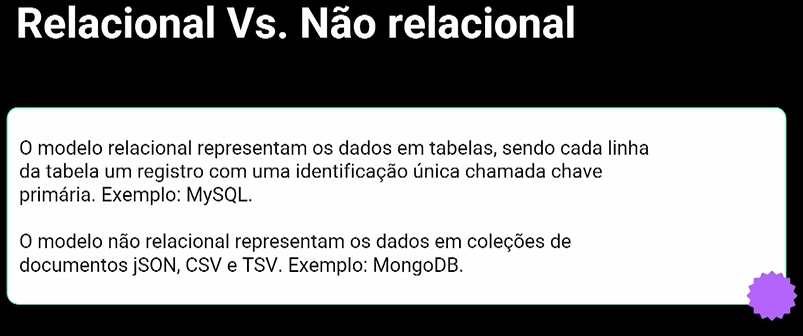
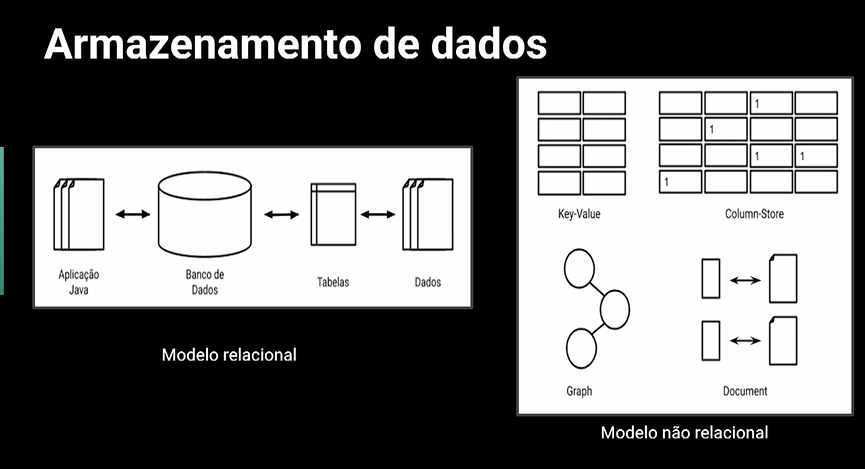
**Banco de Dados:**

Existem N fornecedores de bancos de dados. Ex: Oracle, mas existem vários outros.

Trabalham bom banco de dados relacionais: relaciona dados.

