



**UNIVERSIDADE ESTÁCIO DE SÁ**  
**FULLSTACK**

## **Mundo 03 - Nível 03**

**BackEnd sem banco não tem.**

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

Herval Rosano Dantas  
Matrícula 202205119203

RIO DE JANEIRO – RJ  
2023

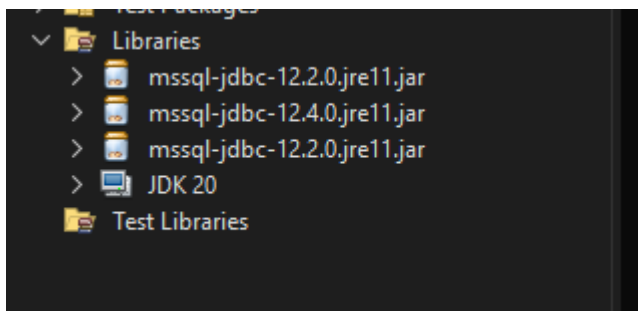
## Objetivo da Prática

- Implementar persistência com base no middleware JDBC.
- Utilizar o padrão DAO (Data Access Object) no manuseio de dados.
- Implementar o mapeamento objeto-relacional em sistemas Java.
- Criar sistemas cadastrais com persistência em banco relacional.

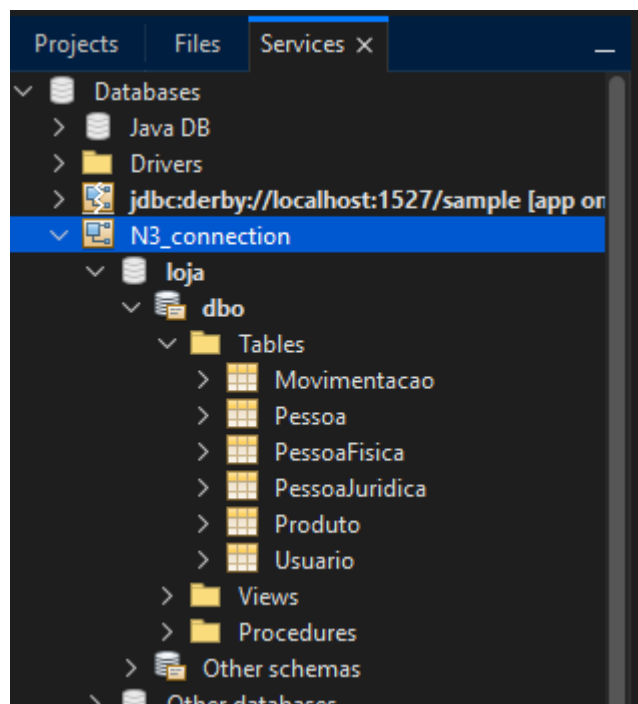
Na conclusão deste trabalho, terá sido criado um aplicativo cadastral com uso do SQL Server na persistência de dados.

## 1º Procedimento – Mapeamento Objeto-Relacional e DAO

1. Criar o projeto e configurar as bibliotecas necessárias



2. Configurar o acesso ao banco pela aba de serviços do NetBeans.

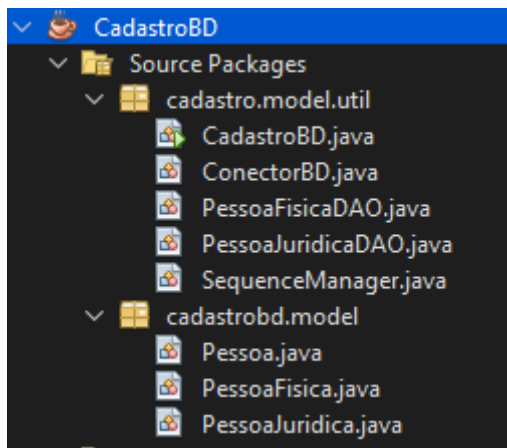


3- Criar o pacote **cadastrobd.model**, e nele criar as classes:

Classe Pessoa, Classe PessoaFisica, herdando de Pessoa, com acréscimo do campo cpf, Classe PessoaJuridica, herdando de Pessoa, com acréscimo do campo cnpj.

