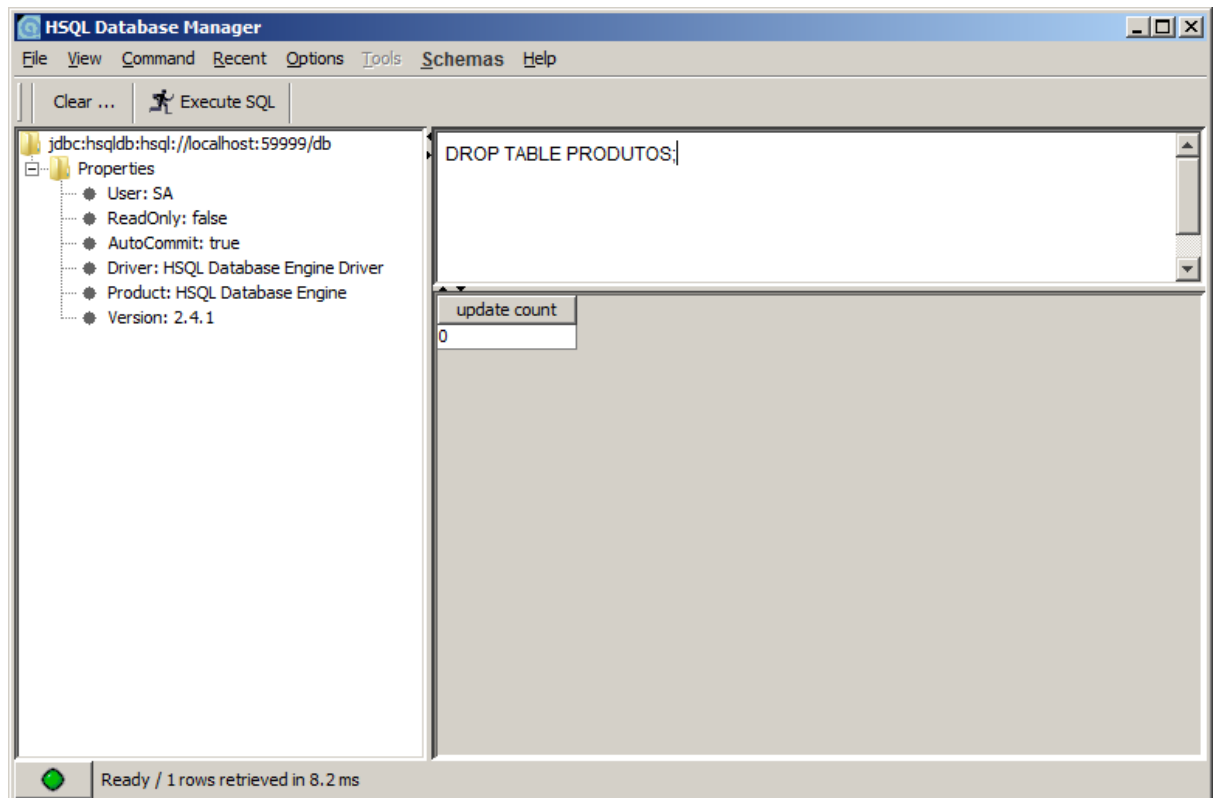


1. Introdução

Neste **exercício** faremos uma modificação na estrutura da tabela produtos no banco de dados **db**.

Inicialmente, vamos proceder à deleção da tabela **produtos**, por meio do comando:

DROP TABLE PRODUTOS;

