



Estácio

Faculdade Estácio

Campus Belford Roxo - RJ

Desenvolvimento Full Stack

Disciplina - Iniciando caminho pelo Java

Turma 2023.2

Semestre - 3

Cristian da Silva de Macena

Repositório: <https://github.com/CristianS09/missao-pratica-java-cadastroPoo.git>

Criação do Cadastro em Modo Texto

Os objetivos dessa prática foram usar a classe scanner para criar interação com o usuário, para que ele possa realizar todas as operações que ele deseje antes de persistir os dados.

Códigos

CadastroPoo

```
package cadastrpoo;
import cadastrpoo.model.PessoaFisica;
import cadastrpoo.model.PessoaJuridica;
import cadastrpoo.model.gerenciadores.PessoaFisicaRepo;
import cadastrpoo.model.gerenciadores.PessoaJuridicaRepo;

import java.util.Scanner;

/**
 *
 * @author Cristian
 */
public class CadastroPOO {
    public static void main(String[] args) {
        //Scanner inicial
        Scanner scanner = new Scanner(System.in);
        //Criado objetos do tipo PessoaFisica e PessoaJuridica
        nulos
        PessoaFisica pessoaFisica = null;
        PessoaJuridica pessoaJuridica = null;

        //Instancinado os repositórios
        PessoaFisicaRepo pessoaFisicaRepo = new PessoaFisicaRepo();
        PessoaJuridicaRepo pessoaJuridicaRepo = new
        PessoaJuridicaRepo();

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



```

        String nome = scanner1.nextLine();
        System.out.println("Cnpj:");
        String cnpj = scanner1.nextLine();
        pessoaJuridica = new
PessoaJuridica(id,nome,cnpj);

pessoaJuridicaRepo.inserir(pessoaJuridica);
        System.out.println("Pessoa Jurídica
criada: " + pessoaJuridica);

    } else{
        System.out.println("Opção inválida,
tente novamente");
    }
} catch (IndexOutOfBoundsException exc){
    System.err.println("O id é inválido");
}
break;
//Alterar Pessoa
case "2":
    try{
        System.out.println("F - Pessoa Física | J -
Pessoa Jurídica");
        String tipoPessoaMudar =
scanner1.nextLine();
        if(pessoaFisica != null &&
tipoPessoaMudar.equalsIgnoreCase("f")) {
            System.out.println("Insira um id valido
para mudar:");

            int idMudar = scanner1.nextInt();
            scanner1.nextLine();
            System.out.println("Nome:");
            String nome = scanner1.nextLine();
            System.out.println("Cpf:");
            String cpf = scanner1.nextLine();
            System.out.println("Idade:");
            int idade = scanner1.nextInt();
            pessoaFisicaRepo.alterar(idMudar,new
PessoaFisica(idMudar,nome,cpf,idade));
        }
        if(pessoaJuridica != null &&
tipoPessoaMudar.equalsIgnoreCase("j")) {
            System.out.println("Insira um id válido
para mudar:");

            int idMudar = scanner1.nextInt();
            scanner1.nextLine();
            System.out.println("Nome:");
            String nome = scanner1.nextLine();

```

```

        System.out.println("Cnpj:");
        String cnpj = scanner1.nextLine();
        pessoaJuridicaRepo.alterar(idMudar, new
PessoaJuridica(idMudar, nome, cnpj));
    }
    }catch (IndexOutOfBoundsException exc){
        System.err.println("Opção inválida ou a
Pessoa ainda não foi definida.");
    }
    break;
case "3":
    //Excluir Pessoa
    try{
        System.out.println("Insira um id válido
para excluir");
        int idPessoa = scanner1.nextInt();
        if(pessoaFisica != null){
            System.out.println("Pessoa Física
removida:\n" + pessoaFisicaRepo.obter(idPessoa));
            pessoaFisicaRepo.excluir(idPessoa);
        }if(pessoaJuridica != null){
            System.out.println("Pessoa Jurídica
removida:\n" + pessoaJuridicaRepo.obter(idPessoa));
            pessoaJuridicaRepo.excluir(idPessoa);
        }
    }catch (IndexOutOfBoundsException exc){
        System.err.println("Id inválido ou não
existente");
    }
    break;
case "4":
    //Buscar Pelo Id
    System.out.println("Insira um id válido:");
    int id = scanner1.nextInt();
    if(pessoaFisica != null){
        try{
            System.out.println("Pessoa Física:");
            System.out.println(pessoaFisicaRepo.obter(id));
        }catch (IndexOutOfBoundsException exc){
            System.out.println("O id é inválido");
        }
    }
    if(pessoaJuridica != null){
        try{
            System.out.println("Pessoa Jurídica:");
            System.out.println(pessoaJuridicaRepo.obter(id));

```

