinkedHashMap para devolver los productos ordenados como se muestra en los ejemplos.

Como funciones tenemos:

- public void addProduct: Para añadir productos.
- String processTicket: Devuelve la solución al problema.

2.2. Product

La clase Product almacena el nombre del producto, su precio, apariciones y si es o no importado.

Como funciones tenemos:

- public double getTaxes(): Para obtener sus impuestos.
- public double getTotalPrice(): Para obtener su coste total.

2.2.1. SpecialProduct

Consiste en una especialización que no tiene en cuenta el 10 % de los impuestos de base.

2.3. LineSolver

Clase con funciones estáticas que resuelven las líneas que se leen.

3. Desarrollo

Una vez se diseñaron las clases el desarrollo fue bastante sencillo ya que toda la codificación resulta bastante intuitiva. Sin embargo, hay determinadas decisiones de diseño que sí resultan interesantes.

3.1. Uso de un LinkedHashMap en Ticket

Como ya he mencionado esto permite pasar por las claves (productos) en el orden en que se han añadido.

3.2. Diferenciación de productos importados

El booleano de los productos importados se marca como true si se encuentra la palabra 'importado' o similar en el nombre. Esto es así porque que un producto sea o no importado se puede dar en todos los casos. Debido a esto no vale la pena poner los productos importados como una clase a parte.

3.3. Diferenciación de los productos especiales

Se han creado tres funciones:

- private static boolean isProductABook(String productName)
- private static boolean isProductFood(String productName)
- private static boolean isProductMedical(String productName)

Estas tres funciones son iguales y buscan palabras diferenciadoras en los nombres de los productos. Se ha optado por esta decisión de diseño porque es lo que mejor se adecua al problema presentado y para permitir diferenciar los tres tipos en un futuro.

Como solución alternativa se plantearía diferenciar estos productos en la entrada o disponer de una base de datos que almacene las diferencias. También puede haber una API en una apliación externa que nos diga el "tipo"de producto que tenemos.

4. Testing

Para testear la aplicación primero probaría los ejemplos proporcionados.

Una vez funcionan pondría casos mezclados con objetos importados y especiales para tener en cuenta los 8 casos posibles:

1. libro importado.
2. libro no importado.
3. medicina importada.
4. medicina no importada.
5. comida importada.
6. comida no importada.
7. otros importados.
8. otros no importados.

Por último algunos ejemplos donde se compre más de una unidad de cada producto. Aunque esto no se contempla en los ejemplos es algo que podría ocurrir.

5. Problemas encontrados y conclusión

El problema no resulta complicado pero hay casos de ejemplo mal resueltos:

5.1. Error en el ejemplo de la entrada 2

Si vemos la entrada 2 vemos que tiene un frasco de perfume importado. Sus impuestos deberían ser de un 15 % sobre su precio de 47.5. Es decir 47.5 * 1.15 = 54.63 redondeando pero el resultado mostrado es de 54.65.

5.2. Error en el ejemplo de la entrada 3

En la entrada 3 vemos que tiene una caja de bombones importados. Sus impuestos deberían ser de un 5 % sobre su precio de 11,25. Es decir 11.25 * 1.05 = 11.81 redondeando pero el resultado mostrado es de 11.85.

De la prueba me ha parecido interesante el hacerme encontrar a mi solo un algoritmo. No ha sido muy difícil pero he tenido que darle unas vueltas. Ilustro algunas ideas descartadas para ver el orden de razonamiento. El realizar tests ha servido para ver algún problema con los casos límite ya que al principio no funcionaba con matrices completamente vacías.