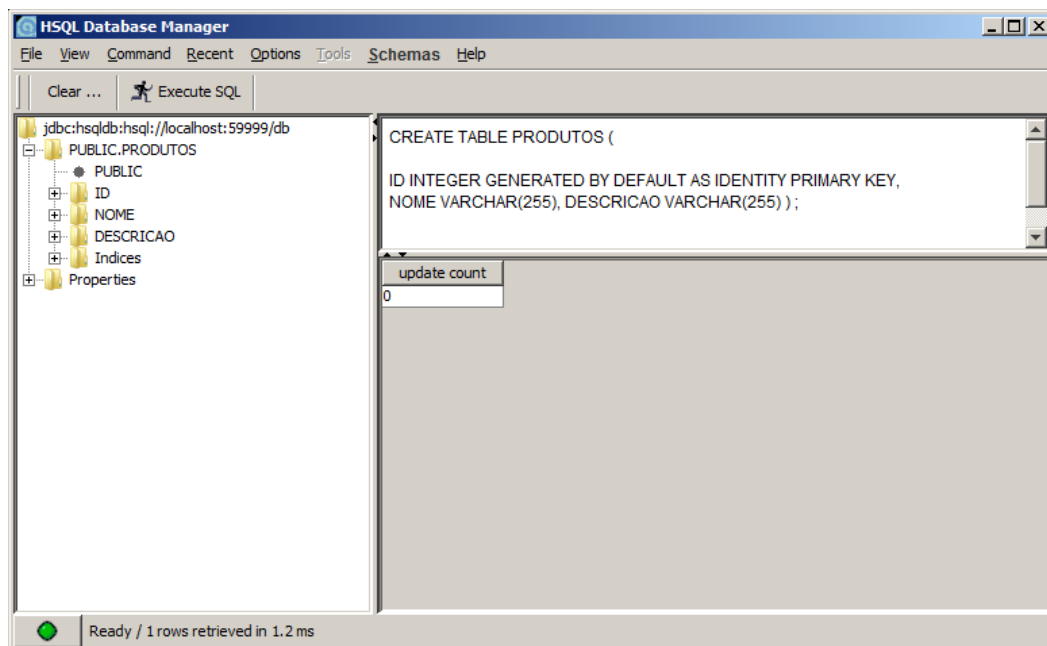


Recriar a tabela produtos com as seguintes definições

- **ID – INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY**
- **NOME – VARCHAR(255)**
- **DESCRICAO – VARCHAR(255)**

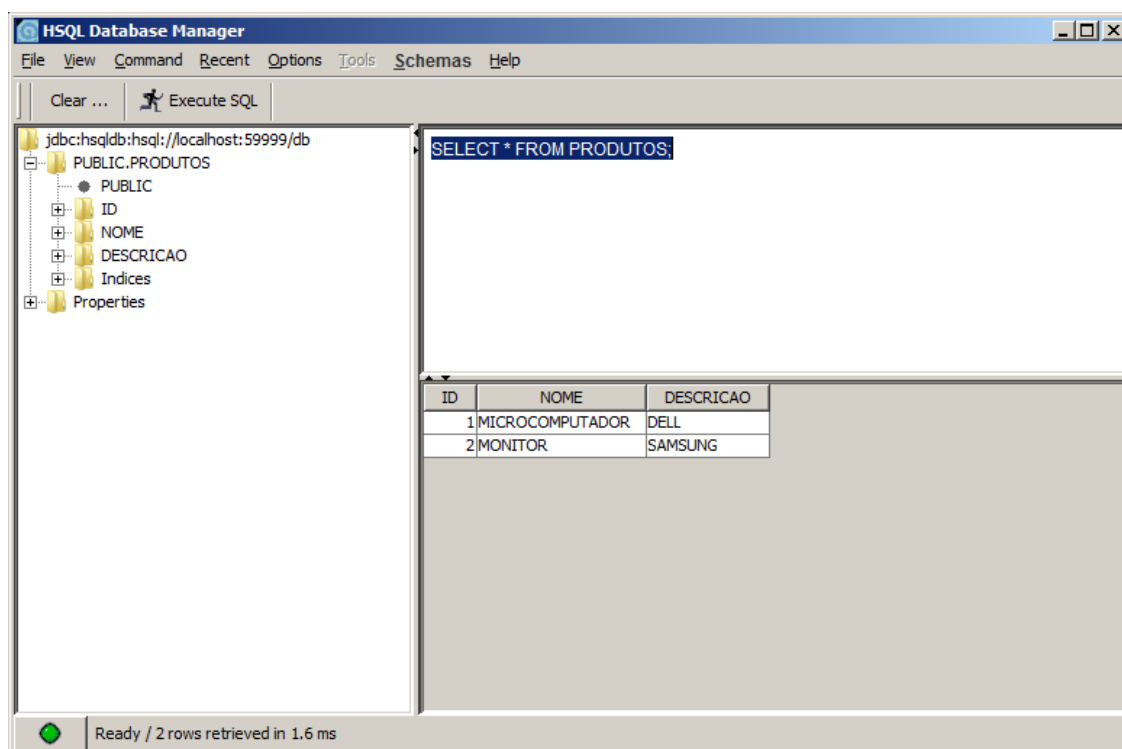
Entrar no cliente **HSQLDB** com o comando **SQL**:

CREATE TABLE PRODUTOS (ID INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY, NOME VARCHAR(255), DESCRICAO VARCHAR(255));



Vamos agora inserir dois novos produtos:

