



Campus: POLO SAGUAÇU - JOINVILLE - SC

Curso: DESENVOLVIMENTO FULL STACK

Disciplina: Back-end Sem Banco Não Tem

Turma: 9001

Semestre: 1º Semestre (2024)

Aluno: Jederson Borges de Oliveira

Link: <https://github.com/JedersonBorges/CadastroBD.git>

BackEnd sem banco não tem

2º Procedimento

Objetivos da prática

1. Implementar persistência com base no middleware JDBC.
2. Utilizar o padrão DAO (Data Access Object) no manuseio de dados.
3. Implementar o mapeamento objeto-relacional em sistemas Java.
4. Criar sistemas cadastrais com persistência em banco relacional.
5. No final do exercício, o aluno terá criado um aplicativo cadastral com uso do SQL
6. Server na persistência de dados.

Código utilizado

Classe CadastroBDTeste

```
package cadastrobd;

import cadastrobd.model.PessoaFisica;
import cadastrobd.model.PessoaFisicaDAO;
import cadastrobd.model.PessoaJuridica;
import cadastrobd.model.PessoaJuridicaDAO;

import java.util.List;
import java.util.Scanner;

public class CadastroBDTeste {

    private static final Scanner scanner = new Scanner(System.in);
```

```

        private static final PessoaFisicaDAO pessoaFisicaDAO = new
PessoaFisicaDAO();
        private static final PessoaJuridicaDAO pessoaJuridicaDAO = new
PessoaJuridicaDAO();

        public static void main(String[] args) {
            while (true) {
                System.out.println("Escolha uma opcao:");
                System.out.println("1. Incluir Pessoa Fisica");
                System.out.println("2. Alterar Pessoa Fisica");
                System.out.println("3. Consultar Pessoas Fisicas");
                System.out.println("4. Excluir Pessoa Fisica");
                System.out.println("5. Incluir Pessoa Juridica");
                System.out.println("6. Alterar Pessoa Juridica");
                System.out.println("7. Consultar Pessoas Juridicas");
                System.out.println("8. Excluir Pessoa Juridica");
                System.out.println("9. Buscar Pessoa");
                System.out.println("0. Sair");
                System.out.print("Opcao: ");
                int opcao = scanner.nextInt();
                scanner.nextLine();

                switch (opcao) {
                    case 1:
                        incluirPessoaFisica();
                        break;
                    case 2:
                        alterarPessoaFisica();
                        break;
                    case 3:
                        consultarPessoasFisicas();
                        break;
                    case 4:
                        excluirPessoaFisica();
                        break;
                    case 5:
                        incluirPessoaJuridica();
                        break;
                    case 6:
                        alterarPessoaJuridica();
                        break;
                    case 7:
                        consultarPessoasJuridicas();
                        break;
                    case 8:
                        excluirPessoaJuridica();
                        break;
                    case 9:
                        buscarPessoa();
                        break;
                    case 0:
                        System.out.println("Saindo...");
                        return;
                    default:
                        System.out.println("Opcao inválida. Tente
novamente.");
                }
            }
        }

        private static void incluirPessoaFisica() {

```

```

        System.out.println("Digite o CPF da Pessoa Fisica:");
        long cpf = scanner.nextLong();
        scanner.nextLine();

        System.out.println("Digite o ID da Pessoa:");
        int idPessoa = scanner.nextInt();
        scanner.nextLine();

        System.out.println("Digite o nome da Pessoa Fisica:");
        String nome = scanner.nextLine();

        System.out.println("Digite o logradouro:");
        String logradouro = scanner.nextLine();

        System.out.println("Digite a cidade:");
        String cidade = scanner.nextLine();

        System.out.println("Digite o estado:");
        String estado = scanner.nextLine();

        System.out.println("Digite o telefone:");
        String telefone = scanner.nextLine();

        System.out.println("Digite o email:");
        String email = scanner.nextLine();

        PessoaFisica pessoaFisica = new PessoaFisica(cpf, idPessoa,
nome, logradouro, cidade, estado, telefone, email);
        pessoaFisicaDAO.incluir(pessoaFisica);
        System.out.println("Pessoa Fisica incluída com sucesso.");
    }

    private static void alterarPessoaFisica() {
        System.out.println("Digite o ID da Pessoa Fisica para
alterar:");
        int idPessoa = scanner.nextInt();
        scanner.nextLine();

        PessoaFisica pessoaFisica = buscarPessoaFisicaPorId(idPessoa);

        if (pessoaFisica != null) {
            System.out.println("Digite o novo nome da Pessoa
Fisica:");
            pessoaFisica.setNome(scanner.nextLine());

            System.out.println("Digite o novo logradouro:");
            pessoaFisica.setLogradouro(scanner.nextLine());

            System.out.println("Digite a nova cidade:");
            pessoaFisica.setCidade(scanner.nextLine());

            System.out.println("Digite o novo estado:");
            pessoaFisica.setEstado(scanner.nextLine());

            System.out.println("Digite o novo telefone:");
            pessoaFisica.setTelefone(scanner.nextLine());

            System.out.println("Digite o novo email:");
            pessoaFisica.setEmail(scanner.nextLine());

            pessoaFisicaDAO.atualizar(pessoaFisica);

