Шмєльова Дар'я

БП-4, Комп'ютерні науки

Теоретичною основою **Neo4j** є модель графів властивостей, яка має наступний склад:

- властивості — пара «ключ - значення», ключ є рядком, значення може бути довільним типом;

- вершина - контейнер для властивостей, кожна додаткова вершина має власний ідентифікатор (аналог первинного ключа у реляційних базах);

- зв’язок - орієнтовне іменоване ребро, яке також може містити властивості та має ідентифікатор.

Маючи в основі нативну базу даних графів, **Neo4j** зберігає та керує даними у своєму більш природному, підключеному стані. База даних графів використовує підхід графа властивостей, що вигідно як для продуктивності обходу, так і для часу виконання операцій.

**Cypher**

Cypher - це мова графічних запитів Neo4j, яка дозволяє користувачам зберігати та отримувати дані з бази даних графіків. Це декларативна, натхненна SQL мова для опису шаблонів на графіках. Синтаксис забезпечує наочний і логічний спосіб зіставлення закономірностей вузлів і відношень на графіку. Cypher був розроблений, щоб бути простим для вивчення, розуміння та використання для всіх, але також включає потужність і функціональність інших стандартних мов доступу до даних.

* **START** визначає вершину або вершини, з якої (яких) буде розпочато обхід.
* **MATCH** описує, які саме частини графа необхідні користувачу, безпосередньо вищезгаданий взірець - «рисунок» підграфа
* **RETURN** визначає, які вершини, зв’язки або властивості повернути як результат

# Практичне завдання

## 1. Створено базову графову модель соціальної мережі, де сутність - користувач з пронумерованим ім’ям, та зв’язок - друзі

CREATE (user1:User{name:"User\_1"})

CREATE (user2:User{name:"User\_2"})

CREATE (user3:User{name:"User\_3"})

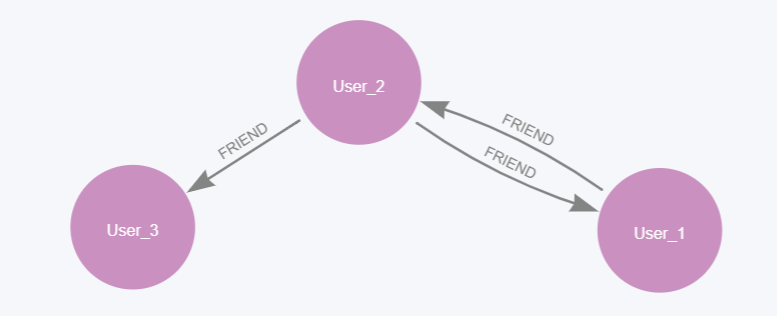
CREATE (user1) - [:FRIEND] -> (user2)

CREATE (user2) - [:FRIEND] -> (user1)

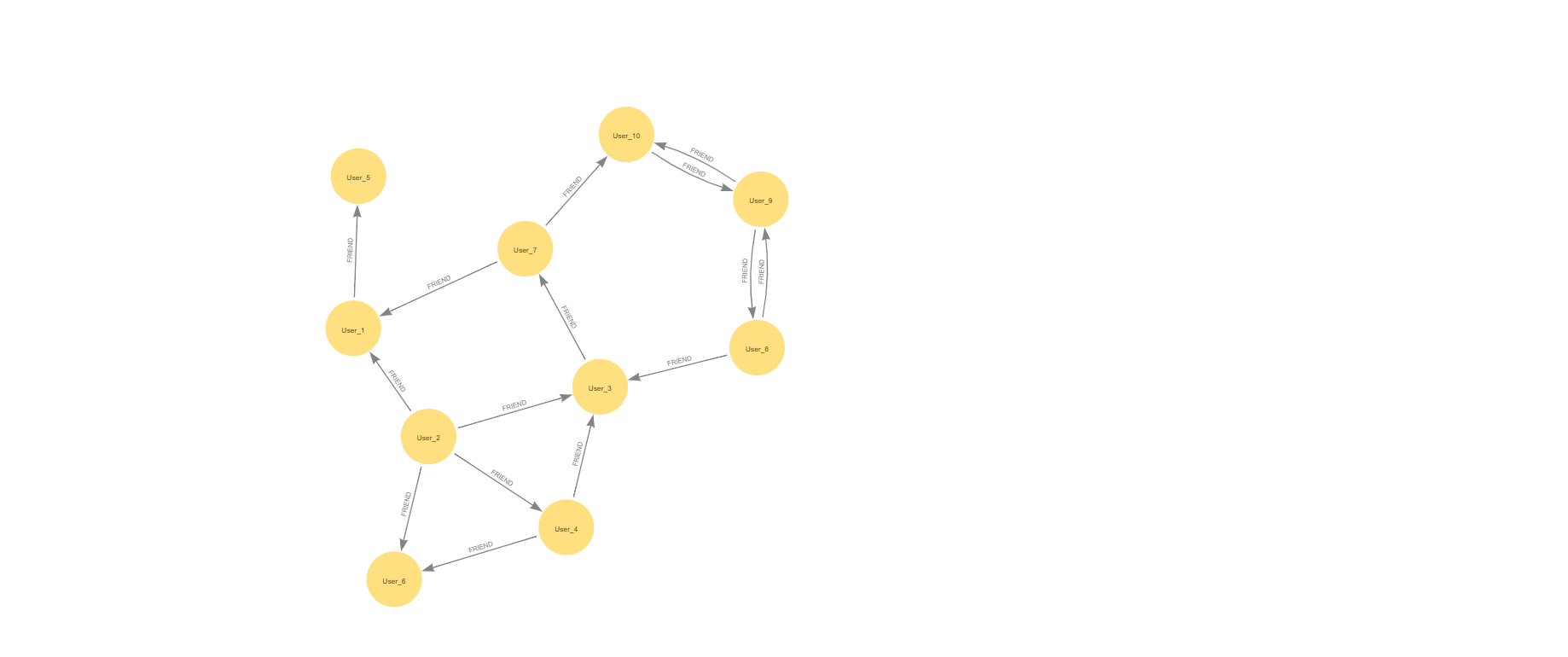
CREATE (user2) - [:FRIEND] -> (user3)

RETURN user1, user2, user3

Результат (граф):



## 2. Базу наповнено більшею кількістю даних



## 3. Написати мовою Cypher запити на пошук «друзів друзів». Візуалізувати засобами бази знайдені кластери друзі

MATCH (user1:User)-[:FRIEND] -> (user2:User)-[:FRIEND] -> (user3:User)

WHERE user1<>user3

CREATE (user1)-[:FRIEND\_OF\_FRIEND]->(user3)

RETURN user1, user3

