**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Базы данных»**

Тема: **Реализация базы данных в СУБД MySQL**

|  |  |  |
| --- | --- | --- |
| Студент гр. 0303 |  | Калмак Д.А. |
| Преподаватель |  | Иванов Д.В. |

Санкт-Петербург

2022

## **Цель работы.**

Необходимо развернуть MySQL локально, создать базу данных, написать запросы для создания таблиц, заполнить тестовыми данными, написать запросы к БД.

## **Задание.**

**Вариант 9**

Пусть требуется создать программную систему, предназначенную для диспетчера автобусного парка. Такая система должна обеспечивать хранение сведений о водителях, о маршрутах и характеристиках автобусов. Каждый водитель характеризуется паспортными данными, классом, стажем работы и окладом, причем оклад зависит от класса и стажа работы. Маршрут автобуса характеризуется номером маршрута, названием начального и конечного пункта движения, временем начала и конца движения, интервалом движения и протяженностью в минутах (время движения от кольца до кольца). Характеристиками автобуса являются: номер государственной регистрации автобуса, его тип и вместимость, причем вместимость автобуса зависит от его типа. Каждый водитель закреплен за отдельным автобусом и работает на определенном маршруте, но в случае поломки своего автобуса или болезни другого водителя может пересесть на другую машину. В базе должен храниться график работы водителей. Необходимо предусмотреть возможность корректировки БД в случаях поступления на работу нового водителя, списания старого автобуса, введения нового маршрута или изменения старого и т.п.

* Список водителей, работающих на определенном маршруте с указанием графика их работы?
* Какие автобусы обслуживают данный маршрут?
* Какие маршруты начинаются или заканчиваются в пункте с заданным названием?
* Когда начинается и заканчивается движение автобусов на каждом маршруте?
* Какова протяженность определенного маршрута?
* Какова общая протяженность маршрутов, обслуживаемых автопарком?
* Какие автобусы не вышли на линию, и по какой причине (неисправность, отсутствие водителя)?

Необходимо развернуть MySQL локально:

* Написать запросы для создания таблиц из предыдущей лабораторный работы
* Заполнить тестовыми данными: 5-10 строк на каждую таблицу, обязательно наличие связи между ними, данные приближены к реальности.
* Написать запросы к БД, отвечающие на вопросы из предыдущей лабораторной работы
* Исходный код выложить на www.db-fiddle.com для проверки работоспособности
* Исходный код в виде .sql файла запушить в виде PR в репо

## **Выполнение работы.**

Основой для создания базы данных является диаграмма базы данных. (см. рис. 1)

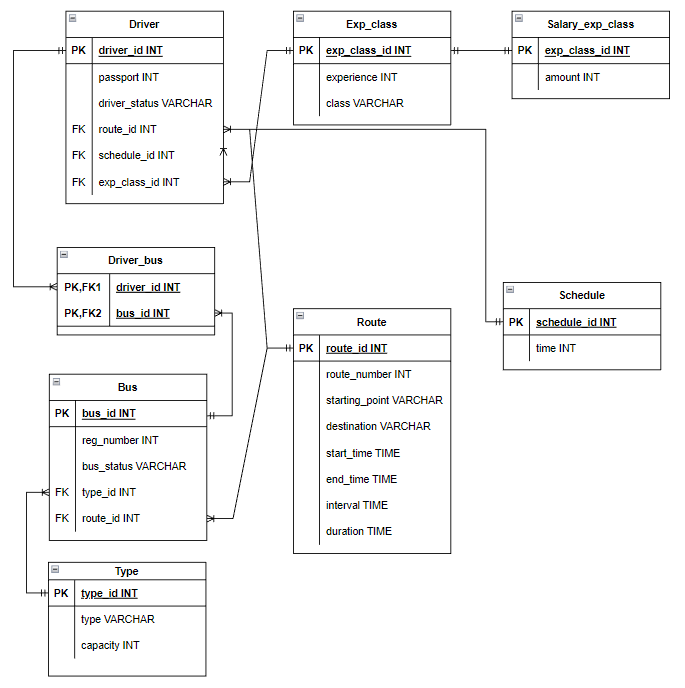


Рисунок 1 – Диаграмма базы данных

Запросы для создания таблиц и заполнения их данными представлены в файле schema.sql. Результаты выполнения запросов представлены на рис. 2-9.

