**Лабораторна робота №9**

**З дисципліни:** Бази даних та інформаційні системи  
**Студента групи МІТ-31:** Добровольського Дмитра

**Тема:** Розширені можливості Redis

**Мета роботи:** Закріпити знання про роботу з Redis та ознайомитися з розширеним функціоналом — транзакціями, скриптами на Lua, публікацією/підпискою (Pub/Sub) та потоками Redis Streams

**Завдання:**

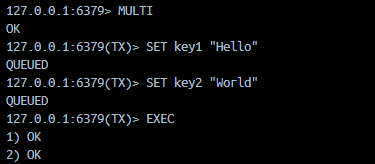
1. Транзакції у Redis

**Ознайомтесь з командами MULTI, EXEC, DISCARD, WATCH.**

WATCH balance  
MULTI  
DECR balance  
INCR purchases  
EXEC

**Створіть просту транзакцію, що додає кілька ключів.**

Приклад базової транкзакції:



**Спробуйте використати WATCH для імітації конкурентного доступу.**

|  |
| --- |
| Після EXEC обидві команди виконуються одночасно. |
| Імітація конкурентного доступу з WATCH У CLI 1:    У CLI 2 (нове вікно):    В CLI 1 вводимо і ось що бачимо:    Redis відповіла (nil) — транзакція не виконалась, бо balance змінився під час WATCH. |

1. Lua-скрипти в Redis

**Напишіть простий Lua-скрипт, який перевіряє наявність ключа і створює його, якщо не існує.**

Наприклад:

|  |
| --- |
| EVAL "if redis.call('exists', KEYS[1]) == 0 then    return redis.call('set', KEYS[1], ARGV[1])  else    return 'exists'  end"  1 testkey "value" |



Якщо запусти скрипт перший раз → створить testkey,якщо запусти ще раз → Redis скаже 'exists'.(на скриншоті 2 варіант)

1. Механізм Pub/Sub

**В одному терміналі запустіть підписку на канал:**

|  |
| --- |
| SUBSCRIBE news |
|  |

**В іншому терміналі виконайте публікацію повідомлення:**

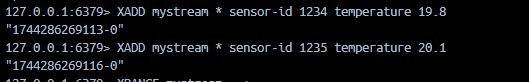
|  |
| --- |
| PUBLISH news "Redis is awesome!" |
| Перше вікно отримало повідомлення: |
|  |

        Вивчіть можливість використання Pub/Sub для чатів або нотифікацій.

1. Redis Streams (потоки даних)

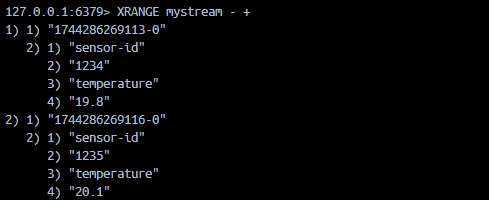
**Додайте події до потоку:**

|  |
| --- |
| XADD mystream \* sensor-id 1234 temperature 19.8 |



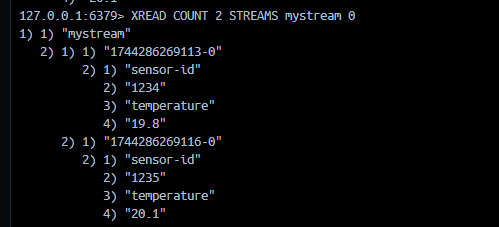
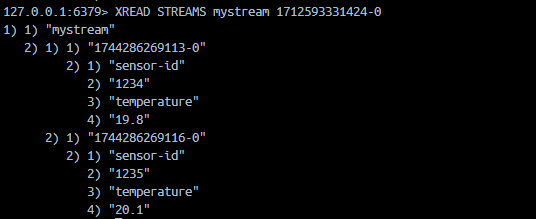
**Прочитайте останні події:**

|  |
| --- |
| XRANGE mystream - + |



**Створіть споживача потоку та зчитайте нові повідомлення:**

|  |
| --- |
| XREAD COUNT 2 STREAMS mystream 0 |

  
Після першого разу можна використовувати ID останнього повідомлення, щоб зчитувати тільки нові:  


**Висновок:**

Під час лабораторної роботи було закріплено знання про роботу з Redis та ознайомився з розширеним функціоналом — транзакціями, скриптами на Lua, публікацією/підпискою (Pub/Sub) та потоками Redis Streams.