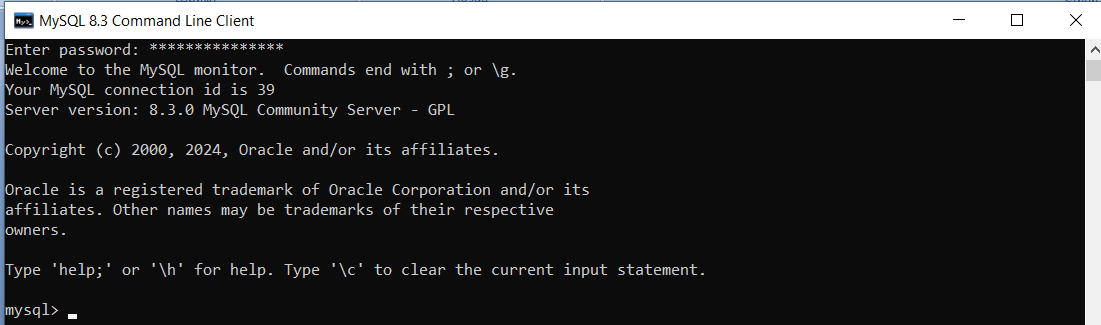
Практична №5. Jdbc

Розпочала виконання даної лабораторної роботи з завантаження MySQL. Спочатку я мала намір створювати таблицю завдяки MySQL command line client  


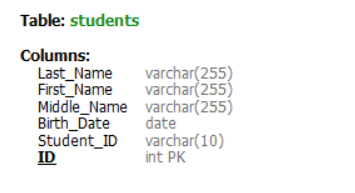
Але потім перейшла на MySQL Workbench.

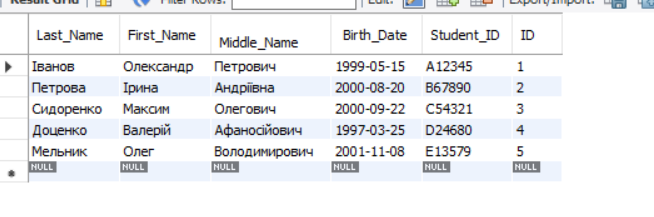
Розпочала з підключення сервера баз даних. Ввела дані для підключення до мого сервера MySQL, такі як ім'я хоста, порт, ім'я користувача (Kate) та пароль(ketkatarina2002). Після введення цих даних, натиснула кнопку "Test Connection", щоб переконатися, що підключення встановлене правильно. Усе пройшло успішно, з’єднання встановлено!

