import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class MySQLDemoWrongNotACID {

/\*\*

\* @param args

\* @throws ClassNotFoundException

\* @throws SQLException

\* @throws ClassNotFoundException

\*/

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

ClassNotFoundException {

// Load the MySQL driver

Class.forName("com.mysql.jdbc.Driver");

// Connect to the database

Connection conn = DriverManager

.getConnection("jdbc:mysql://localhost:3306/test");

// For atomicity

conn.setAutoCommit(false);

// For isolation

conn.setTransactionIsolation(Connection.TRANSACTION\_SERIALIZABLE);

Statement stmt =null;

try{

stmt = conn.createStatement();

String sql = "CREATE DATABASE STORE\_MANAGEMENT";

stmt.executeUpdate(sql);

sql = "CREATE TABLE PRODUCT " +

"(prod VARCHAR(255) not NULL, " +

" pname VARCHAR(255), " +

" price INTEGER, " +

" PRIMARY KEY ( prod ))";

stmt.executeUpdate(sql);

sql = "CREATE TABLE DEPOT " +

"(dep VARCHAR(255) not NULL, " +

" address VARCHAR(255), " +

" volume INTEGER, " +

" PRIMARY KEY ( dep ))";

stmt.executeUpdate(sql);

sql = "CREATE TABLE STOCK " +

"(prod VARCHAR(255) not NULL, " +

" dep VARCHAR(255) not NULL, " +

" quantity INTEGER, " +

" PRIMARY KEY (prod, dep)"+

" FOREIGN KEY (prod) REFERENCES PRODUCT(prod) ON DELETE CASCADE ON UPDATE CASCADE" +

" FOREIGN KEY (dep) REFERENCES DEPOT(dep) ON DELETE CASCADE ON UPDATE CASCADE)";

stmt.executeUpdate(sql);

sql = "INSERT INTO PRODUCT " +

"VALUES ('p1', 'tape', 2.5)";

stmt.executeUpdate(sql);

sql = "INSERT INTO PRODUCT " +

"VALUES ('p2', 'tv', 250)";

stmt.executeUpdate(sql);

sql = "INSERT INTO PRODUCT " +

"VALUES ('p3', 'ver', 80)";

stmt.executeUpdate(sql);

sql = "INSERT INTO DEPOT " +

"VALUES ('d1', 'New York', 9000)";

stmt.executeUpdate(sql);

sql = "INSERT INTO DEPOT " +

"VALUES ('d2', 'Syracuse', 6000)";

stmt.executeUpdate(sql);

sql = "INSERT INTO DEPOT " +

"VALUES ('d4', 'New York', 2000)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p1', 'd1', 1000)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p1', 'd2', -100)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p1', 'd4', 1200)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p3', 'd1', 3000)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p3', 'd4', 2000)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p2', 'd4', 1500)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p2', 'd1', -400)";

stmt.executeUpdate(sql);

sql = "INSERT INTO STOCK " +

"VALUES ('p2', 'd2', 2000)";

stmt.executeUpdate(sql);

//Group 4 â€“ The depot d1 changes its name to dd1 in Depot and Stock.

sql = "UPDATE DEPOT " +

"SET dep = 'dd1' WHERE dep ='d1' ";

stmt.executeUpdate(sql);

}

catch (SQLException e) {

System.out.println("catch Exception");

// For atomicity

conn.rollback();

stmt.close();

conn.close();

return;

} // main

conn.commit();

stmt.close();

conn.close();

}

}