

Programação Orientada a Objetos

Projeto Final

Gestor de compras online

14 Dezembro 2021

Realizado por: André Moreira n 2020239416

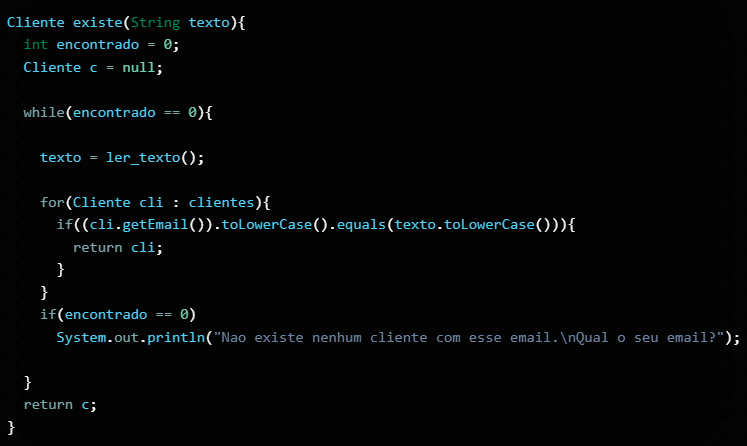
Introdução

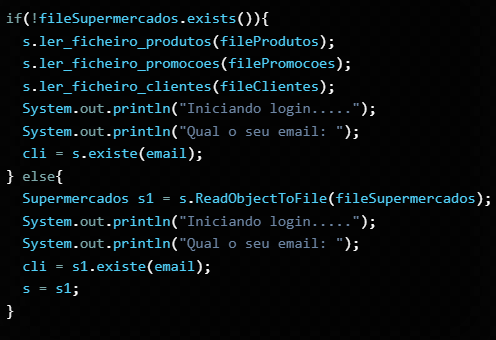
# O objetivo deste projeto consiste em realizar uma aplicação para a venda online de produtos em java. Para realizar esta aplicação existem vários operações que o programa deverá realizar, sendo as mais importantes as seguintes:

1. Cliente poder realizar um login.
2. Cliente poder realizar uma compra.
3. Cliente poder verificar todas as compras realizadas.

Operação 1

A operação 1 consiste, como dito na introdução, em permitir o cliente realizar um **login**. Iniciamos a operação por verificar se o **ficheiro de objeto** já foi criado, cajo ainda não tenha sido criado, lemos os ficheiros de texto para gravar os **clientes**, **produtos** e **promoções** que eles contêm. Cajo tenha sido criado, lemos o ficheiro de objeto para gravar o mesmo.



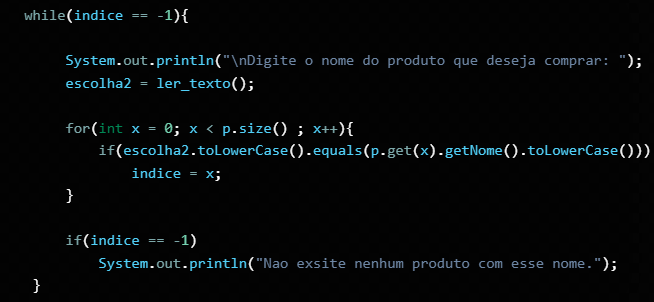
Após realizada a leitura dos ficheiros, iniciamos o login do cliente. Começamos por pedir o seu email usando o método **“ler\_texto**” e, depois, verificando se é um email existente através do método **“existe”** que se encontra na classe **Supermercados** (classe que contem funções do programa).

Este método verifica o email de todos os clientes, verificando se algum coincide com o **“texto”** (email fornecido pelo cliente), cajo não seja encontrado o email, é pedido de novo um email ao cliente, repetindo o loop ate que seja fornecido um email válido, devolvendo o cliente que contem esse email.

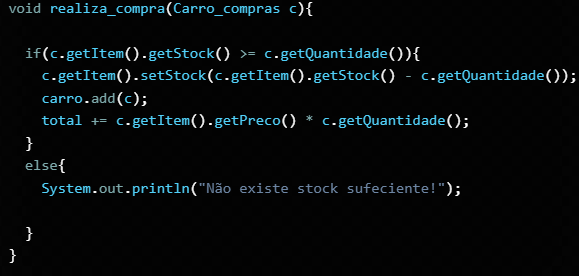
Operação 2

A operação 2 garante a possibilidade do usuário poder realizar uma compra, perguntando ao usuário o produto que ele deseja comprar e verificando se o produto inserido pelo cliente é válido.

E feita uma verificação através de um **loop for**, que verifica todos os nomes dos produtos para ver se algum e equivalente ao fornecido pelo cliente, cajo não seja, avisamos o cliente que não existe o produto que ele forneceu e pedimos outro nome, fazendo isto até que seja fornecido um nome válido.



