

Aula 08 – Estabelecendo conexões

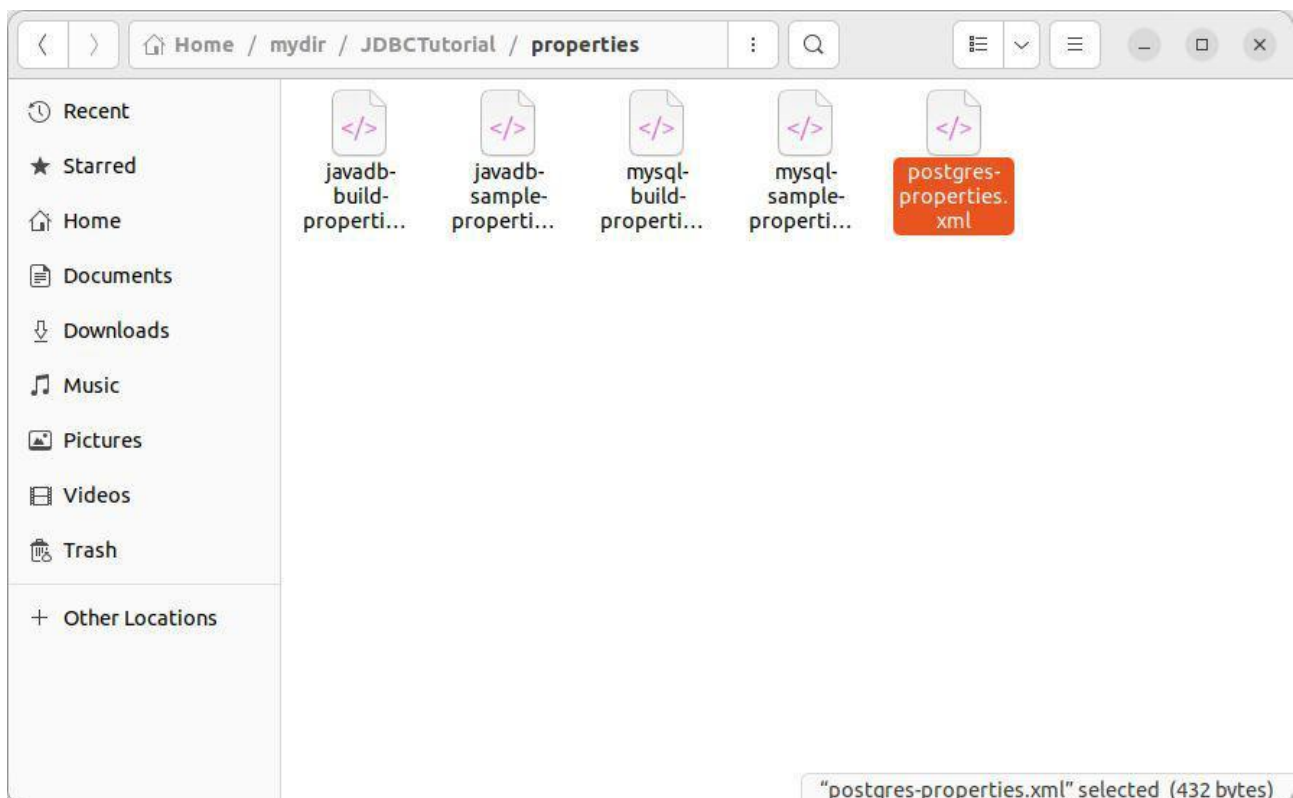
Jefferson Cristino da Costa - 11821BSI227

