

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра интеллектуальных информационных технологий

Тема: «Разработки баз данных и начальная интеграция БД с кодом Java с помощью JDBC»

Отчёт лабораторной работы №3
по дисциплине «Современные платформы программирования»
за II семестр

Выполнил:
студент 3-его курса
VI-го семестр
факультета ЭИС
группы ПО-4(1)
зачётная книжка №190333
Галанин П. И.
«__» _____ 2022 г.

Проверил:
ассистент
кафедры ИИТ
Монтик Н. С.
«__» _____ 2022 г.

Отчёт лабораторной работы №3

Тема: «Разработки баз данных и начальная интеграция БД с кодом Java с помощью JDBC»

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

Что нужно сделать:

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. 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.)

Вариант 5: База данных «Сборка компьютера».

Проектирование базы данных:

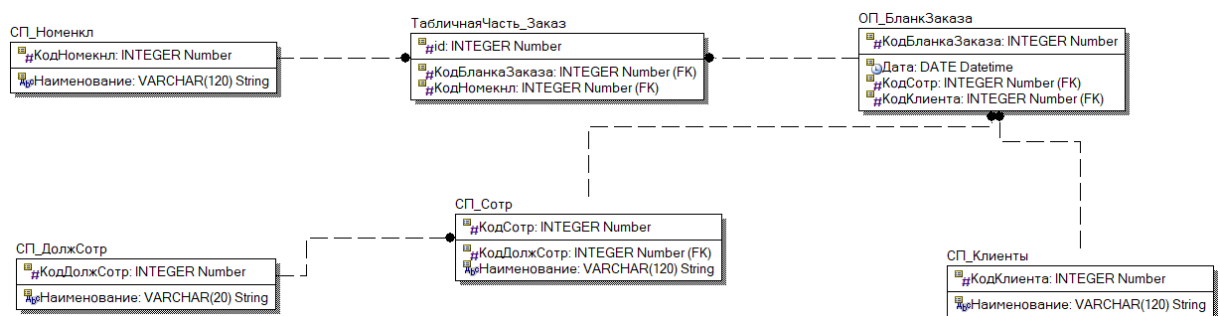


Рисунок 1 – Логическая модель базы данных

Разработка дизайна:

Листинг: Консольное меню

```
1 Menu:
2 1. Create element
3 2. Read table
4 3. Update element
5 4. Delete element
6 7. !!! Create database !!!
7 8. !!! Drop database !!!
8 0. Exit
```

Исходный код:

- а) **Main** - основной класс с логикой программы.
- б) **RB_Nomenclature** - справочник «Номенклатура» с операциями создания таблицы, добавление элемента в таблицу, чтение таблицы, обновление элемента в таблице, удаление элемента в таблице, удаление таблицы.