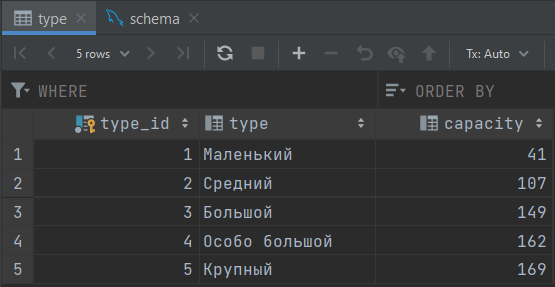


Рисунок 2 – Таблица типов автобусов

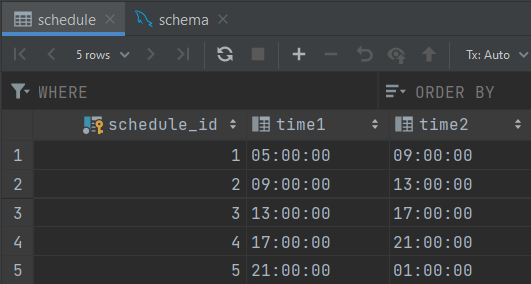


Рисунок 3 – Таблица графика работы

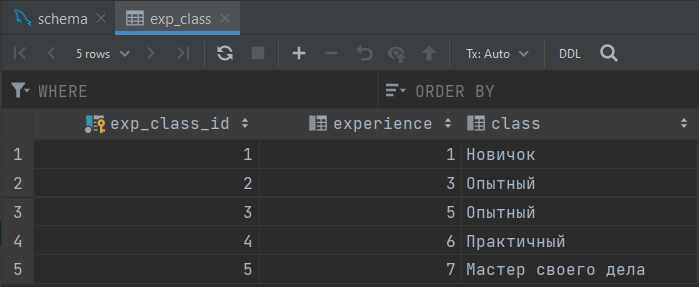


Рисунок 4 – Таблица стажа и класса

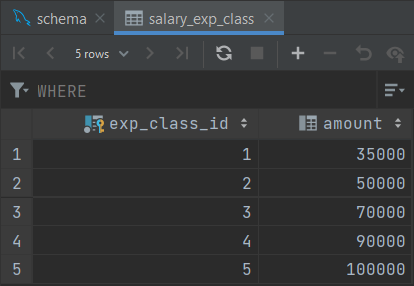


Рисунок 5 – Таблица окладов

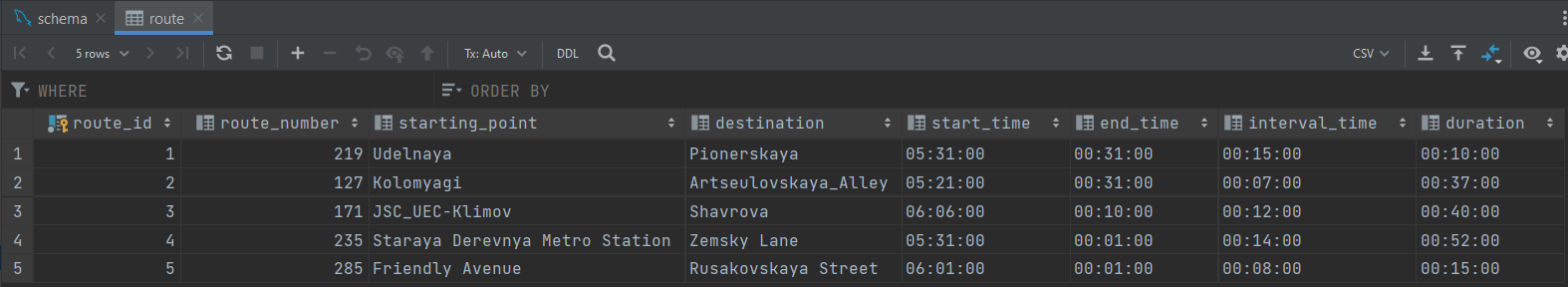


Рисунок 6 – Таблица маршрутов

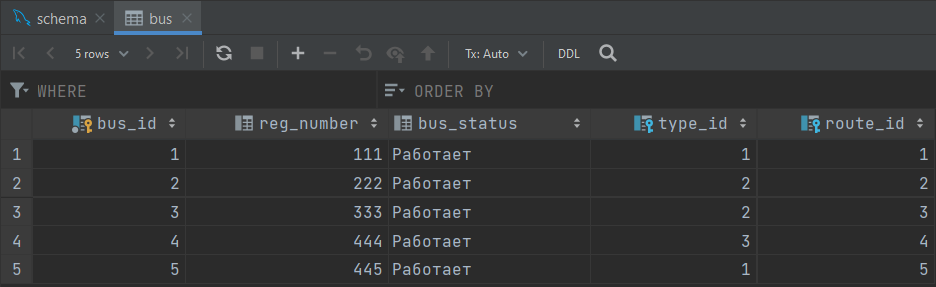


Рисунок 7 – Таблица автобусов

