

Problema:

El dueño de una tienda necesita un sistema que le permita administrar su negocio. Para ello, el sistema deberá contar con una opción que le permita registrar sus productos, dar de baja productos, listar productos perecederos, listar productos no perecederos, registrar las compras que realiza cada cliente y mostrar el total de ganancia que se lleva en total. Los productos perecederos a diferencia de los no perecederos tienen una fecha de caducidad. Cada producto tiene nombre, precio y el porcentaje sobre el precio del producto que será la ganancia que obtendrá por la venta de ese producto (si el porcentaje es 10% y el producto se vendió el \$100.00, entonces \$10.00 de esos \$100.00 son la ganancia). Como los productos pueden variar en precio, entonces el sistema deberá permitir la modificación del precio y el porcentaje de ganancia, el nombre no deberá cambiar.

Para hacer una compra, se deberá buscar el producto en una lista, arreglo o colección, si se encuentra se calcula la ganancia sobre ese producto. La búsqueda de los productos se deberá realizar por nombre. Las ganancias de los productos vendidos se deberán ir almacenando en una lista, arreglo o colección, un dato (ganancia) por cada producto vendido.

Cuando el programa inicie su ejecución se deberá cargar, desde un archivo binario, en memoria en una lista, arreglo o colección, los productos tanto perecederos como no perecederos (no leer el archivo de ganancias en memoria, cada vez que se ejecute el programa se generará una nueva lista). Y sobre esta lista se deberán realizar todas las operaciones con respecto a los productos. Cuando el programa termine su ejecución, la lista de productos, si ha sido modificada, se deberá almacenar en el archivo binario. Con respecto a la lista de ganancias, cuando termine el programa, la lista de ganancias generada se deberá agregar al final del archivo de texto de ganancias, esto es, las ganancias se deberán almacenar en un archivo de texto.

Solución:

Para solucionar el problema se creó una clase abstracta a la cual llamamos “Producto” esta tendrá todas las características generales de los productos de la tienda y heredarán a perecederos y no perecederos.

En perecederos lo que tenemos es la agregación de una fecha de caducidad como un atributo extra a los que ya tiene la clase abstracta y en la clase no perecederos tenemos los mismos atributos de la clase padre sin variación.

Posteriormente se creó una clase con el nombre “Negocio” la cual tiene como atributos un `ArrayList<Producto>` en donde se guardarán los productos dados de alta y un `ArrayList<Double>` en donde se guardarán las ganancias obtenidas en el día, al igual la clase Negocio tendrá como métodos todos los procesos de administración como el dar de alta y de baja productos, enlistar los productos tanto perecederos como no perecederos, las ventas que realizará la tienda y por último mostrar las ganancias del día.



Para la interfaz gráfica tenemos un `JFrame` con el identificador “Home” en el cual lo que se hace es tener como atributos una clase de tipo `Negocio` para administrar las opciones y un `ArrayList<Double>` para las ganancias, al igual la clase correspondiente para el manejo de archivos de texto, en la parte estética se muestran imágenes y botones con cada una de las opciones disponibles en la tienda al igual que un botón con el cual se termina el programa es decir se cierra sesión.

En el JFrame del Alta lo que se hace es tener etiquetas para mostrar lo que lleva o tienen que poner en cada opción por ejemplo nombre, precio, etc. Para el nombre, precio y el porcentaje de ganancias se uso 3 `(jTextField)` para obtener los datos los respectivos espacios, para el precio y el porcentaje de ganancias se utilizan un envoltorio del tipo `double` y `integer` respectivamente.

Para obtener la fecha de caducidad o decir que no es perecedero se utiliza un `jRadioButton` para dichas opciones, al ser perecedero se activan 3 `JComboBox` para elegir el día, mes y año de caducidad del producto final mente al elegir el botón Dar de alta se crea un objeto de tipo perecedero o no perecedero según corresponda y se manda como parámetro a ***home.negocio.altaProducto*** para guardarlo en el arraylist correspondiente.

Si alguna de las opciones no es llenada correctamente lanza una pantalla de advertencia y sale de el apartado, al igual que al presionar cancelar igual sale del apartado Alta y regresa al Home.

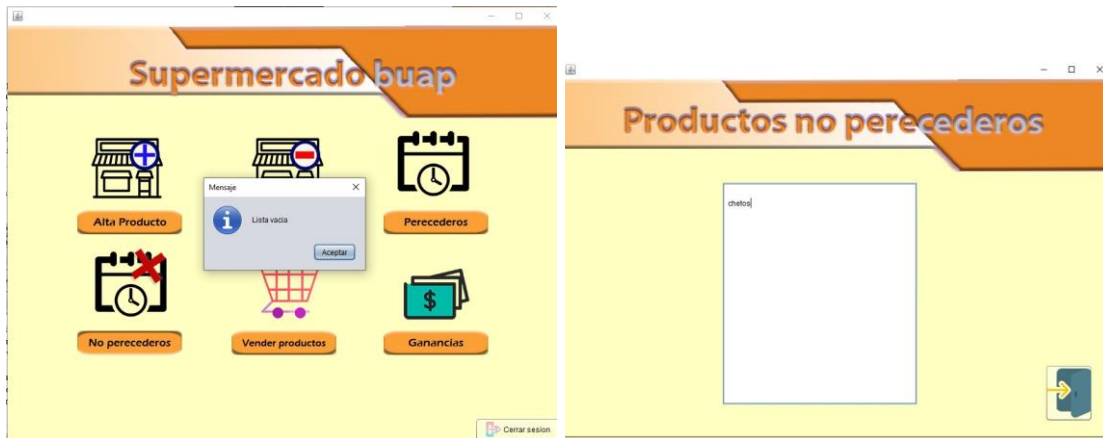


En el apartado de baja igual se tiene como atributo un objeto de tipo negocio y la parte estética lo único que se tiene es un espacio en donde se pondrá el nombre del producto que se quiere dar de alta y dos botones uno para realizar la baja y otro para cancelar.



Al presionar el botón de dar de baja se obtendrá el texto ingresado y se asignará como parámetro a [home.negocio.bajaProducto](#) con el identificador nombre y si se llega a encontrar el producto con dicho nombre se dará de baja correctamente de lo contrario dará el mensaje que no se pudo dar de baja.

Y saldrá de regreso a home, si se presiona cancelar saldrá a home sin realizar ningún otro proceso.



Para el apartado de perecederos y no perecederos se muestra el listado con el nombre de los productos correspondientes a cada lista en caso de haber productos nos muestra un mensaje diciendo que la lista esta vacia en la pestaña igual existe un boton para salir al home



En el apartado de venta lo que se hace es buscar por nombre el producto que se desea comprar e ir mostrando sus nombres en el espacio en blanco cada que damos en comprar finalmente al dar en el botón "Monto total" muestra la cantidad a pagar y sale al home.

Un proceso extra es que guarda las ganancias en el `arraylist<Double>`.



En la pantalla de ganancias se hace una lista de todo lo ganado por producto en el día y al igual que en las anteriores tiene un botón para salir e ir al home del programa