Листинг: src/com/company/Main.java

```
1 package com.company;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.Statement;
6 import java.util.Scanner;
7
8 public class Main {
9     protected static Connection connection;
10    protected static Statement statement;
11
12    public static void main(String[] args) {
13        try {
14            connection = DriverManager.getConnection("jdbc:sqlite:mydatabase.sqlite");
15            System.out.println("Success connect to database");
16            statement = connection.createStatement();
17        }
18        catch (Exception e) {
19            System.out.println(e);
20            return;
21        }
22        init_database();
23        main_menu();
24    }
25
26    // Функция, которая создает таблицы в базе данных
27    protected static void init_database() {
28        ReferenceBook_Nomenclature.create_table(statement);
29    }
30
31    // Функция, которая вызывает главное меню
32    protected static void main_menu() {
33        String key;
34        Scanner in = new Scanner(System.in);
35        while(true) {
36            print_main_menu();
37            key = in.nextLine();
38
39            if (key.equals("0")) {
40                break;
41            }
42
43            if (key.equals("1")) {
44                ReferenceBook_Nomenclature.create_interface(statement);
45                press_any_key();
46                continue;
47            }
48
49            if (key.equals("2")) {
50                ReferenceBook_Nomenclature.read(statement);
51                press_any_key();
52                continue;
```

```

53         }
54
55         if (key.equals("3")) {
56             ReferenceBook_Nomenclature.update_interface(statement);
57             press_any_key();
58             continue;
59         }
60
61         if (key.equals("4")) {
62             ReferenceBook_Nomenclature.delete_interface(statement);
63             press_any_key();
64             continue;
65         }
66
67         if (key.equals("7")) {
68             ReferenceBook_Nomenclature.create_table(statement);
69             press_any_key();
70             continue;
71         }
72
73         if (key.equals("8")) {
74             ReferenceBook_Nomenclature.drop_table(statement);
75             press_any_key();
76             continue;
77         }
78     }
79 }
80
81 // Функция, которая печатает меню
82 protected static void print_main_menu() {
83     System.out.println("Menu:");
84     System.out.println("1. Create element");
85     System.out.println("2. Read table");
86     System.out.println("3. Update element");
87     System.out.println("4. Delete element");
88     System.out.println("7. !!! Create database!!!");
89     System.out.println("8. !!! Drop database!!!");
90     System.out.println("0. Exit");
91     System.out.println("");
92 }
93
94 // Функция, которая ожидает ввода пользователя
95 protected static void press_any_key() {
96     System.out.print("Press any key... ");
97     Scanner in = new Scanner(System.in);
98     in.nextLine();
99 }
100 }

```

Листинг: src/com/company/ReferenceBook_Nomenclature.java

```

1  package com.company;
2
3  import java.sql.ResultSet;
4  import java.sql.Statement;
5  import java.util.Scanner;
6
7  public class ReferenceBook_Nomenclature {
8      //protected int CodeNomenclature;
9      //protected String Name;
10
11      // Функция, которая создает таблицу
12      protected static void create_table(Statement statement) {
13          final String sql = ""

```

```

14         CREATE TABLE IF NOT EXISTS "RB_Nomenclature" (
15             "CodeNomenclature" integer PRIMARY KEY AUTOINCREMENT UNIQUE,
16             "Name" text
17         );
18         """;
19         try {
20             statement.execute(sql);
21         }
22         catch (Exception e) {
23             System.out.println(e);
24         }
25     }
26
27     // Функция, которая добавляет элемент
28     protected static void create(Statement statement, String Name) {
29         final String sql = "INSERT INTO \"RB_Nomenclature\" (Name) VALUES (\n" + Name + "\n)";
30         try {
31             statement.execute(sql);
32         }
33         catch (Exception e) {
34             System.out.println(e);
35         }
36     }
37
38     // Функция, которая позволяет добавлять элемент через консоль
39     protected static void create_interface(Statement statement) {
40         Scanner in = new Scanner(System.in);
41         System.out.println("\n = = = = =");
42         System.out.println("Create nomenclature element");
43         System.out.print("Name: ");
44         String Name = in.nextLine();
45         create(statement, Name);
46     }
47
48     // Функция, которая выводит таблицу в консоль
49     protected static void read(Statement statement) {
50         final String sql = "SELECT * FROM \"RB_Nomenclature\"";
51         try {
52             ResultSet result_set = statement.executeQuery(sql);
53             System.out.println("\n = = = = =");
54             System.out.println("Table - RB_Nomenclature");
55             System.out.printf("| %-8s | %-64s | \n", "#", "Name");
56             System.out.printf("| %-8s | %-64s | \n",
57                 "-----",
58                 "-----");
59             while(result_set.next()) {
60                 int CodeNomenclature = result_set.getInt(1);
61                 String Name = result_set.getString(2);
62                 System.out.printf("| %-8d | %-64s | \n", CodeNomenclature, Name);
63             }
64         }
65         catch (Exception e) {
66             System.out.println(e);
67         }
68     }
69
70     // Функция, которая обновляет запись по id
71     protected static void update(Statement statement, String id, String Name) {
72         final String sql = "UPDATE \"RB_Nomenclature\" SET Name = \n"
73             + Name + "\n WHERE CodeNomenclature = " + id + ";";
74         try {
75             statement.execute(sql);
76         }

