

# Trabalho 10 Banco de dados 2

Matheus Costa Monteiro - 12111BSI821

Carlos Augusto Dantas Marquez - 11811BSI228

## Conectando sem as classes do tutorial JDBC

### 1-Bibliotecas

```
1 import java.util.Properties; //Objeto genérico que armazena propriedades com usuário e senha
2 import java.sql.DriverManager; //Objeto que criará a conexão do sistema de banco de dados
3 import java.sql.Connection; //Objeto que armazenará o objeto de conexão ao banco de dados
4 import java.sql.Statement; //Objeto para disparar um comando para o SGBD
5 import java.sql.ResultSet; //Objeto que armazenará as tuplas resultantes de um comando SQL
6 import java.sql.SQLException; //Objeto para capturar eventos de erro no acesso ao banco de dados
7
```

### 2-Uma classe e quatro métodos para a execução do tutorial

```
8 public class StandAloneJDBCCode {
9     public static Connection getConnection(){
10         //code
11     }
12 }
13     public static void myquery(Connection con) throws SQLException {
14         //code
15     }
16     public static void closeConnection(Connection con) {
17         //code
18     }
19     public static void main(String[] args) {
20         //code
21     }
22 }
23
24
```

### 3-Armazenar a conexão do banco de dados

```
8 public class StandAloneJDBCCode {
9     public static Connection getConnection(){
10         //code
11         Connection con = null;
12         String currentUrlString = null;
13         Properties connectionProps = new Properties();
14         connectionProps.put("user", "postgres");
15         connectionProps.put("password", "1234");
16         currentUrlString = "jdbc:postgresql://localhost:5432/IB2";
17         //atenção para os símbolos de barra "/" na linha acima, não confunda com a letra l
18         try {
19             con = DriverManager.getConnection(currentUrlString, connectionProps);
20         } catch (SQLException e) {
21             e.printStackTrace();
22         }
23         return con;
24     }
25 }
26 }
```

#### 4-Uma consulta no banco de dados

```
27 public static void myquery(Connection con) throws SQLException {
28     //code here
29     Statement stat = null;
30     String query =
31         "select cliente.nome_cliente, sum(deposito.saldo_deposito) as soma_deposito, sum(emprestimo.valor_emprestimo) as soma_emprestimo from cliente inner join deposito on cliente.nome_cliente =
32         deposito.nome_cliente left join emprestimo on cliente.nome_cliente = emprestimo.nome_cliente where deposito.nome_cliente is not null and emprestimo.nome_cliente is not null group by cliente.nome_cliente
33         order by cliente.nome_cliente";
34     try {
35         stat = con.createStatement();
36         ResultSet rs = stat.executeQuery(query);
37         while (rs.next()) {
38             String coffeeName = rs.getString(1);
39             int sumd = rs.getInt(2);
40             int sume = rs.getInt(3);
41             System.out.println("Nome cliente: " + coffeeName + "Soma Depositos: " + sumd + "Soma Empréstimos: " + sume);
42         }
43     } catch (SQLException e) { e.printStackTrace(); }
44     finally { if (stat != null) { stat.close(); } }
45 }
```

5) Finalizar a execução do banco, sendo que o mesmo pode abrir várias sessões (foi necessário adicionar o parâmetro “Connection con”)

```
47 public static void closeConnection(Connection con) {
48     //code
49
50     try {
51         if (con != null) {
52             con.close(); con = null;
53         }
54         System.out.println("Released all database resources.");
55     } catch (SQLException e) { e.printStackTrace(); }
56 }
```

6)A “main” para fazer as chamadas e executar o nosso programa (foi necessário adicionar o parâmetro “MyConnection” para o “closeConnection”)

```
57 public static void main(String[] args) {
58     //code
59     if (args.length == 0) {
60         System.err.println("No arguments.");
61     }
62     Connection myConnection = null;
63     try {
64         myConnection = StandAloneJDBCCode.getConnection();
65         StandAloneJDBCCode.myquery(myConnection);
66     } catch (SQLException e) {
67         e.printStackTrace();
68     } finally {
69         StandAloneJDBCCode.closeConnection(myConnection);
70     }
71 }
72 }
73 }
```

