



Nome do campus: 3366 Polo Centro – Garopaba - SC.

Nome do curso: Desenvolvimento Full Stack.

Nome da disciplina: Iniciando caminho pelo java.

Número da turma: 9001.

Nome: Daniel dos Santos Pereira.

Endereço: [DaniielDev/Entidades Sistema](#)

Título da Prática: Criação do Cadastro em Moto Texto.

Objetivo: Implementar um sistema de cadastro de pessoas físicas e jurídicas, utilizando conceitos de Programação Orientada a Objetos, como herança e polimorfismo.

O que são elementos estáticos e por que o método main os utiliza?

Elementos estáticos (static) pertencem à classe e não a uma instância específica dela. O método main é static porque ele precisa ser executado pela JVM sem que uma instância da classe principal seja criada. Assim, ele pode ser acessado diretamente pelo nome da classe ao iniciar o programa.

Para que serve a classe Scanner?

A classe Scanner é usada para ler entradas do usuário via teclado. No sistema, ela permite capturar dados como nomes, CPFs, CNPJs e IDs digitados pelo usuário para incluir ou manipular os registros.

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

O uso de repositórios (PessoaFisicaRepo e PessoaJuridicaRepo) tornou o código mais modular e organizado. Ele separa claramente a lógica de persistência e gerenciamento de dados da interface com o usuário. Essa separação melhora a manutenção, testes e reusabilidade do código.

Códigos:

CriaçãoDeCadastro.java

```
import model.*;

import java.util.Scanner;

public class Principal {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        PessoaFisicaRepo repoFisica = new PessoaFisicaRepo();

        PessoaJuridicaRepo repoJuridica = new PessoaJuridicaRepo();

        int opcao;

        do {

            System.out.println("\n=== MENU ===");

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

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

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

            System.out.println("4 - Exibir por ID");

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

            System.out.println("6 - Salvar dados");

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

            System.out.println("0 - Sair");

            System.out.print("Escolha uma opção: ");

            opcao = Integer.parseInt(sc.nextLine());

            switch (opcao) {

                case 1 -> {

                    System.out.print("Pessoa Física (F) ou Jurídica (J)? ");

                    String tipo = sc.nextLine().toUpperCase();
```

```

System.out.print("ID: ");
int id = Integer.parseInt(sc.nextLine());
System.out.print("Nome: ");
String nome = sc.nextLine();

if (tipo.equals("F")) {
    System.out.print("CPF: ");
    String cpf = sc.nextLine();
    System.out.print("Idade: ");
    int idade = Integer.parseInt(sc.nextLine());
    repoFisica.inserir(new PessoaFisica(id, nome, cpf, idade));
} else {
    System.out.print("CNPJ: ");
    String cnpj = sc.nextLine();
    repoJuridica.inserir(new PessoaJuridica(id, nome, cnpj));
}
}

case 2 -> {
    System.out.print("Pessoa Física (F) ou Jurídica (J)? ");
    String tipo = sc.nextLine().toUpperCase();
    System.out.print("ID da pessoa para alterar: ");
    int id = Integer.parseInt(sc.nextLine());

    if (tipo.equals("F")) {
        PessoaFisica pf = repoFisica.obter(id);
        if (pf != null) {
            pf.exibir();
            System.out.print("Novo nome: ");
            pf.setNome(sc.nextLine());
        }
    }
}

```

```

        System.out.print("Novo CPF: ");
        pf.setCpf(sc.nextLine());
        System.out.print("Nova idade: ");
        pf.setIdade(Integer.parseInt(sc.nextLine()));
        repoFisica.alterar(pf);
    } else {
        System.out.println("Pessoa física não encontrada.");
    }
} else {
    PessoaJuridica pj = repoJuridica.obter(id);
    if (pj != null) {
        pj.exibir();
        System.out.print("Novo nome: ");
        pj.setNome(sc.nextLine());
        System.out.print("Novo CNPJ: ");
        pj.setCnpj(sc.nextLine());
        repoJuridica.alterar(pj);
    } else {
        System.out.println("Pessoa jurídica não encontrada.");
    }
}
}

case 3 -> {
    System.out.print("Pessoa Física (F) ou Jurídica (J)? ");
    String tipo = sc.nextLine().toUpperCase();
    System.out.print("ID da pessoa para excluir: ");
    int id = Integer.parseInt(sc.nextLine());

    if (tipo.equals("F")) {

```

```

        repoFisica.excluir(id);
    } else {
        repoJuridica.excluir(id);
    }
}

case 4 -> {

    System.out.print("Pessoa Física (F) ou Jurídica (J)? ");

    String tipo = sc.nextLine().toUpperCase();

    System.out.print("ID da pessoa para exibir: ");

    int id = Integer.parseInt(sc.nextLine());

    if (tipo.equals("F")) {

        PessoaFisica pf = repoFisica.obter(id);

        if (pf != null) pf.exibir();

        else System.out.println("Pessoa física não encontrada.");

    } else {

        PessoaJuridica pj = repoJuridica.obter(id);

        if (pj != null) pj.exibir();

        else System.out.println("Pessoa jurídica não encontrada.");

    }

}

case 5 -> {

    System.out.print("Pessoa Física (F) ou Jurídica (J)? ");

    String tipo = sc.nextLine().toUpperCase();

    if (tipo.equals("F")) {

        for (PessoaFisica pf : repoFisica.obterTodos()) {

            pf.exibir();

            System.out.println();

```

```

    }
} else {
    for (PessoaJuridica pj : repoJuridica.obterTodos()) {
        pj.exibir();
        System.out.println();
    }
}
}

case 6 -> {
    System.out.print("Informe o prefixo do arquivo: ");
    String prefixo = sc.nextLine();
    try {
        repoFisica.persistir(prefixo + ".fisica.bin");
        repoJuridica.persistir(prefixo + ".juridica.bin");
        System.out.println("Dados salvos com sucesso!");
    } catch (Exception e) {
        System.out.println("Erro ao salvar dados: " + e.getMessage());
    }
}

case 7 -> {
    System.out.print("Informe o prefixo do arquivo: ");
    String prefixo = sc.nextLine();
    try {
        repoFisica.recuperar(prefixo + ".fisica.bin");
        repoJuridica.recuperar(prefixo + ".juridica.bin");
        System.out.println("Dados recuperados com sucesso!");
    } catch (Exception e) {
        System.out.println("Erro ao recuperar dados: " + e.getMessage());
    }
}

```

```
    }  
    case 0 -> System.out.println("Finalizando...");  
    default -> System.out.println("Opção inválida.");  
    }  
} while (opcao != 0);  
sc.close();  
}  
}
```