```

```

77         catch (Exception e) {
78             System.out.println(e);
79         }
80     }
81
82     // Функция, которая обновляет запись через консоль
83     protected static void update_interface(Statement statement) {
84         Scanner in = new Scanner(System.in);
85         System.out.println("\n = = = = =");
86         System.out.println("Update nomenclature element");
87
88         System.out.print("id: ");
89         String id = in.nextLine();
90
91         System.out.print("Name: ");
92         String Name = in.nextLine();
93
94         update(statement, id, Name);
95     }
96
97     // Функция, которая удаляет запись по id
98     protected static void delete_(Statement statement, String id) {
99         final String sql = "DELETE FROM \"RB_Nomenclature\" WHERE CodeNomenclature = " + id + ";";
100         try {
101             statement.execute(sql);
102         }
103         catch (Exception e) {
104             System.out.println(e);
105         }
106     }
107
108     // Функция, которая удаляет запись через консоль
109     protected static void delete_interface(Statement statement) {
110         Scanner in = new Scanner(System.in);
111         System.out.println("\n = = = = =");
112         System.out.println("Delete nomenclature element");
113
114         System.out.print("id: ");
115         String id = in.nextLine();
116
117         delete_(statement, id);
118     }
119
120     // Функция, которая удаляет таблицу
121     protected static void drop_table(Statement statement) {
122         final String sql = "DROP TABLE \"RB_Nomenclature\";";
123         try {
124             statement.execute(sql);
125         }
126         catch (Exception e) {
127             System.out.println(e);
128         }
129     }
130 }

```

Листинг: Вывод в консоль

```

1 Success connect to database
2 Menu:
3 1. Create element
4 2. Read table
5 3. Update element
6 4. Delete element
7 7. !!! Create database !!!

```

```

8 8. !!! Drop database!!!
9 0. Exit
10
11 2
12
13 = = = = =
14 Table - RB_Nomenclature
15 | #          | Name
16 | ----- | -----
17 Press any key...
18 Menu:
19 1. Create element
20 2. Read table
21 3. Update element
22 4. Delete element
23 7. !!! Create database!!!
24 8. !!! Drop database!!!
25 0. Exit
26
27 1
28
29 = = = = =
30 Create nomenclature element
31 Name: red mouse
32 Press any key...
33 Menu:
34 1. Create element
35 2. Read table
36 3. Update element
37 4. Delete element
38 7. !!! Create database!!!
39 8. !!! Drop database!!!
40 0. Exit
41
42 1
43
44 = = = = =
45 Create nomenclature element
46 Name: grey mouse
47 Press any key...
48 Menu:
49 1. Create element
50 2. Read table
51 3. Update element
52 4. Delete element
53 7. !!! Create database!!!
54 8. !!! Drop database!!!
55 0. Exit
56
57 2
58
59 = = = = =
60 Table - RB_Nomenclature
61 | #          | Name
62 | ----- | -----
63 |          1 | red mouse
64 |          2 | grey mouse
65 Press any key...
66 Menu:
67 1. Create element
68 2. Read table
69 3. Update element
70 4. Delete element

```

```

71 7. !!! Create database!!!
72 8. !!! Drop database!!!
73 0. Exit
74
75 3
76
77 = = = = =
78 Update nomenclature element
79 id: 1
80 Name: black mouse
81 Press any key...
82 Menu:
83 1. Create element
84 2. Read table
85 3. Update element
86 4. Delete element
87 7. !!! Create database!!!
88 8. !!! Drop database!!!
89 0. Exit
90
91 2
92
93 = = = = =
94 Table - RB_Nomenclature
95 | # | Name |
96 | ----- | ----- |
97 | 1 | black mouse |
98 | 2 | grey mouse |
99 Press any key...
100 Menu:
101 1. Create element
102 2. Read table
103 3. Update element
104 4. Delete element
105 7. !!! Create database!!!
106 8. !!! Drop database!!!
107 0. Exit
108
109 4
110
111 = = = = =
112 Delete nomenclature element
113 id: 1
114 Press any key...
115 Menu:
116 1. Create element
117 2. Read table
118 3. Update element
119 4. Delete element
120 7. !!! Create database!!!
121 8. !!! Drop database!!!
122 0. Exit
123
124 2
125
126 = = = = =
127 Table - RB_Nomenclature
128 | # | Name |
129 | ----- | ----- |
130 | 2 | grey mouse |
131 Press any key...
132 Menu:
133 1. Create element