4. Criar o pacotes **cadastro.model.util**, para inclusão das classes:

ConectorBD, SequenceManager, classes no padrão DAO PessoaFisicaDAO e PessoaJuridicaDAO.



Todas as funcionalidades deste sistema de cadastro serão mostradas **no segundo procedimento** onde o mesmo será alimentado através de um cadastro em modo texto.

## Exibindo todas Pessoas Físicas

The image displays two screenshots of a Java application running in an IDE.

**Left Screenshot:** The application window is titled "CadastroBD (run)". It features a menu with options: "1 - Incluir Pessoa", "2 - Alterar Pessoa", "3 - Excluir Pessoa", "4 - Buscar pelo Id", "5 - Exibir todos", and "0 - Sair do Programa". Below the menu, a message "Escolha uma opção: 5" is shown. The "F - Pessoa Física | J - Pessoa Jurídica" option is selected, and the "Exibindo dados Pessoa Física..." section is active. The form displays details for a person with ID 1: "Nome: Herval Rosano Dantas", "CPF: 0111940066-55", "Logradouro: Rua Ituverava 866", "Cidade: Rio de Janeiro", "Estado: RJ", "Telefone: 21991870000", and "Email: hervaldantas@gmail.com".

**Right Screenshot:** The application window is titled "CadastroBD (run)". It displays a table with 9 rows of data. The table has columns: "#", "idPessoaFisica", "idPessoa", "nome", "logradouro", "cidade", "estado", "telefone", "email", and "CPF". The data is as follows:

#	idPessoaFisica	idPessoa	nome	logradouro	cidade	estado	telefone	email	CPF
1	1	1	Herval Rosano Dantas	Rua Ituverava 866	Rio de Janeiro	RJ	21991870000	hervaldantas@gmail.com	0111940066-55
2	2	2	Angela Agostinho da C. Dantas	Rua Araguaia 111	Rio de Janeiro	RJ	11133328909	angeladantas@gmail.com	01220066-20
3	3	3	Antonio Dantas	Rua Tirol 777	Rio de Janeiro	RJ	11991098789	antoniodantas@gmail.com	0444494066-50
4	4	4	Carlos Ronan Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678903	calosronan@gmail.com	0122094005-40
5	5	5	José Ronilson	Rua Cascavel 2	Rio de Janeiro	RJ	12345678904	ronsilsondantas2@gmail.com	022055785-50
6	6	6	Rosângela Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	33448922545	rosangeladantas@gmail.com	0133940066-70
7	7	7	Rossana Daguia Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	4446786666	rossanadaguia@gmail.com	0244005-33
8	8	8	Rosânia Cristina Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678444	rosaniacris@gmail.com	01356209405-69
9	9	9	Francisca Natália Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678111	chica@gmail.com	012334005-11

