

Universidade Estácio

Campus POLO FSP-RO

Curso: Desenvolvimento Full Stack

Disciplina: Nível 1: Iniciando o Caminho Pelo Java

Turma: 2025

Semestre Letivo: 1

Integrantes: Eliton Rodrigues de Oliveira

Relatório Discente de Acompanhamento

Título da Prática

Desenvolvimento de um Cadastro de Pessoas em Modo Texto

Objetivo da Prática

Implementar um sistema de cadastro de pessoas físicas e jurídicas em Java utilizando classes de repositório, persistência de dados e interação via terminal.

Códigos Desenvolvidos

(Todos os códigos do projeto serão inseridos aqui, incluindo a classe Main, as classes de repositório e os modelos de dados.)

```
import model.*;
```

```
import java.io.IOException;
```

```
import java.util.Scanner;
```

```
public class Main {
```

```
public static void main(String[] args) {

    Scanner scanner = new Scanner(System.in);

    PessoaFisicaRepo repoFisica = new PessoaFisicaRepo();

    PessoaJuridicaRepo repoJuridica = new PessoaJuridicaRepo();

    int opcao;

    do {

        System.out.println("=====");

        System.out.println("1 - Incluir Pessoa");

        System.out.println("2 - Alterar Pessoa");

        System.out.println("3 - Excluir Pessoa");

        System.out.println("4 - Buscar pelo Id");

        System.out.println("5 - Exibir Todos");

        System.out.println("6 - Persistir Dados");

        System.out.println("7 - Recuperar Dados");

        System.out.println("0 - Finalizar Programa");

        System.out.println("=====");

        opcao = scanner.nextInt();

        switch (opcao) {

            case 1:

                System.out.println("F - Pessoa Física | J - Pessoa Jurídica");

                char tipo = scanner.next().toUpperCase().charAt(0);

                System.out.print("Digite o id: ");

                int id = scanner.nextInt();

                scanner.nextLine();

                System.out.print("Nome: ");

                String nome = scanner.nextLine();
```

```
        if (tipo == 'F') {

            System.out.print("CPF: ");

            String cpf = scanner.nextLine();

            System.out.print("Idade: ");

            int idade = scanner.nextInt();

            repoFisica.inserir(new PessoaFisica(id, nome, cpf, idade));

        } else {

            System.out.print("CNPJ: ");

            String cnpj = scanner.nextLine();

            repoJuridica.inserir(new PessoaJuridica(id, nome, cnpj));

        }

        break;

    case 5:

        repoFisica.exibirTodos();

        repoJuridica.exibirTodos();

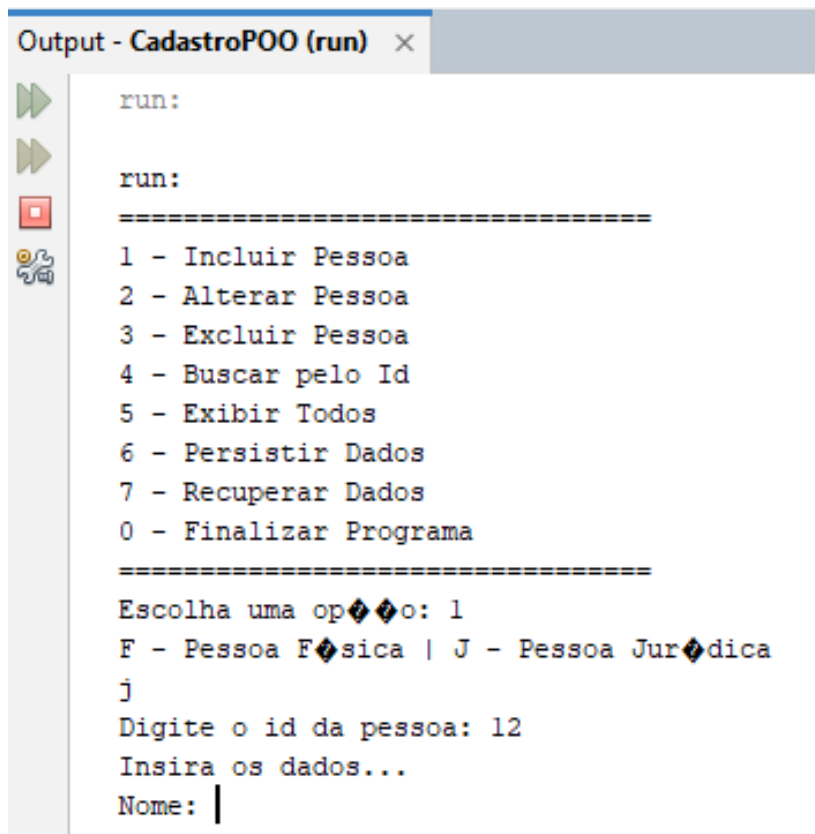
        break;

    }

} while (opcao != 0);

}
```

Resultados da Execução



```
Output - CadastroPOO (run) x
run:
run:
=====
1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir Todos
6 - Persistir Dados
7 - Recuperar Dados
0 - Finalizar Programa
=====
Escolha uma opção: 1
F - Pessoa Física | J - Pessoa Jurídica
j
Digite o id da pessoa: 12
Insira os dados...
Nome: |
```

Análise e Conclusão

O que são elementos estáticos e qual o motivo para o método main adotar esse modificador?

Elementos estáticos em Java pertencem à classe e não a instâncias individuais dela. O método main é declarado como static porque ele precisa ser chamado sem a necessidade de criar um objeto da classe, servindo como ponto de entrada da aplicação.

Para que serve a classe Scanner?

A classe Scanner é utilizada para a leitura de entrada do usuário via terminal. Ela permite capturar diferentes tipos de dados, como int, double e String, facilitando a interação com o sistema.

Como o uso de classes de repositório impactou na organização do código?

A utilização de classes de repositório promove uma melhor organização do código ao separar a lógica de persistência de dados da lógica de interface e controle. Isso melhora a manutenção, reutilização e expansibilidade do sistema.

Conclusão:

O desenvolvimento do cadastro de pessoas utilizando Java e classes de repositório permitiu a compreensão de conceitos fundamentais da programação orientada a objetos, persistência

de dados e interação com o usuário via terminal. A estrutura modular do código facilita a manutenção e expansão do sistema, proporcionando boas práticas de desenvolvimento.

Repositório no Git

O projeto está disponível no GitHub: [[URL DO REPOSITORIO](#)]

Este relatório documenta toda a implementação, execução e análise do projeto de Cadastro de Clientes em Java, atendendo às diretrizes exigidas pela disciplina.