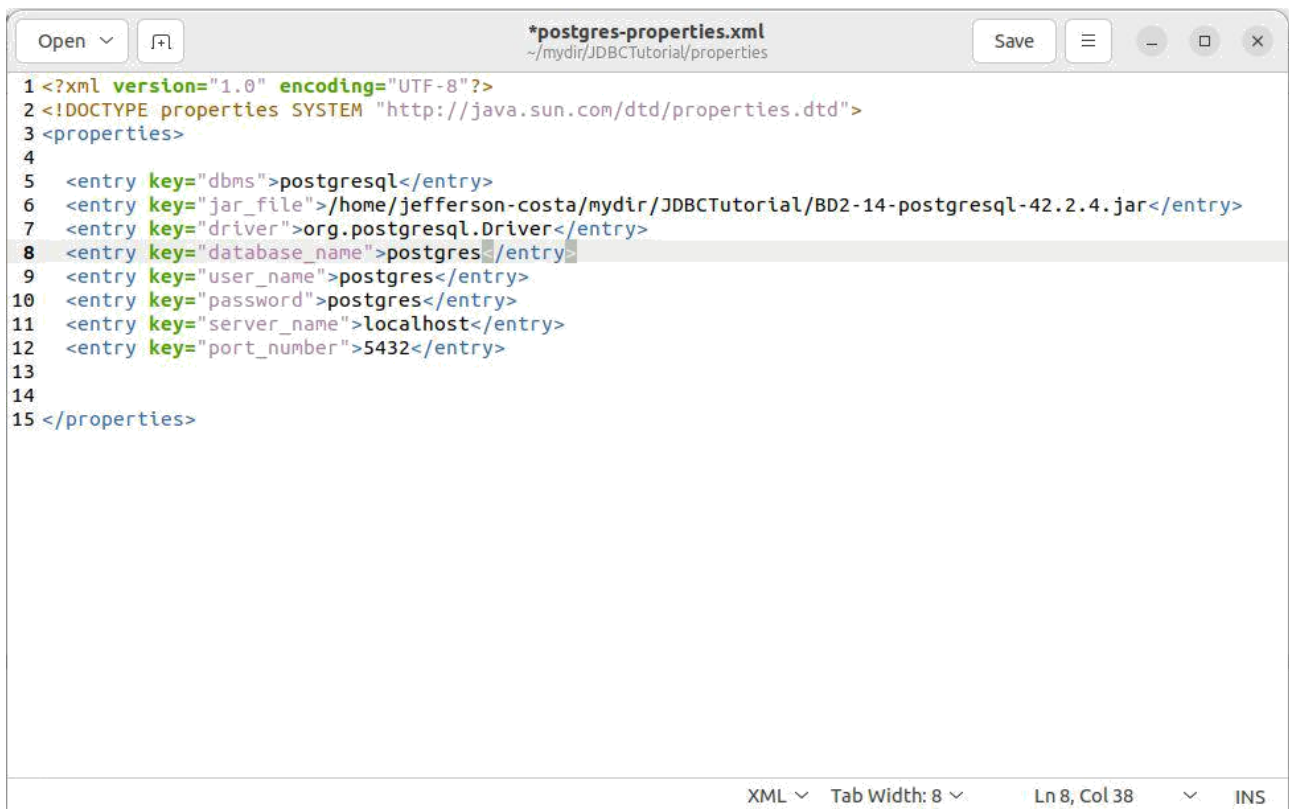
Baixe o arquivo postgresql-42.2.4.jar do moodle. Depois abra o terminal de comandos do linux e posicione-se no diretório em que baixou o arquivo postgresql-42.2.4.jar para esta aula. Para que não se recorde: use para isso o comando cd.

Copie o drive do postgres de nome postgresql-42.2.4.jar para a pasta raiz do tutorial do java digitando o comando cp abaixo:

```
jefferson-cost@jeffersoncosta:~$ cd Downloads/
jefferson-cost@jeffersoncosta:~/Downloads$ ls
BD2-13-source_code  BD2-13-source_code.zip  BD2-14-postgresql-b2.2.4.jar  JDBCTutorial.zip  jdk-8u31-linux-x64.tar.gz
jefferson-cost@jeffersoncosta:~/Downloads$ cd BD2-14-postgresql-b2.2.4.jar
jefferson-cost@jeffersoncosta:~/BD2-14-postgresql-b2.2.4.jar$ cd /home/jefferson-cost/mydir/JDBCTutorial/
jefferson-cost@jeffersoncosta:~/Downloads$ cd /home/jefferson-cost/mydir/JDBCTutorial/ls
bash: cd: /home/jefferson-cost/mydir/JDBCTutorial/ls: No such file or directory
jefferson-cost@jeffersoncosta:~/Downloads$ cd /home/jefferson-cost/mydir/JDBCTutorial/
jefferson-cost@jeffersoncosta:~/mydir/JDBCTutorial$ ls
BD2-14-postgresql-b2.2.4.jar  build.xml  classes  com  JDBCUtillities.java  lib  MyQueries.java  properties  sql  src  test  xml
jefferson-cost@jeffersoncosta:~/mydir/JDBCTutorial$
```

