МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Изображение выглядит как текст

Автоматически созданное описаниеНаціональний технічний університет «Дніпровська політехніка»

Кафедра ПЗКС

ЗВІТ

Лабораторна робота №5

з дисципліни «Поглиблене програмування в середовищі Java»

Виконав: ст. гр. 122-21-1

Карапиш Ілля Володимирович

Перевірили:   
Мінєєв О.С.

Шевченко Ю.О.

Дніпро

2025

**Лабораторна робота № 5**

**Тема: «**JDBC».

**Завдання:** Створити базу даних в будь  якому сервері баз даних. Створити таблицю з переліком студентів вказати їх прізвище, ім'я, по батькові, день народження номер залікової книжки та ID.

Створити програму  що буде дозволяти   виводити на екран  інформацію про студентів  які народилися в тому чи іншому місяці року.  Програма повинна завдяки системі jdbc під'єднатися до вашої бази даних та робити до неї запроси.  Вимог до розробки бази даних немає.  Програма ж має бути написана за усіма стандартами ООП.  Та може бути спроектована за двох принципів:

- при будь-якій ситуації буде забиратися весь перелік студентів, а вже на стороні java буде зроблено пошук  необхідного

 - SQL запрос буде сформований згідно запросу який зробив користувач і вже сервер управління баз даних буде вирішувати, які самі студенти народилися в тому чи іншому місяці.

 У висновку обов'язково пояснити чому вибрали той чи інший принцип, які в нього переваги та недоліки.  Оцінка не залежить від того який сервер управління баз даних вибрали.  Перелік студентів зробити не менше 20 людей. Місяць червень зробити місяцем, коли в жодного зі студентів немає дня народження.

 SQL код створення бази даних розмістити проекті 6 лабораторної роботи  в файлі database в пекеджі resources.  Для використання цієї лабораторної роботи рекомендується активно використовувати знання отримані на дисципліні що стосуються  розробки баз даних.

 До паперового звіту обов'язково додати принтскрин з програми  в якій ви дивитесь інформацію  вашого  сервера управління баз даних, де показати створену таблицю, її ім'я та загальні відомості бази даних, наприклад назва, ім'я, назва користувача адміністратора, пароль тощо.  Для роботи з сервером управління баз даних рекомендуємо використовувати програмне забезпечення компанії jetbrains datagrip. Або вбудовану панель користування базами даних, що міститься у середовищі intellij Idea,  яка на сьогоднішній день підтримує майже всі сервери управління баз даних.

**Хід роботи**

Код програми:

Student.java:

package ntudp.pzks.lr5.model;

import java.util.Objects;

public class Student {

private String lastName;

private String firstName;

private String middleName;

private String birthDate;

private String recordBookNumber;

public Student(String lastName, String firstName, String middleName, String birthDate, String recordBookNumber) {

this.lastName = lastName;

this.firstName = firstName;

this.middleName = middleName;

this.birthDate = birthDate;

this.recordBookNumber = recordBookNumber;

}

@Override

public String toString() {

return lastName + " " + firstName + " " + middleName + ", was born: " + birthDate + ". Student number: " + recordBookNumber;

}

@Override

public boolean equals(Object o) {

if (this == o) return true;

if (o == null || getClass() != o.getClass()) return false;

Student student = (Student) o;

return Objects.equals(lastName, student.lastName) && Objects.equals(firstName, student.firstName) &&

Objects.equals(middleName, student.middleName) && Objects.equals(birthDate, student.birthDate) &&

Objects.equals(recordBookNumber, student.recordBookNumber);

}

@Override

public int hashCode() {

return Objects.hash(lastName, firstName, middleName, birthDate, recordBookNumber);

}

}

DatabaseConnector.java:

package ntudp.pzks.lr5.database;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DatabaseConnector {

private static final String URL = "jdbc:postgresql://localhost:5432/students\_db";

private static final String USER = "illia";

private static final String PASSWORD = "illia";

public static Connection getConnection() throws SQLException {

return DriverManager.getConnection(URL, USER, PASSWORD);

}

}

StudentDatabaseManager.java:

package ntudp.pzks.lr5.database;

import ntudp.pzks.lr5.model.Student;

import java.sql.\*;

import java.util.ArrayList;

import java.util.List;

public class StudentDatabaseManager {

public List<Student> getStudentsByBirthMonth(int month) {

List<Student> students = new ArrayList<>();

String query = "SELECT last\_name, first\_name, middle\_name, birth\_date, record\_book\_number FROM students WHERE EXTRACT(MONTH FROM birth\_date) = ?";

try (Connection connection = DatabaseConnector.getConnection();

PreparedStatement preparedStatement = connection.prepareStatement(query)) {

preparedStatement.setInt(1, month);

ResultSet resultSet = preparedStatement.executeQuery();

while (resultSet.next()) {

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

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

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

String birthDate = resultSet.getString("birth\_date");