```

        } catch (IndexOutOfBoundsException exc) {
            System.out.println("O id é inválido");
        }
    }
    if(pessoaJuridica == null && pessoaJuridica ==
null) {
        System.out.println("Nenhuma pessoa foi
inserida.");
    }
    break;
    case "5":
        //Exibir todas as Pessoas
        System.out.println("Pessoa Jurídica: ");
        pessoaJuridicaRepo.obterTodos();
        System.out.println("Pessoa Física: ");
        pessoaFisicaRepo.obterTodos();
        break;
    case "6":
        //Persistir dados
        try{
            if (pessoaJuridica != null) {
                pessoaJuridicaRepo.persistir("C:\\Users\\crist\\OneDrive\\Área de
Trabalho\\CadastroPOO\\src\\cadastropoo\\model\\arquivos\\pessoaJu
ridica.dat");
            }
            if(pessoaFisica != null){
                pessoaFisicaRepo.persistir("C:\\Users\\crist\\OneDrive\\Área de
Trabalho\\CadastroPOO\\src\\cadastropoo\\model\\arquivos\\pessoaFi
sica.dat");
            } else{
                System.err.println("Nenhuma dado foi
inserido para persistência.");
            }
        } catch (Exception exc){
            System.out.println(exc.getMessage());
        }
        break;
    case "7":
        //Recuperar dados
        try{
            pessoaJuridicaRepo.recuperar("C:\\Users\\crist\\OneDrive\\Área de
Trabalho\\CadastroPOO\\src\\cadastropoo\\model\\arquivos\\pessoaJu
ridica.dat");
            pessoaFisicaRepo.recuperar("C:\\Users\\crist\\OneDrive\\Área de

```

```
Trabalho\\CadastroPOO\\src\\cadastropoo\\model\\arquivos\\pessoaFi
sica.dat");
        }catch (Exception exc){
            System.err.println("Arquivo não
encontrado");
        }
        break;
        case "8":
            //Monstrar as instruções do programa

System.out.println("=====");
            System.out.println("1 - Incluir pessoa \n2 -
Alterar Pessoa \n3 - Excluir Pessoa" +
                " \n4 - Buscar pelo Id \n5 - Exibir
Todos \n6 - Persistir Dados \n7 - Recuperar Dados " +
                "\n0 - Finalizar programa ");
            break;
        case "0":
            //Encerrar o programa
            status = false;
            System.out.println("=====");
            System.out.println("Progama finalizado =)");
            scanner1.close();
            break;
        default:
            System.out.println("Opção inválida, por favor,
tente novamente");
    }
}
}
}
```

Resultado da execução

```
run:
=====
1 - Incluir pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir Todos
6 - Persistir Dados
7 - Recuperar Dados
8 - Monstrar Instruções
0 - Finalizar programa
1
F - Pessoa Física | J - Pessoa Jurídica
|
```

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

Elementos estáticos são elementos que não pertencem a classe, por isso o método main usa static.

Para que serve a classe Scanner?

A classe scanner serve para receber entradas dos usuários e atribuir para uma determinada variável.

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

O uso da classe de repositório foi útil para se separar das classes padrões, e também para organizar a lista para sua respectiva pessoa, fazendo uma classe de repositório para cada.