



Curso
Desenvolvimento full Stack
Turma

Campus
POLO INGL RIO VERM – FLORIANOPOLIS - SC
Nome
Yuri Frederick de Sousa Cunha Bernardo

Disciplina
Nível 3: BackEnd sem banco não tem

3ºSemestre

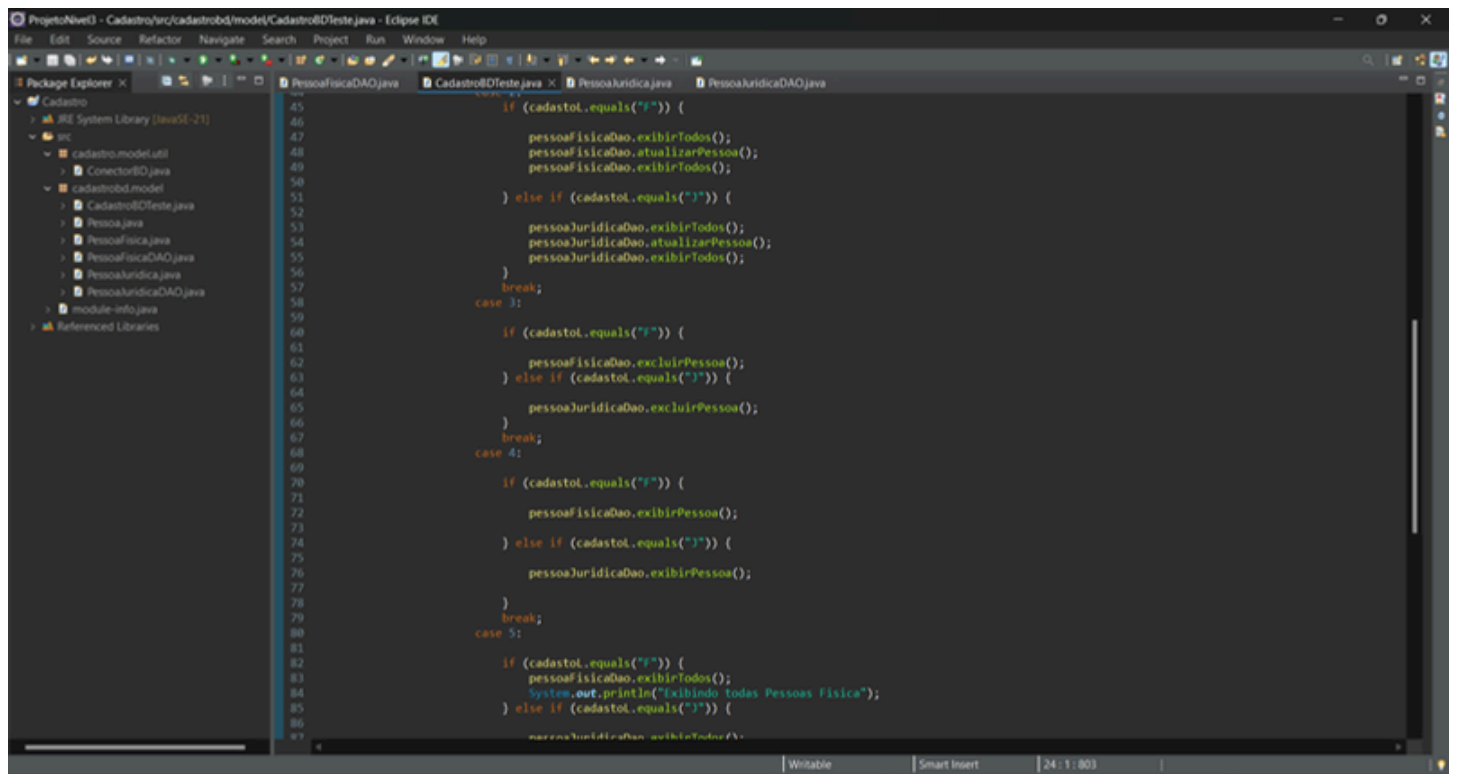
Matricula
2023 0407 7282

BackEnd sem banco não tem

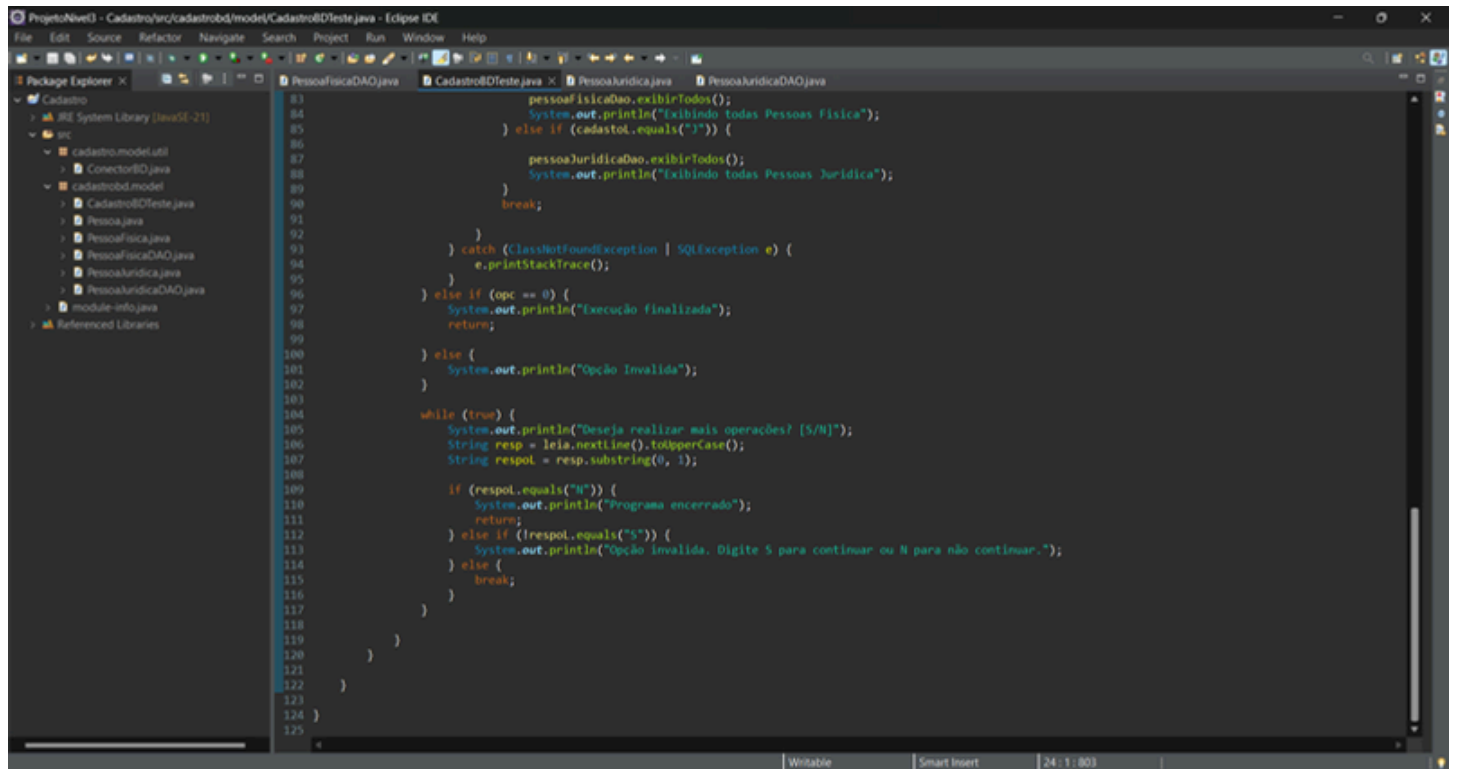
Criação de aplicativo Java, com acesso ao banco dedados SQL Server através do middleware JDBC.