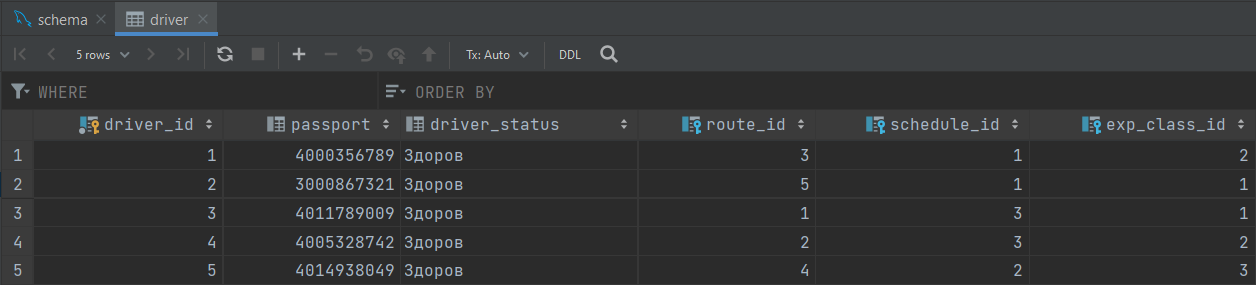


Рисунок 8 – Таблица водителей

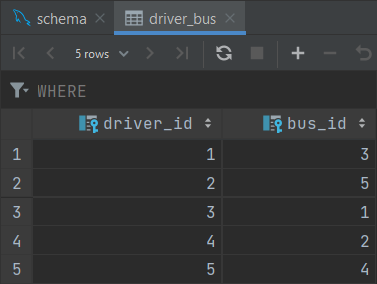


Рисунок 9 – Таблица водителей и автобусов

Запросы к БД, отвечающие на вопросы из задания, представлены в файле query.sql. Результаты выполнения запросов представлены на рис. 10-19.

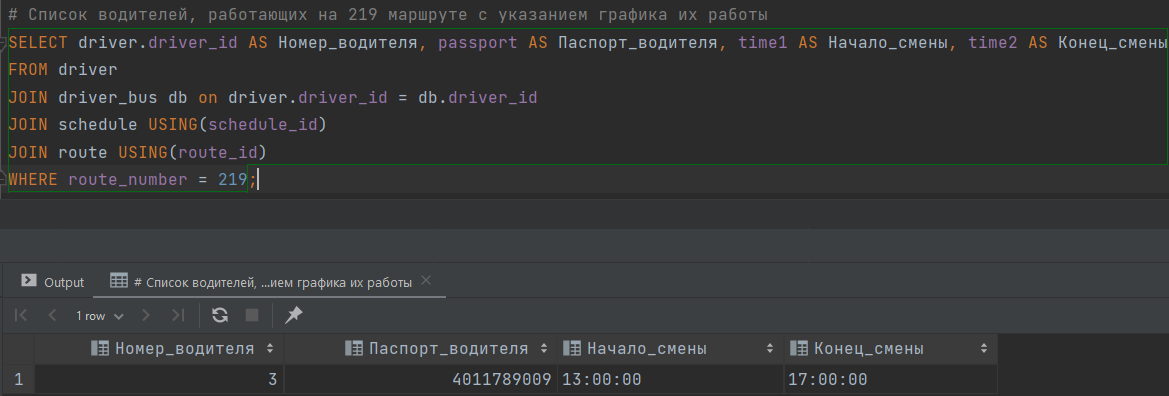


Рисунок 10 – Список водителей, работающих на 219 маршруте с указанием графика их работы