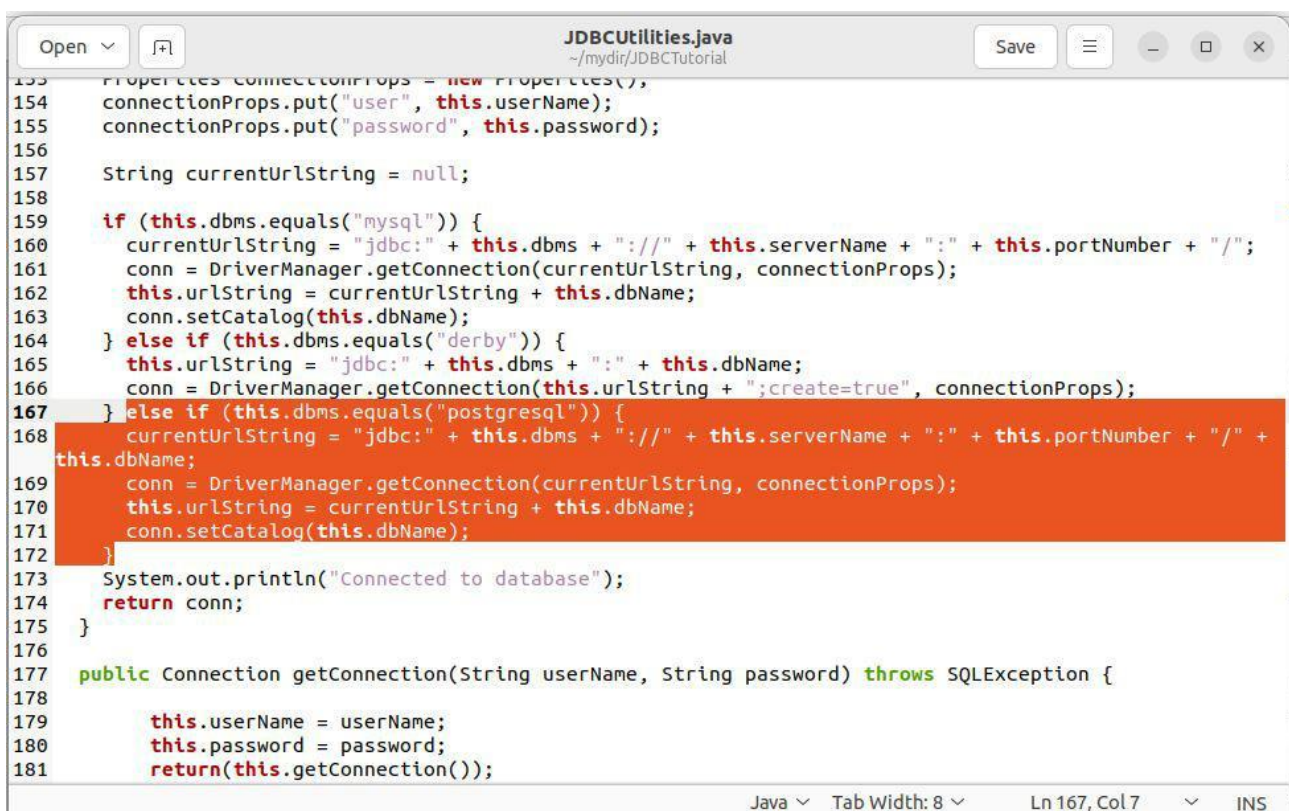
Faça uma cópia do arquivo `mysql-sample-properties.xml`, que está na pasta `properties`, com o nome de `postgres-properties.xml`. Esse novo arquivo será utilizado para se conectar ao banco de dados do postgres. Terminada a cópia: edite o novo arquivo com os parâmetros corretos, lembre-se do usuário e senha do usuário postgres e da porta de conexão padrão. Se você não estiver conseguindo se lembrar, tente criar uma conexão usando, por exemplo, o NetBeans. Parâmetros iguais devem ser configurados no arquivo `postgres-properties.xml`.





