

Programação Orientada a Objetos

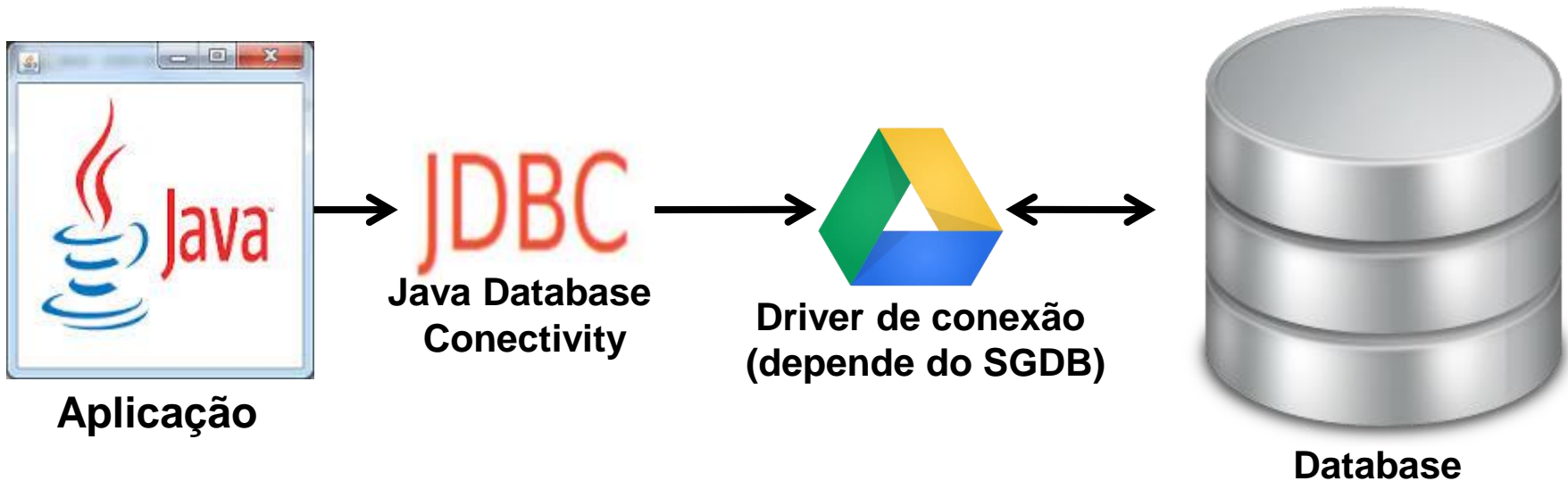
Semana 11 Banco de Dados

Reflexão

“O principal objetivo da educação é criar pessoas capazes de fazer coisas novas e não simplesmente repetir o que outras gerações fizeram.”

