

Sérgio Koerich Pereira

<https://github.com/sergiokoerich/Miss-o-Pr-tica-N-vel-1-Mundo-3>

Título da Prática: Desenvolvimento de Aplicação de Cadastro com Persistência em Java

Objetivo da Prática: O objetivo desta prática é desenvolver uma aplicação de cadastro utilizando Java, aplicando conceitos de orientação a objetos, herança, persistência em arquivos e interação com o usuário via console. Será implementado um sistema que permite gerenciar cadastros de pessoas físicas e jurídicas, com capacidade de inserir, alterar, excluir, buscar, exibir, persistir e recuperar dados armazenados em arquivos binários.

Todos os códigos solicitados neste roteiro de aula: Aqui estão os códigos implementados conforme o roteiro:

1. Pacote `model`:

- `Pessoa.java`
- `PessoaFisica.java`
- `PessoaJuridica.java`
- `PessoaFisicaRepo.java`
- `PessoaJuridicaRepo.java`

2. Classe `Main`:

- `Main.java`

3. Resultados da Execução dos Códigos:

- A aplicação foi testada com sucesso, permitindo interações para inclusão, alteração, exclusão, busca, exibição, persistência e recuperação de dados de pessoas físicas e jurídicas.
- Os dados foram corretamente armazenados e recuperados dos arquivos binários especificados.

Análise e Conclusão:

1. Elementos Estáticos e o Método `Main`:

- Elementos estáticos em Java pertencem à classe em vez de instâncias individuais. O método `main` é estático para que possa ser invocado pelo sistema Java sem a necessidade de instanciar um objeto da classe `Main`. Isso é fundamental porque o ponto de entrada do programa Java (`main`) precisa ser acessível diretamente pelo Java Virtual Machine (JVM) durante a execução.

2. Classe `Scanner`:

- A classe `Scanner` em Java é usada para ler entradas do usuário no console. Ela permite que o programa receba e manipule dados inseridos pelo usuário durante a execução. No contexto deste projeto, o `Scanner` é utilizado para

capturar opções de menu e dados das entidades (como nome, CPF, CNPJ, etc.) durante a interação do usuário com a aplicação.

3. Impacto das Classes de Repositório na Organização do Código:

- As classes de repositório (`PessoaFisicaRepo` e `PessoaJuridicaRepo`) têm um papel crucial na organização do código. Elas encapsulam a lógica de gerenciamento dos dados das entidades (`PessoaFisica` e `PessoaJuridica`), fornecendo métodos padronizados para inserção, alteração, exclusão, busca e persistência de dados. Isso promove uma separação clara entre as responsabilidades de gerenciamento de dados e as classes de entidade, seguindo os princípios de encapsulamento e coesão.

Esta documentação resume o desenvolvimento e os conceitos explorados na implementação da aplicação de cadastro em Java. Se precisar de mais detalhes ou ajustes, estou à disposição para ajudar!