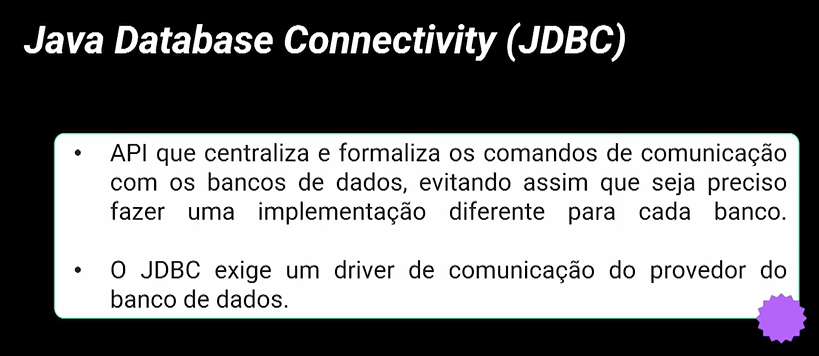


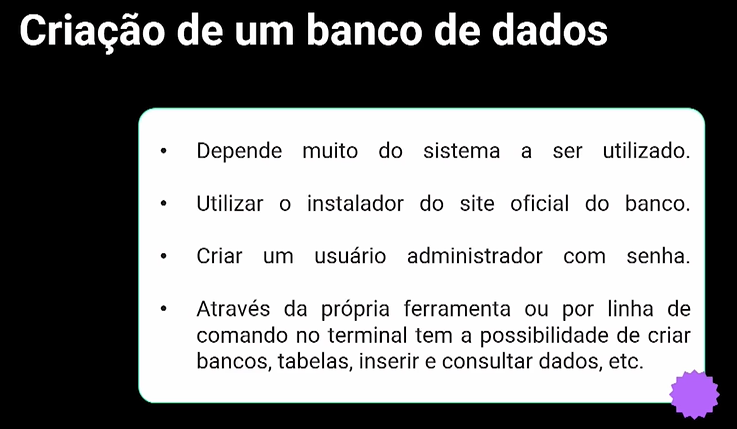


Nas aulas iremos falar sobre os modelos relacionais.

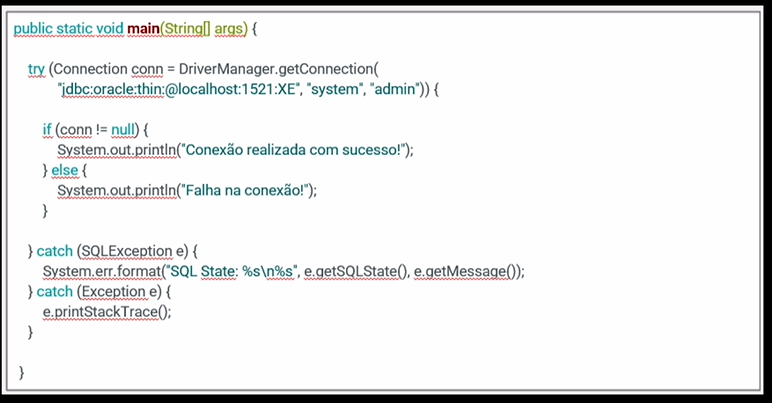
**JDBC: JAVA DATA BASE CONECTED.**

Padrão (ponte) que permite por meio das minhas aplicações javas conectar, manipular de forma bem tranquila que estão armazenado em um SGBD-Data Base Management System ou Sistema de Gerenciamento de Banco de Dados.





Ex:

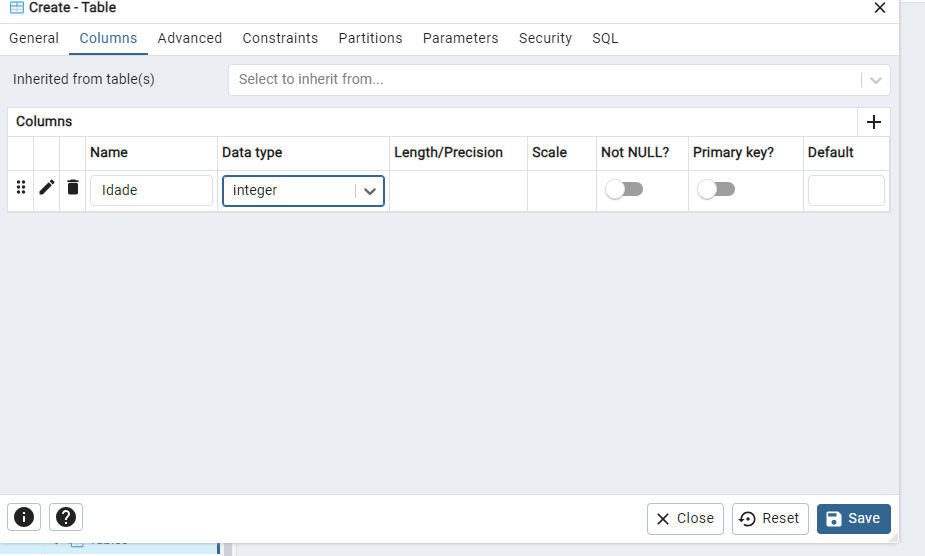


Podemos usar vários bancos de dados, na aula vamos usar o PostgreSQL. (instala como pg admin).