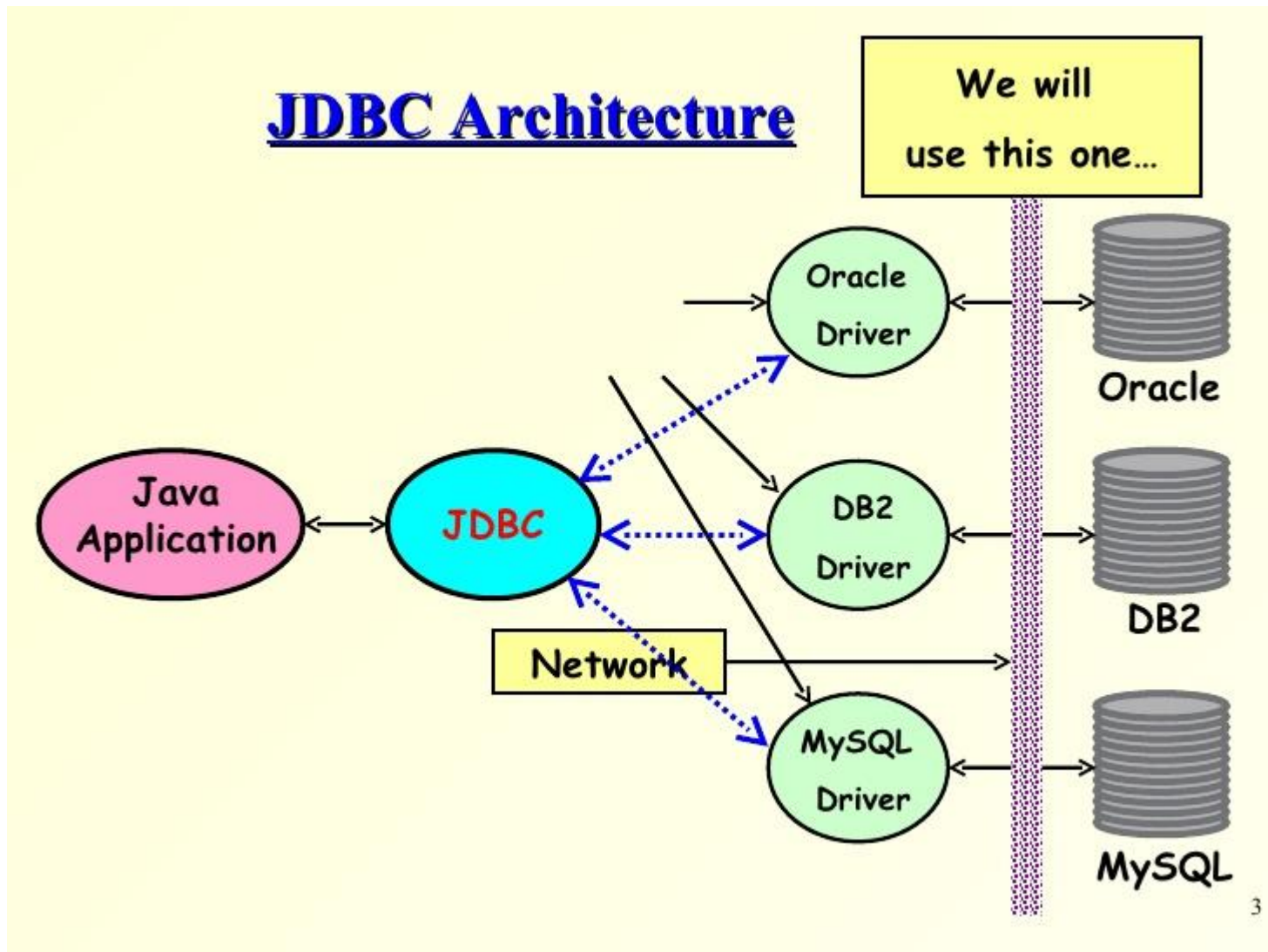
Jean Piaget

Uso do Banco de Dados em Java



- ✓ A aplicação chama a biblioteca JDBC
- ✓ A biblioteca carrega o driver que “entende” o SGDB
- ✓ Depois disso, a aplicação pode se conectar e enviar requisições ao SGDB

Uso do Banco de Dados em Java



Conexão

Para abrir uma conexão com um banco de dados, é necessário utilizar um driver.

A classe DriverManager é responsável por se comunicar com o driver.

É necessário importar uma interface de conexão:

```
import java.sql.Connection;
```

Statement

Permite definir um objeto para executar instruções SQL.

Métodos principais:

1. `executeQuery()` → para realizar uma pesquisa no banco de dados
2. `executeUpdate()` → para realizar uma atualização no banco de dados.

É necessário importar a interface `Statement`:

```
import java.sql.Statement;
```

ResultSet

Permite definir um objeto para receber o resultado de uma consulta em SQL.

Métodos principais:

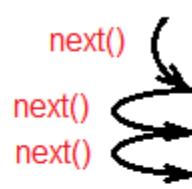
1. `next()` → move para o próximo registro
2. `getX()` → retorna dados do banco.

É necessário importar a interface `ResultSet`:

```
import java.sql.ResultSet;
```

ResultSet

```
Select Emp_Id, Emp_No, Emp_Name from Employee
```



	1	2	3
	EMP_ID	EMP_NO	EMP_NAME
▶	1	7839	E7839 ... KING ...
	2	7566	E7566 ... JONES ...
	3	7902	E7902 ... FORD ...
	4	7369	E7369 ... SMITH ...
	5	7698	E7698 ... BLAKE ...
	6	7499	E7499 ... ALLEN ...
	7	7521	E7521 ... WARD ...
	8	7654	E7654 ... MARTIN ...
	9	7782	E7782 ... CLARK ...
	10	7788	E7788 ... SCOTT ...
	11	7844	E7844 ... TURNER ...
	12	7876	E7876 ... ADAMS ...
	13	7900	E7900 ... ADAMS ...
	14	7934	E7934 ... MILLER ...

