

RELATÓRIO DISCENTE DE ACOMPANHAMENTO - PRÁTICA: CADASTRO POO

1. Trabalho Prático | DGT2821 Programação Back-end com Java

Implementação de Sistema de Cadastro com Herança, Persistência Binária e Interface em Modo Texto.

2. Objetivo da Prática

O objetivo deste trabalho foi desenvolver um sistema completo de gestão de pessoas (Físicas e Jurídicas) aplicando os conceitos de herança e polimorfismo. O sistema permite o armazenamento persistente em arquivos binários e oferece uma interface de linha de comando para operações de CRUD (Create, Read, Update, Delete).

3. Códigos do Projeto

O projeto foi estruturado no pacote model para a lógica de negócio e repositórios, e na classe principal para a interface.

3.1 Entidades e Repositórios

As classes Pessoa, PessoaFisica e PessoaJuridica utilizam herança para compartilhar atributos comuns. Os repositórios implementam a lógica de ArrayList e os métodos de entrada/saída de arquivos utilizando ObjectOutputStream e ObjectInputStream.

3.2 Classe Principal (CadastroPOO.java)

Implementada com a classe Scanner para interação com o usuário, utilizando uma estrutura de repetição do-while e um bloco switch-case para processar as 8 opções do menu (incluindo tratamento de exceções para segurança dos dados).

4. Resultados da Execução

A execução no console demonstra o fluxo completo de cadastro, listagem e persistência:

```
=====
1 - Incluir | 2 - Alterar | 3 - Excluir
4 - Exibir ID | 5 - Todos | 6 - Salvar
7 - Recuperar | 0 - Sair
=====
Opção: 1
Tipo (F - Fisica, J - Juridica): F
ID: 10
Nome: Julio Silva
CPF: 12345678900
Idade: 30

Opção: 6
Prefixo dos arquivos: backup
Dados salvos com sucesso.
```

5. Análise e Conclusão

Quais as vantagens e desvantagens do uso de herança?

- **Vantagens:** Permite a reutilização de código e a criação de uma hierarquia lógica. Facilita o polimorfismo, onde um repositório pode tratar diferentes subclasses como o tipo base.
- **Desvantagens:** Pode gerar um acoplamento excessivo. Alterações na classe Pessoa impactam obrigatoriamente PessoaFisica e PessoaJuridica, o que exige cuidado na manutenção.

Por que a interface `Serializable` é necessária ao efetuar persistência em arquivos binários?

Ela atua como um sinalizador para a JVM, indicando que o estado daquele objeto pode ser "achado" em uma sequência de bytes para ser gravado em disco ou enviado pela rede. Sem ela, ocorre a exceção `NotSerializableException`.

Como o paradigma funcional é utilizado pela API `Stream` no Java?

O paradigma funcional é utilizado de forma declarativa através de métodos como `forEach()`, `filter()` e `map()`. Em vez de descrever o passo a passo da iteração (como num `for` tradicional), o desenvolvedor define a operação que deseja realizar sobre a coleção de dados.

Quando trabalhamos com Java, qual padrão de desenvolvimento é adotado na persistência de dados em arquivos?

O padrão adotado é o **Repository** (ou **DAO - Data Access Object**). Ele separa a lógica de acesso aos dados (como ler e escrever no arquivo `.bin`) da lógica de interface e das regras de negócio das entidades.