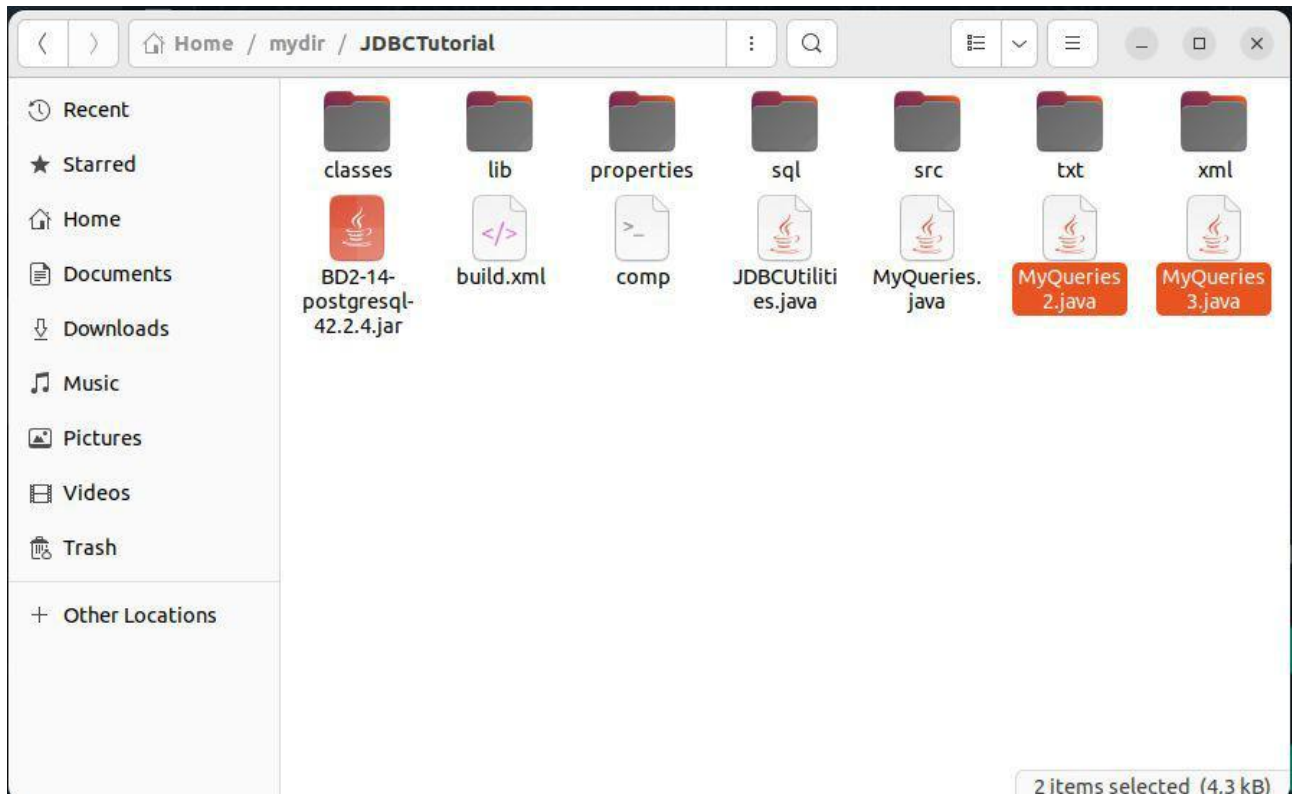
```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
3 <properties>
4
5   <entry key="dbms">postgresql</entry>
6   <entry key="jar_file">/home/jefferson-costa/mydir/JDBCTutorial/BD2-14-postgresql-42.2.4.jar</entry>
7   <entry key="driver">org.postgresql.Driver</entry>
8   <entry key="database_name">postgres</entry>
9   <entry key="user_name">postgres</entry>
10  <entry key="password">postgres</entry>
11  <entry key="server_name">localhost</entry>
12  <entry key="port_number">5432</entry>
13
14
15 </properties>
```