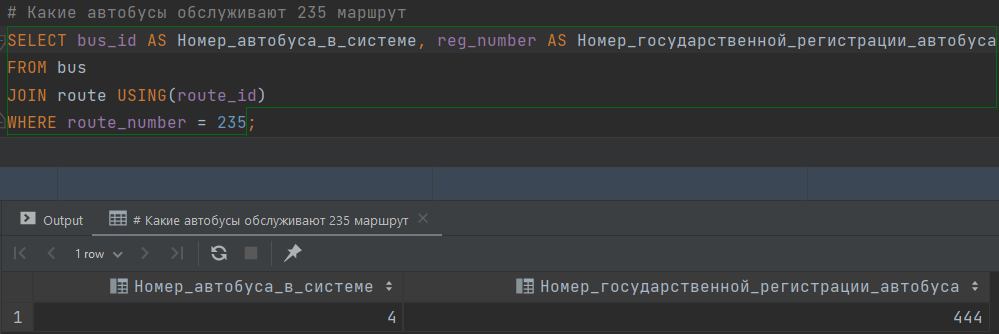


Рисунок 11 – Какие автобусы обслуживают 235 маршрут

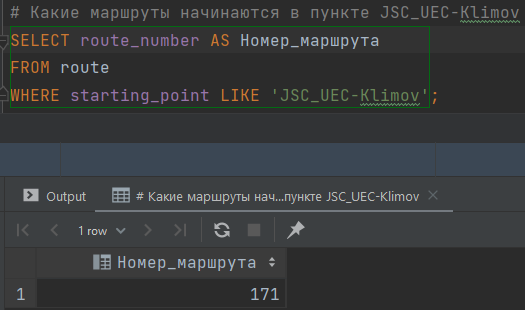


Рисунок 12 – Какие маршруты начинаются в пункте JSC\_UEC-Klimov

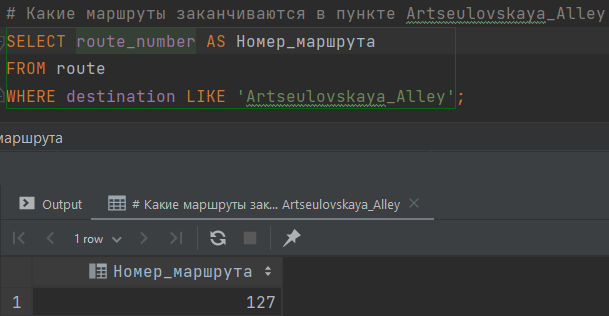


Рисунок 13 – Какие маршруты заканчиваются в пункте Artseulovskaya\_Alley

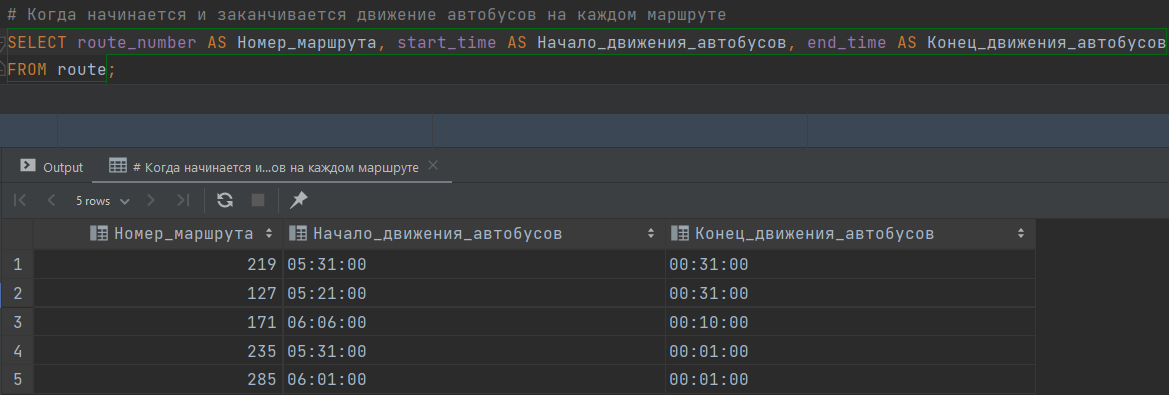


Рисунок 14 – Когда начинается и заканчивается движение автобусов на каждом маршруте