```

```

        System.out.println("Pessoa Fisica atualizada com
sucesso.");
    } else {
        System.out.println("Pessoa Fisica nao encontrada.");
    }
}

private static void consultarPessoasFisicas() {
    List<PessoaFisica> pessoasFisicas =
    pessoaFisicaDAO.consultarTodos();
    System.out.println("Pessoas Fisicas:");
    for (PessoaFisica pf : pessoasFisicas) {
        pf.exibir();
    }
}

private static void excluirPessoaFisica() {
    System.out.println("Digite o ID da Pessoa Fisica para
excluir:");
    int idPessoa = scanner.nextInt();
    scanner.nextLine();

    PessoaFisica pessoaFisica = buscarPessoaFisicaPorId(idPessoa);

    if (pessoaFisica != null) {
        pessoaFisicaDAO.excluir(pessoaFisica);
        System.out.println("Pessoa Fisica excluida com sucesso.");
    } else {
        System.out.println("Pessoa Fisica nao encontrada.");
    }
}

private static void incluirPessoaJuridica() {
    System.out.println("Digite o CNPJ da Pessoa Juridica:");
    long cnpj = scanner.nextLong();
    scanner.nextLine();

    System.out.println("Digite o ID da Pessoa:");
    int idPessoa = scanner.nextInt();
    scanner.nextLine();

    System.out.println("Digite o nome da Pessoa Juridica:");
    String nome = scanner.nextLine();

    System.out.println("Digite o logradouro:");
    String logradouro = scanner.nextLine();

    System.out.println("Digite a cidade:");
    String cidade = scanner.nextLine();

    System.out.println("Digite o estado:");
    String estado = scanner.nextLine();

    System.out.println("Digite o telefone:");
    String telefone = scanner.nextLine();

    System.out.println("Digite o email:");
    String email = scanner.nextLine();

    PessoaJuridica pessoaJuridica = new PessoaJuridica(cnpj,
idPessoa, nome, logradouro, cidade, estado, telefone, email);
}

```

```

        pessoaJuridicaDAO.incluir(pessoaJuridica);
        System.out.println("Pessoa Juridica incluída com sucesso.");
    }

    private static void alterarPessoaJuridica() {
        System.out.println("Digite o ID da Pessoa Juridica para
alterar:");
        int idPessoa = scanner.nextInt();
        scanner.nextLine();

        PessoaJuridica pessoaJuridica =
buscarPessoaJuridicaPorId(idPessoa);

        if (pessoaJuridica != null) {
            System.out.println("Digite o novo nome da Pessoa
Juridica:");
            pessoaJuridica.setNome(scanner.nextLine());

            System.out.println("Digite o novo logradouro:");
            pessoaJuridica.setLogradouro(scanner.nextLine());

            System.out.println("Digite a nova cidade:");
            pessoaJuridica.setCidade(scanner.nextLine());

            System.out.println("Digite o novo estado:");
            pessoaJuridica.setEstado(scanner.nextLine());

            System.out.println("Digite o novo telefone:");
            pessoaJuridica.setTelefone(scanner.nextLine());

            System.out.println("Digite o novo email:");
            pessoaJuridica.setEmail(scanner.nextLine());

            pessoaJuridicaDAO.atualizar(pessoaJuridica);
            System.out.println("Pessoa Juridica atualizada com
sucesso.");
        } else {
            System.out.println("Pessoa Juridica não encontrada.");
        }
    }

    private static void consultarPessoasJuridicas() {
        List<PessoaJuridica> pessoasJuridicas =
pessoaJuridicaDAO.consultarTodos();
        System.out.println("Pessoas Juridicas:");
        for (PessoaJuridica pj : pessoasJuridicas) {
            pj.exibir();
        }
    }

    private static void excluirPessoaJuridica() {
        System.out.println("Digite o ID da Pessoa Juridica para
excluir:");
        int idPessoa = scanner.nextInt();
        scanner.nextLine();

        PessoaJuridica pessoaJuridica =
buscarPessoaJuridicaPorId(idPessoa);

        if (pessoaJuridica != null) {
            pessoaJuridicaDAO.excluir(pessoaJuridica);
        }
    }

