import java.sql.\*;

public class Base {

// JDBC driver name and database URL

static final String JDBC\_DRIVER = "org.h2.Driver";

static final String DB\_URL = "jdbc:h2:~/test";

// Database credentials

static final String USER = "sa";

static final String PASS = "";

public static void main(String[] args) {

Connection connection = null;

Statement statement = null;

try {

// STEP 1: Register JDBC driver

Class.forName(JDBC\_DRIVER);

//STEP 2: Open a connection

System.out.println("Connecting to database...");

connection = DriverManager.getConnection(DB\_URL, USER, PASS);

//STEP 3: Execute a query

8

System.out.println("Creating table in given database...");

statement = connection.createStatement();

String sql = "CREATE TABLE IF NOT EXISTS Employees(id\_employee

BIGINT AUTO\_INCREMENT, surname VARCHAR(255), name VARCHAR(255),

patronymic VARCHAR(255),position VARCHAR(255),phone VARCHAR(255),address

VARCHAR(255), PRIMARY KEY ( id\_employee ))";

String sql1 = "CREATE TABLE IF NOT EXISTS Clients(id\_client

BIGINT AUTO\_INCREMENT, nameCompany VARCHAR(255), phone

VARCHAR(255), address VARCHAR(255), PRIMARY KEY ( id\_client ))";

String sql2 = "CREATE TABLE IF NOT EXISTS Orders(id\_orders

BIGINT AUTO\_INCREMENT, id\_employee INTEGER, id\_client INTEGER,

datePlacement DATE,dateCompletion DATE,amount INTEGER, PRIMARY KEY (

id\_orders ))";

statement.executeUpdate(sql);

statement.executeUpdate(sql1);

statement.executeUpdate(sql2);

statement.execute("ALTER TABLE Orders ADD FOREIGN KEY

(id\_employee) REFERENCES Employees(id\_employee)");

statement.execute("ALTER TABLE Orders ADD FOREIGN KEY

(id\_client) REFERENCES Clients(id\_client)");

System.out.println("Created tables in given database...");

sql = "INSERT INTO Employees " + "VALUES (1, 'Фамилия1', 'Имя1',

'Отчество1', 'Должность1','123-45-67', 'адрес1')";

statement.executeUpdate(sql);

sql = "INSERT INTO Employees " + "VALUES (2, 'Фамилия2', 'Имя2',

'Отчество2', 'Должность2','123-45-67', 'адрес2')";

statement.executeUpdate(sql);

sql = "INSERT INTO Employees " + "VALUES (3, 'Фамилия3', 'Имя3',

'Отчество3', 'Должность3','123-45-67', 'адрес3')";

statement.executeUpdate(sql);

sql1 = "INSERT INTO Clients " + "VALUES (1, 'Компания1', '123-45-

67', 'адрес1')";

statement.executeUpdate(sql1);

sql1 = "INSERT INTO Clients " + "VALUES (2, 'Компания2', '123-45-

67', 'адрес2')";

statement.executeUpdate(sql1);

sql1 = "INSERT INTO Clients " + "VALUES (3, 'Компания3', '123-45-

67', 'адрес3')";

statement.executeUpdate(sql1);

sql2 = "INSERT INTO Orders " + "VALUES (1, 1, 1, '2021-04-04', '2022-

02-20', 100000)";

statement.executeUpdate(sql2);

sql2 = "INSERT INTO Orders " + "VALUES (2, 1, 2, '2021-03-03', '2022-

01-05', 20000)";

statement.executeUpdate(sql2);

sql2 = "INSERT INTO Orders " + "VALUES (3, 3, 3, '2021-02-02', '2022-

02-02', 30000)";

statement.executeUpdate(sql2);

ResultSet resultSet = statement.executeQuery("SELECT \* FROM

Employees");

9

statement = connection.createStatement();

ResultSet resultSet1 = statement.executeQuery("SELECT \* FROM

Clients");

statement = connection.createStatement();

ResultSet resultSet2 = statement.executeQuery("SELECT \* FROM

Orders");

while (resultSet.next()) {

int id = resultSet.getInt(1);

String surname = resultSet.getString(2);

String name = resultSet.getString(3);

String patronymic = resultSet.getString(4);

String position = resultSet.getString(5);

String phone = resultSet.getString(6);

String address = resultSet.getString(7);

System.out.printf("%d. %s %s %s %s %s %s \n", id, surname, name,

patronymic, position, phone, address);

}

while (resultSet1.next()) {

int id = resultSet1.getInt(1);

String nameCompany = resultSet1.getString(2);

String phone = resultSet1.getString(3);

String address = resultSet1.getString(4);

System.out.printf("%d. %s %s \n", id, nameCompany, phone, address);

}

while (resultSet2.next()) {

int id = resultSet2.getInt(1);

int id1 = resultSet2.getInt(2);

int id2 = resultSet2.getInt(3);

String datePlacement = resultSet2.getString(4);

String dateCompletion = resultSet2.getString(5);

int amount = resultSet2.getInt(6);

System.out.printf("%d. %d %d %s %s %d\n", id, id1, id2,

datePlacement, dateCompletion, amount);

}

// STEP 4: Clean-up environment

resultSet.close();

resultSet1.close();

resultSet2.close();

connection.close();

statement.close();

} catch(

SQLException se)

{

//Handle errors for JDBC

se.printStackTrace();

} catch(

Exception e)

{

//Handle errors for Class.forName

e.printStackTrace();

} finally

{

//finally block used to close resources

try {

if (statement != null) statement.close();

} catch (SQLException se2) {

} // nothing we can do

try {

if (connection != null) connection.close();

} catch (SQLException se) {

se.printStackTrace();

} //end finally try

} //end try

System.out.println("Goodbye!");

}

}