Exibindo todas Pessoas Jurídicas

The screenshot displays a Java application running in an IDE. The application has a menu with options: 1 - Incluir Pessoa, 2 - Alterar Pessoa, 3 - Excluir Pessoa, 4 - Buscar pelo Id, 5 - Exibir todos, and 0 - Sair do Programa. The user has selected option 5, and the application displays the details of three people: Digital Minda, Britex, and Digital Minda. The details include CNPJ, Logradouro, Cidade, Estado, Telefone, and Email. The application is running in a terminal window titled 'Output - CadastroBD (run) X'. The database query result is shown in a table with columns: #, idPessoaJuridica, idPessoa, nome, logradouro, cidade, estado, telefone, email, and CNPJ. The query is 'SELECT TOP 100 \* FROM dbo... X' and it fetched 4 rows.

```
Output - CadastroBD (run) X
Menu do programa:
-----
1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir todos
0 - Sair do Programa
-----

Escolha uma opção: 5

F - Pessoa Fisica | J - Pessoa Juridica J
Exibindo dados Pessoa Juridica...
-----
ID: 1
Nome: Digital Minda
CNPJ: 11163258000146
Logradouro: Rua Ituverava 21
Cidade: Rio de Janeiro
Estado: RJ
Telefone: 1111234242
Email: digitalmind1@gmail.com
-----
ID: 2
Nome: Britex
CNPJ: 11163258000146
Logradouro: Rua Ituverava 21
Cidade: Rio de Janeiro
Estado: RJ
Telefone: 1111234242
Email: digitalmind1@gmail.com
-----
ID: 3
Nome: Digital Minda
CNPJ: 11163258000146
Logradouro: Rua Ituverava 21
Cidade: Rio de Janeiro
Estado: RJ
Telefone: 1111234242
Email: digitalmind1@gmail.com
-----

SELECT TOP 100 * FROM dbo... X
Max. rows: 100    Fetched Rows: 4

#   idPessoaJuridica  idPessoa  nome          logradouro      cidade          estado          telefone        email           CNPJ
1   1                 1         Digital Minda  Rua Ituverava 21 Rio de Janeiro  RJ             1111234242     digitalmind1@gmail.com 11163258000146
2   2                 2         Britex        Rua Ituverava 21 Rio de Janeiro  RJ             1111234242     digitalmind1@gmail.com 11163258000146
3   3                 3         Digital Minda Rua Ituverava 21 Rio de Janeiro  RJ             1111234242     digitalmind1@gmail.com 11163258000146
4   4                 4         H2A Design    Rua Tirol 2      Rio de Janeiro  RJ             12345678902     h2adesign@gmail.com   18263258000146
```

## Exclusão baseado no ID – caso abaixo foi o ID 10

#	idPessoaFisica	idPessoa	nome	logradouro	cidade	estado	telefone	email	CPF
1	1	1	Herval Rosano Dantas	Rua Ituverava 866	Rio de Janeiro	RJ	21991870000	hervaldantas@gmail.com	0111940066-55
2	2	2	Angela Agostinho da C. Dantas	Rua Araguaia 111	Rio de Janeiro	RJ	11133328909	angeladantas@gmail.com	01220066-20
3	3	3	Antonio Dantas	Rua Tirol 777	Rio de Janeiro	RJ	11991098789	antoniodantas@gmail.com	0444494066-50
4	4	4	Carlos Ronan Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678903	calosronan@gmail.com	0122094005-40
5	5	5	José Ronilson	Rua Cascavel 2	Rio de Janeiro	RJ	12345678904	ronsilsondantas2@gmail.com	022055785-50
6	6	6	Rosângela Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	33448822545	rosangeladantas@gmail.com	0133940066-70
7	7	7	Rossana Daguia Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	4446786666	rossanadaguia@gmail.com	0244005-33
8	8	8	Rosânia Cristina Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678444	rosaniacris@gmail.com	01356209405-69
9	9	9	Francisca Natália Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678111	chica@gmail.com	012334005-11
10	10	10	Ledir Oliveira	Rua do Flamengo, 88	Rio de Janeiro	RJ	2187878548	lediroliv@gmail.com	939365636

```

Menu do programa:
-----
1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir todos
0 - Sair do Programa
-----

Escolha uma opção: 3

F - Pessoa Fisica | J - Pessoa Juridica F
Digite o ID da Pessoa Fisica a ser excluida:
10
Pessoa física excluIda com sucesso.