Modifique o programa JDBCUtilities.java para incorporar estas linhas que o habilitam a se conectar a um banco de dados postgres. Veja que no código existem duas funções com uma estrutura if para se conectar ao derby e ao mysql. Altere ambas as funções acrescentando este código:

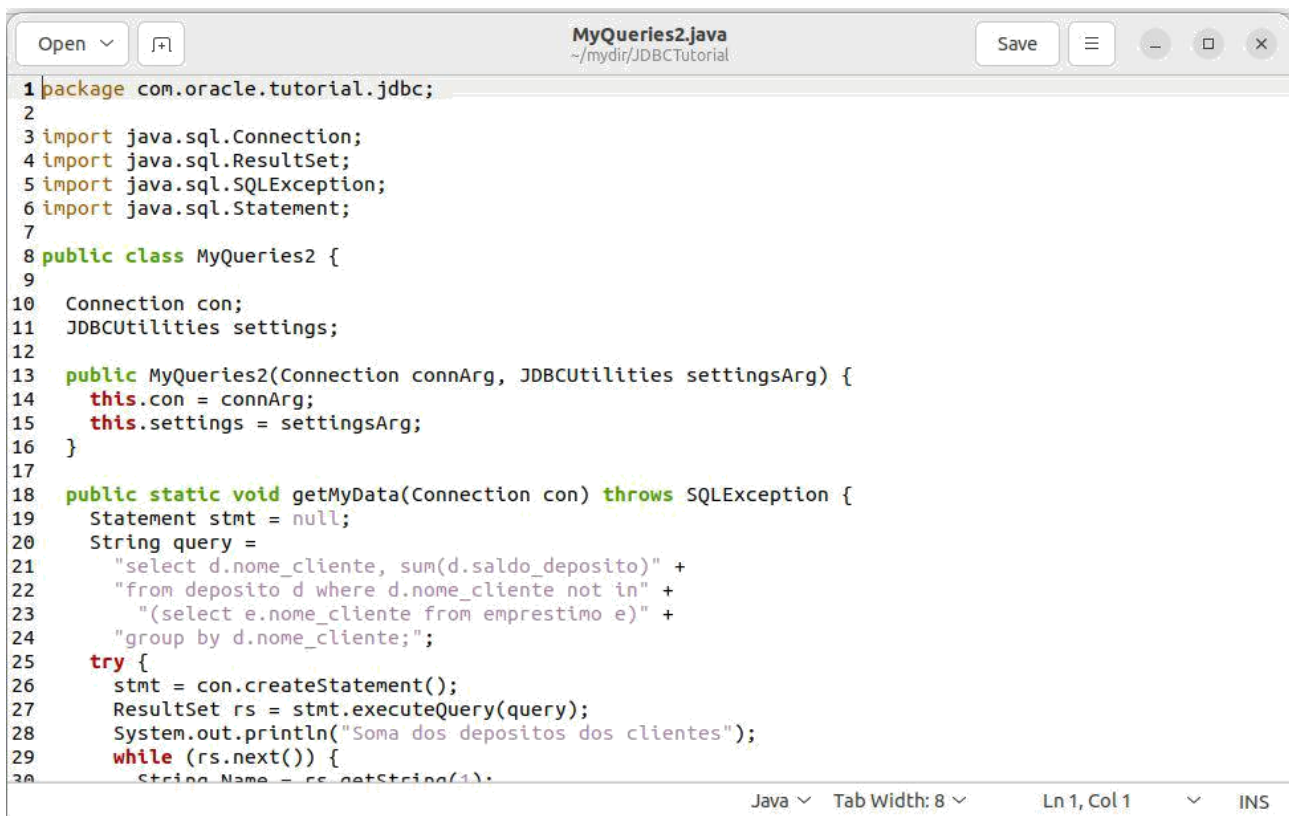


```
153 Properties connectionProps = new Properties();
154 connectionProps.put("user", this.userName);
155 connectionProps.put("password", this.password);
156
157 String currentUrlString = null;
158
159 if (this.dbms.equals("mysql")) {
160     currentUrlString = "jdbc:" + this.dbms + "://" + this.serverName + ":" + this.portNumber + "/" +
161     conn = DriverManager.getConnection(currentUrlString, connectionProps);
162     this.urlString = currentUrlString + this.dbName;
163     conn.setCatalog(this.dbName);
164 } else if (this.dbms.equals("derby")) {
165     this.urlString = "jdbc:" + this.dbms + ":" + this.dbName;
166     conn = DriverManager.getConnection(this.urlString + ";create=true", connectionProps);
167 } else if (this.dbms.equals("postgresql")) {
168     currentUrlString = "jdbc:" + this.dbms + "://" + this.serverName + ":" + this.portNumber + "/" +
169     this.dbName;
170     conn = DriverManager.getConnection(currentUrlString, connectionProps);
171     this.urlString = currentUrlString + this.dbName;
172     conn.setCatalog(this.dbName);
173 }
174 System.out.println("Connected to database");
175 return conn;
176 }
177
178 public Connection getConnection(String userName, String password) throws SQLException {
179     this.userName = userName;
180     this.password = password;
181     return(this.getConnection());
182 }
```