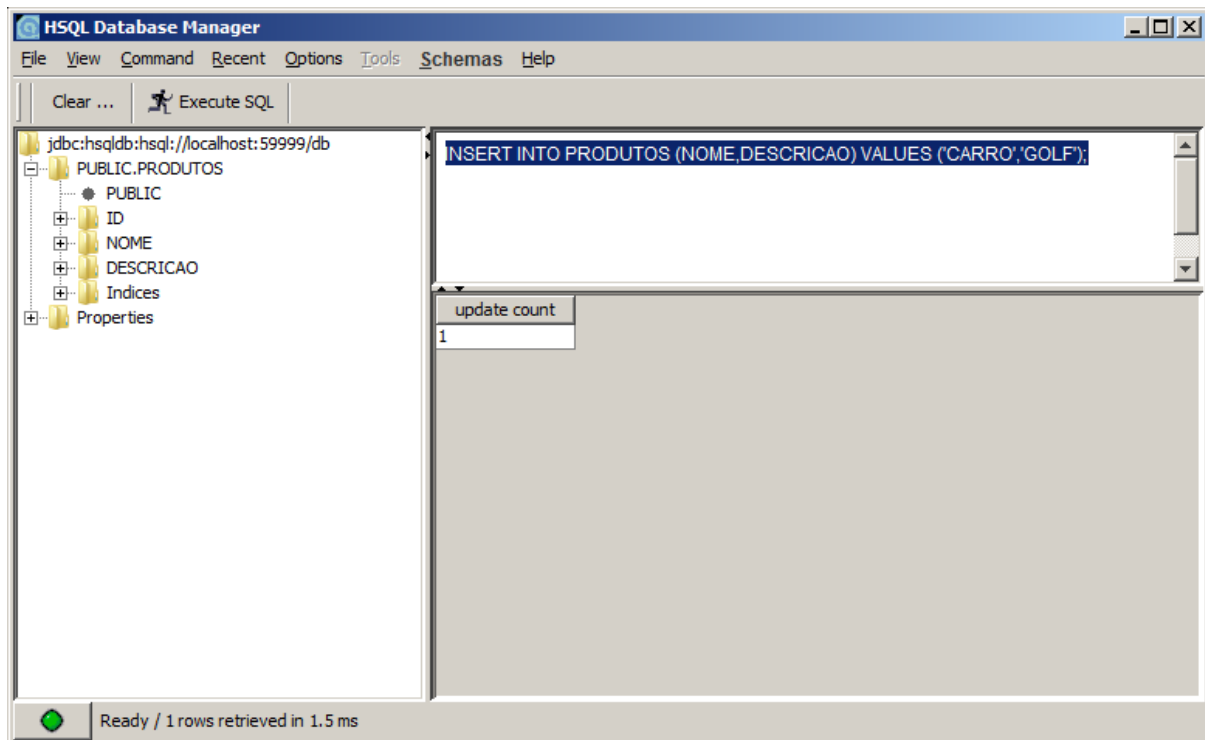
- **INSERT INTO PRODUTOS VALUES(1, 'MICROCOMPUTADOR', 'DELL');**
- **INSERT INTO PRODUTOS VALUES(2, 'MONITOR', 'SAMSUNG');**



Vamos agora inserir um novo produto com o comando **SQL**:

INSERT INTO PRODUTOS (NOME,DESCRICAO) VALUES ('CARRO', 'GOLF');

Embora o comando tenha sido executado com sucesso, não sabemos de antemão qual o código do produto que acabou de ser inserido.



Voltemos agora para o nosso código **Java**:

Para descobrirmos qual o **ID** gerado sem a necessidade de se executar novamente o **SELECT**, pode-se empregar a função **getGeneratedKeys()** do objeto **Statement**.

Vejamos como essa função é chamada no código:

```
package br.maua;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class InsereProdutos {

    public static void main(String[] args) throws SQLException {

        try {
            Connection c =
DriverManager.getConnection("jdbc:hsqldb:hsqldb://localhost:59999/db" , "SA" ,
null);

            System.out.println("conexao ao HSQLDB feita com SUCESSO ! ");
```

```

Statement stmt = c.createStatement();

        boolean resultado = stmt.execute("INSERT INTO PRODUTOS
(NOME,DESCRICAO) VALUES ('NOTEBOOK','SONY')", Statement.RETURN_GENERATED_KEYS);

        System.out.println("resultado = " + resultado);

        ResultSet resultSet = stmt.getGeneratedKeys();

        while (resultSet.next()) {

            Integer id = resultSet.getInt("id");
            System.out.println("ID " + id + "   gerado...");

        }

        stmt.close();

        c.close();

    }

    catch (SQLException e) {

        System.out.println("Erro SQLException....");

    }

    catch ( Exception e) {
        System.out.println("Problemas na conexao ao HSQLDB....");
    }

}

}

```

Caso o código acima seja executado várias vezes, teremos como resultado a gravação no console:

```

conexao ao HSQLDB feita com SUCESSO !
resultado = false
ID 7   gerado...

```

Observação:

A variável resultado retorna o valor **false**, pois, o método **execute** retornou apenas **um** resultado. A variável retornaria **true** quando se obtivesse uma lista de resultados, tal qual se faz com o comando **SELECT**.

Agora faremos operação de deleção de todos os **ID's** que são superiores a 3.

Mas, antes de se executar o **DELETE**, como descobrir quantas linhas foram removidas?

Uma remoção de tuplas na tabela, também é uma operação de atualização do banco de dados.

Para se obter essa informação (quantidade de tuplas afetadas pela operação de **delete**) pode-se empregar o método **getUpdateCount()** que informa o número de **tuplas** atualizadas.

Complemente o código abaixo para exibir a quantidade de linhas removidas:

```
package br.maua;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;

public class DeletaProduto {

    public static void main(String[] args) throws SQLException {

        try {
```

```
// Complemente o código aqui
```

```
}
```

```
catch (SQLException e) {

    System.out.println("Erro SQLException....");
```

```

        }

        catch ( Exception e) {
            System.out.println("Problemas na conexao ao HSQLDB....");
        }
    }
}

```

Após a execução do código acima, teremos como resultado a gravação na console:

```

conexao ao HSQLDB feita com SUCESSO !
Linhas removidas: 8
DELETE executado com sucesso...

```