```

#	idPessoaFisica	idPessoa	nome	logradouro	cidade	estado	telefone	email	CPF
1	1	1	Herval Rosano Dantas	Rua Ituverava 866	Rio de Janeiro	RJ	21991870000	hervaldantas@gmail.com	0111940066-55
2	2	2	Angela Agostinho da C. Dantas	Rua Araguaia 111	Rio de Janeiro	RJ	11133328909	angeladantas@gmail.com	01220066-20
3	3	3	Antonio Dantas	Rua Tirol 777	Rio de Janeiro	RJ	11991098789	antoniodantas@gmail.com	0444494066-50
4	4	4	Carlos Ronan Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678903	calosronan@gmail.com	0122094005-40
5	5	5	José Ronilson	Rua Cascavel 2	Rio de Janeiro	RJ	12345678904	ronsilsondantas2@gmail.com	022055785-50
6	6	6	Rosângela Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	33448822545	rosangeladantas@gmail.com	0133940066-70
7	7	7	Rossana Daguia Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	4446786666	rossanadaguia@gmail.com	0244005-33
8	8	8	Rosânia Cristina Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678444	rosaniacris@gmail.com	01356209405-69
9	9	9	Francisca Natália Dantas	Rua Jose Venâncio 596	Rio de Janeiro	RJ	12345678111	chica@gmail.com	012334005-11

Alteração baseado no ID – caso abaixo foi o ID 9

```
CadastroBD(run) x
```

```
run:
```

```
Menu do programa:
```

```
-----
```

```
1 - Incluir Pessoa
```

```
2 - Alterar Pessoa
```

```
3 - Excluir Pessoa
```

```
4 - Buscar pelo Id
```

```
5 - Exibir todos
```

```
0 - Sair do Programa
```

```
-----
```

```
Escolha uma opção: 2
```

```
F - Pessoa Fisica | J - Pessoa Juridica F
```

```
Digite o ID da pessoa fisica a ser alterada: 9
```

```
Pessoa Fisica localizada:
```

```
ID: 9
```

```
Nome: Francisco Rafael Dantas
```

```
CPF: 012334005-11
```

```
Lugradouro: Rua Cascavel 2
```

```
Cidade: Rio de Janeiro
```

```
Estado: RJ
```

```
Telefone: 12345678904
```

```
Email: ronsilsondantas@gmail.com
```

```
-----
```

```
Digite o nome da Pessoa Fisica: Francisco Rafael Dantas
```

```
Digite o CPF da Pessoa Fisica: 111111111
```

```
Digite o Lugradouro da Pessoa Fisica: Sitio Volta do Rio
```

```
Digite a cidade da Pessoa Fisica: Carmauba dos Dantas
```

```
Digite o estado da Pessoa Fisica: RN
```

```
Digite o telefone da Pessoa Fisica: 333333333
```

```
Digite o email da Pessoa Fisica: francas@gmail.com
```

```
Pessoa Fisica incluida com sucesso!
```

```
Pessoa Fisica alterada com sucesso!
```

SELECT TOP 100 \* FROM dbo... X

Max. rows: 100      Fetched Rows: 9

#	idPessoaFisica	idPessoa	nome	logradouro	cidade	estado	telefone	email	CPF
1	1	1	Herval Rosano Dantas	Rua Ituverava 866	Rio de Janeiro	RJ	21991870000	hervaldantas@gmail.com	0111940066-55
2	2	2	Angela Agostinho da C. Dantas	Rua Araguaia 111	Rio de Janeiro	RJ	11133328909	angeladantas@gmail.com	01222066-20
3	3	3	Antonio Dantas	Rua Tirol 777	Rio de Janeiro	RJ	11991098789	antoniodantas@gmail.com	0444494066-50
4	4	4	Carlos Ronan Dantas	Rua Jose Venancio 596	Rio de Janeiro	RJ	12345678903	calosronan@gmail.com	0122294005-40
5	5	5	Jose Ronilson	Rua Cascavel 2	Rio de Janeiro	RJ	12345678904	ronilsondantas2@gmail.com	022055785-50
6	6	6	Rosângela Dantas	Rua Jose Venancio 596	Rio de Janeiro	RJ	33448822545	rosangeladantas@gmail.com	0133940066-70
7	7	7	Rossana Daguia Dantas	Rua Jose Venancio 596	Rio de Janeiro	RJ	4446786666	rossanadaguia@gmail.com	0244005-33
8	8	8	Rosânia Cristina Dantas	Rua Jose Venancio 596	Rio de Janeiro	RJ	12345678444	rosaniacris@gmail.com	01356209405-69
9	9	9	Francisco Rafael Dantas	Sítio Volta do Rio	Carnaúba dos Dantas	RN	333333333	franca@gmail.com	111111111