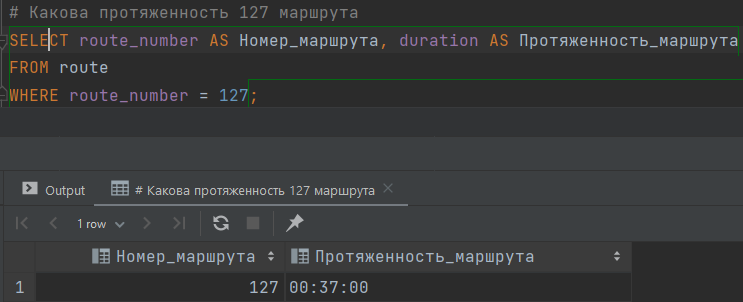


Рисунок 15 – Какова протяженность 127 маршрута

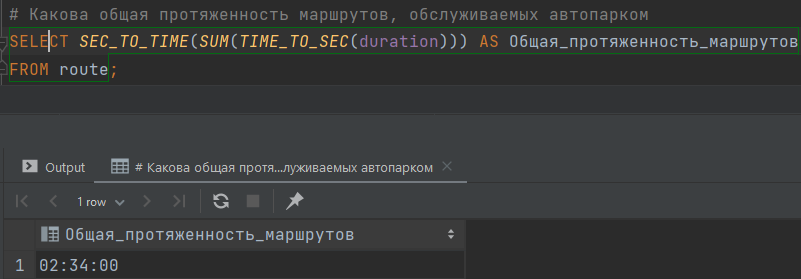


Рисунок 16 – Какова общая протяженность маршрутов, обслуживаемых автопарком

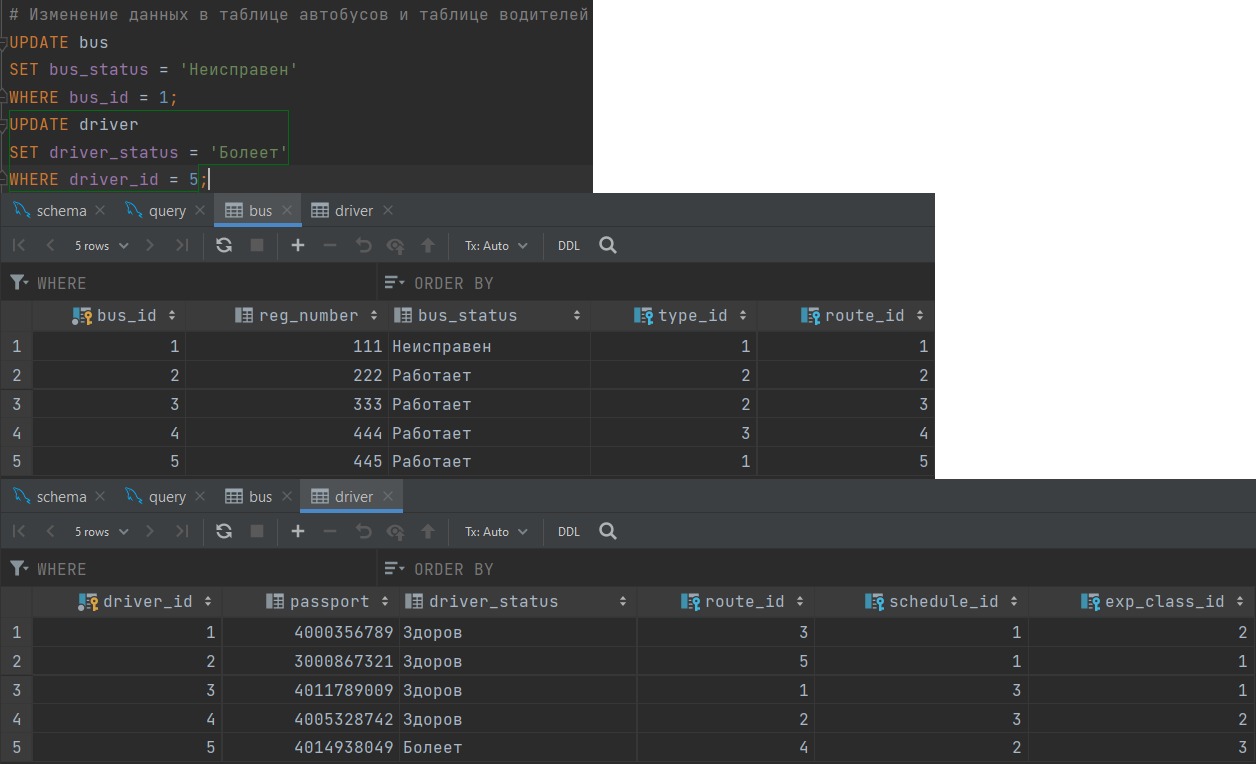


Рисунок 17 – Изменение данных в таблице автобусов и таблице водителей

