



Campus: Madureira

Curso: Desenvolvimento full stack

Disciplina: Back-end sem banco não tem

Turma: 9001

Semestre: 3º

Nome: Lincoln Miguel da Cruz Antonio

Missão Prática - Nível 3

Todos os códigos usados:

```
package cadastrbd;
```

```
import cadastrbd.model.PessoaFisica;  
import cadastrbd.model.PessoaJuridica;  
import cadastrbd.model.PessoaFisicaDAO;  
import cadastrbd.model.PessoaJuridicaDAO;
```

```
public class CadastroBDTeste {
```

```
    public static void main(String[] args) {  
        PessoaFisicaDAO pessoaFisicaDAO = new  
        PessoaFisicaDAO();
```

```
PessoaJuridicaDAO pessoaJuridicaDAO = new
PessoaJuridicaDAO();
```

```
PessoaFisica pessoaFisica = new PessoaFisica(o, "Joao
Silva", "Rua A", "Cidade A", "Estado A", "123456789",
"joao@example.com", "123.456.789-00");
pessoaFisicaDAO.incluir(pessoaFisica);
```

```
pessoaFisica.setNome("Joao Silva Alterado");
pessoaFisicaDAO.alterar(pessoaFisica);
```

```
for (PessoaFisica pf : pessoaFisicaDAO.getPessoas()) {
    pf.exibir();
}
```

```
pessoaFisicaDAO.excluir(pessoaFisica.getId());
```

```
PessoaJuridica pessoaJuridica = new PessoaJuridica(o,
"Empresa ABC", "Avenida B", "Cidade B", "Estado B",
"987654321", "contato@empresa.com",
"12.345.678/0001-99");
pessoaJuridicaDAO.incluir(pessoaJuridica);
```

```
pessoaJuridicaDAO.alterar(pessoaJuridica);
```

```
for (PessoaJuridica pj : pessoaJuridicaDAO.getPessoas()) {
    pj.exibir();
}
```

```
pessoaJuridicaDAO.excluir(pessoaJuridica.getId());
}
}
```

```
package cadastrbd;

import java.util.Scanner;
import cadastrbd.model.PessoaFisica;
import cadastrbd.model.PessoaJuridica;
import cadastrbd.model.PessoaFisicaDAO;
import cadastrbd.model.PessoaJuridicaDAO;

public class CadastroBD {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        PessoaFisicaDAO pessoaFisicaDAO = new
PessoaFisicaDAO();
        PessoaJuridicaDAO pessoaJuridicaDAO = new
PessoaJuridicaDAO();
        int opcao;

        do {
            System.out.println("=== Sistema de Cadastro ===");
            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("0. Sair");
            System.out.print("Escolha uma opção: ");
            opcao = scanner.nextInt();
            scanner.nextLine();

            switch (opcao) {
                case 1:
                    System.out.print("Escolha o tipo (Fisica/Juridica):
");
```

```
String tipoInclusao = scanner.nextLine();
if (tipoInclusao.equalsIgnoreCase("Fisica")) {
    incluirPessoaFisica(scanner, pessoaFisicaDAO);
} else if (tipoInclusao.equalsIgnoreCase("Juridica"))
{
    incluirPessoaJuridica(scanner,
pessoaJuridicaDAO);
} else {
    System.out.println("Tipo invalido.");
}
break;
```

case 2:

```
System.out.print("Escolha o tipo (Fisica/Juridica):
");
```

```
String tipoAlteracao = scanner.nextLine();
if (tipoAlteracao.equalsIgnoreCase("Fisica")) {
    alterarPessoaFisica(scanner, pessoaFisicaDAO);
} else if
(tipoAlteracao.equalsIgnoreCase("Juridica")) {
    alterarPessoaJuridica(scanner,
pessoaJuridicaDAO);
} else {
    System.out.println("Tipo invalido.");
}
break;
```