java -cp "/home/carlos/Downloads/JDBCTutorial/BD2-14-postgresql-42.2.4.jar" StandAloneJDBCCode.java

```
carlos@carlos-VirtualBox:~/Downloads/JDBCTutorial$ java -cp "/home/carlos/Downloads/JDBCTutorial/BD
2-14-postgresql-42.2.4.jar" StandAloneJDBCCode.java
No arguments.
Nome cliente: Alexandre Márcio de SouzaSoma Depositos: 4227Soma Empréstimos: 1030
Nome cliente: Andrade de FreitasSoma Depositos: 20436Soma Empréstimos: 9442
Nome cliente: André Cabral da SilvaSoma Depositos: 22305Soma Empréstimos: 9579
Nome cliente: Bruno Miranda Pacheco de CastroSoma Depositos: 49523Soma Empréstimos: 21987
Nome cliente: Bruno Tadeu PitaSoma Depositos: 23565Soma Empréstimos: 22237
Nome cliente: Everardo Monfort LeitãoSoma Depositos: 18574Soma Empréstimos: 8741
Nome cliente: Felipe Peres Bichara JúniorSoma Depositos: 2970Soma Empréstimos: 1675
Nome cliente: Frederico de Oliveira GuimarãesSoma Depositos: 210Soma Empréstimos: 1992
Nome cliente: Geraldo OliveiraSoma Depositos: 3897Soma Empréstimos: 2109
Nome cliente: Germano Luiz de PaulaSoma Depositos: 2598Soma Empréstimos: 1415
Nome cliente: Gilmar Negreiros CarvalhoSoma Depositos: 2509Soma Empréstimos: 1640
Nome cliente: João BoiadeiroSoma Depositos: 13638Soma Empréstimos: 14632
Nome cliente: Joaquim Carlos ReisSoma Depositos: 17120Soma Empréstimos: 13092
Nome cliente: Josealdo Teixeira SoaresSoma Depositos: 3105Soma Empréstimos: 1380
Nome cliente: Manuel OliveiraSoma Depositos: 2207Soma Empréstimos: 1396
Nome cliente: Marcos AndradeSoma Depositos: 243082Soma Empréstimos: 94620
Nome cliente: Marcos CláudioSoma Depositos: 7400Soma Empréstimos: 2897
Nome cliente: Maria Raquel Teixeira de CastroSoma Depositos: 2968Soma Empréstimos: 582
Nome cliente: Marta Avelar SantosSoma Depositos: 145425Soma Empréstimos: 49756
Nome cliente: Rafael Mesquita GondimSoma Depositos: 27364Soma Empréstimos: 10429
Nome cliente: Thiago Andrade FigueiraSoma Depositos: 6012Soma Empréstimos: 672
```

```
java -cp "/home/carlos/Downloads/JDBCTutorial/BD2-14-postgresql-42.2.4.jar:./"
StandAloneJDBCCode.java
```

```
carlos@carlos-VirtualBox:~/Downloads/JDBCTutorial$ java -cp "/home/carlos/Downloads/JDBCTutorial/BD
2-14-postgresql-42.2.4.jar:./" StandAloneJDBCCode.java
No arguments.
Nome cliente: Alexandre Márcio de SouzaSoma Depositos: 4227Soma Empréstimos: 1030
Nome cliente: Andrade de FreitasSoma Depositos: 20436Soma Empréstimos: 9442
Nome cliente: André Cabral da SilvaSoma Depositos: 22305Soma Empréstimos: 9579
Nome cliente: Bruno Miranda Pacheco de CastroSoma Depositos: 49523Soma Empréstimos: 21987
Nome cliente: Bruno Tadeu PitaSoma Depositos: 23565Soma Empréstimos: 22237
Nome cliente: Everardo Monfort LeitãoSoma Depositos: 18574Soma Empréstimos: 8741
Nome cliente: Felipe Peres Bichara JúniorSoma Depositos: 2970Soma Empréstimos: 1675
Nome cliente: Frederico de Oliveira GuimarãesSoma Depositos: 210Soma Empréstimos: 1992
Nome cliente: Geraldo OliveiraSoma Depositos: 3897Soma Empréstimos: 2109
Nome cliente: Germano Luiz de PaulaSoma Depositos: 2598Soma Empréstimos: 1415
Nome cliente: Gilmar Negreiros CarvalhoSoma Depositos: 2509Soma Empréstimos: 1640
Nome cliente: João BoiadeiroSoma Depositos: 13638Soma Empréstimos: 14632
Nome cliente: Joaquim Carlos ReisSoma Depositos: 17120Soma Empréstimos: 13092
Nome cliente: Josenaldo Teixeira SoaresSoma Depositos: 3105Soma Empréstimos: 1380
Nome cliente: Manuel OliveiraSoma Depositos: 2207Soma Empréstimos: 1396
Nome cliente: Marcos AndradeSoma Depositos: 243082Soma Empréstimos: 94620
Nome cliente: Marcos CláudioSoma Depositos: 7400Soma Empréstimos: 2897
Nome cliente: Maria Raquel Teixeira de CastroSoma Depositos: 2968Soma Empréstimos: 582
Nome cliente: Marta Avelar SantosSoma Depositos: 145425Soma Empréstimos: 49756
Nome cliente: Rafael Mesquita GondimSoma Depositos: 27364Soma Empréstimos: 10429
Nome cliente: Thiago Andrade FiuzaSoma Depositos: 6012Soma Empréstimos: 672
```

7) Todo tipo de alteração no código foi notificada como uma pequena explicação do que cada método executa.