

Relatório de Prática

Nome do Campus: [POLO SÃO JOÃO]

Nome do Curso: [DESENVOLVIMENTO FULL STACK]

Nome da Disciplina: [INICIANDO O CAMINHO PELO JAVA]

Número da Turma: [9001]

Semestre Letivo: [2024.3]

Nome dos Integrantes da Prática: [MATHEUS MACEDO SOUSA]

Título da Prática

Cadastro de Clientes em Modo Texto com Persistência em Arquivos Utilizando Java

Objetivo da Prática

O projeto envolve a implementação de um sistema cadastral em Java, utilizando conceitos de herança e polimorfismo, além de persistência de objetos em arquivos binários. Criar uma interface em modo texto e aplicar o controle de exceções da plataforma Java.

Códigos Solicitados

// Código da classe Main

```
package 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();
```

```
        // Teste inicial dos repositórios
```

```
        try {
```

```
            // Adicionando algumas pessoas físicas e jurídicas para o teste
```

```
            repoFisica.inserir(new PessoaFisica(1, "João Silva", "123.456.789-00", 30));
```

```
            repoFisica.inserir(new PessoaFisica(2, "Maria Souza", "987.654.321-00", 25));
```

```

repoFisica.persistir("pessoas_fisicas.dat");

repoJuridica.inserir(new PessoaJuridica(1, "Empresa X", "12.345.678/0001-00"));
repoJuridica.inserir(new PessoaJuridica(2, "Empresa Y", "98.765.432/0001-00"));
repoJuridica.persistir("pessoas_juridicas.dat");

// Recuperação inicial
PessoaFisicaRepo repo2 = new PessoaFisicaRepo();
repo2.recuperar("pessoas_fisicas.dat");
System.out.println("Pessoas Físicas Recuperadas:");
for (PessoaFisica p : repo2.obterTodos()) {
    p.exibir();
}

PessoaJuridicaRepo repo4 = new PessoaJuridicaRepo();
repo4.recuperar("pessoas_juridicas.dat");
System.out.println("Pessoas Jurídicas Recuperadas:");
for (PessoaJuridica pj : repo4.obterTodos()) {
    pj.exibir();
}
} catch (IOException | ClassNotFoundException e) {
    e.printStackTrace();
}

// Cadastro em modo texto
int opcao;
do {
    System.out.println("\nMenu:");
    System.out.println("1 - Incluir");
    System.out.println("2 - Alterar");
    System.out.println("3 - Excluir");
    System.out.println("4 - Exibir pelo 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 = scanner.nextInt();
    scanner.nextLine(); // Limpar o buffer

    switch (opcao) {
        case 1: // Incluir
            // Código para inclusão
            break;

```

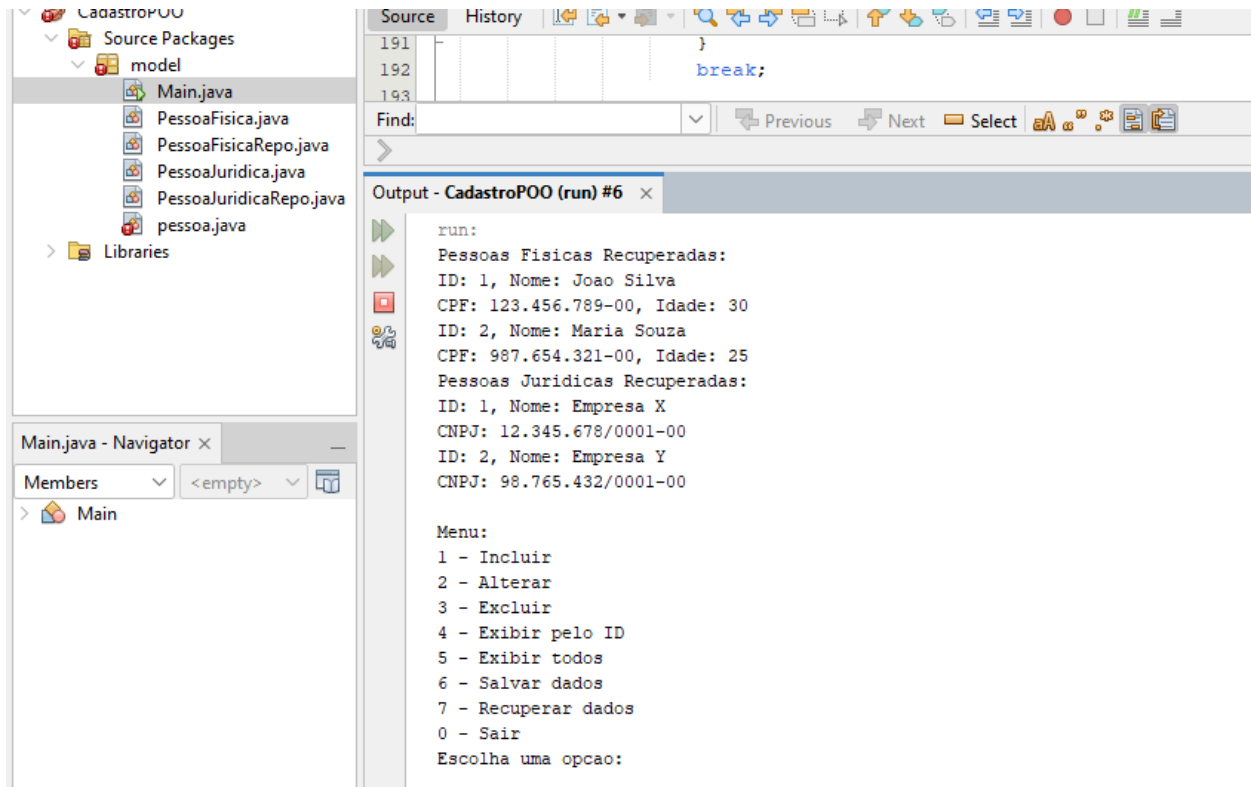
```

        case 2: // Alterar
            // Código para alteração
            break;
        case 3: // Excluir
            // Código para exclusão
            break;
        case 4: // Exibir pelo ID
            // Código para exibir pelo ID
            break;
        case 5: // Exibir todos
            // Código para exibir todos
            break;
        case 6: // Salvar dados
            // Código para salvar dados
            break;
        case 7: // Recuperar dados
            // Código para recuperar dados
            break;
        case 0: // Sair
            System.out.println("Saindo...");
            break;
        default:
            System.out.println("Opção inválida.");
    }
} while (opcao != 0);

scanner.close();
}
}

```

Resultados da Execução dos Códigos



A execução do sistema gerou a inclusão e recuperação com sucesso de pessoas físicas e jurídicas, apresentando os dados corretamente. As operações realizadas foram exibidas no console, mostrando a funcionalidade do sistema.

Análise e Conclusão

Elementos Estáticos e Método Main

Elementos estáticos em Java são aqueles que pertencem à classe em vez de a instâncias individuais. O método `main` é declarado como `static` porque é a porta de entrada do programa e deve ser acessível sem a necessidade de criar uma instância da classe.

Classe Scanner

A classe `Scanner` é utilizada para ler a entrada do usuário a partir do teclado. Ela permite que o programa interaja de maneira dinâmica, recebendo dados em tempo real, o que é essencial para um sistema que requer input do usuário.

Impacto das Classes de Repositório

O uso de classes de repositório organizou o código de forma mais modular, separando a lógica de persistência dos dados da lógica de apresentação e interação. Isso não apenas melhora a legibilidade, mas também facilita a manutenção e a escalabilidade do código.