Trabalho 08 - BD2

Guilherme Rimoldi Kameoka
Júlio César de Jesus Batista

1. **Importação das bibliotecas:**

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

Aqui apenas estamos fazendo a importação das bibliotecas necessárias para o funcionamento do programa.

2. **Método getConnection():**

```java
public class StandAloneJDBCCode {
    Codeium: Refactor | Explain | Generate Javadoc
    public static Connection getConnection() {
        Connection con = null;
        String currentUrlString = null;
        Properties connectionProps = new Properties();

        connectionProps.put(key:"user", value:"guilhermekameoka");
        connectionProps.put(key:"password", value:"postgres");

        currentUrlString = "jdbc:postgresql://localhost:5432/IB2";

        try {
            con = DriverManager.getConnection(currentUrlString, connectionProps);
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return con;

    }
```

A função getConnection() inicializa e retorna uma instância da conexão com o banco de dados. Neste caso, a conexão está sendo feita no banco IB2 através do localhost. As credenciais de acesso são atribuídas às chaves "user" e "password".

3. **Método myquerry:**

```
public static void myquery(Connection con) throws SQLException {
    Statement stmt = null;
    String query = "SELECT DISTINCT NOME_CLIENTE, SUM(SALDO_DEPOSITO) AS TOTAL_DEP " +
            "FROM DEPOSITO " +
            "WHERE NOME_CLIENTE NOT IN (SELECT DISTINCT NOME_CLIENTE FROM EMPRESTIMO) " +
            "GROUP BY NOME_CLIENTE";
    try {
        stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery(query);
        System.out.println(x:"\nClientes que possuem apenas depositos:\n");
        while (rs.next()) {
            String cliente = rs.getString(columnLabel:"NOME_CLIENTE");
            float soma = rs.getFloat(columnLabel:"TOTAL_DEP");
            System.out.printf(format:"     Nome: %35s - Depositos: R$ %6.2f\n", cliente, soma);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        if (stmt != null) {
            stmt.close();
        }
    }
}
```

A função myquery é responsável por interagir com o banco de dados, fazendo uso do objeto de conexão obtido por meio da função getConnection(). Neste processo, ela realiza a execução de uma consulta SQL, cuja estrutura está definida na variável "query". Ao executar a consulta, os resultados são recuperados um a um e, posteriormente, são exibidos no console.O propósito desse código é identificar e listar os clientes que possuem exclusivamente depósitos em suas contas bancárias. Para cada cliente identificado, o programa apresenta seus nomes e o montante total dos depósitos.

4. **Método closeConnection**

```java
public static void closeConnection(Connection con) {
    try {
        System.out.println(x:"Released all database resources.");
        if (con != null) {
            con.close();
            con = null;
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}
```

Este método serve para fechar a conexão com o banco de dados. Primeiro, ele checa se a conexão está aberta, e se estiver, fecha e a deixa como se não existisse. Se algo der errado durante o processo de fechamento, como um problema no banco de dados, ele mostra uma mensagem de erro no prompt.

4. Função main

```
public static void main(String[] args) {
    if (args.length == 0) {
        System.err.println(x:"No arguments.");
    }
    Connection myConnection = null;
    try {
        myConnection = StandAloneJDBCCode.getConnection();
        StandAloneJDBCCode.myquery(myConnection);
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        StandAloneJDBCCode.closeConnection(myConnection);
    }
}
```

Essa função verifica se argumentos foram passados na linha de comando. Se não houver argumentos, uma mensagem de erro será impressa no prompt. Em seguida, a função tenta estabelecer uma conexão com o banco de dados usando a função getConnection() e, se for bem sucedida, ela chamará a função myquery() para realizar a consulta do item 3 no banco de dados. Se ocorrer algum erro durante a execução, ela irá imprimir a mensagem de erro. Por fim, mesmo que tudo tenha corrido bem ou não, o programa garante que a conexão seja fechada usando a função closeConnection().

5. **Compilando e executando o arquivo StandAloneJDBCCode para efetuar a consulta no banco de dados:**

```
guilhermekameoka@192 JDBCTutorial % java -cp "/Users/guilhermekameoka/Desktop/my
dir/JDBCTutorial/BD2-14-postgresql-42.2.4.jar:./" StandAloneJDBCCode
No arguments.

Clientes que possuem apenas depositos:

     Nome:                    Maria das Dores - Depositos: R$ 3252.82
     Nome:                 Pedro Alvares Sousa - Depositos: R$ 4901.22
     Nome:                   Maria Lúcia Alves - Depositos: R$ 2354.96
     Nome:               Carolina Soares Souza - Depositos: R$ 4549.06
     Nome:              Clayton Pereira Bonfim - Depositos: R$ 9683.09
     Nome:              Eurides Alves da Silva - Depositos: R$ 17537.87
     Nome:                  Fábio Couto Amorim - Depositos: R$ 1244.07
     Nome:            Norton Saint Clair Silva - Depositos: R$ 3527.83
     Nome: Lorena Albuquerque Gonçalves Galdin - Depositos: R$ 5471.91
     Nome:                  Jefferson Oliveira - Depositos: R$ 3712.37
     Nome:                 Adilson de Oliveira - Depositos: R$ 6381.16
     Nome:                Marco Aurélio Santos - Depositos: R$ 4110.75
     Nome:          Reinaldo Pereira da Silva - Depositos: R$ 5460.60
     Nome:          Gustavo Baer Albuquerque  - Depositos: R$ 7179.74
     Nome:                 Cláudia Santos Mota - Depositos: R$ 10034.17
     Nome:            Elvis Fernando Da Silva - Depositos: R$ 7147.54
Released all database resources.
guilhermekameoka@192 JDBCTutorial %
```

A execução do programa efetua a query do item 3 e devolve no prompt o resultado da consulta.