



Campus: 202301037751 - POLO CENTRO - PORTO REAL - RJ

Curso: Desenvolvimento Full Stack

Disciplina: Nível 3: BackEnd sem banco não tem

Turma: 9001

Semestre letivo: 3º Semestre

Nome: Leonardo Naves de Lima Araujo

Missão Prática | Nível 3 | Mundo 3

Github do projeto:

<https://github.com/Navesz/Missao-pratica-2-mundo-3-BancoDeDados>

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 Server na persistência de dados.

1º Procedimento | Mapeamento Objeto-Relacional e DAO

Resultado

Cadastro de pessoa física:

```
Relacao de pessoas fisicas cadastradas:  
ID: 2  
Nome: Leonardo  
Endereco: Rua das Andurinhas  
Cidade: Volta Redonda  
Estado: RS  
Telefone: 21998340238  
E-mail: leonardo@hotmail.com  
CPF: 28194533494
```

Cadastro de pessoa jurídica:

Relacao de pessoas juridicas cadastradas:
ID: 3
Nome: Estacio
Endereco: Rua 12, Centro
Cidade: Barra Mansa
Estado: RJ
Telefone: 24 99380238
E-mail: estacio@gmail.com
CNPJ: 28390444839

Pessoa Editada:

Relacao de pessoas fisicas cadastradas:
ID: 2
Nome: Leonardo Marques
Endereco: Rua dos Sonhos, 22
Cidade: Quatiz
Estado: RJ
Telefone: 2399843839
E-mail: leonardomarques@gmail.com
CPF: 1928383774944

Relacao de pessoas juridicas cadastradas:
ID: 3
Nome: Google
Endereco: Rua das Andurinhas, 84
Cidade: Itatiaia
Estado: MG
Telefone: 43998320038
E-mail: google@gmail.com
CNPJ: 1928474394848

Análise e Conclusão

Qual a importância dos componentes de middleware, como o JDBC? O JDBC é uma ferramenta indispensável no universo Java, atuando como elo entre aplicações e bancos de dados. Ele simplifica a vida dos desenvolvedores, que podem usar as mesmas interfaces para acessar diferentes bancos de dados, garantindo que a aplicação funcione de maneira eficiente em diferentes ambientes sem precisar se adaptar a cada novo banco de dados.

Qual a diferença no uso de Statement ou PreparedStatement para a manipulação de dados? Utilizar Statement ou PreparedStatement depende do que você precisa. Se for executar uma consulta simples várias vezes, Statement serve. Mas se suas consultas mudam um pouco ou precisam de mais segurança, o PreparedStatement é a melhor escolha. Ele não só melhora a performance por pré-compilar o SQL, mas também protege contra ataques de injeção SQL, cuidando de maneira segura dos dados inseridos.

Como o padrão DAO melhora a manutenibilidade do software? Implementar o padrão DAO é como colocar um intermediário entre sua aplicação e a fonte de dados. Isso limpa e organiza o código, tornando mais fácil fazer mudanças no banco de dados sem mexer no restante da aplicação. Essa organização traz clareza ao código e facilita a manutenção e os testes.

Como a herança é refletida no banco de dados, quando lidamos com um modelo estritamente relacional? Herança em bancos de dados relacionais: Tratar de herança em bancos de dados que seguem o modelo relacional pode ser um quebra-cabeça. Como esse modelo não foi feito para entender a herança naturalmente, os desenvolvedores precisam escolher entre criar uma tabela para cada tipo de classe ou uma tabela que serve para todas, cada uma com suas vantagens e desvantagens em termos de organização de dados e desempenho das consultas.

2º Procedimento | Alimentando a Base

Resultado

```
#####
1 - Incluir pessoa
2 - Alterar pessoa
3 - Excluir pessoa
4 - Buscar pelo Id
5 - Listar todos
6 - Lista de pessoas fisicas
7 - Lista de pessoas juridicas
0 - Sair
#####
Escolha uma opcao:
```

Incluir pessoa:

```
(F) - Pessoa Fisica | (J) - Pessoa juridica
Escolha uma opcao:
f
Digite o nome para a pessoa fisica:
Leonardo Silva
Digite o endereco:
Rua das Arvores, 44
Digite a cidade:
Rezende
Digite o estado:
RJ
Digite o telefone:
998370429
Digite o e-mail:
leonardosilva@gmail.com
Digite o CPF:
18309477840
## > Pessoa Fisica cadastrada com sucesso!
```

Alterar pessoa:

```
F - Alterar pessoa fisica | J - Alterar pessoa juridica
Escolha uma opcao:
f
Digite o ID da pessoa fisica que deseja alterar:
7
Digite o novo nome da pessoa fisica:
João Mirante
Digite o novo endereco:
Rua dos Diamantes, 55
Digite a nova cidade:
Rezende
Digite o novo estado:
RS
Digite o novo telefone:
99380433
Digite o novo e-mail:
joaomirante@gmail.com
Digite o novo CPF:
38394048847
## > Pessoa fisica atualizada com sucesso.
```

Excluir pessoa:

```
(F) - Excluir pessoa fisica | (J) - Excluir pessoa juridica
Escolha uma opcao:
f
Digite o ID da pessoa fisica para ser removida:
7
## > Pessoa fisica removida com sucesso.
```

Listar todos:

```
Relacao de pessoas fisicas cadastradas:
ID: 2
Nome: Leonardo Marques
Endereco: Rua dos Sonhos, 22
Cidade: Quatiz
Estado: RJ
Telefone: 2399843839
E-mail: leonardomarques@gmail.com
CPF: 1928383774944

Relacao de pessoas juridicas cadastradas:
ID: 3
Nome: Google
Endereco: Rua das Andurinhas, 84
Cidade: Itatiaia
Estado: MG
Telefone: 43998320038
E-mail: google@gmail.com
CNPJ: 1928474394848
```

Análise e Conclusão:

Quais as diferenças entre a persistência em arquivo e a persistência em banco de dados? A escolha entre guardar dados em um arquivo ou em um banco de dados depende do que você precisa fazer com esses dados. Se você só precisa de algo simples, como armazenar configurações ou dados que não mudam muito, um arquivo pode ser suficiente. Por outro lado, bancos de dados são melhores para gerenciar grandes volumes de dados, permitindo consultas complexas, melhor segurança e integridade dos dados. Além disso, os bancos de dados oferecem recursos como backup e recuperação mais eficientes que os sistemas baseados em arquivos.

Como o uso de operador lambda simplificou a impressão dos valores contidos nas **entidades, nas versões mais recentes do Java?** Nas versões mais recentes do Java, o operador lambda realmente mudou o jogo ao simplificar a maneira como manipulamos coleções de dados, incluindo a impressão de valores de entidades. Por exemplo, você pode imprimir todos os elementos de uma lista com uma simples linha de código usando lambda, substituindo os loops tradicionais. Isso torna o código não apenas mais limpo e mais direto, mas também mais fácil de ler e manter.

Por que métodos acionados diretamente pelo método main, sem o uso de um objeto, precisam ser marcados como static? Quando você vê um método que é chamado diretamente do método main sem criar um objeto, esse método precisa ser marcado como static. Isso acontece porque o método main é estático e, em Java, métodos estáticos não podem chamar métodos não-estáticos diretamente. Eles pertencem à classe, não a uma instância da classe. Isso significa que métodos estáticos podem ser usados sem criar um objeto da classe, o que é útil para operações gerais que não dependem do estado atual dos objetos.