```

```

        System.out.println("Pessoa Juridica excluida com
sucesso.");
    } else {
        System.out.println("Pessoa Juridica nao encontrada.");
    }
}

private static void buscarPessoa() {
    System.out.println("Escolha o tipo: F - Pessoa Fisica | J -
Pessoa Juridica");
    String tipo = scanner.nextLine().toUpperCase();

    if (tipo.equals("F")) {
        System.out.println("Digite o ID da Pessoa Fisica para
buscar:");
        int id = scanner.nextInt();
        scanner.nextLine();
        buscarPessoaFisicaPorId(id);
    } else if (tipo.equals("J")) {
        System.out.println("Digite o ID da Pessoa Juridica para
buscar:");
        int id = scanner.nextInt();
        scanner.nextLine();
        buscarPessoaJuridicaPorId(id);
    } else {
        System.out.println("Tipo de pessoa inválido.");
    }
}

private static PessoaFisica buscarPessoaFisicaPorId(int id) {
    PessoaFisica pessoaFisica = pessoaFisicaDAO.buscarPorId(id);
    if (pessoaFisica != null) {
        pessoaFisica.exibir();
    } else {
        System.out.println("Pessoa Fisica nao encontrada.");
    }
    return pessoaFisica;
}

private static PessoaJuridica buscarPessoaJuridicaPorId(int id) {
    PessoaJuridica pessoaJuridica =
pessoaJuridicaDAO.buscarPorId(id);
    if (pessoaJuridica != null) {
        pessoaJuridica.exibir();
    } else {
        System.out.println("Pessoa Juridica nao encontrada.");
    }
    return pessoaJuridica;
}
}

```

Resultados:

Menu:

```
run:
[java] Escolha uma opcao:
[java] 1. Incluir Pessoa Fisica
[java] 2. Alterar Pessoa Fisica
[java] 3. Consultar Pessoas Fisicas
[java] 4. Excluir Pessoa Fisica
[java] 5. Incluir Pessoa Juridica
[java] 6. Alterar Pessoa Juridica
[java] 7. Consultar Pessoas Juridicas
[java] 8. Excluir Pessoa Juridica
[java] 9. Buscar Pessoa
[java] 0. Sair
```

Incluir Pessoa Fisica:

VSCode

```
[java] 9. Buscar Pessoa
[java] 0. Sair
1
[java] Opcao: Digite o CPF da Pessoa Fisica:
11122233344
[java] Digite o ID da Pessoa:
23
[java] Digite o nome da Pessoa Fisica:
Fulano da Silva
[java] Digite o logradouro:
Rua do fim do mundo
[java] Digite a cidade:
Joinville
[java] Digite o estado:
SC
[java] Digite o telefone:
4799999995
[java] Digite o email:
FULANO@HOTMAIL.COM
[java] ... 12:30:23 PM ...
```

SSMS

Resultados		Mensagens						
	idPessoa	nome	logradouro	cidade	estado	telefone	email	cpf
1	7	Joao	Rua 12, casa 3, Quitanda	Riacho do Sul	PA	1111-1111	joao@riacho.com	11111111111
2	23	Fulano da Silva	Rua do fim do mundo	Joinville	SC	4799999995	FULANO@HOTMAIL.COM	11122233344

Incluir Pessoa Juridica:

```
run:
[java] Escolha uma opcao:
[java] 1. Incluir Pessoa Fisica
[java] 2. Alterar Pessoa Fisica
[java] 3. Consultar Pessoas Fisicas
[java] 4. Excluir Pessoa Fisica
[java] 5. Incluir Pessoa Juridica
[java] 6. Alterar Pessoa Juridica
[java] 7. Consultar Pessoas Juridicas
[java] 8. Excluir Pessoa Juridica
[java] 9. Buscar Pessoa
[java] 0. Sair
5
[java] Opcao: Digite o CNPJ da Pessoa Juridica:
11112222333344
[java] Digite o ID da Pessoa:
55
[java] Digite o nome da Pessoa Juridica:
Estacio
[java] Digite o logradouro:
Rua XV de Novembro
[java] Digite a cidade:
Joinville
[java] Digite o estado:
SC
[java] Digite o telefone:
11112222
[java] Digite o email:
estacio@estacio.com
```

