

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
БЕЛАРУСЬ УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ» КАФЕДРА «ИИТ»

Лабораторная работа №9

По дисциплине «Современные платформы программирования»

Выполнил: студент ФЭИС

группы ПО-3

Григорьева В.А.

Проверил:

Крощенко А.А.

Брест, 2021

Вариант №8

Цель: приобрести практические навыки разработки баз данных и начальной интеграции БД с кодом Java с помощью JDBC.

Ход работы:

Задание:

Реализовать базу данных из не менее 5 таблиц на заданную тематику. При реализации продумать типизацию полей и внешние ключи в таблицах. Визуализировать разработанную БД с помощью схемы, на которой отображены все таблицы и связи между ними (пример, схема на рис. 1). На языке Java с использованием JDBC реализовать подключение к БД и выполнить основные типы запросов, продемонстрировать результаты преподавателю и включить тексты составленных запросов в отчет. Основные типы запросов – 1. На выборку/на выборку с упорядочиванием (SELECT); 2. На добавление (INSERT INTO); 3. На удаление (DELETE FROM); 4. На модификацию (UPDATE).
Базу данные можно реализовать в любой СУБД (MySQL, PostgreSQL, SQLite и др.)

8) База данных «Прокат товаров и услуг»

Код программы:

Main.java:

```
public class Main {  
  
    public static void main(String[] args) throws java.io.IOException,  
        java.sql.SQLException {  
  
        Menu menu=new Menu();  
        menu.start();  
    }  
}
```

Menu.java:

```
import javax.xml.crypto.Data;  
import java.util.Scanner;
```

```

public class Menu {
    public void start() throws java.io.IOException, java.sql.SQLException {
        DatabaseController controller=new DatabaseController();
        boolean key=true;
        while(key) {
            System.out.println("1 - вывести все товары и услуги");
            System.out.println("2 - вывести всех клиентов с сортировкой по фамилии");
            System.out.println("3 - вывести все товары и услуги в прокате");
            System.out.println("4 - добавить нового клиента");
            System.out.println("5 - изменить имя продукта");
            System.out.println("6 - удалить клиента");
            System.out.println("7 - выйти");
            Scanner in = new Scanner(System.in);
            int num = in.nextInt();
            switch(num)
            {
                case 1: controller.showProducts(); System.in.read(); break; case 2:
                controller.showClients(); System.in.read(); break; case 3:
                controller.showProductsClients(); System.in.read(); break; case 4:
                controller.addNewClient(); System.in.read(); break; case 5:
                controller.updateNameProduct(); System.in.read(); break; case 6:
                controller.deleteClient(); System.in.read(); break; case 7: key=false;
            }
        }
    }
}

```

DatabaseController.java:

```

import java.sql.*;
import java.util.Scanner;

public class DatabaseController {
    Connection connection;
    Statement state;
    Scanner in;

    public DatabaseController () throws SQLException {
        connection=

        DriverManager.getConnection("jdbc:postgresql://localhost:5432/lab3?user=postgres&password=
        1111&ssl=false");

        state = connection.createStatement();
        in = new Scanner(System.in);
    }

    public void showProducts() throws SQLException
    {
        ResultSet result = state.executeQuery("SELECT * FROM products;");
        System.out.printf("%30s%30s%30s\n", "Название", "Стоимость", "Тип");
    }
}

```

```

while(result.next()) {
    System.out.printf("%30s%30s", result.getString(2), result.getString(3));
    if(Integer.parseInt(result.getString(4))==0)
        System.out.printf("%30s\n", "товар");
    else
        System.out.printf("%30s\n", "услуга");
}
}

```

```

public void showClients() throws SQLException
{
    ResultSet result = state.executeQuery("SELECT * FROM clients ORDER BY surname;");
    System.out.printf("%30s%30s%30s\n", "Имя", "Фамилия", "Телефон"); while(result.next())
    System.out.printf("%30s%30s%30s\n", result.getString(2), result.getString(3),
result.getString(4));
}