case 3:

```
System.out.print("Escolha o tipo (Fisica/Juridica):
");
```

```
String tipoExclusao = scanner.nextLine();
if (tipoExclusao.equalsIgnoreCase("Fisica")) {
    excluirPessoaFisica(scanner, pessoaFisicaDAO);
```

```

        } else if (tipoExclusao.equalsIgnoreCase("Juridica"))
    {
        excluirPessoaJuridica(scanner,
    pessoaJuridicaDAO);
        } else {
            System.out.println("Tipo invalido.");
        }
        break;

    case 4:
        System.out.print("Escolha o tipo (Fisica/Juridica):
    ");

        String tipoExibicaoPorId = scanner.nextLine();
        if (tipoExibicaoPorId.equalsIgnoreCase("Fisica")) {
            exibirPessoaFisica(scanner, pessoaFisicaDAO);
        } else if
    (tipoExibicaoPorId.equalsIgnoreCase("Juridica")) {
            exibirPessoaJuridica(scanner,
    pessoaJuridicaDAO);
        } else {
            System.out.println("Tipo invalido.");
        }
        break;

    case 5:
        System.out.print("Escolha o tipo (Fisica/Juridica):
    ");

        String tipoExibicaoTodos = scanner.nextLine();
        if (tipoExibicaoTodos.equalsIgnoreCase("Fisica")) {
            exibirTodasPessoasFisicas(pessoaFisicaDAO);
        } else if
    (tipoExibicaoTodos.equalsIgnoreCase("Juridica")) {

```

```
exibirTodasPessoasJuridicas(pessoaJuridicaDAO);
    } else {
        System.out.println("Tipo invalido.");
    }
    break;

    case 0:
        System.out.println("Saindo do sistema...");
        break;

    default:
        System.out.println("Opção invalida. Tente
novamente.");
    }
} while (opcao != 0);
```

```
    scanner.close();
}
```

```
    public static void incluirPessoaFisica(Scanner scanner,
PessoaFisicaDAO pessoaFisicaDAO) {
        System.out.print("Nome: ");
        String nome = scanner.nextLine();
    }
```

```
    public static void incluirPessoaJuridica(Scanner scanner,
PessoaJuridicaDAO pessoaJuridicaDAO) {
        System.out.print("Nome: ");
        String nome = scanner.nextLine();

    }
```

```
public static void alterarPessoaFisica(Scanner scanner,
PessoaFisicaDAO pessoaFisicaDAO) {
}
```

```
public static void alterarPessoaJuridica(Scanner scanner,
PessoaJuridicaDAO pessoaJuridicaDAO) {
}
```

```
public static void excluirPessoaFisica(Scanner scanner,
PessoaFisicaDAO pessoaFisicaDAO) {
    System.out.print("Informe o ID da Pessoa Fisica para
excluir: ");
    int id = scanner.nextInt();
}
```

```
public static void excluirPessoaJuridica(Scanner scanner,
PessoaJuridicaDAO pessoaJuridicaDAO) {
    System.out.print("Informe o ID da Pessoa Juridica para
excluir: ");
    int id = scanner.nextInt();
}
```

```
public static void exibirPessoaFisica(Scanner scanner,
PessoaFisicaDAO pessoaFisicaDAO) {
    System.out.print("Informe o ID da Pessoa Fisica: ");
    int id = scanner.nextInt();
}
```

```
public static void exibirPessoaJuridica(Scanner scanner,
PessoaJuridicaDAO pessoaJuridicaDAO) {
    System.out.print("Informe o ID da Pessoa Juridica: ");
    int id = scanner.nextInt();
}
```

```

    public static void
    exibirTodasPessoasFisicas(PessoaFisicaDAO pessoaFisicaDAO)
    {
    }

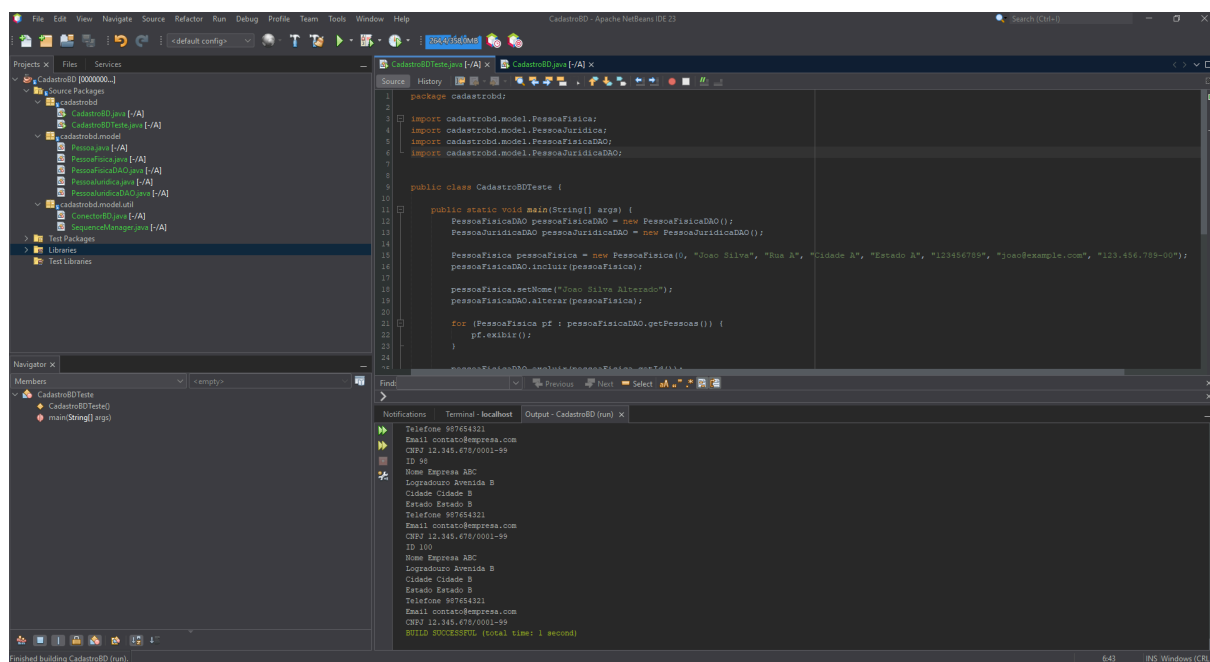
```