Classe Principal:

```
1 package cadastrbd.model;
2
3 import java.sql.SQLException;
4
5 public class CadastroBDTeste {
6
7     public static void main(String[] args) throws ClassNotFoundException, SQLException {
8
9         PessoaFisicaDAO pessoaFisicaDao = new PessoaFisicaDAO();
10        PessoaJuridicaDAO pessoaJuridicaDao = new PessoaJuridicaDAO();
11
12        try (Scanner leia = new Scanner(System.in)) {
13            while (true) {
14
15                try {
16                    System.out.println("Cadastro De Pessoas");
17                    System.out.println();
18                    System.out.println("1 - Incluir Pessoa\n" + "2 - Alterar Pessoa\n" + "3 - Excluir Pessoa\n"
19                        + "4 - Buscar pelo Id\n" + "5 - Exibir Todos Cadastrados\n" + "0 - Finalizar Programa ");
20                    System.out.println();
21                    System.out.println("Escolha uma das opções á cima: ");
22
23                    Integer opc = leia.nextInt();
24                    leia.nextLine();
25
26                    if (opc >= 1 && opc <= 5) {
27
28                        System.out.println("Digite:\n F - Pessoa Fisica | J - Pessoa Juridica");
29                        String cadast = leia.nextLine().toUpperCase();
30                        String cadastol = cadast.substring(0, 1);
31
32                        try {
33                            switch (opc) {
34                                case 1:
35                                    if (cadastol.equals("F")) {
36                                        pessoaFisicaDao.inserindoPessoa();
37                                    } else if (cadastol.equals("J")) {
38                                        pessoaJuridicaDao.inserirPessoa();
39                                    }
40                                }
41                            }
42                            break;
43                        }
44                    }
45                }
46            }
47        }
48    }
49 }
```



```
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125
```



```
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125
```

Resultados da execução dos códigos:

```
Problems @ Javadoc Declaration Console ×
CadastroBDTeste [Java Application] C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot
Cadastro De Pessoas

1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir Todos Cadastrados
0 - Finalizar Programa

Escolha uma das opções á cima:
```

Instanciar uma pessoa e persistir no banco de dados.

Pessoa Física

```
Escolha uma das opções á cima:
1
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
f
Nome:
Maria
Logradouro:
Casa
Cidade:
Florianopolis
Estado:
Santa Catarina
Telefone:
48 97070-7070
Email:
maria.floripa@gmail.com
CPF:
000000000000
Cadastrando Pessoa Fisica
Deseja realizar mais operações? [S/N]
```

Pessoa Juridica

```
Escolha uma das opções á cima:
1
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
j
Nome:
NIKE
Logradouro:
Avenida
Cidade:
São Paulo
Estado:
São Paulo
Telefone:
11 27070-7070
Email:
nike.sp@hotmail.com
CNPJ:
11111111111111
Cadastrando Pessoa Juridica
Deseja realizar mais operações? [S/N]
```

Alterar os dados de pessoa no banco.

Pessoa Física

```
CadastrobDTeste [Java Application] C:\Program Files\Eclipse Adoptium
Digite o Id da pessoa que deseja atualizar:
ID:
2
Nome:
João
Logradouro:
Rua
Estado:
São Paulo
Cidade:
Guarulhos
Telefone:
11 9090-9090
Email:
joaozinho@hotmail.com
CPF:
111111111111111
Alterado com sucesso!!!
=====
ID: 2
Nome: João
Logradouro: Rua
Cidade: Guarulhos
Telefone: 11 9090-9090
Estado: São Paulo
Email: joaozinho@hotmail.com
CPF: 111111111111111
=====
ID: 7
Nome: Maria
Logradouro: Casa
Cidade: Florianopolis
Telefone: 48 97070-7070
Estado: Santa Catarina
Email: maria.floripa@gmail.com
CPF: 000000000000
=====
Deseja realizar mais operações? [S/N]
```

