

UNIVERSIDADE ESTÁCIO DE SÁ

POLO CENTRO - IGREJINHA - RS

Curso: Desenvolvimento Fulll Stack

Disciplina: Iniciando o caminho pelo Java

(RPG0014)

Turma: 2024.1 | 3º semestre

Nome: Emily Duarte Watthier

Relatório de Prática: Cadastro com POO e Persistência Binária

Objetivos da Prática:

- Utilizar herança e polimorfismo na definição de entidades.
- Utilizar persistência de objetos em arquivos binários.
- Implementar uma interface cadastral em modo texto.
- Utilizar o controle de exceções da plataforma Java.

Códigos Implementados:

1. Entidades (model)

- Pessoa.java
- PessoaFisica.java
- PessoaFisica.java

2. Repositórios (model)

- PessoaFisicaRepo.java
- PessoaJuridicaRepo.java

3. Classe Principal

• CadastroPOO.java (com método main() contendo as etapas do passo 5)

Resultados da Execução:

As execuções foram realizadas com sucesso, gerando os seguintes arquivos:

Pessoas_fisicas.dat

Pessoas_juridicas.dat

A saída do console apresentou a leitura correta dos dados armazenados e recuperados. Foram exibidas as informações de duas pessoas físicas e duas jurídicas.

Print da Execução:

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
                                                                                                     CadastroPOO - Apache NetBeans IDE 25
  🚪 💾 🔩 : 🤚 🤃 <default config> 🔻 🕒 🚏 🌠 🕨 🛊 🖟 🕩 🔹 : 169,2/354,0MB 🗞 🕼
                                                   💶 ....A] 📓 PessoaFisica.java [-/A] 🗴 📓 PessoaJuridica.java [-/A] 🗴 📓 PessoaFisicaRepo.java [-/A] 🗴
  © cadastroPOO [0000000...]

✓ In Source Packages

✓ In Scadastropoo

CadastroPOO.java [-/A]
                                                       Source History 🔯 🔯 🔻 💆 💆 🔭 🖶 🕠 🤻 📞 💺 🔩 💆 🛑 🔳 🌿
                                                                        FileOutputStream fos = new FileOutputStream(nomeArquivo);
ObjectOutputStream(fos);
oos.writeObject(listaPessoaJuridica);
                                                         2
-
      Arquivo salvo em: C:\Users\emily\OneDrive\Documentos\missaoNV1\CadastroPOO\pessoas_fisicas.dat Dados de Pessoa Fisica Armazenados.
      Dados carregados de: C:\Users\emily\OneDrive\Documentos\missaoNV1\CadastroFOO\pessoas_fisicas.dat
Dados de Pessoa Fisica Recuperados.
      Pessoa F∳sica - ID: 1, Nome: Ana, CPF: 11111111111, Idade: 25
      Pessoa F♦sica - ID: 2, Nome: Carlos, CPF: 2222222222, Idade: 52
      Dados de Pessoa Juridica Recuperados
      Pessoa Jur∳dica - ID: 3, Nome: XPTO Sales, CNPJ: 3333333333333
```

Repositório do Projeto:

O projeto completo está hospedado no GitHub, no seguinte endereço:

https://github.com/Watthier09/missaoNV1JAVA/tree/main/missaoNV1/CadastroPOO

Análise e Conclusão:

a. Quais as vantagens e desvantagens do uso de herança?

Vantagens:

- Reutilização de código entre classes.
- Estruturação hierárquica mais clara.
- Facilita extensões de funcionalidades.

Desvantagens:

- Forte acoplamento entre super e subclasses.
- Pode haver uso indevido em lugar de composição.
- Rastreio de comportamentos herdados pode ficar complexo.
- b. Por que a interface Serializable é necessária ao efetuar persistência em arquivos binários?

A interface Serializable permite transformar objetos em uma sequência de bytes, essencial para que possam ser salvos e recuperados de arquivos binários. Sem ela, a JVM não permite a serialização.

c. Como o paradigma funcional é utilizado pela API stream no Java?

A API Stream permite operar sobre coleções de forma declarativa, utilizando funções como filter, map, reduce. Isso melhora a legibilidade e permite paralelismo. Embora não tenha sido utilizado nesta prática, é ideal para manipular listas.

d. Qual padrão de desenvolvimento é adotado na persistência de dados em arquivos?

O padrão DAO (Data Access Object) é o mais comum. Ele separa a lógica de acesso aos dados da lógica de negócios, como feito nas classes PessoaFisicaRepo e PessoaJuridicaRepo.