Crie duas novas classes a partir da classe MyQueries.java (MyQueries2.java e MyQueries3.java) para realizar as seguintes consultas e exibir os resultados na tela da linha de comando. Para esse fim é necessário tanto mudar o nome do arquivo quanto das referências à classe que está definida dentro do arquivo ou a Java vai acusar erro de compilação. Estas são as consultas que os novos programas devem fazer

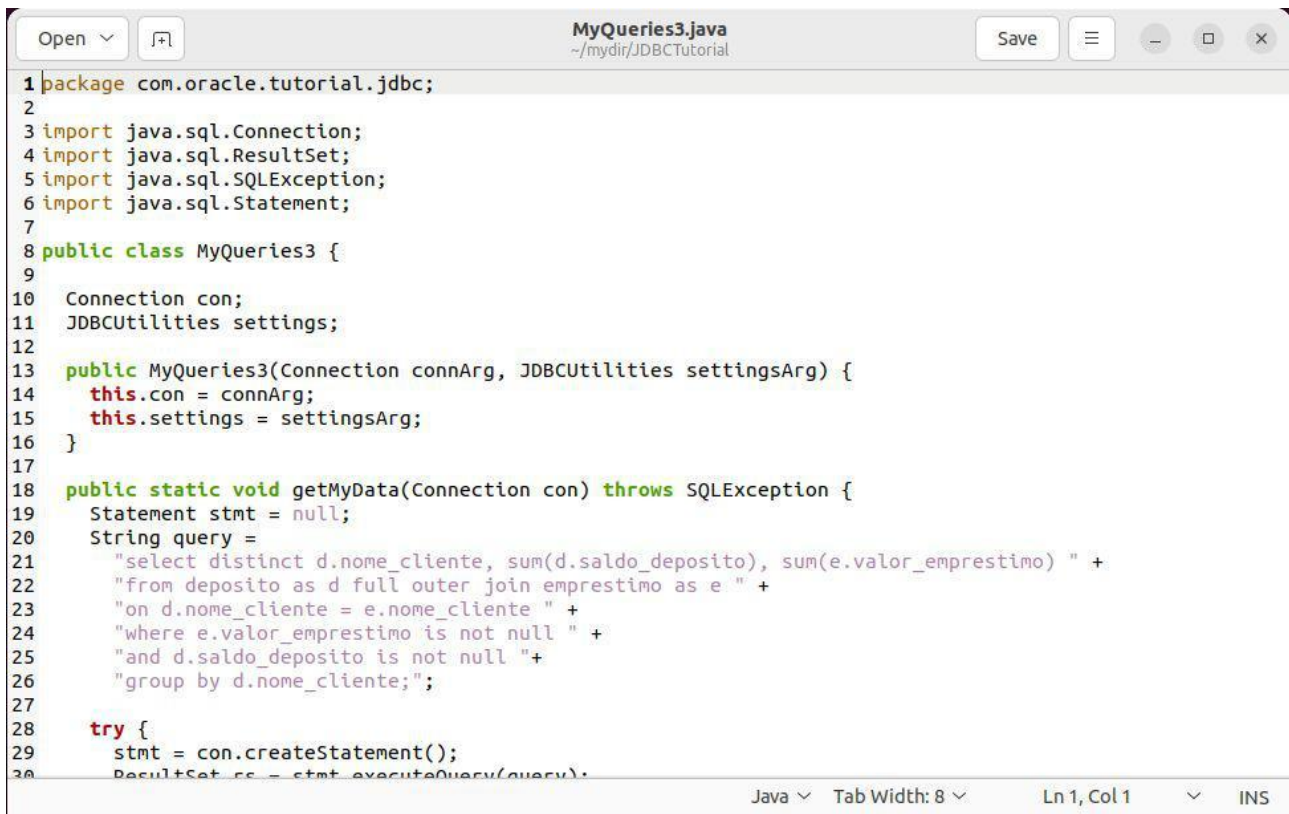


a. Retorne os nomes de todos os clientes que possuem apenas depósitos, bem como a soma de depósitos de cada cliente (Implementar no MyQueries2.java);



```
1 package com.oracle.tutorial.jdbc;
2
3 import java.sql.Connection;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6 import java.sql.Statement;
7
8 public class MyQueries2 {
9
10     Connection con;
11     JDBCUtilities settings;
12
13     public MyQueries2(Connection connArg, JDBCUtilities settingsArg) {
14         this.con = connArg;
15         this.settings = settingsArg;
16     }
17
18     public static void getMyData(Connection con) throws SQLException {
19         Statement stmt = null;
20         String query =
21             "select d.nome_cliente, sum(d.saldo_deposito)" +
22             "from deposito d where d.nome_cliente not in" +
23             "(select e.nome_cliente from emprestimo e)" +
24             "group by d.nome_cliente;";
25         try {
26             stmt = con.createStatement();
27             ResultSet rs = stmt.executeQuery(query);
28             System.out.println("Soma dos depositos dos clientes");
