

T07-Recuperando e modificando valores

Jefferson Cristino da Costa - 11821BSI227

1-

```
1 SELECT cliente.nome_cliente,  
2     sum(emprestimo.valor_emprestimo),  
3     sum(deposito.saldo_deposito)  
4 FROM cliente  
5 inner join deposito on cliente.nome_cliente = deposito.nome_cliente  
6 inner join emprestimo on cliente.nome_cliente = emprestimo.nome_cliente  
7 inner join conta on conta.nome_cliente = cliente.nome_cliente  
8 group by cliente.nome_cliente, conta.nome_agencia, conta.numero_conta;
```

Data Output

	nome_cliente [PK] character varying	sum double precision	sum double precision
1	João Boiadeiro	14632.68	13571.039999999999
2	Geraldo Oliveira	2109.7400000000002	3877.96
3	Josenaldo Teixeira Soares	1380.92	3090.27
4	Marcos Andrade	94620.15000000002	241873.17000000025
5	André Cabral da Silva	9579.359999999999	22194.440000000002
6	Marcos Cláudio	2897.9700000000003	7363.9800000000005
7	Marcos Andrade	94620.15000000002	241873.17000000025
8	Joaquim Carlos Reis	13092.399999999998	17034.999999999996

2- Adicionando getData3 ao MyQueries e compilando:

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml  
Set the following properties:  
dbms: postgresql  
driver: com.mysql.cj.jdbc.Driver  
dbName: IB2  
userName: postgres  
serverName: localhost  
portNumber: 5432  
Connected to database  
Contas da Instituicao Bancaria:  
55282, Central  
50073, Cidade Jardim  
39864, Gameleira  
17477, Glória  
95617, Central  
30273, Glória  
66119, Pampulha  
50906, Central  
69118, Central  
24035, Pampulha  
70719, Glória  
77 e.printStackTrace();
```

3- Modificando a consulta do MyQueries para a mesma do item 2:

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
Consulta exercício 2:
João Boiadeiro, 14632.68, 13571.039999999999
Geraldo Oliveira, 2109.7400000000002, 3877.96
Josenaldo Teixeira Soares, 1380.92, 3090.27
Marcos Andrade, 94620.15000000002, 241873.17000000025
André Cabral da Silva, 9579.359999999999, 22194.440000000002
Marcos Cláudio, 2897.9700000000003, 7363.9800000000005
Marcos Andrade, 94620.15000000002, 241873.17000000025
Joaquim Carlos Reis, 13092.399999999998, 17034.999999999996
Marta Avelar Santos, 49756.28, 144701.639999999996
Andrade de Freitas, 9442.89, 20335.08
Thiago Leôncio Guimarães, 26114.800000000003, 48497.94
Bruno Miranda Pacheco de Castro, 21987.4, 49276.799999999996
```

4- EXPLICAR

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
ResultSet.HOLD_CURSORS_OVER_COMMIT = 1
ResultSet.CLOSE_CURSORS_AT_COMMIT = 2
Default cursor holdability: 1
Supports HOLD_CURSORS_OVER_COMMIT? true
Supports CLOSE_CURSORS_AT_COMMIT? true
Releasing all open resources ...
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$
```

ResultSet.HOLD_CURSORS_OVER_COMMIT: O objeto ResultSet, continua aberto depois de fazer o

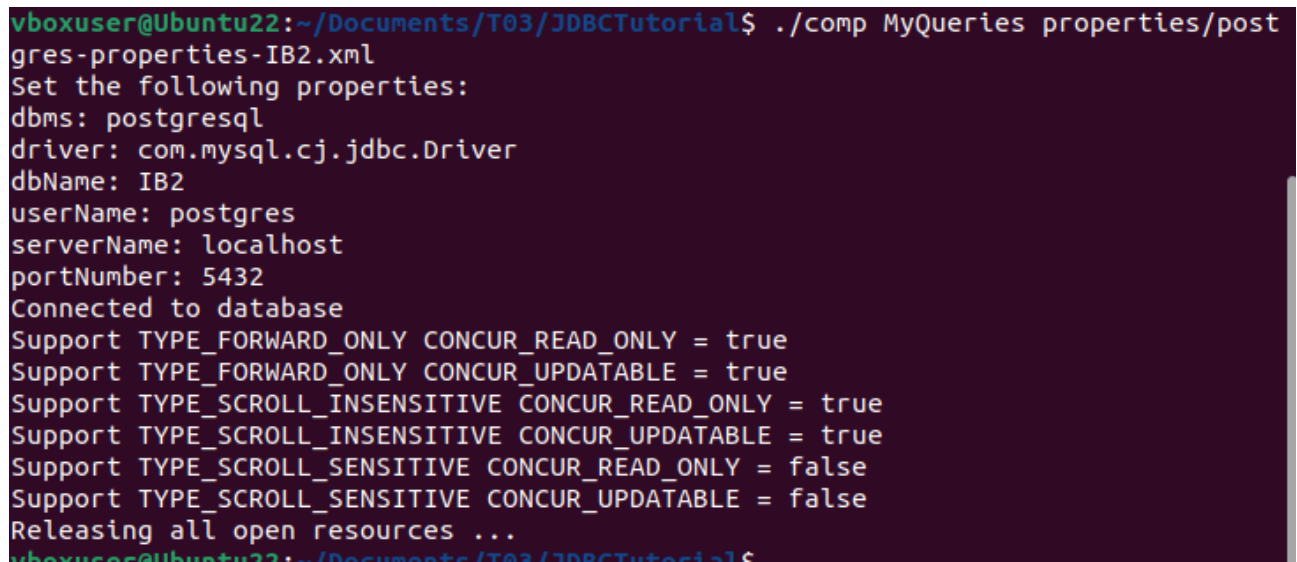
commit das alterações ou busca.

ResultSet.CLOSE_CURSORS_AT_COMMIT: O objeto ResultSet, é fechado depois de fazer o commit

das alterações ou busca, melhor para performance.

Holdability: mostra se o SGBD funciona por padrão com `HOLD_CURSORS_OVER_COMMIT` ou `.CLOSE_CURSORS_AT_COMMIT`.

5- Verificando suporte do banco de dados para atualização de um ResultSet:

A terminal window on Ubuntu 22.04 showing the execution of a Java program. The program sets JDBC properties for a PostgreSQL database and then checks for support of different cursor types for update operations. The output shows that forward-only cursors support updates, while scroll-sensitive cursors do not.

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
Support TYPE_FORWARD_ONLY CONCUR_READ_ONLY = true
Support TYPE_FORWARD_ONLY CONCUR_UPDATABLE = true
Support TYPE_SCROLL_INSENSITIVE CONCUR_READ_ONLY = true
Support TYPE_SCROLL_INSENSITIVE CONCUR_UPDATABLE = true
Support TYPE_SCROLL_SENSITIVE CONCUR_READ_ONLY = false
Support TYPE_SCROLL_SENSITIVE CONCUR_UPDATABLE = false
Releasing all open resources ...
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$
```

