Teste Dev Operacional

Modelo de Negócio

O modelo de negócio parece fornecer uma boa base para um comércio eletrônico, consegui ver alguns detalhes:

Classe Venda

Tirei o Acento do atributo `código` , pois pode acarretar problemas no futuro.

- Colocar um tempo de seção para o usuário.
  Para isso iremos realizar a modificação da classe usuário, iremos adicionar 2 atributos e 3 método:
  Atributos
  - `sessaoExpirada` que ira armazenar o tempo de expiração em minutos, vou deixar esse atributo fixo com 45 minutos, podendo ser mudado se for para cada usuário,
    dependendo da regra de negócio, vou testar com 10 minutos só pra não ter que esperar 45 minutos nos testes realizados no ambiente de dev.
  - `inicioSessao` que ira armazenar a hora atual em que a sessão começou

  Metodos
  - Vamos migrar a funcionalidade de Login para o Usuário adicionando um método chamado `login()` que vai esperar 2 parâmetros: e-mail e senha e irá retornar um `boolean`
    Nesse método iremos inicializar os atributos descritos acima.
  - Iremos verificar se o usuário ainda está logado sempre que for realizar alguma operação novamente com o método `estaLogado` que não espera nenhum parâmero.
    Nele iremos verificar se o tempo de sessão expirou verificando entre o tempo de seção e a hora atual em que o método está sendo chamado.
  - Métódo para deslogar que atribuirá aos atributos os valores **iniciais**

- Métodos para adicionar e remover produtos da lista de produtos vendidos por uma empresa.
- Métodos para calcular o valor total de uma venda, incluindo a comissão cobrada pelo sistema.
- Métodos para listar as vendas realizadas por uma empresa e suas respectivas taxas de comissão.
- Eu adicionaria a data da venda também
- Mascarar o CPF e CNPJ

Não tem uma validação para "**registrar**" um usuário com usuário e senha iguais, então teria vários usuários iguais podendo acarretar em conflitos.
Daria pra fazer a mesma coisa com empresa tendo CNPJ iguais, nomes iguais dependendo dos requisitos e da lei de cada local.
A mesma coisa serve para registrar clientes com CPF iguais.

Correções no código fora do modelo de negócios

Vou transferir toda nossa simulação de banco de dados para uma classe chamada 'BancoDados' que terá todos os atributos já fornecidos anteriormente

Classe BancoDados

- método chamado 'iniciar()' que irá criar todos os objetos que utilizaremos/já utilizamos anteriormente
- Faremos um método genérico para validar dependendo do objeto (usuario, cliente, empresa) se o cnpj, cpf ou username existe, passando um objeto do tipo T, um predicate que utilizaremos no `anyMatch()` com auxílio do stream api para verificar de acordo com o predicate se algum objeto existe. Adicionaremos na lista se não houver.

```
public static <T> void verificarExiste(T objeto, Predicate predicate, List<T> list) {
        if (list.stream().anyMatch(predicate))
            return;
```

```
        list.add(objeto);
    }
```

- Faremos um método para cadastrar todos os atributos de forma genérica que irá esperar um argumento 'varargs' do tipo Usuário

```
public static <T> List<T> cadastrar(T... objetos)
```

Classe Main

- Não precisamos fazer um `filter()` e retornar uma lista, dado que queremos somente um usuário que tenha as informações passadas, retornaremos somente o usuário que vier da lista com o método do **stream api** `first()`
- O usuário está "deslogando" após realizar alguma operação, como: "Listar Vendas", etc.
  Correção: Manter o usuário logado utilizando a seção.
- Fiz um `Loop While` para verificar enquanto o usuário estiver logado para fazer operações básicas.
- Adicionei o statement `brake` depois de cada switch case, menos no de deslogar.

Logado como uma empresa, ao mostrar as vendas, fiz um método na própria classe **Venda** para realizar a mesma operação feita no método `executar`

Irei fazer o mesmo para as outras operações.

No menu de realizar uma compra ao escolher uma empresa vou retirar a parte de obter os objetos do stream em uma lista para obter o primeiro elemento da mesma e substituir pelo método `findFirst()` que retorna um `Optional`

A mesma coisa com o cliente

Adicionei um condição para verificar se o usuário logado é um admin, se for não terá acesso para comprar produtos. Foi o que eu entendi do modelo de negócio.

Adicionei a condição de que se for um Admin ou uma empresa poderá ver as informações da mesma.

Trocando o padrão das funções `IsEmpresa()` e `IsAdmin()` de PascalCase para camelCase

Fiz um menu um pouco diferente que só os admins podem ver, nos quais tem as mesmas opções de perfils do tipo empresa só que eles podem ver todos os produtos, todas as vendas.

Fiz um for loop que vai percorrer todas as empresas.

após isso vou fazer um stream nos produtos e ver quais produtos tem o id das empresas e pegar somentes os produtos das mesmas ficando assim:

```
Produto da empresa SafeWay Padaria - código 2
1 - Pão Frances - R$ 3,50 - 5 unidades
8 - Sonho - R$ 8,50 - 5 unidades
9 - Croissant - R$ 6,50 - 7 unidades
10 - Chá Gelado - R$ 5,50 - 4 unidades

Produto da empresa Level Varejo - código 1
2 - Coturno - R$ 400,00 - 10 unidades
3 - Jaqueta Jeans - R$ 150,00 - 15 unidades
4 - Calça Sarja - R$ 150,00 - 15 unidades

Produto da empresa SafeWay Restaurante - código 3
5 - Prato feito - Frango - R$ 25,00 - 10 unidades
6 - Prato feito - Carne - R$ 25,00 - 10 unidades
7 - Suco Natural - R$ 10,00 - 30 unidades
```

Alterando função que mostrava todos os valores de venda, estava mostrando muitos números após a virgula, colocando somente 2

```
this.itens.forEach(x -> {
    System.out.printf("%d - %s - R$ %.2f", x.getId(), x.getNome(), x.getPreco());
    System.out.println(x.getId() + " - " + x.getNome() + "    R$"  + x.getPreco());
});
System.out.printf("Total Venda: R$ %.2f", this.valor);
System.out.printf("Total Taxa a ser paga: R$ %.2f", this.comissaoSistema);
System.out.printf("Total Liquido  para empresa: R$ %.2f",(this.valor - this.comissaoSistema));
System.out.println("*******************************************************");
```

Alterando o “apelido” de **x** para **item** fica mais intuitivo.

Adicionei uma opção de listar todas as empresas se for um usuário admin