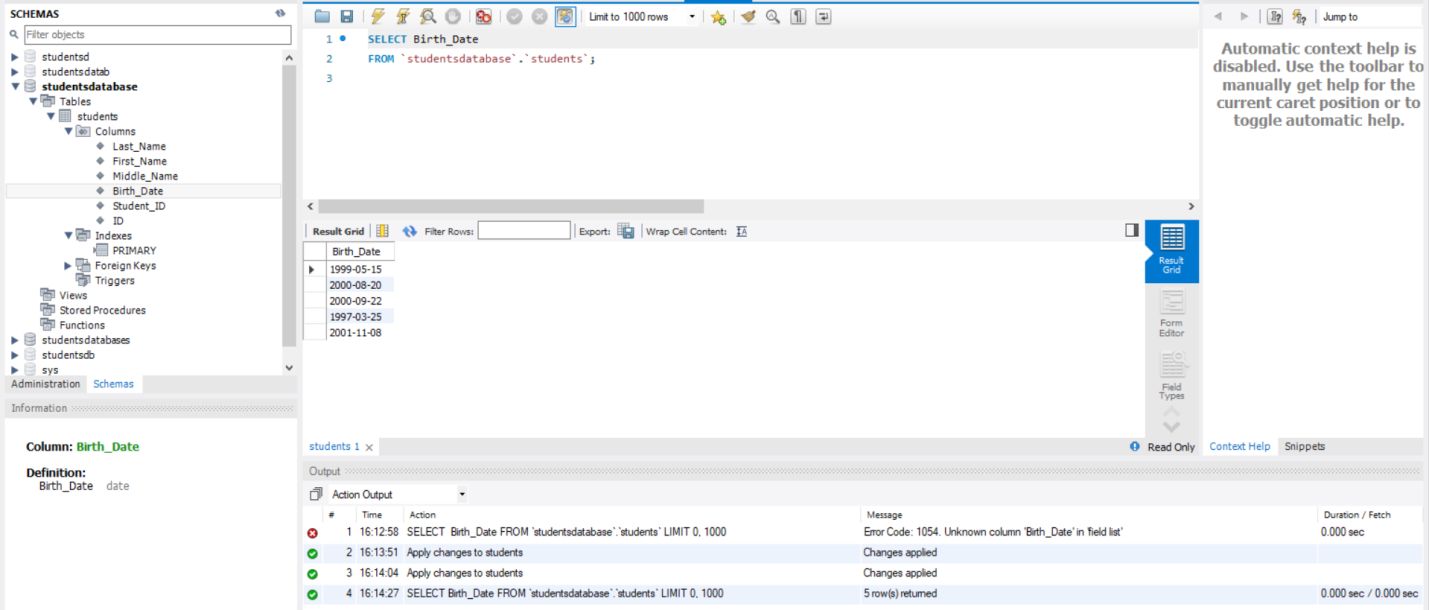
**Далі я створила нову базу даних, на** вкладці "Navigator". Так я створила нову схему. Дала назву «studentsdatabase» та натиснула Apply.  




Після стоврення схеми, я розпочала роботу над базою даних. Створила стопчики з наступними даними : прізвище, ім'я, по батькові, день народження номер залікової книжки та ID.



Після розпочала вносити дані до таблиці, заповнювати її, та отримала дану таблицю  


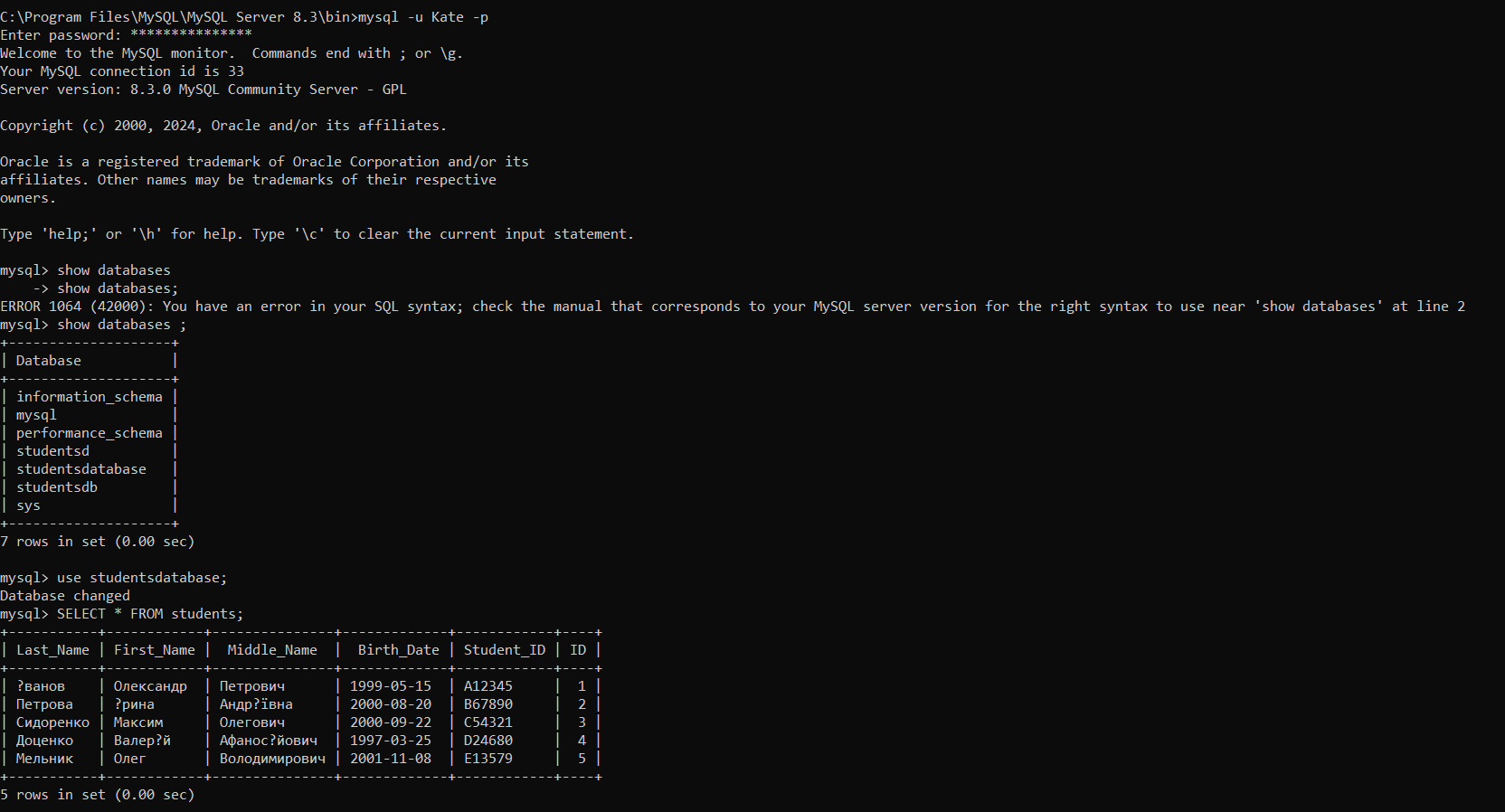
Потім перевірила чи буде працювати таблиця за критерієм «Дата народження»  