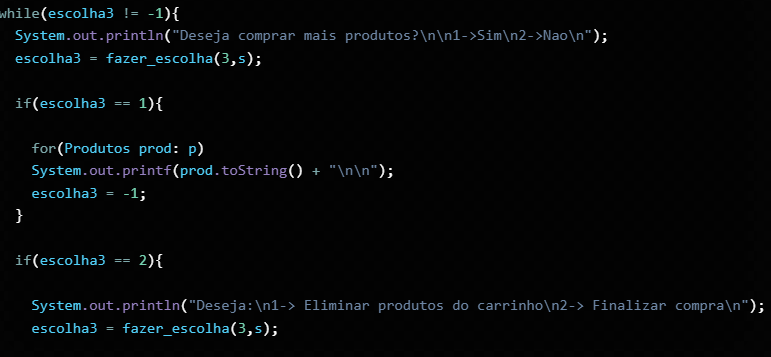
Após ser inserido um produto valido iniciamos o processo da compra (Fig.1). Começamos por perguntar ao cliente qual a quantidade do produto deseja comprar, verificando o valor inserido para ver se é um valor válido através do método **“ler\_inteiro”,** depois de receber um valor válido, verificamos se o produto já foi comprado anteriormente ou se é um produto novo.

Cajo já tenha sido comprado, verificamos se temos **stock** suficiente desse produto e, se tivermos stock suficiente, aumentamos a **quantidade do produto** na compra, o **custo total** e a variável **encontrado** (que nos permite saber que o produto já existia, para não criarmos uma nova compra).

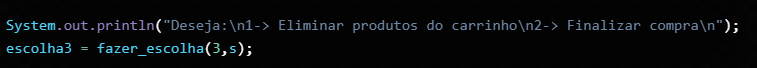
Se ainda não tiver sido comprado, criamos uma nova compra, com o produto e a quantidade inserida e realizamos a compra através do método **“realiza\_compra”** pertencente à classe **Compra,** este método começa por verificar se o **stock** é suficiente, se sim, retira ao **stock** a **quantidade** comprada, adiciona a **compra** ao **carrinho** e, adiciona ao **total** o **valor da compra**.

 Fig.1 – Processo da compra

Finalizada a compra do produto, começamos por perguntar ao cliente se ele deseja comprar mais produtos ou não utilizando o método **“fazer\_escolha”** para verificar se a escolha encontra-se entre os números apresentados.

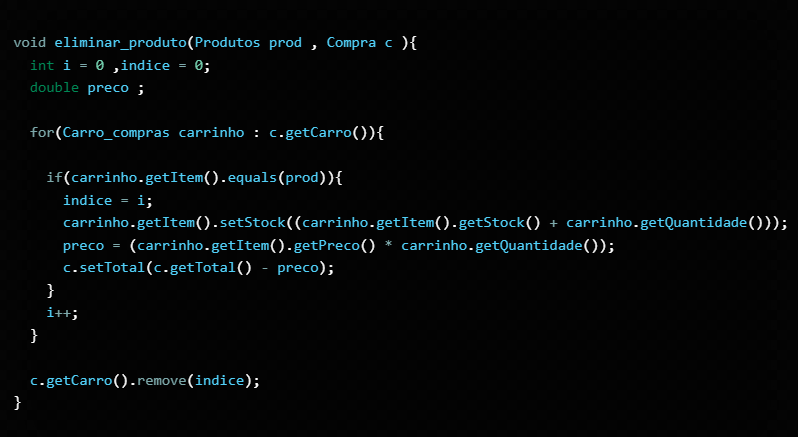


Cajo o cliente escolha comprar mais produtos, mostramos o catálogo de produtos de novo e, realizamos a **compra do produto** novamente. Se escolher que não deseja comprar mais produtos, damos duas opções, **eliminar produtos** **do carrinho** ou **finalizar a compra**.



Se o cliente escolher eliminar um produto (Fig. 2), começamos por perguntar ao cliente se deseja eliminar só um produto ou todos os produtos.

Se o cliente escolher eliminar um produto, começamos por representar o catálogo e perguntar qual o produto que deseja eliminar. Depois de inserido o nome do produto, verificamos se foi inserido uma string (através do método **“ler\_texto”)**, após a verificação corremos o carrinho de compra para confirmar que o produto esta a ser comprado e, corremos os produtos para obter o índice do produto dentro do **arraylist “Produtos”**, depois de obter o índice, chamamos o método “**eliminar\_produto”,** que ira eliminar o produto escolhido pelo cliente.



\*\*(Guarda índice para remover fora do for)



Fig. 2 – Processo de eliminação de um produto

Operação 3

Verifica se o carrinho não esta vazio e se tem alguma compra, se tiver da print se não diz que não tem.

Muda Data

Le\_data() verifica se a data inserida e valida se não for faz loop

Ficheiros

No fim por imagens do (le\_data , le\_string() , le\_int())