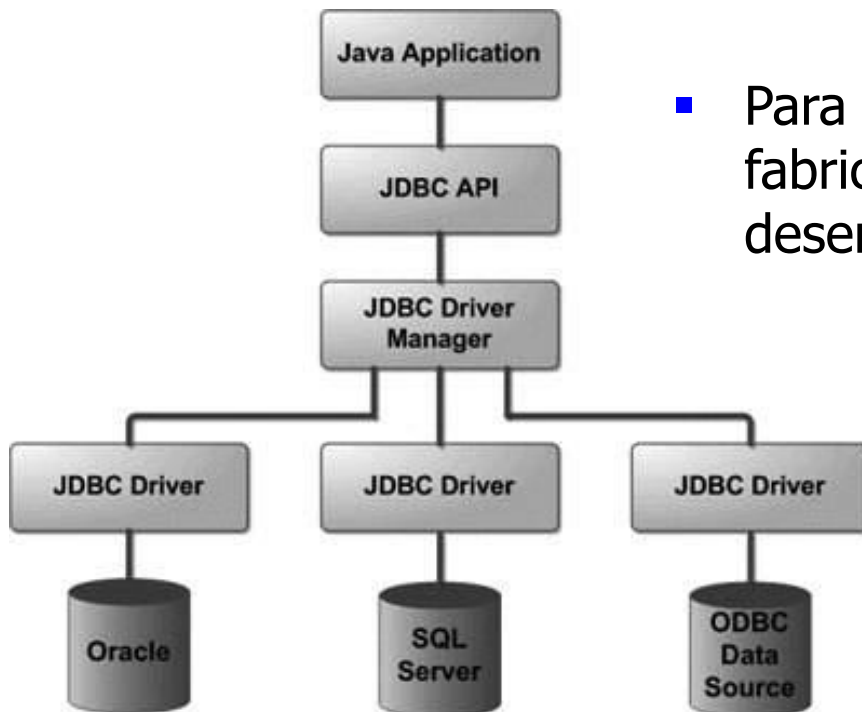


Introdução

- O JDBC (Java Data Base Connectivity) é uma especificação que prove acessibilidade de aplicações Java com bancos de dados.
- Trata-se de um padrão de acesso a dados a bancos de dados.



- Para seguir este padrão, os diferentes fabricantes distribuem drivers JDBC aos desenvolvedores Java.

Exemplos de drivers JDBC

- Microsoft SQL Server JDBC

<https://docs.microsoft.com/pt-br/sql/connect/jdbc/download-microsoft-jdbc-driver-for-sql-server?view=sql-server-ver15>

- MySQL Connector / J

<https://dev.mysql.com/downloads/connector/j/5.1.html>

- Oracle

<https://www.oracle.com/database/technologies/appdev/jdbc-downloads.html>

- PostgreSQL

<https://jdbc.postgresql.org/download.html>

Pacote java.sql

- Contém classes e interfaces utilizadas no acesso e manipulação de dados via JDBC.

Connection: Realiza a conexão ao banco de dados.

Statement: Executa instruções SQL simples e pré-definidas.

PreparedStatement: Executa instruções SQL parametrizadas.

CallableStatement: Executa stored procedures e functions.

ResultSet: Permite manipular consultas.

PreparedStatement

- Ao criar um PreparedStatement pode-se especificar a declaração SQL contendo parâmetros (?) que serão substituídos por valores no momento da execução.
- Cada interrogação contida na declaração SQL representa um parâmetro.
- Antes da execução, todos os parâmetros devem ter sido assinalados.

Consultas com ResultSet

- ResultSet representa um cursor proveniente da base de dados.
- Representa uma “tabela” em memória que possui um ponteiro apontando para uma de suas linhas (registro atual).

cidade X			
	codigo	nome	estado
	3520509	Indaiatuba	SP
	3520608	Indiana	SP
	3520707	Indiaporã	SP
	3520806	Inúbia Paulista	SP
	3520905	Ipaussu	SP
	3521002	Iperó	SP
	3521101	Ipeúna	SP
	3521150	Ipiguá	SP
	3521200	Iporanga	SP
	3521309	Ipuã	SP
	3521408	Iracemápolis	SP

Transações

- Conjunto de operações realizadas no banco de dados, agrupadas de forma única, que podem ser confirmado (commit) ou cancelado em caso de falha (rollback).
- O objetivo é garantir a integridade dos dados contidos no banco.
- Métodos da interface `java.sql.Connection`:
 - `setAutoCommit(boolean)`
 - `commit()`
 - `rollback()`