

=====

TRABALHO DE PROGRAMAÇÃO JAVA

TEMA: SISTEMA DE PESHOP SIMPLS

=====

1. INTRODUÇÃO

Este projeto foi desenvolvido em Java e representa um sistema básico de gerenciamento de um PetShop.

O objetivo é permitir o cadastro de clientes, animais, serviços e rações, além de possibilitar o agendamento de serviços e a compra de rações. Ao final da execução, o programa também gera um recibo

simples com o resumo das operações realizadas.

O sistema foi feito de maneira simples e direta, com foco em aplicar os conceitos básicos de Programação Orientada a Objetos (POO): classes, objetos, atributos, métodos, listas e herança.

2. ESTRUTURA DO PROJETO

O projeto é composto pelas seguintes classes principais:

1. Pessoa.java

- Classe base que representa qualquer pessoa no sistema (nome e telefone).
- É a classe “mãe” de Cliente e Funcionário.

2. Cliente.java

- Herda de Pessoa.
- Guarda uma lista de animais cadastrados pelo cliente.

3. Funcionario.java

- Também herda de Pessoa.
- Representa os funcionários do PetShop (nome, telefone e cargo).

4. Animal.java

- Guarda os dados dos animais: nome, espécie, idade e dono (Cliente).

5. Servico.java

- Representa um serviço oferecido pelo PetShop (ex: Banho, Tosa).
- Cada serviço tem nome, preço e horários disponíveis.

6. Racao.java

- Representa uma ração vendida no PetShop (nome, preço e estoque).

7. Agendamento.java

- Armazena os dados dos serviços marcados pelos clientes (cliente, animal, serviço e horário).

8. Compra.java

- Representa a compra de uma ração feita por um cliente, com quantidade e valor total.

9. PetShop.java

- Classe principal de controle do sistema.
- Guarda as listas de clientes, serviços, rações, agendamentos e compras.
- Contém métodos para adicionar, listar e gerar recibos com totais gastos.

10. Main.java

- É o ponto de entrada do programa.
- Possui um menu interativo em texto com as opções:
 1. Cadastrar Cliente
 2. Listar Clientes
 3. Cadastrar Serviço
 4. Listar Serviços

5. Ver horários disponíveis
6. Cadastrar Ração
7. Listar Rações
8. Agendar Serviço
9. Comprar Ração
0. Sair e gerar recibos

3. FUNCIONAMENTO DO SISTEMA

Ao iniciar o programa, já existem dois serviços (Banho e Tosa) e cinco tipos de rações cadastradas

automaticamente, variando entre as mais baratas e as Premium.

O usuário interage com o sistema por meio de números digitados no console.

Exemplo:

- Se digitar 1, entra na função de cadastrar cliente.
- Se digitar 8, agenda um serviço.

Durante o agendamento, o sistema:

- Pede o cliente e o animal.
- Pede o serviço e o horário desejado.
- Caso o horário já esteja ocupado, o sistema avisa e pede outro.

Durante a compra de ração:

- O usuário escolhe o cliente e a ração.
- Informa a quantidade desejada.
- Se houver estoque, o sistema registra a compra e atualiza o estoque.

Ao encerrar o programa (opção 0), o sistema mostra um RECIBO com:

- Nome e telefone do cliente.
- Lista de serviços realizados e preços.
- Lista de rações compradas e valores.
- Totais parciais e o total geral gasto.

4. CONCEITOS DE PROGRAMAÇÃO APLICADOS

- CLASSES E OBJETOS:

Cada classe representa uma entidade do mundo real (Cliente, Animal, Serviço, etc.).

Os objetos são as instâncias criadas dessas classes.

- HERANÇA:

A classe Pessoa é a “mãe” de Cliente e Funcionário.

Isso evita repetição de código, já que ambos compartilham nome e telefone.

- ENCAPSULAMENTO:

Os atributos das classes são acessados por métodos (get/set), garantindo segurança e organização.

- COMPOSIÇÃO:

Um Cliente possui uma lista de Animais.

Um PetShop possui listas de Serviços, Clientes, Rações, etc.

- COLEÇÕES (ArrayList):

O sistema usa ArrayList para guardar as listas de objetos, permitindo adicionar e percorrer elementos facilmente.

- ESTRUTURAS DE CONTROLE:

O menu principal usa um laço while e um switch para navegar entre as opções.

5. CONCLUSÃO

Este projeto demonstra de forma prática os conceitos fundamentais da programação orientada a objetos.

Embora simples, ele simula um pequeno sistema real de PetShop, sendo uma ótima base para alunos que

estão aprendendo Java.

Possíveis melhorias futuras:

- Salvar os dados em arquivos para manter o histórico.
- Criar uma interface gráfica (JavaFX ou Swing).
- Adicionar autenticação de funcionários.
- Implementar relatórios com filtros e datas.

6. AUTORES / CRÉDITOS

Trabalho desenvolvido para fins didáticos.

Disciplina: Programação Orientada a Objetos

Linguagem: Java

Ano: 2025