Отримала позитивний результат. База даних працює!

Після цього у пошуковому рядку комп’ютера ввела cmd та перевірила чи існує моя таблиця даних у MySQl. Для цього я ввела наступну команду   
mysql –u Kate –p

Password: ketkatarina2002

Обрала свою базу даних та отримала такий результат :



Що повністю відповідало моїм очікуванням.

Для того щоб підключити базу даних до коду мені потрібні JDBC драйвер. Я завантажила їх на офіційному сайті MySQL. Та розмістила у своїй папці з LR\_5.

<https://dev.mysql.com/downloads/connector/j/>

Далі розпочалась робота з кодом. Для початку я створила код який забезпечить зв'язок з базою даних та виконає запити користувача.

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class StudentDatabaseApp {

    // URL адреса бази даних

    private static final String DB\_URL = "jdbc:mysql://localhost/studentsdatabase";

    // Ім'я користувача бази даних

    private static final String USERNAME = "Kate";

    // Пароль для доступу до бази даних

    private static final String PASSWORD = "ketkatarina2002";

    public static void main(String[] args) {

        Connection connection = null;

        try {

            // Завантажуємо драйвер для MySQL

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

            // Встановлюємо з'єднання з базою даних

            connection = DriverManager.getConnection(DB\_URL, USERNAME, PASSWORD);

            // Якщо з'єднання успішно встановлено, виводимо повідомлення

            System.out.println("З'єднання з базою даних успішно!");

            // Виводимо інформацію про студентів, які народилися у січні

            printStudentsBornInMonth(connection, 1); // 1 - січень

        } catch (SQLException | ClassNotFoundException e) {

            // Виводимо помилку, якщо вона виникла

            e.printStackTrace();

        } finally {

            // Закриваємо з'єднання

            if (connection != null) {

                try {

                    connection.close();

                } catch (SQLException e) {

                    e.printStackTrace();

                }

            }

        }

    }

    // Метод для виведення інформації про студентів, які народилися в певному місяці

    private static void printStudentsBornInMonth(Connection connection, int month) throws SQLException {

        // SQL запит для отримання даних про студентів, які народилися у вказаному місяці