6- O código propõe a fazer uma atualização em todos os preços de café

O erro do código está no segundo `createStatement()` pois no segundo parâmetro está utilizando `CONCUR_READ_ONLY`, logo como o código busca alterar valores no banco de dados é necessário utilizar o `CONCUR_UPDATABLE`.

7- Modificando código do exercício 6 para utilizar no banco IB2:

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
Depositos Atualizados
Releasing all open resources ...
```

8- Modificando código do exercício 7, para usuário informar o aumento dos depósitos:

```
public static void modifyPrices(Connection con) throws SQLException, IOException {
    System.out.println("Digite o multiplicador como um numero real (Ex.: 5% = 1,05):");
    Scanner in = new Scanner(System.in);
    double percentage = in.nextDouble();

    Statement stmt = null;
    try {
        stmt = con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE, ResultSet.CONCUR_UPDATABLE);
        ResultSet uprs = stmt.executeQuery( "SELECT * FROM deposito");
        while (uprs.next()) {
            float f = uprs.getFloat("saldo_deposito");
            uprs.updateDouble("saldo_deposito", f * percentage);
            uprs.updateRow();
        }
        System.out.println("Depositos Atualizados");
    } catch (SQLException e) {
        JDBCTutorialUtilities.printStackTrace(e);
    } finally {
        if (stmt != null) {
            stmt.close();
        }
    }
}
```

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
Digite o multiplicador como um numero real (Ex.: 5% = 1,05):
0,1
Depositos Atualizados
Releasing all open resources ...
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$
```


9- Fazendo inserções com batch do ultimo trabalho na populateTable:

```
try {
    stmt = con.createStatement();

    inputStream = new BufferedReader(new FileReader("/home/vboxuser/Downloads/debito-populate-table.txt"));
    stmt.executeUpdate("truncate table debito;");

    while ((line = inputStream.readLine()) != null) {
        countv=0;
        scanned_line = new Scanner(line);
        scanned_line.useDelimiter("\t");
        while (scanned_line.hasNext()) {
            value[countv++]=scanned_line.next();
        } //while
        if (scanned_line != null) { scanned_line.close(); }

        stmt.addBatch("insert into debito (numero_debito, valor_debito, motivo_debito, data_debito, numero_conta, no
nome_cliente) " + "values (" + value[0] + ", " + value[1] + ", " + value[2] + ", " + value[3] + ", " + value[4] + ", " + val
" + value[6] + " + " + "));";
    } //while

    int[] updateCounts = stmt.executeBatch();
    con.commit();
}
```

```
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
Releasing all open resources ...
```

10- Inserindo linhas na tabela Debito, com variáveis no programa para posterior adição ao BD:

```
public static void insertRow(Connection con, Double valor_debito, int motivo, String data_debito, int Conta, String agencia,
String Cliente ) throws SQLException, IOException {
    Statement stmt = null;
    try {
        stmt = con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE, ResultSet.CONCUR_UPDATABLE);
        ResultSet uprs = stmt.executeQuery("SELECT * FROM debito");
        uprs.last();
        int numero_debito = uprs.getInt("numero_debito");
        uprs.moveToInsertRow(); //posiciona no ponto de inserção da tabela
        uprs.updateInt("numero_debito", numero_debito + 1);
        uprs.updateDouble("valor_debito", valor_debito);
        uprs.updateInt("motivo_debito", motivo);
        uprs.updateDate("data_debito", Date.valueOf(data_debito));
        uprs.updateInt("numero_conta", Conta);
        uprs.updateString("nome_agencia", agencia);
        uprs.updateString("nome_cliente", Cliente);
        uprs.insertRow(); //insere a linha na tabela
        uprs.beforeFirst(); //posiciona-se novamente na posição anterior ao primeiro registro

        System.out.println("Debito feito com sucesso: " + Cliente);

    } catch (SQLException e) {
        JDBCTutorialUtilities.printSQLException(e);
    } finally {
        if (stmt != null) {
            stmt.close();
        }
    }
}
```

```
1 error
vboxuser@Ubuntu22:~/Documents/T03/JDBCTutorial$ ./comp MyQueries properties/postgres-properties-IB2.xml
Set the following properties:
dbms: postgresql
driver: com.mysql.cj.jdbc.Driver
dbName: IB2
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
Debito feito com sucesso: Carla Soares Sousa
Debito feito com sucesso: Carolina Soares Souza
Debito feito com sucesso: Eurides Alves da Silva
Releasing all open resources ...
```