Pessoa Juridica

```
CadastrobDTeste [Java Application] C:\Program Files\Eclipse Adoptium
Digite o Id da pessoa que deseja atualizar:
2
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
j
=====
ID: 4
Nome: NIKE
Logradouro: Avenida
Cidade: São Paulo
Telefone: 11 27070-7070
Estado: São Paulo
Email: nike.sp@hotmail.com
CNPJ: 111111111111111
=====
Digite o Id da pessoa que deseja atualizar:
ID:
4
Nome:
NIKE
Logradouro:
Rodovia
Estado:
São Paulo
Cidade:
Barra Funda
Telefone:
222222222222222
Email:
nike.new@hotmail.com
CNPJ:
2222222222222222
Alterado com sucesso!!!
=====
ID: 4
Nome: NIKE
Logradouro: Rodovia
Cidade: Barra Funda
Telefone: 222222222222222
Estado: São Paulo
Email: nike.new@hotmail.com
CNPJ: 2222222222222222
=====
Deseja realizar mais operações? [S/N]
```

Consultar todas as pessoas do banco de dados e listar no console

Pessoa Física

```
Escolha uma das opções á cima:
5
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
f
=====
ID: 2
Nome: Yuri
Logradouro: Apartamento
Cidade: Florianopolis
Telefone: (11) 97021-2190
Estado: SC
Email: yuri.bernardo@alunos.sc.senac.com
CPF: 41992258899
=====
ID: 7
Nome: Maria
Logradouro: Casa
Cidade: Florianopolis
Telefone: 48 97070-7070
Estado: Santa Catarina
Email: maria.floripa@gmail.com
CPF: 000000000000
=====
Exibindo todas Pessoas Fisica
Deseja realizar mais operações? [S/N]
```

Pessoa Jurídica

```
Escolha uma das opções á cima:
5
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
j
=====
ID: 4
Nome: NIKE
Logradouro: Avenida
Cidade: São Paulo
Telefone: 11 27070-7070
Estado: São Paulo
Email: nike.sp@hotmail.com
CNPJ: 11111111111111
=====
Exibindo todas Pessoas Juridica
Deseja realizar mais operações? [S/N]
```

Excluir a pessoa criada anteriormente no banco.

Pessoa Física

```
Cadastro De Pessoas

1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir Todos Cadastrados
0 - Finalizar Programa

Escolha uma das opções á cima:
3
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
f
Digite o ID da pessoa que deseja excluir:
7
Cadastro excluido
```

Pessoa Jurídica

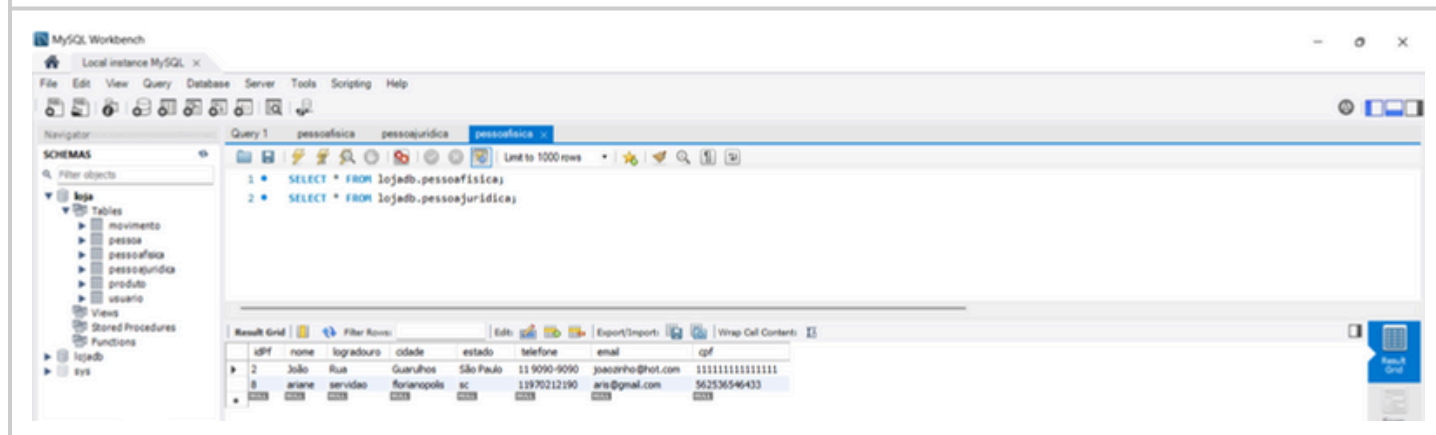
```
Cadastro De Pessoas

1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir Todos Cadastrados
0 - Finalizar Programa

Escolha uma das opções á cima:
3
Digite:
F - Pessoa Fisica | J - Pessoa Juridica
j
Digite o ID da pessoa que deseja excluir:
4
Cadastro excluido
Deseja realizar mais operações? [S/N]
```