        String sql = "SELECT \* FROM students WHERE MONTH(date\_of\_birth) = ?";

        // Підготовка SQL запиту

        try (PreparedStatement statement = connection.prepareStatement(sql)) {

            statement.setInt(1, month);

            // Виконання запиту і отримання результатів

            try (ResultSet resultSet = statement.executeQuery()) {

                // Виведення результатів

                System.out.println("Студенти, які народилися у місяці " + month + ":");

                while (resultSet.next()) {

                    String lastName = resultSet.getString("last\_name");

                    String firstName = resultSet.getString("first\_name");

                    String middleName = resultSet.getString("middle\_name");

                    String dateOfBirth = resultSet.getString("date\_of\_birth");

                    String studentId = resultSet.getString("student\_id");

                    String studentDatabaseId = resultSet.getString("student\_database\_id");

                    System.out.println("Прізвище: " + lastName + ", Ім'я: " + firstName + ", По батькові: " + middleName

                            + ", Дата народження: " + dateOfBirth + ", Номер залікової книжки: " + studentId

                            + ", ID: " + studentDatabaseId);

                }

            }

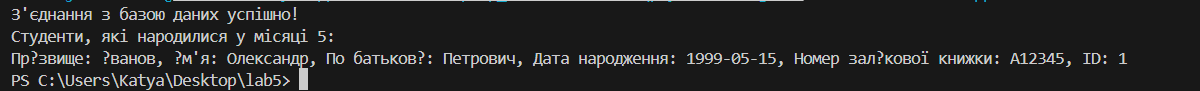
        }

    }

}

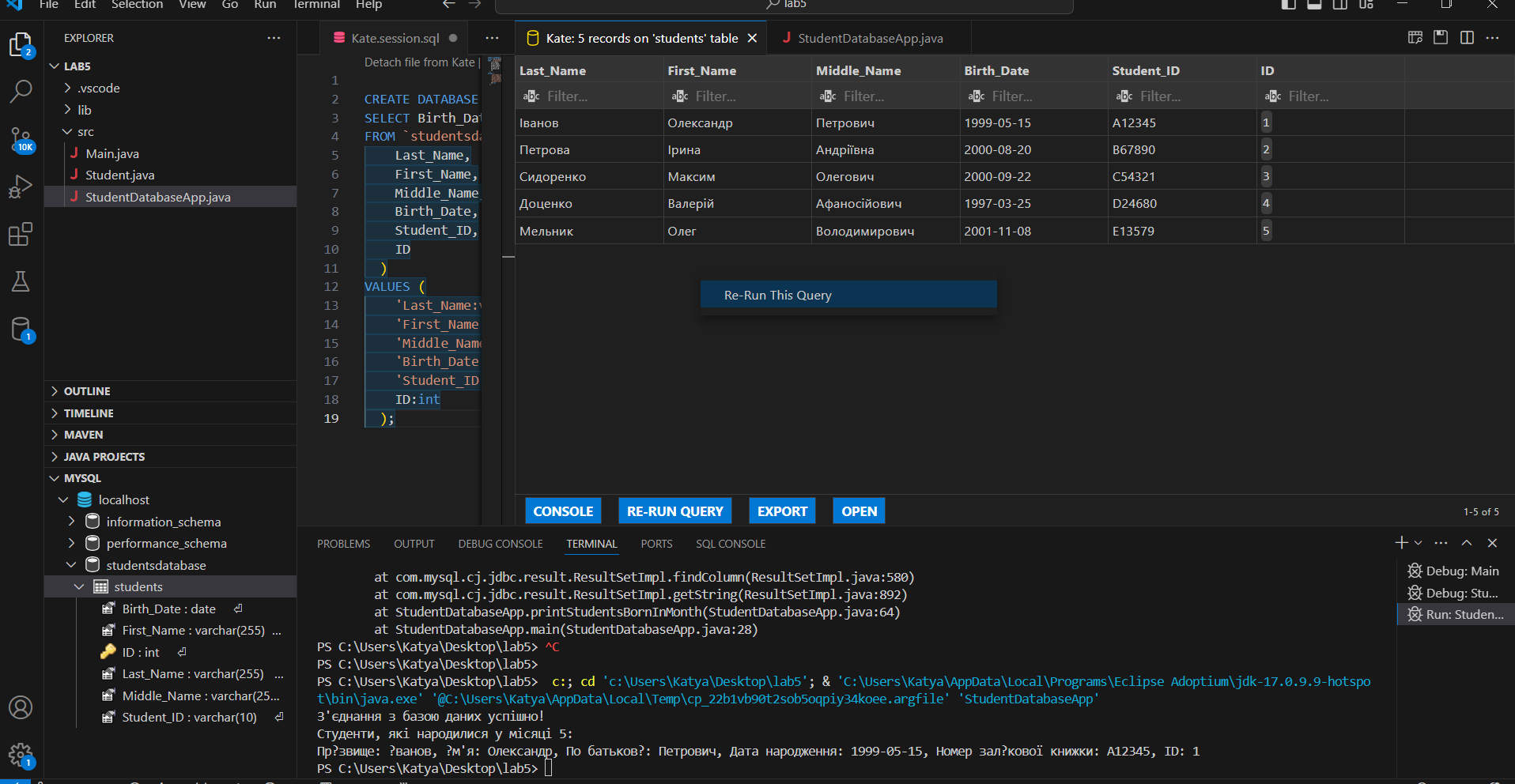
Цей код дозволить користувачеві вводити місяць, після чого виводитимуться всі студенти, народжені в цьому місяці.