29             while (rs.next()) {
30                 String Name = rs.getString(1);
```

b. Retorne os nomes dos clientes que possuem depósitos e empréstimos (ambos) com as respectivas somas (Implementar no MyQueries3.java);



```
1 package com.oracle.tutorial.jdbc;
2
3 import java.sql.Connection;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6 import java.sql.Statement;
7
8 public class MyQueries3 {
9
10     Connection con;
11     JDBCUtilities settings;
12
13     public MyQueries3(Connection connArg, JDBCUtilities settingsArg) {
14         this.con = connArg;
15         this.settings = settingsArg;
16     }
17
18     public static void getMyData(Connection con) throws SQLException {
19         Statement stmt = null;
20         String query =
21             "select distinct d.nome_cliente, sum(d.saldo_deposito), sum(e.valor_emprestimo) " +
22             "from deposito as d full outer join emprestimo as e " +
23             "on d.nome_cliente = e.nome_cliente " +
24             "where e.valor_emprestimo is not null " +
25             "and d.saldo_deposito is not null " +
26             "group by d.nome_cliente;";
27
28         try {
29             stmt = con.createStatement();
30             ResultSet rs = stmt.executeQuery(query);
```

Modifique o arquivo 'comp' da aula anterior para criar uma variável postgres com a configuração adequada à estrutura de diretórios de sua máquina. Use a

variável derby como exemplo de como deve ficar a variável postgres dentro do arquivo comp.