Banco de dados:

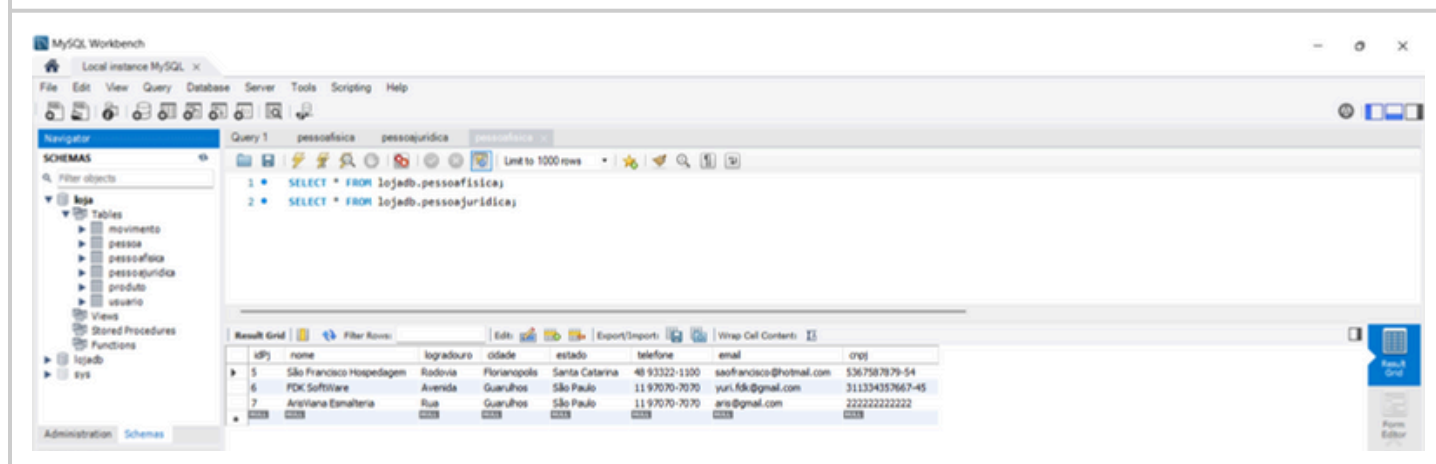
Pessoa Física:



The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying two SQL queries: `SELECT * FROM lojadb.pessoafisica;` and `SELECT * FROM lojadb.pessoajuridica;`. The 'Result Grid' shows the results of the first query, which includes columns: idpf, nome, logradouro, cidade, estado, telefone, email, and cpf. The results are as follows:

idpf	nome	logradouro	cidade	estado	telefone	email	cpf
2	João	Rua	Guarulhos	São Paulo	11 9090-9090	joaozinho@hotmail.com	111111111111111111
8	ariane	serviço	foranópolis	sc	11970212190	aria@gmail.com	562536546433

Pessoa Jurídica:



The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying two SQL queries: `SELECT * FROM lojadb.pessoafisica;` and `SELECT * FROM lojadb.pessoajuridica;`. The 'Result Grid' shows the results of the second query, which includes columns: idpj, nome, logradouro, cidade, estado, telefone, email, and cnpj. The results are as follows:

idpj	nome	logradouro	cidade	estado	telefone	email	cnpj
5	São Francisco Hospedagem	Rodovia	Foranópolis	Santa Catarina	48 93322-1100	saofrancisco@hotmail.com	5367587979-54
6	PDF Software	Avenida	Guarulhos	São Paulo	11 97070-7070	yuri.fdk@gmail.com	311334357667-45
7	Arquitana Esmalteria	Rua	Guarulhos	São Paulo	11 97070-7070	aria@gmail.com	222222222222

Análise e Conclusão:

Importância dos Componentes de Middleware (como o JDBC):

O middleware é um software que permite que diferentes aplicações se comuniquem entre si de forma inteligente e eficiente.

Ele atua como uma ponte entre tecnologias, ferramentas e bancos de dados, integrando-os em um único sistema.

Sem o middleware, seria necessário criar módulos de troca de dados para cada componente de software, o que é desafiador em sistemas modernos com vários microsserviços¹.

O JDBC (Java Database Connectivity) é um exemplo de middleware usado para conectar aplicativos Java a bancos de dados relacionais.

Diferença entre Statement e PreparedStatement:

O Statement é usado para executar consultas SQL estáticas, onde os valores dos parâmetros não mudam.

O PreparedStatement, por outro lado, é usado para consultas parametrizadas, permitindo que os valores dos parâmetros sejam definidos dinamicamente.

O PreparedStatement é mais seguro contra injeção de SQL, pois os parâmetros são tratados automaticamente.

Padrão DAO (Data Access Object):

O padrão DAO melhora a manutenibilidade do software, separando a lógica de acesso a dados da lógica de negócios.

Ele fornece uma abstração para o acesso a dados, permitindo que as mudanças no banco de dados sejam isoladas.

Com o DAO, é mais fácil substituir ou atualizar o acesso a dados sem afetar outras partes do sistema.

Herança no Banco de Dados Relacional:

Quando lidamos com um modelo estritamente relacional, a herança é geralmente representada por tabelas relacionadas.

Por exemplo, uma tabela “Pessoa” pode ter subtipos como “Cliente” e “Funcionário”, cada um com suas próprias colunas específicas.

Essa abordagem permite consultas eficientes e mantém a integridade dos dados.

Persistência em Arquivo vs. Persistência em Banco de Dados:

A persistência de dados refere-se à garantia de que as informações sejam armazenadas de forma consistente e não sejam perdidas após o encerramento da sessão.

Persistência em arquivo:

Os dados são armazenados em arquivos no sistema de arquivos do computador.

Os arquivos podem ser acessados e manipulados mesmo após o encerramento do programa ou sistema.

Persistência em banco de dados:

Os dados são armazenados em bancos de dados, como MySQL, PostgreSQL ou Oracle.

Essa abordagem oferece maior segurança, escalabilidade e gerenciamento de dados.

Os registros são permanentes e podem ser recuperados sempre que necessário¹.

Operadores Lambda no Java:

As expressões lambda foram introduzidas no Java 8 para adicionar técnicas de linguagens funcionais.

Uma função lambda é uma função sem declaração explícita, permitindo escrever código de forma mais concisa.

Elas simplificam a escrita de funções, como classes internas, Listeners e Threads2.

Métodos Acionados pelo Método Main (static):

Os métodos acionados diretamente pelo método main precisam ser marcados como static.

Isso ocorre porque o método main é estático e não requer uma instância da classe para ser chamado.

Métodos estáticos pertencem à classe como um todo, enquanto métodos não estáticos estão associados a instâncias específicas (objetos)3

Link GitHub: [Yuri6406/3Semestre-ProjetoNivel3 \(github.com\)](https://github.com/Yuri6406/3Semestre-ProjetoNivel3).