Buscar baseado no ID – caso abaixo foi o ID 4

Menu do programa:

- ```
1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir todos
0 - Sair do Programa
```

Escolha uma opção: 4

F - Pessoa Física | J - Pessoa Jurídica **F**  
 Digite o ID da Pessoa Física a ser exibida:

Nome: Carlos Ronan Dantas

Logradouro: Rua Jose Venancio 596

Cidade: Rio de Janeiro

Estado: RJ

Telephone: 12345678903

Email: calosronan@gmail.com

-----

```
SELECT TOP 100 * FROM dbo... X
```

Max. rows: 100 | Fetched Rows: 9

[illegible]

## Incluir uma Pessoa Física

Menu do programa:

```
1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Buscar pelo Id
5 - Exibir todos
0 - Sair do Programa
```

Escolha uma opção: 1

F - Pessoa Física | J - Pessoa Jurídica F

Digite o nome da Pessoa Física: **Andrea Gomes de Souza**

Digite o CPF da Pessoa Física: 8888899999

**Digite o Logradouro da Pessoa Física: Rua Tirol, 222**

Digite a cidade da Pessoa Física: Rio de Janeiro

Digite o estado da Pessoa Física: RJ

Digite o telefone da Pessoa Física: 1

Digite o telefone da Pessoa Física: 2187878987  
 Digite o e-mail da Pessoa Física: andrea\_nunes@mail.com

Pessoa Física incluída com sucesso!

[illegible]



## Análise e Conclusão:

### 1 - Qual a importância dos componentes de middleware, como o JDBC?

O JDBC (Java Database Connectivity), desempenha um importante papel quando se trata de conexões com Banco de Dados, e como se trata de um middleware (Middle = meio) o mesmo consegue fazer esta conexão com diferentes Sistemas de Gerenciamento de Banco de Dados (SGBD) do tipo: MySQL, Oracle, SQL Server, PostgreSQL, etc.

O JDBC fornece uma camada de abstração que permite se desenvolva a programação dos códigos Java independente do SGBD o que o torna bastante portátil. Algumas das principais funcionalidade do JDBC são:

2. Gerenciamento de Conexões principalmente abrindo e fechando as conexões com os BD, garantindo economia e eficácia nos recursos de processamento, e evitando vazamentos nas conexões.
3. Execução de Consultas e Comandos que permite seja enviada consultas SQL e comandos para o banco de dados. Isso é essencial para realizar operações de leitura e gravação.
4. Preparação de Declarações SQL o JDBC suporta a preparação de declarações SQL parametrizadas, evitando a injeção de SQL o que melhora a segurança das consultas.
5. Gerenciamento de Transações dá suporte as transações de forma a permitir que se inicie, comprometa ou até mesmo que se reverta as transações (rollback) caso haja inconsistência dos dados nas operações complexas.

### 2 - Qual a diferença no uso de Statement ou PreparedStatement para a manipulação de dados?

A diferença principal é que PreparedStatement é mais seguro e tem um melhor desempenho especialmente para consultas dinâmicas. Já o Statement pode ser mais simples, mas requer cuidados adicionais para evitar vulnerabilidades de injeção de SQL.