Para começamos a trabalhar precisamos criar tabelas! Mas como faço?

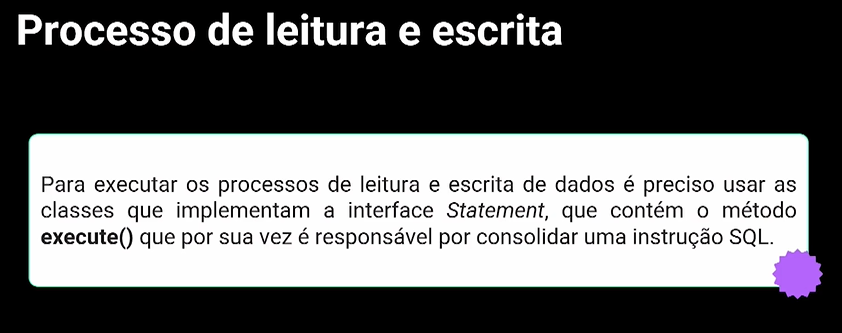
Vou em Servers> Postgre> databases> postgress> schemas> public> tables (e crio).

Ai salvo o nome e vou em coluna:

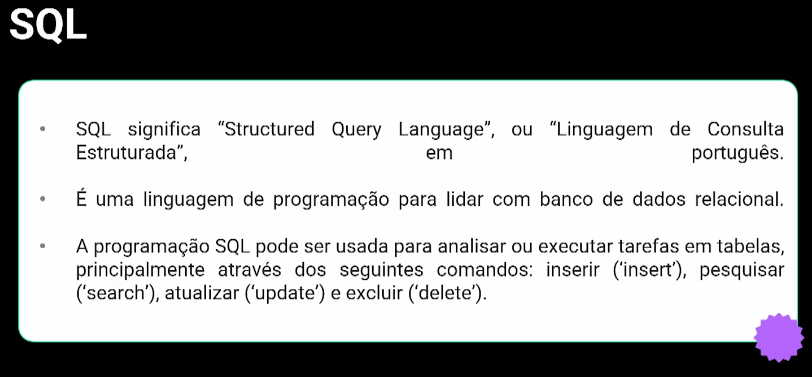


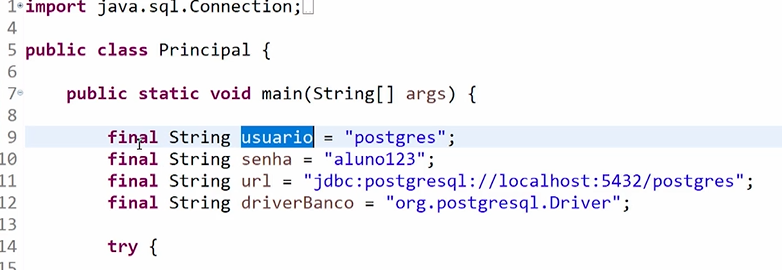
**Declarações:**

Quando vamos manipular o banco de dados fazemos processo de leitura e escrita no banco de dados. Escrever algo diferente. (input e output). *Chamado de persistência de dados.*