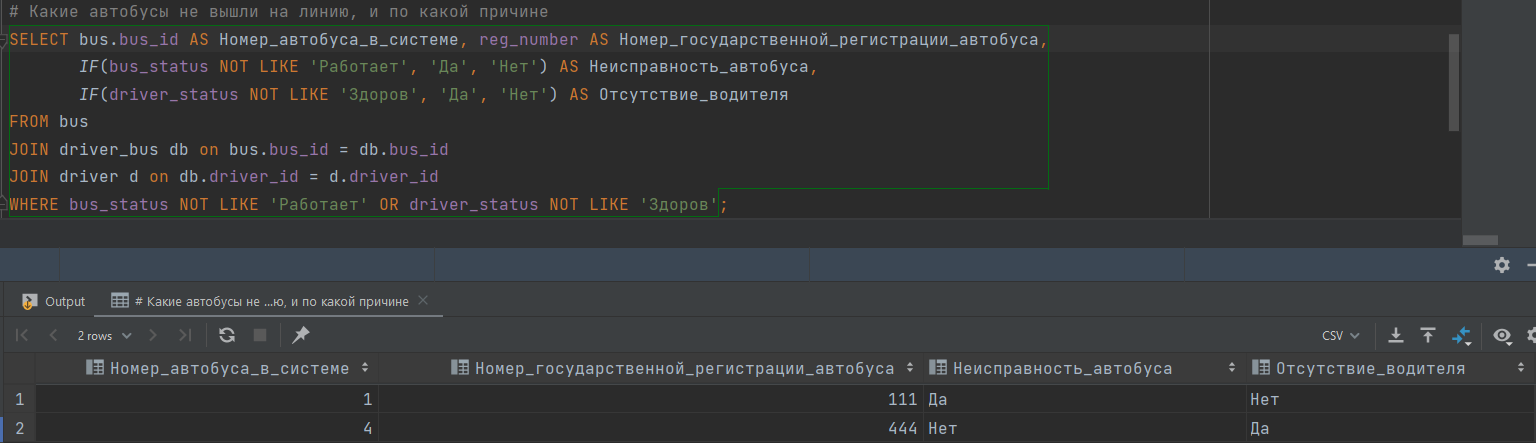


Рисунок 18 – Какие автобусы не вышли на линию, и по какой причине

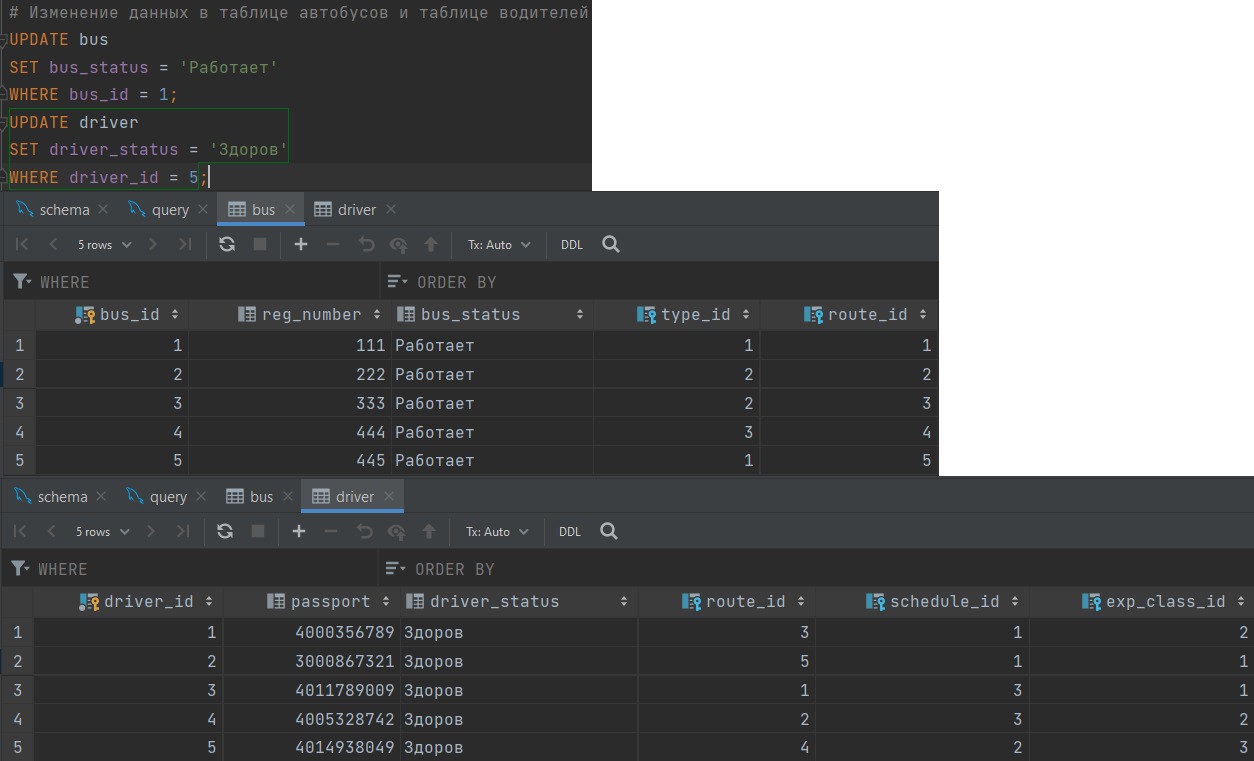


Рисунок 19 – Изменение данных в таблице автобусов и таблице водителей

Ссылка на пулл-реквест, ссылка на db-fiddle, исходный код в виде .sql файлов представлены в Приложении А.