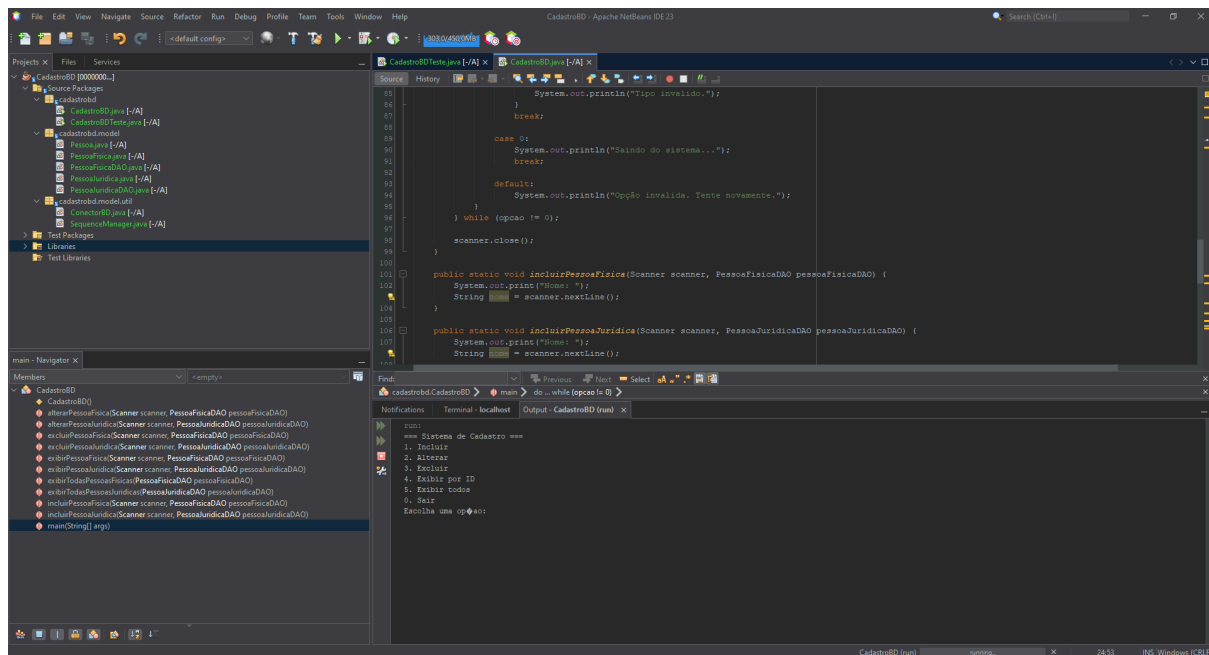
```

    public static void
    exibirTodasPessoasJuridicas(PessoaJuridicaDAO
    pessoaJuridicaDAO) {
    }
}

```

Resultados dos códigos:





Análise e Conclusão:

- a. Quais as diferenças entre a persistência em arquivo e a persistência em banco de dados?**

Arquivo armazena os dados em formatos mais simples e banco de dados já usa um armazenamento mais organizado.

- b. Como o uso de operador lambda simplificou a impressão dos valores contidos nas entidades, nas versões mais recentes do Java?**

Melhorou de forma que ficou mais funcional e menos complicado pra escrever e fazer manutenção em códigos.

- c. Por que métodos acionados diretamente pelo método main, sem o uso de um objeto, precisam ser marcados como static?**

Porque eles pertencem apenas a aquela classe em si e não em uma classe específica.