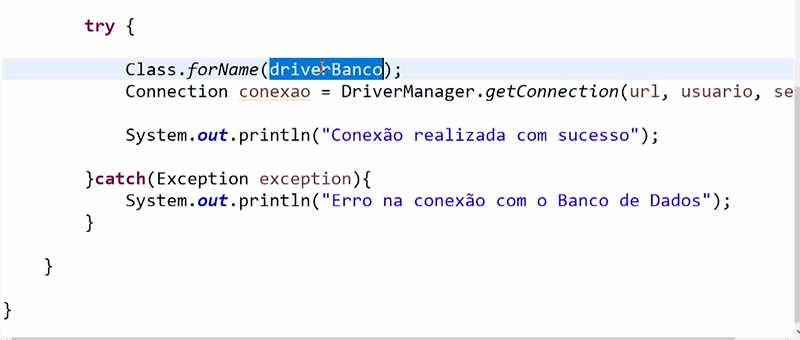


**SQL**: Linguagem padrão de bancos de dados;



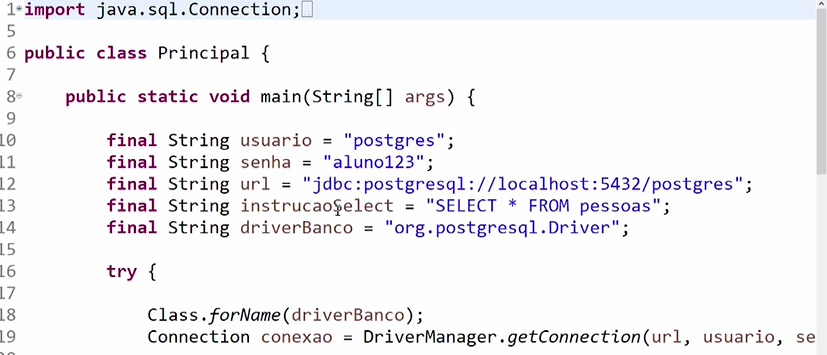


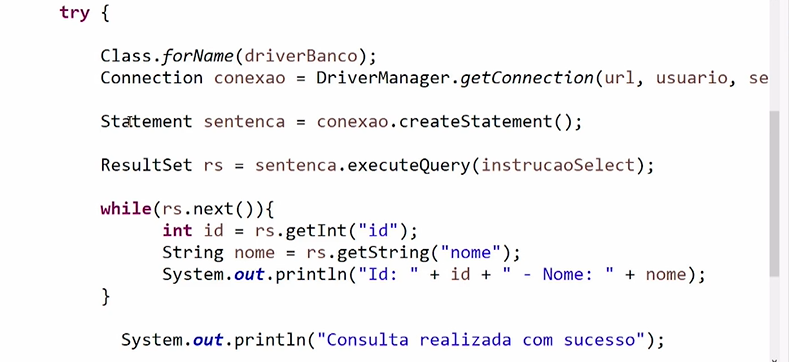
Final: faz com que o atributo seja algo constante.



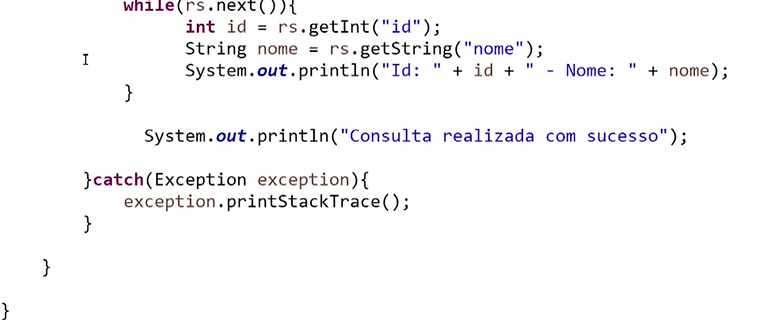
Class.forname (drive que esta escrito ali).

Outro exemplo:

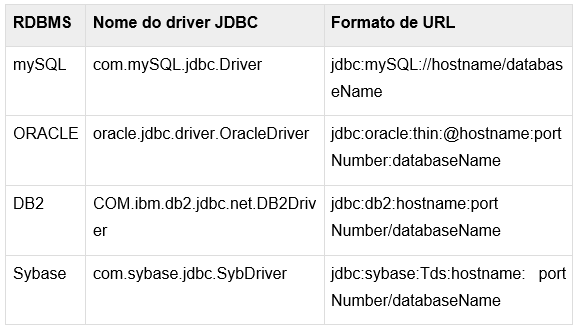




Statement: permite acessar os dados que estão no dados.