String recordBookNumber = resultSet.getString("record\_book\_number");

students.add(new Student(lastName, firstName, middleName, birthDate, recordBookNumber));

}

} catch (SQLException e) {

e.printStackTrace();

}

return students;

}

}

Run.java:

package ntudp.pzks.lr5;

import ntudp.pzks.lr5.database.StudentDatabaseManager;

import ntudp.pzks.lr5.model.Student;

import java.util.List;

public class Run {

public static void main(String[] args) {

StudentDatabaseManager manager = new StudentDatabaseManager();

int searchMonth = 7;

List<Student> students = manager.getStudentsByBirthMonth(searchMonth);

if (students.isEmpty()) {

System.out.println("There are no students born this month.");

} else {

System.out.println("Students born in " + searchMonth + " month:");

for (Student student : students) {

System.out.println("\t" + student);

}

}

}

}

database.sql:

CREATE TABLE students (

id SERIAL PRIMARY KEY,

last\_name VARCHAR(50) NOT NULL,

first\_name VARCHAR(50) NOT NULL,

middle\_name VARCHAR(50),

birth\_date DATE NOT NULL,

record\_book\_number VARCHAR(20) UNIQUE NOT NULL

);

INSERT INTO students (last\_name, first\_name, middle\_name, birth\_date, record\_book\_number) VALUES

('Smith', 'Daniel', 'Oliver', '2000-01-25', 'S001'),

('Johnson', 'William', 'Isaac', '2001-02-18', 'S002'),

('Brown', 'Roman', 'Sebastian', '2002-03-09', 'S003'),

('Davis', 'Ethan', 'Michael', '2000-04-30', 'S004'),

('Miller', 'Anna', 'Evelyn', '2001-05-12', 'S005'),

('Wilson', 'James', 'Benjamin', '2002-07-27', 'S006'),

('Moore', 'Olivia', 'Charlotte', '2000-08-14', 'S007'),

('Taylor', 'Timothy', 'Gregory', '2001-09-25', 'S008'),

('Anderson', 'Sophia', 'Madeline', '2002-10-07', 'S009'),

('Thomas', 'George', 'Dmitri', '2000-11-10', 'S010'),

('Jackson', 'Arthur', 'Alexander', '2001-12-28', 'S011'),

('White', 'Maria', 'Stephanie', '2002-01-05', 'S012'),

('Harris', 'Alexander', 'Frederick', '2000-02-21', 'S013'),

('Martin', 'Alice', 'Victoria', '2001-03-14', 'S014'),

('Thompson', 'Michael', 'Andrew', '2002-04-22', 'S015'),

('Garcia', 'Emma', 'Patricia', '2000-05-26', 'S016'),

('Martinez', 'Brandon', 'Julian', '2001-07-11', 'S017'),

('Robinson', 'Irene', 'Stella', '2002-08-19', 'S018'),

('Clark', 'Victor', 'Nicholas', '2000-09-03', 'S019'),

('Rodriguez', 'Mason', 'Eugene', '2001-10-29', 'S020');

Результат виконання програми:

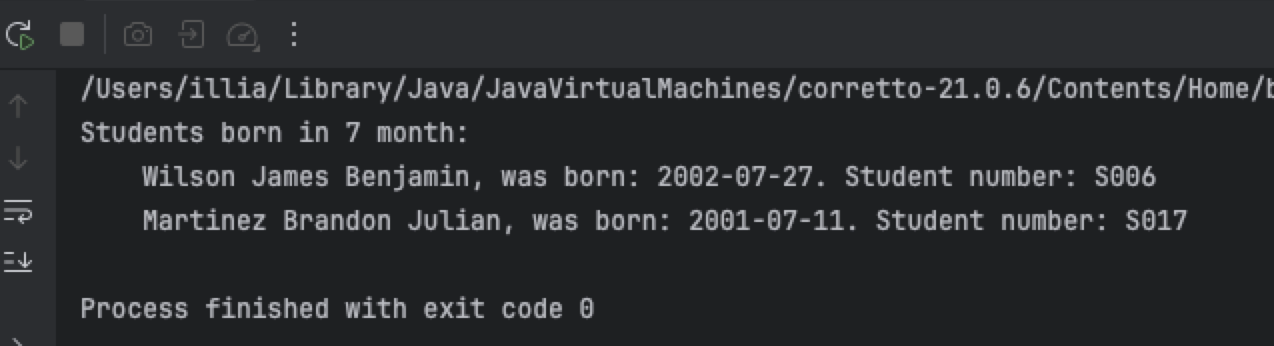


Рис. 5.1 – Результат виконання програми

Репозиторій з кодом програми: <https://github.com/illyaXD/Karapysh-javalabs>

**Висновки:** у ході виконання даної лабораторної роботи ми навчилися працювати з JDBC у Java.