

Estudo de Caso: Petshop 'Amigo Fiel'

Contexto:

O Petshop 'Amigo Fiel' tem crescido em popularidade na cidade e busca modernizar seus processos internos. A proprietária, Dona Marisa, sentiu a necessidade de um sistema que auxilie na gestão dos animais, produtos e serviços oferecidos. Ela contratou você, para criar este sistema.

Requisitos do Sistema:

1. Cadastro de animais:
 - Cada animal possui nome, espécie (cachorro, gato, pássaro, etc.), raça, data de nascimento e proprietário.
2. Cadastro de produtos:
 - Cada produto possui nome, categoria (alimento, brinquedo, medicamento, etc.), preço e quantidade em estoque.
3. Cadastro de serviços:
 - Os serviços podem ser banho, tosa, consulta veterinária, etc. Cada serviço tem um nome, descrição e preço.
4. Agenda para serviços:
 - Deve ser possível agendar um serviço para um animal, indicando a data, horário e o serviço a ser prestado.
5. Venda de produtos:
 - Registrar a venda de um produto, atualizando a quantidade em estoque.

Desafio:

Com base no aprendizado sobre algoritmos e orientação a objetos, desenvolva classes para representar os elementos citados nos requisitos, considerando:

1. Classe Animal:
 - Atributos: nome, espécie, raça, data de nascimento, proprietário.
 - Métodos: getters e setters para todos os atributos.
2. Classe Produto:
 - Atributos: nome, categoria, preço, quantidade em estoque.
 - Métodos: vender (diminui a quantidade em estoque), getters e setters.
3. Classe Servico (Abstrata):
 - Métodos abstratos: descricao() e preco().
 - Subclasses específicas: Banho, Tosa, ConsultaVeterinaria, etc., cada uma implementando os métodos abstratos de sua maneira única.
4. Classe Agenda:

- Atributos: animal, serviço, data, horário.
 - Métodos: agendar (define um novo agendamento), getters e setters.
5. Implemente a herança. Por exemplo, crie classes específicas para diferentes tipos de animais (Cachorro, Gato, etc.) que herdem da classe Animal, ou diferentes tipos de produtos que herdem da classe Produto.
 6. Polimorfismo: Implemente polimorfismo e abstração no sistema da maneira que achar mais correta.
 7. Bônus: Utilize a classe ArrayList do Java para armazenar os produtos, serviços e registros de venda. Lembre-se de importar `java.util.ArrayList`.

Restrições:

- Não permitir agendamentos em horários já ocupados.
- Não permitir a venda de produtos que não estejam em estoque.

Entrega:

Desenvolva um algoritmo que simule a interação de um usuário com o sistema. Por exemplo, o usuário pode cadastrar novos animais, agendar um serviço para um animal, vender um produto, etc. Garanta que todas as funcionalidades sejam testadas.

Ao finalizar este estudo de caso, os estudantes devem ser capazes de demonstrar sua compreensão sobre os conceitos de orientação a objetos e a aplicação destes conceitos na resolução de problemas do mundo real.