## **Выводы.**

Таким образом, был развернут MySQL локально, создана база данных на основе диаграммы базы данных, а именно написаны запросы для создания таблиц, таблицы заполнены тестовыми данными. Также были написаны запросы к БД, отвечающие на вопросы задания.

# **ПРИЛОЖЕНИЕ А**

**Ссылка на пулл реквест:**

<https://github.com/moevm/sql-2022-0303/pull/32>

**Ссылка на db-fiddle:**

<https://www.db-fiddle.com/f/chi1UXwBKM6u5HtqEut41T/1#&togetherjs=cjEpbCAz80>

**Исходный код**

**Файл schema.sql**

# создание базы данных

CREATE DATABASE bus\_park\_dispatcher;

USE bus\_park\_dispatcher;

# тип автобуса

CREATE TABLE type (

type\_id INT PRIMARY KEY AUTO\_INCREMENT,

type VARCHAR(100),

capacity INT

);

# заполнение таблицы типов автобуса данными

INSERT INTO type(type, capacity)

VALUES ('Маленький', 41),

('Средний', 107),

('Большой', 149),

('Особо большой', 162),

('Крупный', 169);

# график работы

CREATE TABLE schedule (

schedule\_id INT PRIMARY KEY AUTO\_INCREMENT,

time1 TIME,

time2 TIME

);

# заполнение таблицы графика работы данными

INSERT INTO schedule(time1, time2)

VALUES ('05:00:00', '09:00:00'),

('09:00:00', '13:00:00'),

('13:00:00', '17:00:00'),

('17:00:00', '21:00:00'),

('21:00:00', '01:00:00');

# стаж и класс

CREATE TABLE exp\_class (

exp\_class\_id INT PRIMARY KEY AUTO\_INCREMENT,

experience INT,

class VARCHAR(100)

);

# заполнение таблицы стажа и класса данными

INSERT INTO exp\_class(experience, class)

VALUES (1, 'Новичок'),

(3, 'Опытный'),

(5, 'Опытный'),

(6, 'Практичный'),

(7, 'Мастер своего дела');

# оклады

CREATE TABLE salary\_exp\_class (

exp\_class\_id INT PRIMARY KEY AUTO\_INCREMENT,

amount INT,

FOREIGN KEY (exp\_class\_id) REFERENCES exp\_class (exp\_class\_id) ON DELETE CASCADE

);

# заполнение таблицы окладов данными

INSERT INTO salary\_exp\_class(amount)

VALUES (35000),

(50000),

(70000),

(90000),

(100000);

# маршруты

CREATE TABLE route (

route\_id INT PRIMARY KEY AUTO\_INCREMENT,

route\_number INT,

starting\_point VARCHAR(100),

destination VARCHAR(100),

start\_time TIME,

end\_time TIME,

interval\_time TIME,

duration TIME

);

# заполнение таблицы маршрутов данными

INSERT INTO route(route\_number, starting\_point, destination, start\_time, end\_time, interval\_time, duration)

VALUES (219, 'Udelnaya', 'Pionerskaya', '05:31:00', '00:31:00', '00:15:00', '00:10:00'),

(127, 'Kolomyagi', 'Artseulovskaya\_Alley', '05:21:00', '00:31:00', '00:07:00', '00:37:00'),

(171, 'JSC\_UEC-Klimov', 'Shavrova', '06:06:00', '00:10:00', '00:12:00', '00:40:00'),

(235, 'Staraya Derevnya Metro Station', 'Zemsky Lane', '05:31:00', '00:01:00', '00:14:00', '00:52:00'),

(285, 'Friendly Avenue', 'Rusakovskaya Street', '06:01:00', '00:01:00', '00:08:00', '00:15:00');

# автобусы

CREATE TABLE bus (

bus\_id INT PRIMARY KEY AUTO\_INCREMENT,

reg\_number INT,

bus\_status VARCHAR(100),

type\_id INT,

route\_id INT,

FOREIGN KEY (type\_id) REFERENCES type (type\_id) ON DELETE SET NULL,

FOREIGN KEY (route\_id) REFERENCES route (route\_id) ON DELETE SET NULL

);

# заполнение таблицы автобусов данными

INSERT INTO bus(reg\_number, bus\_status, type\_id, route\_id)

VALUES (111, 'Работает', 1, 1),

(222, 'Работает', 2, 2),

(333, 'Работает', 2, 3),

(444, 'Работает', 3, 4),

(445, 'Работает', 1, 5);

