**Resolução - Quinta Lista de Exercícios - JDBC**

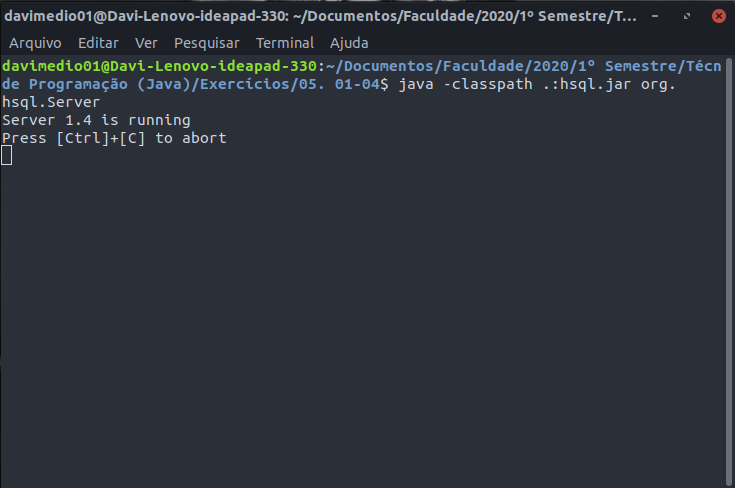
* **Server.properties (criar/abrir banco de dados no servidor local)**

//Criado por Davi Augusto - Exercício JDBC (Servidor)

//Abrir servidor com: java -classpath .:hsql.jar org.hsql.Server

port=8080

database=jdbc\_davineves



* **Criar.java** **(criar tabela)**

import java.sql.\*;

class Criar {

public static void main(String[] args) {

try{

//Carregando o driver do HypersonicSQL

Class.forName("org.hsql.jdbcDriver");

//Abrindo a conexão (porta 8080 definida em Server.properties)

Connection con = DriverManager.getConnection("jdbc:HypersonicSQL:hsql://localhost:8080", "sa", "");

//Criando variável para realizar comandos ao Banco de Dados (Statement)

Statement stmt = con.createStatement();

//Criando tabela e campos

//Tabela: Estudantes

//Campos: RA (Primary Key); Nome; Telefone; Curso

stmt.executeUpdate("CREATE TABLE ESTUDANTES (" +

"RA INT NOT NULL PRIMARY KEY," +

"NOME VARCHAR(50) NOT NULL," +

"TELEFONE CHAR(11)," +

"CURSO VARCHAR(30) NOT NULL" +

");");

//Fechando a conexão e a variável de comandos

stmt.close();

con.close();

}catch(Exception e){

System.out.println("Erro: " + e);

}

}

}

* **Inserir.java (inserir dados na tabela)**

import java.sql.\*;

class Inserir {

public static void main(String[] args) {

try{

// Carregando driver da HypersonicSQL

Class.forName("org.hsql.jdbcDriver");

//Abrindo a conexão

Connection con = DriverManager.getConnection("jdbc:HypersonicSQL:hsql://localhost:8080", "sa", "");

//Criando variável genérica (PreparedStatement) para comandos

PreparedStatement pStmt = con.prepareStatement("INSERT INTO ESTUDANTES VALUES (?, ?, ?, ?);");

//Inserindo alguns campos...

// Campo 1

pStmt.setInt(1, 191027383);

pStmt.setString(2, "Davi Neves");

pStmt.setString(3, "11958682744");

pStmt.setString(4, "BCC");

pStmt.executeUpdate(); // Atualizando tabela

// Campo 2

pStmt.setInt(1, 191128494);

pStmt.setString(2, "Luis Henrique");

pStmt.setString(3, "14999992176");

pStmt.setString(4, "Matemática");

pStmt.executeUpdate(); // Atualizando tabela

// Campo 3

pStmt.setInt(1, 204829942);

pStmt.setString(2, "Gustavo Stahl");

pStmt.setString(3, "19992740000");

pStmt.setString(4, "Química");

pStmt.executeUpdate(); // Atualizando tabela

// Campo 4

pStmt.setInt(1, 902948215);

pStmt.setString(2, "Rafael Kawagoe");

pStmt.setString(3, "11970005555");

pStmt.setString(4, "Design Gráfico");

pStmt.executeUpdate(); // Atualizando tabela

//Fechando banco de dados e comandos

pStmt.close();

con.close();

}catch(Exception e){

System.out.println("Erro: " + e);

}

}

}

* **Mostrar.java (mostrar os dados da tabela com comando select)**

import java.sql.\*;

class Mostrar {

public static void main(String[] args) {

try{

//Carregando driver do HypersonicSQL

Class.forName("org.hsql.jdbcDriver");

//Abrindo a conexão

Connection con = DriverManager.getConnection("jdbc:HypersonicSQL:hsql://localhost:8080", "sa", "");

//Variável para comandos no Banco de Dados

Statement stmt = con.createStatement();

//Criando variável para recuperar CADA campo da tabela (com Select)

ResultSet rs = stmt.executeQuery("SELECT \* FROM ESTUDANTES");

System.out.println("RA\t\tNOME\t\tTELEFONE\tCURSO\n");

while(rs.next()){

//Mostrando campos (por linha)

System.out.println(rs.getInt("RA") + "\t" + rs.getString("NOME") + "\t"

+ rs.getString("TELEFONE") + "\t" + rs.getString("CURSO"));

}

//Fechando a conexão e os comandos

stmt.close();

con.close();

}catch(Exception e){

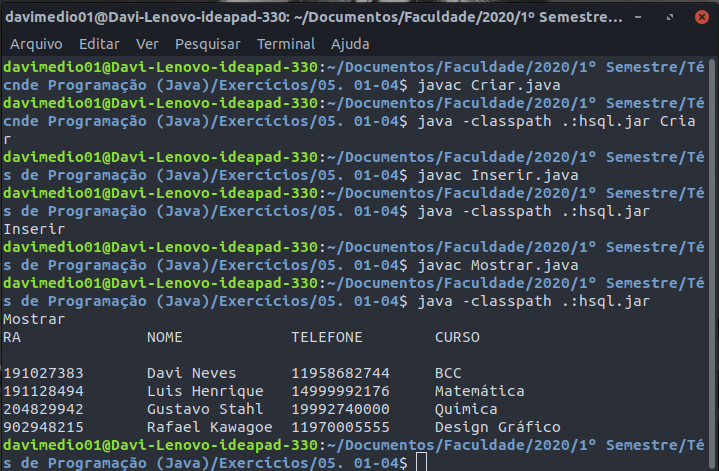
System.out.println("Erro: " + e);

}

}

}

**Imagem com comandos acima:**



**Pasta final (após comandos):**