```



```
134 2. Read table
135 3. Update element
136 4. Delete element
137 7. !!! Create database!!!
138 8. !!! Drop database!!!
139 0. Exit
140
141 0
142
143 Process finished with exit code 0
```

Вывод:

Спроектировали базу данных из 6 таблиц.

Подключились к базе данных SQLite через код java, используя DriverManager.

Реализовали операции создания и удаления таблицы.

Реализовали операции CRUD: c - create, r - read, u - update, d - delete. Создания, чтение, обновления, удаления.

Список использованных источников

1. IntelliJ database connection error: java.sql.SQLException: No suitable driver found. SOLVED - YouTube [Электронный ресурс] - Режим доступа: <https://www.youtube.com/watch?v=lks1NZCnvL8>. Дата доступа: 02.05.2022.
2. Releases · xerial/sqlite-jdbc [Электронный ресурс] - Режим доступа: <https://github.com/xerial/sqlite-jdbc/releases>. Дата доступа: 02.05.2022.
3. JDBC CRUD Example Tutorial [Электронный ресурс] - Режим доступа: <https://www.javaguides.net/2018/10/jdbc-crud-example-tutorial.html>. Дата доступа: 03.05.2022.
4. java - Проверить наличие таблицы в БД и если она не существует создать. JDBC - Stack Overflow на русском [Электронный ресурс] - Режим доступа: <https://ru.stackoverflow.com/questions/623653/%D0%9F%D1%80%D0%BE%D0%B2%D0%B5%D1%80%D0%B8%D1%82%D1%8C-%D0%BD%D0%B0%D0%BB%D0%B8%D1%87%D0%B8%D0%B5-%D1%82%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D1%8B-%D0%B2-%D0%91%D0%94-%D0%B8-%D0%B5%D1%81%D0%BB%D0%B8-%D0%BE%D0%BD%D0%B0-%D0%BD%D0%B5-%D1%81%D1%83%D1%89%D0%B5%D1%81%D1%82%D0%B2%D1%83%D0%B5%D1%82-%D1%81%D0%BE%D0%B7%D0%B4%D0%B0%D1%82%D1%8C-jdbc>. Дата доступа: 03.05.2022.
5. SQL INSERT INTO Statement [Электронный ресурс] - Режим доступа: https://www.w3schools.com/sql/sql_insert.asp. Дата доступа: 03.05.2022.
6. SQL SELECT Statement [Электронный ресурс] - Режим доступа: https://www.w3schools.com/sql/sql_select.asp. Дата доступа: 03.05.2022.
7. SQL UPDATE Statement [Электронный ресурс] - Режим доступа: https://www.w3schools.com/sql/sql_update.asp. Дата доступа: 03.05.2022.
8. SQL DELETE Statement [Электронный ресурс] - Режим доступа: https://www.w3schools.com/sql/sql_delete.asp. Дата доступа: 03.05.2022.

9. SQL DROP TABLE Statement [Электронный ресурс] - Режим доступа: https://www.w3schools.com/sql/sql_drop_table.asp. Дата доступа: 03.05.2022.
10. Как создать исполняемый jar файл в IntelliJ IDEA - YouTube [Электронный ресурс] - Режим доступа: https://www.youtube.com/watch?v=tA8rEz_xFrQ. Дата доступа: 01.05.2022.
11. Export JavaFX 11, 15 or 17 projects into an executable jar file with IntelliJ [2022] - YouTube [Электронный ресурс] - Режим доступа: <https://www.youtube.com/watch?v=F8ahBtXkQzU>. Дата доступа: 01.05.2022.