# водители

CREATE TABLE driver (

driver\_id INT PRIMARY KEY AUTO\_INCREMENT,

passport INT UNSIGNED,

driver\_status VARCHAR(100),

route\_id INT,

schedule\_id INT,

exp\_class\_id INT,

FOREIGN KEY (route\_id) REFERENCES route (route\_id) ON DELETE SET NULL,

FOREIGN KEY (schedule\_id) REFERENCES schedule (schedule\_id) ON DELETE SET NULL,

FOREIGN KEY (exp\_class\_id) REFERENCES exp\_class (exp\_class\_id) ON DELETE SET NULL

);

# заполнение таблицы водителей данными

INSERT INTO driver(passport, driver\_status, route\_id, schedule\_id, exp\_class\_id)

VALUES (4000356789, 'Здоров', 3, 1, 2),

(3000867321, 'Здоров', 5, 1, 1),

(4011789009, 'Здоров', 1, 3, 1),

(4005328742, 'Здоров', 2, 3, 2),

(4014938049, 'Здоров', 4, 2, 3);

# водители и автобусы

CREATE TABLE driver\_bus (

driver\_id INT,

bus\_id INT,

FOREIGN KEY (driver\_id) REFERENCES driver (driver\_id) ON DELETE SET NULL,

FOREIGN KEY (bus\_id) REFERENCES bus (bus\_id) ON DELETE SET NULL

);

# заполнение таблицы водителей и автобусов данными

INSERT INTO driver\_bus(driver\_id, bus\_id)

VALUES (1, 3),

(2, 5),

(3, 1),

(4, 2),

(5, 4);

**Файл query.sql**

# Список водителей, работающих на 219 маршруте с указанием графика их работы

SELECT driver.driver\_id AS Номер\_водителя, passport AS Паспорт\_водителя, time1 AS Начало\_смены, time2 AS Конец\_смены

FROM driver

JOIN driver\_bus db on driver.driver\_id = db.driver\_id

JOIN schedule USING(schedule\_id)

JOIN route USING(route\_id)

WHERE route\_number = 219;

# Какие автобусы обслуживают 235 маршрут

SELECT bus\_id AS Номер\_автобуса\_в\_системе, reg\_number AS Номер\_государственной\_регистрации\_автобуса

FROM bus

JOIN route USING(route\_id)

WHERE route\_number = 235;

# Какие маршруты начинаются в пункте JSC\_UEC-Klimov

SELECT route\_number AS Номер\_маршрута

FROM route

WHERE starting\_point LIKE 'JSC\_UEC-Klimov';

# Какие маршруты заканчиваются в пункте Artseulovskaya\_Alley

SELECT route\_number AS Номер\_маршрута

FROM route

WHERE destination LIKE 'Artseulovskaya\_Alley';

# Когда начинается и заканчивается движение автобусов на каждом маршруте

SELECT route\_number AS Номер\_маршрута, start\_time AS Начало\_движения\_автобусов, end\_time AS Конец\_движения\_автобусов

FROM route;

# Какова протяженность 127 маршрута

SELECT route\_number AS Номер\_маршрута, duration AS Протяженность\_маршрута

FROM route

WHERE route\_number = 127;

# Какова общая протяженность маршрутов, обслуживаемых автопарком

SELECT SEC\_TO\_TIME(SUM(TIME\_TO\_SEC(duration))) AS Общая\_протяженность\_маршрутов

FROM route;

# Изменение данных в таблице автобусов и таблице водителей

UPDATE bus

SET bus\_status = 'Неисправен'

WHERE bus\_id = 1;

UPDATE driver

SET driver\_status = 'Болеет'

WHERE driver\_id = 5;

# Какие автобусы не вышли на линию, и по какой причине

SELECT bus.bus\_id AS Номер\_автобуса\_в\_системе, reg\_number AS Номер\_государственной\_регистрации\_автобуса,

IF(bus\_status NOT LIKE 'Работает', 'Да', 'Нет') AS Неисправность\_автобуса,

IF(driver\_status NOT LIKE 'Здоров', 'Да', 'Нет') AS Отсутствие\_водителя

FROM bus

JOIN driver\_bus db on bus.bus\_id = db.bus\_id

JOIN driver d on db.driver\_id = d.driver\_id

WHERE bus\_status NOT LIKE 'Работает' OR driver\_status NOT LIKE 'Здоров';

# Изменение данных в таблице автобусов и таблице водителей

UPDATE bus

SET bus\_status = 'Работает'

WHERE bus\_id = 1;

UPDATE driver

SET driver\_status = 'Здоров'

WHERE driver\_id = 5;