```

```

public void showProductsClients() throws SQLException
{
    ResultSet result = state.executeQuery("SELECT * FROM products_clients;");
    System.out.printf("%30s%30s%30s\n", "Фамилия клиента", "Продукт или услуга",
"Количество");
    while(result.next()) {
        Statement state2= connection.createStatement();
        Statement state3= connection.createStatement();
        String client1="null", product1="null";
        ResultSet client = state2.executeQuery("SELECT * FROM clients WHERE
idclients="+result.getString(3)+"");
        if(client.next())
            client1=client.getString(3);

        ResultSet product = state3.executeQuery("SELECT * FROM products WHERE
idproducts="+result.getString(2)+"");
        if(product.next())
            product1=product.getString(2);
        System.out.printf("%30s%30s%30s\n", client1, product1, result.getString(4));
    }
}

```

```

public void addNewClient() throws SQLException
{
    String name, surname, phone;
    System.out.println("Введите имя:");
    name=in.nextLine();
    System.out.println("Введите фамилию:");
    surname=in.nextLine();
    System.out.println("Введите телефон:");
    phone=in.nextLine();
    String query=String.format("INSERT INTO clients (name, surname, phone) VALUES
('%s', '%s', '%s');", name, surname, phone);
    try {
        state.executeUpdate(query);
        System.out.println("Успешно добавлено.");
    }
    catch(SQLIntegrityConstraintViolationException E)

```

```

{
    System.out.println("Произошла ошибка добавления.");
}

}

public void updateNameProduct() throws SQLException
{
    ResultSet result = state.executeQuery("SELECT * FROM products;");
    System.out.printf("%30s%30s\n", "id", "Название");
    while(result.next())
        System.out.printf("%30s%30s\n", result.getString(1), result.getString(2));
    System.out.println("Введите id товара:");
    String id=in.nextLine();
    System.out.println("Введите новое название:");
    String name=in.nextLine();
    String query=String.format("UPDATE products SET name = '%s' WHERE idproducts = '%s';", name, id);
    try
    {
        state.executeUpdate(query);
        System.out.println("Успешно обновлено");
    }
    catch(SQLException E)
    {
        System.out.println("Произошла ошибка обновления.");
    }
}

public void deleteClient() throws SQLException
{
    ResultSet result = state.executeQuery("SELECT * FROM clients;");
    System.out.printf("%30s%30s\n", "id", "Фамилия");
    while(result.next())
        System.out.printf("%30s%30s\n", result.getString(1), result.getString(3));
    System.out.println("Введите id клиента:");
    String id=in.nextLine();
    String query=String.format("DELETE FROM clients WHERE idclients = '%s';", id);
    try
    {
        state.executeUpdate(query);
        System.out.println("Успешно удалено");
    }
    catch(SQLException E)
    {
        System.out.println("Произошла ошибка удаления.");
    }
}
}

```

Результаты работы программы:

- 1 - вывести все товары и услуги
- 2 - вывести всех клиентов с сортировкой по фамилии
- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

1

Название	Стоимость	Тип
Chips	4	товар
Juice	21	товар
Nails	3434	услуга

- 1 - вывести все товары и услуги
- 2 - вывести всех клиентов с сортировкой по фамилии
- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

2

Имя	Фамилия	Телефон
Viktoriya	Grygorieva	+375297777777
Zylfiya	Ibrgimova	+375299999999

- 1 - вывести все товары и услуги
- 2 - вывести всех клиентов с сортировкой по фамилии
- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

3

Фамилия клиента	Продукт или услуга	Количество
Grygorieva	Chips	3213
Grygorieva	Juice	666
Ibrgimova	Chips	67

- 1 - вывести все товары и услуги
- 2 - вывести всех клиентов с сортировкой по фамилии
- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

4

Введите имя:

Ilya

Введите фамилию:

Sobakin

Введите телефон:

+375338769900

- 1 - вывести все товары и услуги
- 2 - вывести всех клиентов с сортировкой по фамилии
- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

2

Имя	Фамилия	Телефон
Viktoriya	Grygorieva	+375297777777
Zylfiya	Ibrgimova	+375299999999
Ilya	Sobakin	+375338769900

- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

5

id	Название
1	Chips
2	Juice
3	Nails

Введите id товара:

3

Введите новое название:

Haier

- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

6

id	Фамилия
2	Ibrgimova
1	Grygorieva
6	Sobakin

Введите id клиента:

6

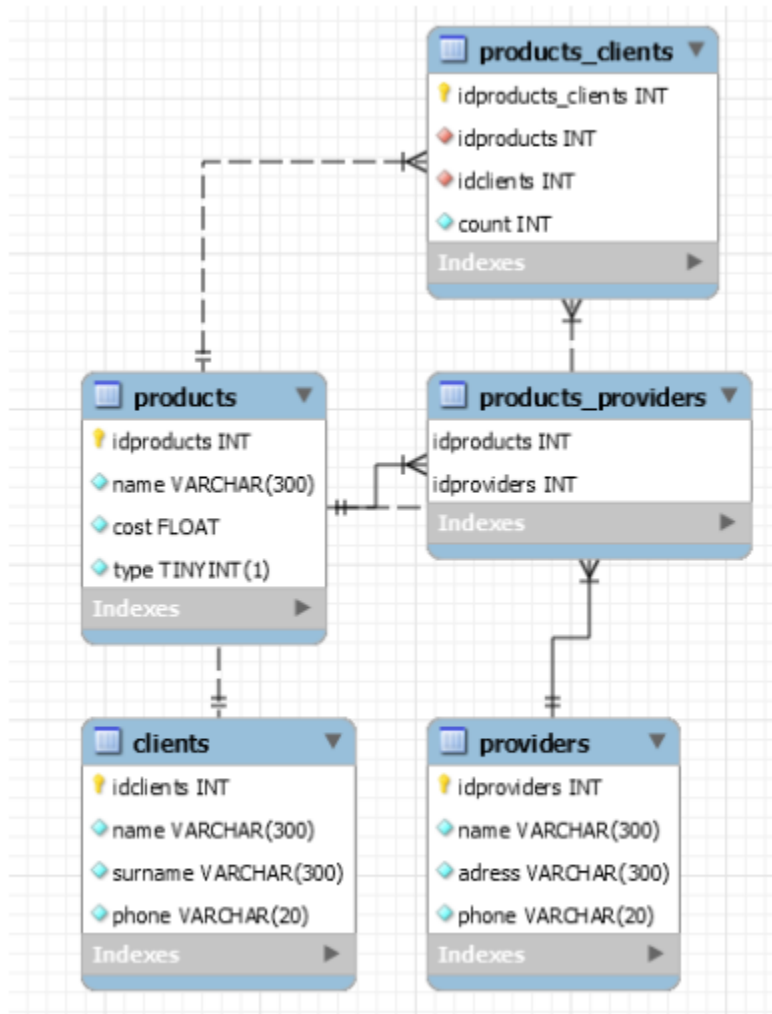
Успешно удалено

- 1 - вывести все товары и услуги
- 2 - вывести всех клиентов с сортировкой по фамилии
- 3 - вывести все товары и услуги в прокате
- 4 - добавить нового клиента
- 5 - изменить имя продукта
- 6 - удалить клиента
- 7 - выйти

2

Имя	Фамилия	Телефон
Viktoriya	Grygorieva	+375297777777
Zylfiya	Ibrgimova	+375299999999

Схема БД:



Выводы:

В результате выполнения работы были приобретены практические навыки разработки баз данных и начальной интеграции БД с кодом Java с помощью JDBC и написана программа по заданной задаче. Положительные результаты работы написанной программы говорят о правильном ее функционировании.