Veja estas comparações:

| Aspecto           | Statement                                                       | PreparedStatement                                                              |
|-------------------|-----------------------------------------------------------------|--------------------------------------------------------------------------------|
| Segurança         | Vulnerável a ataques de injeção de SQL                          | Protege contra injeção de SQL automaticamente                                  |
| Desempenho        | Menos eficiente para consultas repetidas                        | Mais eficiente para consultas repetidas devido à pré-compilação                |
| Usabilidade       | Simples de usar para consultas estáticas                        | Mais flexível para consultas dinâmicas                                         |
| Leitura de Código | Pode tornar o código menos legível devido à concatenação de SQL | Tende a tornar o código mais legível, pois consulta e parâmetros são separados |

### **3 - Como o padrão DAO melhora a manutenibilidade do software?**

O padrão DAO ajuda a melhorar a manutenibilidade do software pois o mesmo oferece uma arquitetura bem organizada, reutilização de código e a abstração do banco de dados além de tornar o código mais limpo e seguro. Este padrão promove a separação clara das responsabilidades pois isola a lógica de acesso ao banco de dados em uma camada, facilitando a compreensão e a manutenção do código, uma vez que cada componente tem um propósito bem definido.

Ao encapsularmos a lógica de acesso ao banco de dados em objetos DAO, podemos reutilizar estes códigos em várias partes do aplicativo. Isso evita a duplicação e facilita a manutenção, uma vez que qualquer mudança nas operações de banco de dados poderá ser feita em apenas um lugar. Existem muitas outras melhorias que o padrão DAO proporciona.

### **4 - Como a herança é refletida no banco de dados, quando lidamos com um modelo estritamente relacional?**

Existem duas abordagens principais para implementar herança em um modelo de banco de dados relacional: herança de tabela única (single-table inheritance) e herança de tabela múltipla (multiple-table inheritance). A herança de tabela única usa uma única tabela com uma coluna adicional para indicar o tipo de entidade. É um esquema simples, principalmente quando se refere a facilidade de consulta. Todavia, pode levar a uma tabela muito ampla e de esquema denormalizado. Ou seja: tudo numa tabela.

Já a herança de tabela múltipla usa uma tabela separada para cada classe, mantendo a normalização, e usando-se as chaves estrangeiras se faz os relacionamentos, mas também pode complicar as consultas na hora de fazer as junções. Mas com certeza é mais fácil de adicionar novas entidades e de fazer manutenção.

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

A persistência em arquivo é mais simples e adequada para casos de uso simples ou armazenamento de dados temporários, enquanto a persistência em banco de dados oferece recursos mais avançados, segurança, escalabilidade e eficiência para aplicativos que requerem gerenciamento e consulta de dados de maneira mais sofisticada e robusta.

Só posso acreditar que a persistência de dados em arquivo foi útil no processo inicial da computação, pois não tem como se comparar com a robustez da persistência em banco de dados que tem um sistema de gerenciamento muito versátil, o famoso SGBD. Além de ter também uma poderosa linguagem de consulta SQL que permite consultas poderosas e flexíveis usando uma linguagem de alto nível (mais humana). Tudo bem que tá em inglês, mas por exemplo: selecione todas colunas da tabela tal, cujo campo valor unitário seja igual a 100 ==> `SELECT * FROM dbo.carros WHERE valueUnid = 100;` - pra mim tá mais em linguagem humana do que em linguagem de máquina.

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

O uso de operadores lambda no Java simplificou significativamente. Antes da introdução de lambdas, era comum usar expressões anônimas e iterar manualmente sobre coleções para realizar operações como a impressão de valores. Agora com o operador lambda reduziu-se muito a quantidade de código, a sintaxe é mais concisa além de tornar o código mais legível e de fácil compreensão. Exemplo bem simples:

### Sem Lambda:

```
// Imprimir os nomes ordenados
```

```
for (String nome : nomes) {  
    System.out.println(nome);  
}
```

### Com Lambda:

```
nomes.forEach(nome -> System.out.println(nome));
```

Veja como a iteração ficou muito mais simples.

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

Métodos acionados diretamente pelo método main em Java precisam ser marcados como static porque o main é o ponto de entrada do programa, e o modificador static permite que os métodos sejam chamados diretamente pelas suas próprias classes sem que haja a necessidade de instanciá-lo, no caso a classe main.