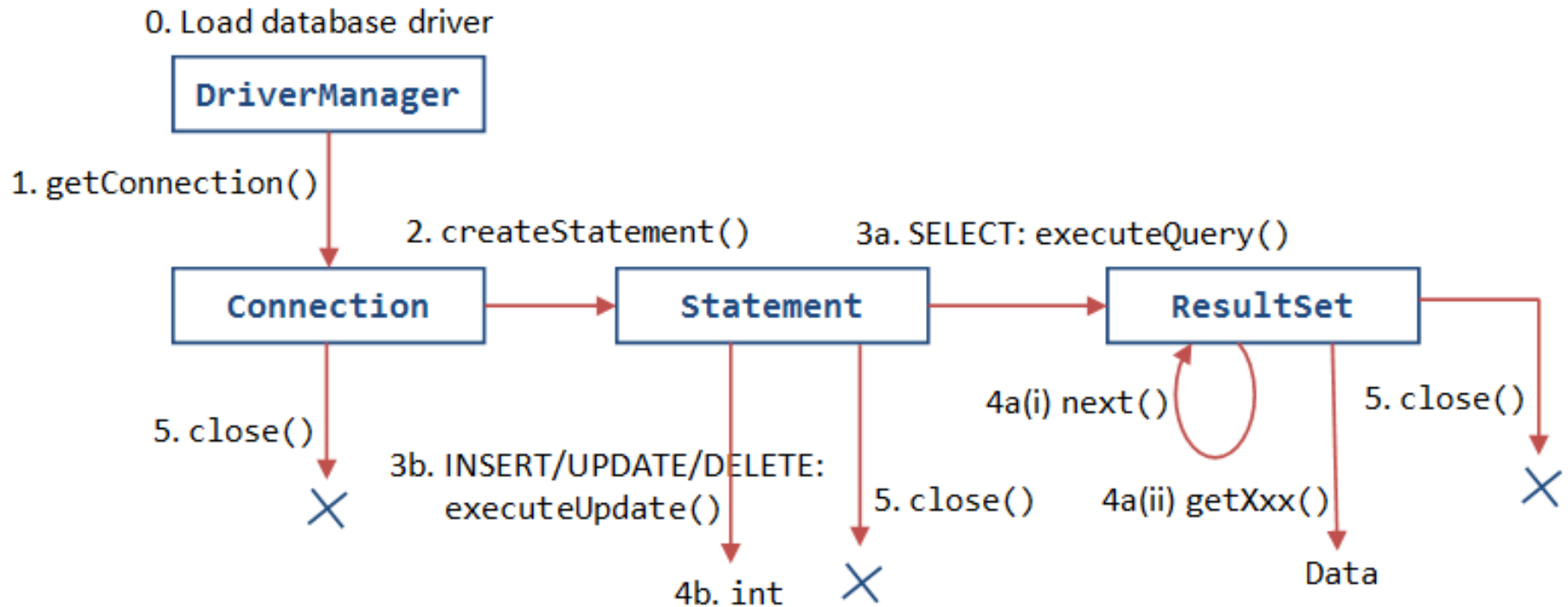
```
Connection connection = ....;
```

```
Statement statement =  
    connection.createStatement();
```

```
ResultSet rs = statement.executeQuery(sql);
```

```
while (rs.next()) {  
    int empId = rs.getInt(1);  
    String empNo = rs.getString(2);  
    String empName = rs.getString("Emp_Name");  
}
```


O ciclo do JDBC



Exercícios

1. Crie um banco de dados e adicione uma tabela chamada clientes com a seguinte estrutura:

id	int
nome	varchar(50)
idade	int
sexo	char(1)

Por meio da linguagem SQL, insira alguns registros na tabela.

Faça uma aplicação em Java que liste em tela todos os registros da tabela clientes.

Exercícios

2. Crie no mesmo banco de dados do exercício anterior uma tabela chamada produtos com a seguinte estrutura:

id	int
produto	varchar(50)
quantidade	int
valor	float

Por meio da linguagem SQL, insira alguns registros na tabela. A seguir, faça uma aplicação em Java que liste um registro de cada vez, cada vez que uma tecla for pressionada.

3. Faça uma aplicação que liste os dados do produto (do exercício 2) a partir da entrada do seu id.
4. Faça uma aplicação para somar o valor de todos os produtos do exercício 2.

Exercícios

5. Ainda com base na tabela de produtos criada no exercício 2, crie uma aplicação que solicite um limite de preço e apresente apenas os dados dos produtos que estão dentro do intervalo solicitado.
6. Crie uma aplicação que implemente uma interface gráfica para o cadastro de produtos, solicite os dados do produto e insira no banco de dados.
7. Crie uma aplicação que permita atualizar o valor de um determinado produto, por meio do seu id.
8. Crie uma aplicação que permita excluir um produto a partir de seu Id.
9. Crie um arquivo texto (no bloco de notas) contendo dados de diversos produtos de acordo com a tabela do exercício 2. A seguir, elabore uma aplicação que leia os dados do arquivo texto e os insira na tabela.
10. Faça uma aplicação que, a partir da tabela de produtos do exercício 2, gere um arquivo texto contendo todos os produtos da tabela.