Geralmente para url tem:

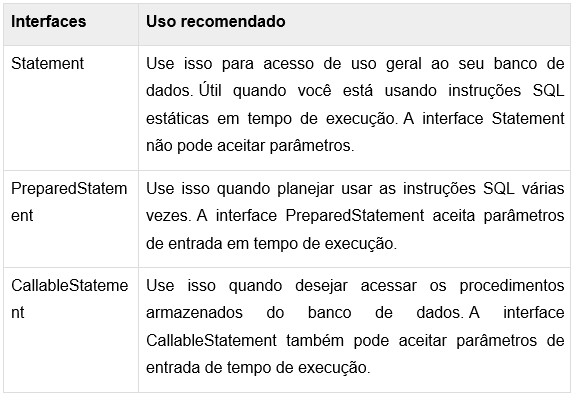


getConnection (URL da string)

getConnection (URL da string, propriedades Prop)

getConnection (URL da string, usuario da String, senha da String)

Outras inf:



**Escrita de dados no banco:**

**Escrita/Inclusão:**

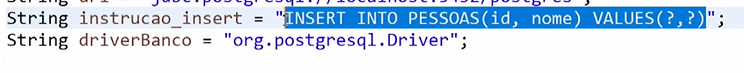
No código a seguir, um comando de insert é feito na tabela alunos do banco descomplica\_db.

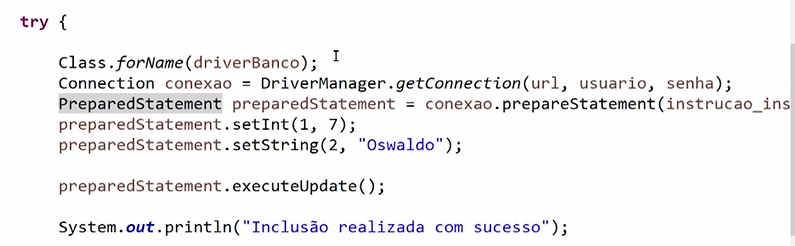
EX:

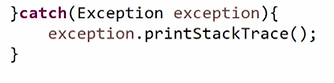


Pratica:



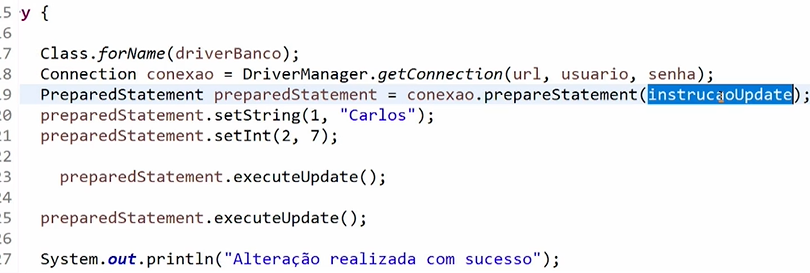
**





Mas se eu tenho uma pessoa no meu banco e quero atualizar:





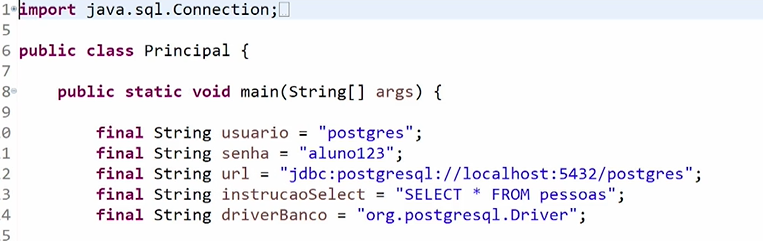
**Leitura de dados:**

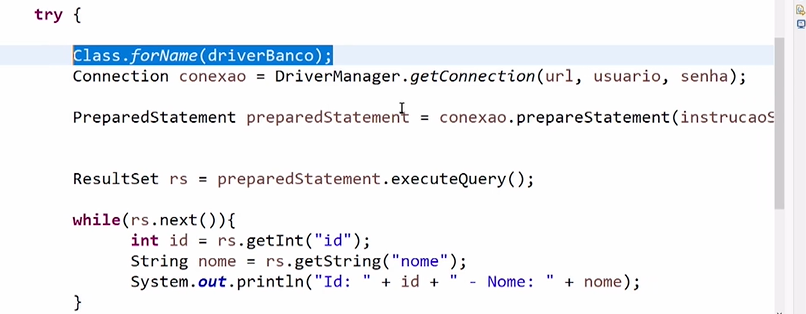
Um comando de leitura é muito semelhante, mas é necessário o uso da interface ResultSet para iterar os possíveis resultados da consulta.

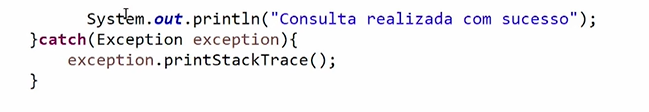
EX:



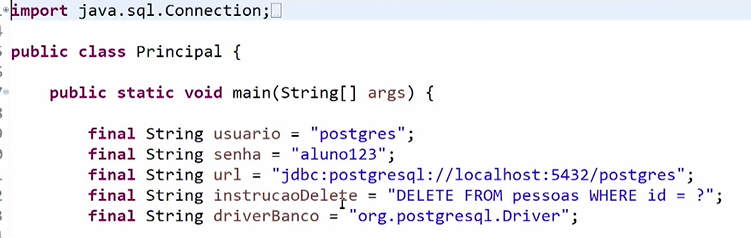
Pratica:

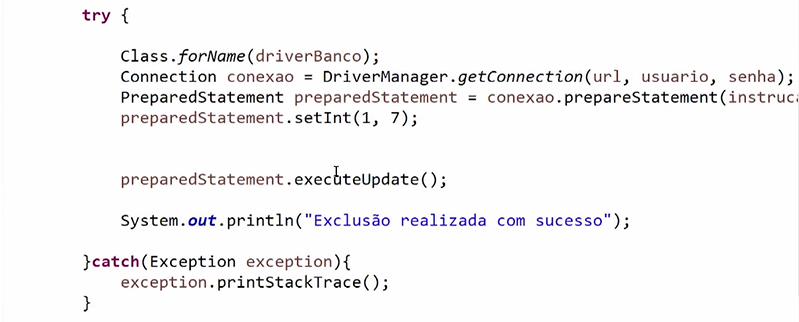






**Exclusão:**





**Resumo:**

**DriverManager**: Esta classe gerencia uma lista de drivers de banco de dados. Corresponde às solicitações de conexão do aplicativo java com o driver de banco de dados apropriado usando o subprotocolo de comunicação. O primeiro driver que reconhece um determinado subprotocolo no JDBC será usado para estabelecer uma conexão com o banco de dados.

**Driver:** Essa interface lida com as comunicações com o servidor de banco de dados. Você interage diretamente com os objetos Driver muito raramente. Em vez disso, você usa objetos DriverManager, que gerencia objetos desse tipo. Ele também abstrai os detalhes associados ao trabalho com objetos Driver.

**Connection:** Essa interface com todos os métodos para entrar em contato com um banco de dados. O objeto de conexão representa o contexto de comunicação, ou seja, toda a comunicação com o banco de dados é apenas através do objeto de conexão.

**Statement:** Você usa objetos criados nessa interface para enviar as instruções SQL ao banco de dados. Algumas interfaces derivadas aceitam parâmetros além de executar procedimentos armazenados.

**ResultSet**: esses objetos mantêm os dados recuperados de um banco de dados após a execução de uma consulta SQL usando objetos Statement. Ele atua como um iterador para permitir que você mova seus dados.

**SQLException:** esta classe lida com quaisquer erros que ocorram em um aplicativo de banco de dados.