- a. postgres=/home/myuser/mydir/JDBCTutorial/postgresql-42.2.4.jar
- b. Acrescente a variável \$postgres no parâmetro -classpath (-cp) ao acionar o compilador java (denominado javac) e também ao acionar a máquina virtual java (denominado java). Modifique o script comp para este fim.
- c. Lembre-se que se o arquivo comp foi copiado agora e ainda não utilizado então ele ainda não tem permissão de execução. Neste caso o comando chmod 755 comp vai habilitá-lo para este fim;

```

1 #!/bin/sh -
2
3 postgres=/home/jefferson-costa/mydir/JDBCTutorial/postgresql-42.2.4.jar
4 mypath=/home/jefferson-costa/mydir/JDBCTutorial
5 mypackage=com/oracle/tutorial/jdbc
6
7 if [ -r $mypath/src/$mypackage/$1.java -a -r $mypath/$2 ]
8 then
9
10     javac -cp "$postgres:$mypath/classes:$mypath/lib/JDBCTutorial.jar" $mypath/src/$mypackage/
11     $1.java
12
13     if [ -r $mypath/src/$mypackage/$1.class ]
14     then
15         mv $mypath/src/$mypackage/$1.class $mypath/classes/$mypackage/
16
17         java -cp "$derby:$mypath/classes" $mypackage/$1 $mypath/$2
18     fi
19 else
20     echo Missing file, check:
21     ls $mypath/src/$mypackage/$1.java
22     ls $mypath/$2
23 fi
24
25

```

Saving file "/home/jefferson-costa/mydir/JDBCTutorial/comp"... sh Tab Width: 8 Ln 10, Col 29 INS

Compile primeiro a classe JDBCUtilities.java que foi modificada nas etapas anteriores:
./comp JDBCUtilities properties/javadb-sample-properties.xml:

```

jefferson-costa@jeffersoncosta:~/mydir/JDBCTutorial$ ./comp JDBCUtilities.java properties/postgres-properties.xml
Missing file, check:
ls: cannot access '/home/jefferson-costa/mydir/JDBCTutorial/src/com/oracle/tutorial/jdbc/JDBCUtilities.java.java': No such file or directory
/home/jefferson-costa/mydir/JDBCTutorial/properties/postgres-properties.xml
jefferson-costa@jeffersoncosta:~/mydir/JDBCTutorial$ ./comp JDBCUtilities properties/postgres-properties.xml
Reading properties file /home/jefferson-costa/mydir/JDBCTutorial/properties/postgres-properties.xml
Set the following properties:
dbms: postgresql
driver: org.postgresql.Driver
dbName: postgres
userName: postgres
serverName: localhost
portNumber: 5432
java.sql.SQLException: No suitable driver found for jdbc:postgresql://localhost:5432/postgres
    at java.sql.DriverManager.getConnection(DriverManager.java:689)
    at java.sql.DriverManager.getConnection(DriverManager.java:288)
    at com.oracle.jdbc.JDBCUtilities.getConnection(JDBCUtilities.java:169)
    at com.oracle.jdbc.JDBCUtilities.main(JDBCUtilities.java:214)
SQLState: 08001
Error Code: 0
Message: No suitable driver found for jdbc:postgresql://localhost:5432/postgres
Releasing all open resources ...
jefferson-costa@jeffersoncosta:~/mydir/JDBCTutorial$

```

Compile e execute em seguida as duas classes MyQueries2 e MyQueries3. Lembre-se que o novo arquivo postgres -properties.xml (passo 4) deve ser passado para todos os programas compilados e executados com o programa comp

```
Set the following properties:
dbms: postgresql
driver: org.postgresql.Driver
dbName: postgres
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
org.postgresql.jdbc.PgConnection@387c703b
Soma dos depositos dos clientes
    Cláudia Santos Mota      3599.29
    Thiago Andrade Fiuza    479.66
    Wantuil Diniz e Souza    84.17
    Alexandre Marcio de Souza 1201.37
    Felipe Peres Bichara Junior 7417.87
Releasing all open resources ...
```

```
Set the following properties:
dbms: postgresql
driver: org.postgresql.Driver
dbName: postgres
userName: postgres
serverName: localhost
portNumber: 5432
Connected to database
org.postgresql.jdbc.PgConnection@387c703b
Soma de depositos e emprestimos dos clientes:
    Marcos Andrade
        DEP - 3493.63
        EMP - 886.85
    Germano Luiz de Paula
        DEP - 9635.44
        EMP - 5026.95
    Gilmar Negreiros Carvalho
        DEP - 10230.12
        EMP - 4234.1
    Everardo Monfort Leitão
        DEP - 3562.54
        EMP - 3773.96
```