

Лабораторна робота №8

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

Тема: Робота з Redis

Мета роботи: Закріпити розуміння роботи Redis та навчитися використовувати його основні можливості.

Завдання:

1. Встановлення Redis та запуск

- ✓ Якщо ви на Windows – використовуйте WSL або Docker.

Після встановлення ми створюємо та запускаємо **Redis** командою:
`docker run --name redis-container -d -p 6379:6379 redis`

Перевіряємо чи працює :

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
	NAMES				
1856b53682f0	redis	"docker-entrypoint.s..."	8 seconds ago	Up 6 seconds	0.0.0
.0:6379->6379/tcp	redis-container				

Після чого ми перевіряємо підключення :

```
PS C:\Users\kingg> docker exec -it redis-container redis-cli
127.0.0.1:6379> PING
PONG
```

2. Операції з рядками

- ✓ Створіть ключ зі своїм ім'ям: SET student "<Ваше ім'я>".
- ✓ Отримайте значення командою GET student.
- ✓ Виконайте INCR mycounter (двічі) та перевірте його значення.

```
127.0.0.1:6379> SET student "Dmytro"
OK
127.0.0.1:6379> GET student
"Dmytro"
127.0.0.1:6379> INCR mycounter
(integer) 1
127.0.0.1:6379> INCR mycounter
(integer) 2
127.0.0.1:6379> GET mycounter
"2"
127.0.0.1:6379> INCR mycounter
(integer) 3
127.0.0.1:6379> INCR mycounter
(integer) 4
127.0.0.1:6379> GET mycounter
"4"
```

3. Структури даних

✓ Список (List):

- LPUSH tasks "Task1", LPUSH tasks "Task2"
- LRANGE tasks 0 -1, LPOP tasks

```
127.0.0.1:6379> LPUSH tasks "Task1"
(integer) 1
127.0.0.1:6379> LPUSH tasks "Task2"
(integer) 2
127.0.0.1:6379> LPUSH tasks "Task3"
(integer) 3
127.0.0.1:6379> LRANGE tasks 0 -1
1) "Task3"
2) "Task2"
3) "Task1"
127.0.0.1:6379> LPOP tasks
"Task3"
127.0.0.1:6379> LRANGE tasks 0 -1
1) "Task2"
2) "Task1"
```

✓ Множина (Set):

- SADD tech:set "Redis", SADD tech:set "PostgreSQL", SADD tech:set "MongoDB"
- SMEMBERS tech:set, SISMEMBER tech:set "Redis"

```
127.0.0.1:6379> SADD tech:set "Redis"
(integer) 1
127.0.0.1:6379> SADD tech:set "PostgreSQL"
(integer) 1
127.0.0.1:6379> SADD tech:set "MongoDB"
(integer) 1
127.0.0.1:6379> SMEMBERS tech:set
1) "Redis"
2) "PostgreSQL"
3) "MongoDB"
127.0.0.1:6379> SISMEMBER tech:set "Redis"
(integer) 1
```

✓ Хеш (Hash):

- HSET profile name "<Ваше ім'я>", HSET profile city "<Ваше місто>"
- HGET profile name, HGETALL profile

```
127.0.0.1:6379> SISMEMBER tech:set "Redis"
(integer) 1
127.0.0.1:6379> HSET profile name "Dmytro"
(integer) 1
127.0.0.1:6379> HSET profile city "Zhytomyr"
(integer) 1
127.0.0.1:6379> HGET profile name
"Dmytro"
127.0.0.1:6379> HGETALL profile
1) "name"
2) "Dmytro"
3) "city"
4) "Zhytomyr"
```

✓ Sorted Set:

- ZADD scores 85 "Student1", ZADD scores 92 "Student2", ZADD scores 74 "Student3"
- ZREVRANGE scores 0 -1 WITHSCORES, ZRANK scores "Student1"

```
127.0.0.1:6379> ZADD scores 85 "Student1"
(integer) 1
127.0.0.1:6379> ZADD scores 92 "Student2"
(integer) 1
127.0.0.1:6379> ZADD scores 74 "Student3"
(integer) 1
127.0.0.1:6379> ZADD scores 76 "Student4"
(integer) 1
127.0.0.1:6379> ZREVRANGE scores 0 -1 WITHSCORES
127.0.0.1:6379> ZRANK scores "Student1"
(integer) 2
```

4. Працюємо з TTL

- SET temp:data "Hello" EX 10
- GET temp:data, TTL temp:data

```
127.0.0.1:6379> SET temp:data "Hello" EX 10
OK
127.0.0.1:6379> GET temp:data
"Hello"
127.0.0.1:6379> TTL temp:data
(integer) 1
127.0.0.1:6379> GET temp:data
(nil)
```

5. Міні-програма з Redis (додатково)

- ✓ Python або JavaScript-код для роботи з Redis.

```
import redis

# Підключення до Redis (локально, порт 6379)
r = redis.Redis(host='localhost', port=6379, decode_responses=True)

# Збільшуємо лічильник
r.incr('mycounter')
print("Лічильник:", r.get('mycounter'))

# Додаємо задачі в список
r.lpush('tasks', 'Написати звіт', 'Пройти захист')
print("Список задач:")
print(r.lrange('tasks', 0, -1))

# Публікуємо повідомлення (наприклад, у канал)
r.publish('updates', 'Користувач виконав завдання!')
```

[1] ✓ 0.1s Python

```
... Лічильник: 5
Список задач:
['Пройти захист', 'Написати звіт', 'Task2', 'Task1']
```

Висновок:

У ході виконання лабораторної роботи було успішно встановлено та запущено Redis, проведено практичну роботу з основними типами даних (рядки, списки, множини, хеші та відсортовані множини), а також опрацьовано механізм встановлення часу життя ключів (TTL). Завдяки цьому вдалося закріпити теоретичні знання та здобути практичні навички роботи з Redis, які можуть бути корисними у реальних проєктах для кешування, обробки сесій та зберігання структурованих даних.