

Ciência da Computação – Programação 3

Trabalho 1

Informações Gerais

- **Data Limite de Entrega:** 12/06/2022 (23:59).
- **Pontuação:** 10 pontos (peso 15% na média do semestre).
- Os trabalhos podem ser desenvolvidos individualmente ou em duplas.
- **Formato de Entrega:** Enviar para o e-mail filipemtz@gmail.com um arquivo .zip contendo os códigos-fonte escritos na linguagem Java. O assunto do e-mail deve ter o formato “[CC] – [Prog3] – [Trab 1] - <nome dos estudantes>”. No corpo do e-mail devem ser informados os nomes e números de matrícula dos integrantes do grupo.
- **Importante:** Trabalhos entregues após a data limite sem justificativa com comprovação documental (atestado médico, etc.), ou que não seguirem o formato especificado de entrega receberão nota zero!

Especificação

O projeto consiste em desenvolver um programa para simular uma loja virtual simplificada.

O programa deve atender os seguintes requisitos:

- Exibir repetidamente um menu ao usuário do programa informando as ações que podem ser realizadas, solicitar que uma opção seja escolhida e realizar a ação selecionada. Se for escolhida uma opção inválida, o usuário deve ser exibida uma mensagem informativa e o menu deve ser exibido novamente.
- Deve ser criada uma classe Loja que contém sequências (ArrayLists) de produtos, clientes e notas fiscais.
- Permitir gerenciar cadastrar, listar todos e remover produtos, clientes e notas fiscais (compras).
 - Cada produto possui um código de identificação, nome, categoria, preço, quantidade em estoque e desconto. O desconto é um valor entre 0 e 1 indicando a porcentagem de desconto.
 - Cada cliente possui CPF, nome e telefone.
 - Cada nota fiscal possui um código de identificação, o usuário que fez a compra, data de realização, meio de pagamento e uma lista (ArrayList) de itens comprados.
- Nos cadastros, as informações devem ser solicitadas ao usuário do programa e ele digitará os valores dos atributos usando o teclado.
- Nas opções de listar todos, os dados devem ser exibidos em formato tabular.
- Nas opções de atualizar a remover, o usuário do programa deve digitar o identificador do elemento a ser atualizado ou removido.
- No cadastro de uma nota fiscal, deve ser informado apenas o CPF do usuário e o meio de pagamento. A data de realização deve ser definida como a data atual.
- Notas fiscais podem possuir vários produtos e um produto pode existir em várias notas fiscais. Deve ser criada uma classe ItemCompra para representar um produto em uma nota fiscal. A nota fiscal deve possuir uma lista (ArrayList) de objetos do tipo ItemCompra.

- Os atributos da classe ItemCompra são o produto adquirido, a quantidade de itens na compra e o valor unitário no momento da compra (para armazenar como histórico; dependendo do desconto e de mudanças no preço do produto, ao longo do tempo o valor de aquisição pode ser diferente do preço atual do produto).
- Adicionar produtos na nota fiscal. O usuário deve digitar o código do produto, o código da nota e a quantidade de itens. Usando estas informações, deve ser criado um objeto do tipo ItemCompra e o item será adicionado no ArrayList da nota fiscal. A quantidade comprada de um produto deve ser subtraída da quantidade em estoque e o valor atual do produto (considerando o desconto) deve ser atribuído ao ItemCompra.
- Permitir atualizar a quantidade em estoque, o preço e o desconto de um produto.
- Listar na tela todos os produtos de uma categoria digitada pelo usuário.
- Exibir a nota fiscal de uma compra. O programa deve solicitar o código da nota e exibir os nomes dos produtos, a quantidade comprada, os valores individuais e totais por produto e o valor total da compra.
- Exibir relatório em que sejam exibidos na tela o número de produtos e usuários cadastrados, além do número de compras. O relatório deve exibir ainda o valor total de todas as compras realizadas e os 5 produtos mais vendidos.

Classes, atributos e métodos

Cliente	NotaFiscal	ItemCompra
cpf: String nome: String telefone: String	codigo: int comprador: Cliente data: String metodoPagamento: String itens: ArrayList<ItemCompra>	produto: Produto quantidade: int custoUnidade: double
Cliente(String cpf, String nome, String telefone); String toString();	NotaFiscal(int codigo, Cliente comprador, String metodoPagamento); String toString(); void adicionarItem(ItemCompra item); void custoTotal();	ItemCompra(Produto produto, int quantidade, double custo); String toString(); double custoTotal();

Produto
codigo: int nome: String preço: double categoria: String desconto: double estoque: int
Produto(int codigo, String nome, double preço, String categoria, double desconto, int estoque); Produto(int codigo, String nome, double preço, int estoque); double custo(); void atualizarEstoque(int qtd); void atualizarDesconto(double v); void atualizarPreço(double v); String toString();

Loja
produtos: ArrayList<Produto> clientes: ArrayList<Cliente> notas: ArrayList<NotaFiscal>
void cadastrarCliente(); void cadastrarProduto(); void cadastrarNota(); void listarClientes(); void listarProdutos(); void listarNotas(); Cliente buscarCliente(String cpf); Produto buscarProduto(int codigo); NotaFiscal buscarNota(int codigo); void removerCliente(String cpf); void removerProduto(int codigo); void removerNota(int codigo); void adicionarItemNota(); void exibirRelatorio(); void listarItensCategoria();