

МIНIСТЕРСТВО ОСВIТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ “КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

Лабораторна робота № 3

з дисципліни “Бази даних”

тема “Практика використання графової бази даних Neo4J”

Виконала

студентка III курсу групи КП-81

Мозгова Катерина Олегівна

варіант №12

Перевірив “ ” “ ” 20 р.

Петрашенко Андрій Васильович

Київ 2021

# Завдання

*Мета роботи:* здобуття практичних навичок створення програм, орієнтованих на використання графової бази даних Neo4J за допомогою мови Python.

*Завдання*: реалізувати можливості формування графової бази даних в онлайн-режимі на основі модифікованої програми лабораторної роботи

№2. На основі побудованої графової бази даних виконати аналіз сформованих даних.

*Окремі програмні компоненти*

1. Інфраструктура лабораторної роботи №2:
   1. Redis server.
   2. Програма емуляції активності користувачі (вхід/вихід, відправка/отримання повідомлення).
   3. Виконувач задач (Worker).
2. Сервер Neo4J.
3. Інтерфейс користувача Neo4J.

*Порядок виконання роботи*

1. В ЛР№2 залишити єдиний режим роботи - емуляція активності.
2. Внести доповнення у програму ЛР№2 шляхом додавання у повідомлення тегу або тегів з переліку, заданого у вигляді констант, обраних студентом.
3. Встановити сервер [Neo4J Community Edition](https://neo4j.com/download-center/#community).
4. Розробити схему бази даних Neo4J для збереження інформації про активності користувачів (вхід/вихід, відправлення/отримання повідомлень) та Worker (перевірка на спам). Визначити вузли та зв’язки між ними на графі.
5. Розширити функціональність ЛР№2 шляхом збереження будь-якої активності (див. п. 4) у базу даних Neo4J у момент збереження даних у Redis.
6. У програмі “Інтерфейс користувача Neo4J” виконати і вивести результат наступних запитів до сервера Neo4J:
   1. Задано список тегів (*tags*). Знайти всіх користувачів, що відправили або отримали повідомлення з набором тегів *tags.*
   2. Задано довжину зв’язку N - кількість спільних повідомлень між користувачами. Знайти усі пари користувачів, що мають зв’язок довжиною N через відправлені або отримані повідомлення. Наприклад, якщо користувач A відправив повідомлення користувачу B, а B відправив повідомлення С, то довжина зв’язку між А і С є N=2.
   3. Задано два користувача. Знайти на графі найкоротший шлях між ними через відправлені або отримані повідомлення.
   4. Знайти авторів повідомлень, які пов’язані між собою лише повідомленнями, позначеними як “спам”.
   5. Задано список тегів (*tags*). Знайти всіх користувачів, що відправили або отримали повідомлення з набором тегів tags, але ці користувачі не пов’язані між собою.

*Вимоги до засобів емуляції даних*

Забезпечити генерацію даних відносно невеликого обсягу, що підтверджують коректність виконання завдання пунктів 6.1 - 6.5.

*Вимоги до інтерфейсу користувача*

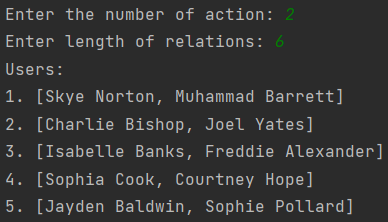
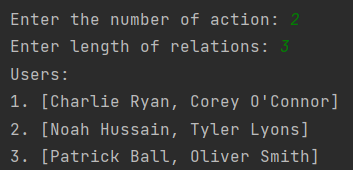
Використовувати консольний (текстовий) інтерфейс користувача.

# [GitHub посилання](https://github.com/Kyanka/DB/tree/main/lab3)

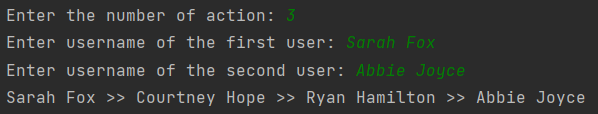
# Результати роботи програми:



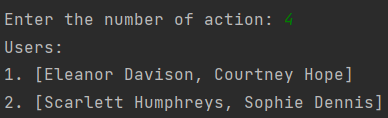
*Рис. 1.* Виконання завдання 6.1.



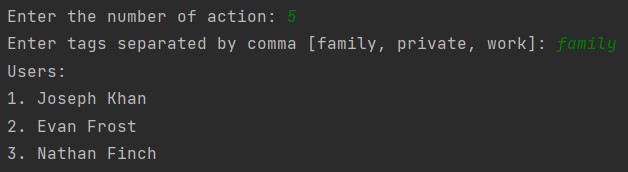
*Рис. 2-3.* Виконання завдання 6.2.



*Рис. 4.* Виконання завдання 6.3.



*Рис. 5.* Виконання завдання 6.4.



*Рис. 6.* Виконання завдання 6.5.

**Висновок:** в процесі виконання лабораторної роботи я здобула практичні навички створення програм, орієнтованих на використання графової бази даних Neo4J за допомогою мови Python.