Ось результат коду:

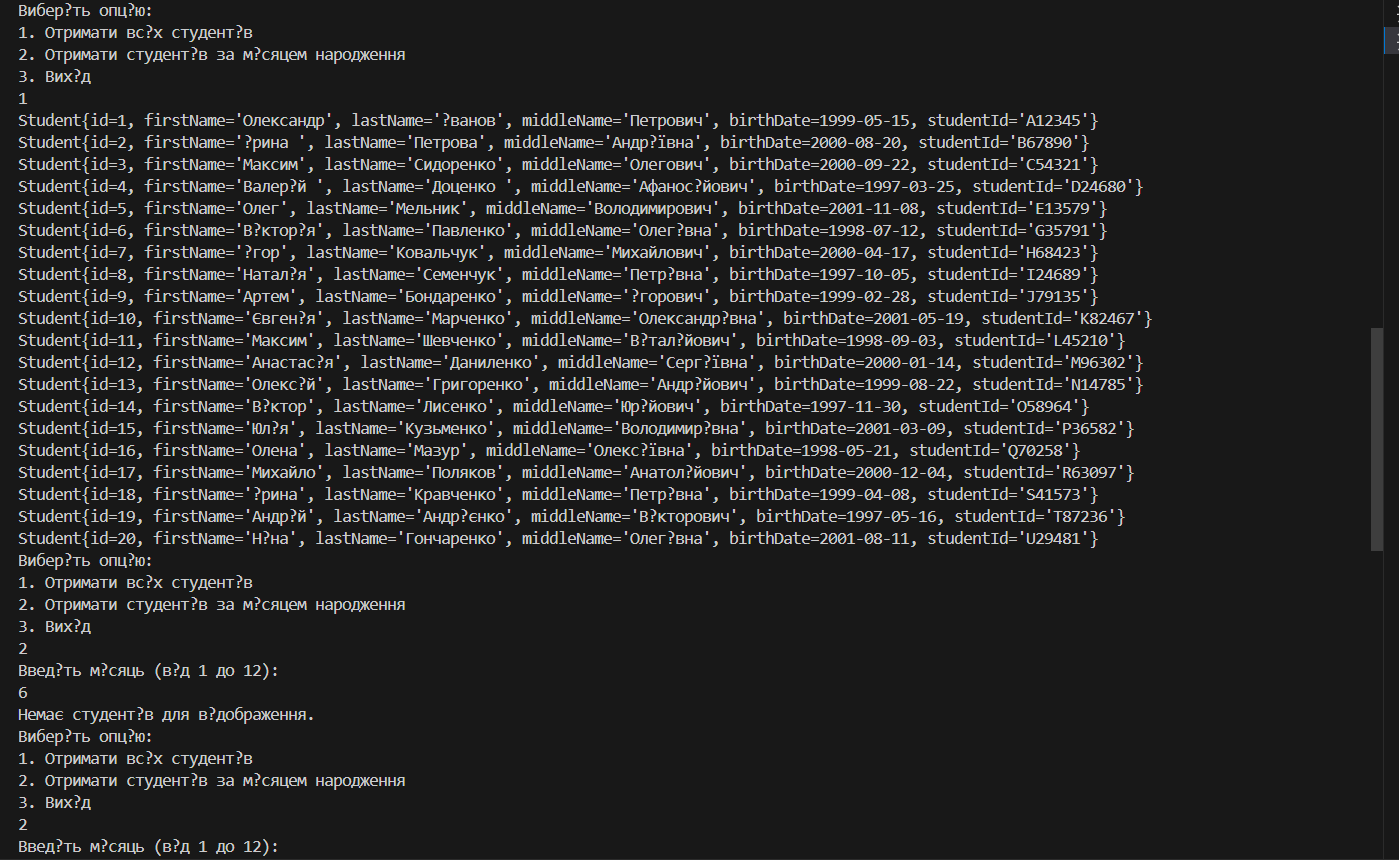
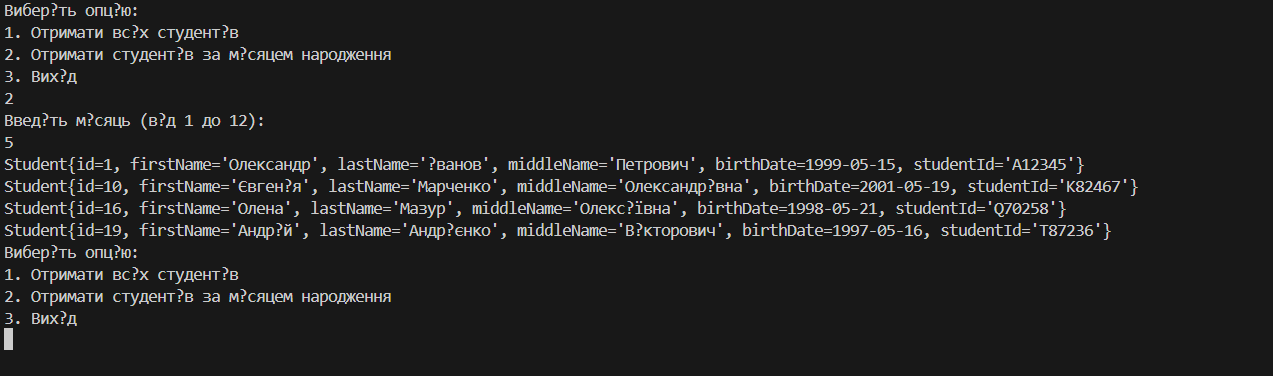


Був обраний 5 місяць, програма вивела мені Іванова Олександра Петровича, його день народження, залік.

Тобто завдання яке стояли перед мною – виконане!

Також я підключила до Visual SC розширення MySQL, де переглядала стан таблиці.  


Висновок:

Оскільки все працює відповідно до вимог, ось контрольний результат роботи:   
   
  
Обрано підхід до проектування програми для управління базою даних студентів, який передбачає формування SQL-запитів на стороні Java-коду, а не на стороні сервера баз даних. Цей підхід дозволяє нам забезпечити більшу гнучкість та контроль над SQL-запитами, а також полегшити відлагодження та підтримку коду. Програма виконує вивід інформації про студентів, які народилися у вказаному місяці року, здійснюючи підключення до бази даних та виконання SQL-запитів за допомогою JDBC.