SSMS

Resultados		Mensagens						
	idPessoa	nome	logradouro	cidade	estado	telefone	email	cnpj
1	55	Estacio	Rua XV de Novembro	Joinville	SC	11112222	estacio@estacio.com	11112222333344

run:

```
[java] Escolha uma opcao:  
[java] 1. Incluir Pessoa Fisica  
[java] 2. Alterar Pessoa Fisica  
[java] 3. Consultar Pessoas Fisicas  
[java] 4. Excluir Pessoa Fisica  
[java] 5. Incluir Pessoa Juridica  
[java] 6. Alterar Pessoa Juridica  
[java] 7. Consultar Pessoas Juridicas  
[java] 8. Excluir Pessoa Juridica  
[java] 9. Buscar Pessoa  
[java] 0. Sair
```

3

```
[java] Opcao: Pessoas Fisicas:  
[java] ID: 7  
[java] Nome: Joao  
[java] Logradouro: Rua 12, casa 3, Quitanda  
[java] Cidade: Riacho do Sul  
[java] Estado: PA  
[java] Telefone: 1111-1111  
[java] Email: joao@riacho.com  
[java] CPF: 11111111111  
[java] ID: 1  
[java] Nome: Jederson  
[java] Logradouro: Rua XV de Novembro  
[java] Cidade: Joinville  
[java] Estado: SC  
[java] Telefone: 47999999191  
[java] Email: jederson@estacio.com.br  
[java] CPF: 12345678901  
[java] Escolha uma opcao:  
[java] 1. Incluir Pessoa Fisica  
[java] 2. Alterar Pessoa Fisica  
[java] 3. Consultar Pessoas Fisicas  
[java] 4. Excluir Pessoa Fisica  
[java] 5. Incluir Pessoa Juridica  
[java] 6. Alterar Pessoa Juridica  
[java] 7. Consultar Pessoas Juridicas  
[java] 8. Excluir Pessoa Juridica  
[java] 9. Buscar Pessoa  
[java] 0. Sair
```

Consultar Pessoas Juridicas:

run:

```
[java] Escolha uma opcao:  
[java] 1. Incluir Pessoa Fisica  
[java] 2. Alterar Pessoa Fisica  
[java] 3. Consultar Pessoas Fisicas  
[java] 4. Excluir Pessoa Fisica  
[java] 5. Incluir Pessoa Juridica  
[java] 6. Alterar Pessoa Juridica  
[java] 7. Consultar Pessoas Juridicas  
[java] 8. Excluir Pessoa Juridica  
[java] 9. Buscar Pessoa  
[java] 0. Sair
```

7

```
[java] Opcao: Pessoas Juridicas:  
[java] ID: 15  
[java] Nome: JJC  
[java] Logradouro: Rua 11, Centro  
[java] Cidade: Riacho do Norte  
[java] Estado: PA  
[java] Telefone: 1212-1212  
[java] Email: jcc@riacho.com  
[java] CNPJ: 22222222222222  
[java] Escolha uma opcao:  
[java] 1. Incluir Pessoa Fisica  
[java] 2. Alterar Pessoa Fisica  
[java] 3. Consultar Pessoas Fisicas  
[java] 4. Excluir Pessoa Fisica  
[java] 5. Incluir Pessoa Juridica  
[java] 6. Alterar Pessoa Juridica  
[java] 7. Consultar Pessoas Juridicas  
[java] 8. Excluir Pessoa Juridica  
[java] 9. Buscar Pessoa  
[java] 0. Sair
```

Análise e Conclusão

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

Persistência em arquivo é mais simples e direta, mas menos eficiente para grandes volumes de dados, enquanto a persistência em banco de dados oferece estrutura, segurança e consultas avançadas, sendo mais adequada para aplicações com mais dados.

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

Os operadores lambda facilitam a impressão de valores porque deixam o código mais direto e fácil de ler, sem a necessidade de criar classes anônimas, tornando tudo mais limpo e simples.

Porque métodos acionados diretamente pelo método main, sem o uso de um objeto, precisam ser marcados como static?

Métodos acionados diretamente pelo main precisam ser static porque não há uma instância do objeto para chamar esses métodos.