

Київський національний університет ім. Тараса Шевченка

Кафедра мережеских та інтернет технологій

Лабораторна робота № 8

Дисципліна: бази даних та інформаційні системи

Тема: Робота з Redis

Виконав: Студент групи МІТ-31

Пугач Назар

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

Хід роботи

1. Встановлення Redis та запуск

- а. Встановіть Redis на свій комп'ютер (або використайте Docker).

```
PS C:\Users\Назар> docker run --name redis-container -d -p 6379:6379 redis
Unable to find image 'redis:latest' locally
cbb99c664e48: Pull complete
a7e27cf18de4: Pull complete
Digest: sha256:fbdbaea47b9ae4ecc2082ecdb4e1cea81e32176ffb1dcf643d422ad07427e5d
Digest: sha256:fbdbaea47b9ae4ecc2082ecdb4e1cea81e32176ffb1dcf643d422ad07427e5d
Status: Downloaded newer image for redis:latest
f30df20326cdce0aa1ad392fbd7150e3560b3521d27be09f26f6ad7deafe0d91
PS C:\Users\Назар> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS
f30df20326cd   redis    "docker-entrypoint.s..." 17 seconds ago Up 15 seconds
ntainer
PS C:\Users\Назар> docker exec -it redis-container redis-cli
127.0.0.1:6379>
```

Рисунок 1 – встановлення Redis.

- б. Запустіть Redis Server і підключіться до нього через redis-cli.
с. Переконайтесь, що команда PING повертає PONG.

```
(error) ERR unknown command 'redis-cli'
127.0.0.1:6379> PING
PONG
127.0.0.1:6379>
```

Рисунок 2 – сервер Redis працює.

2. Операції з рядками

- а. Створіть ключ зі своїм ім'ям: SET student "<Ваше ім'я>".
б. Отримайте значення командою GET student.

```
127.0.0.1:6379> set student Nazar
OK
127.0.0.1:6379> get student
"Nazar"
127.0.0.1:6379> incr student
(error) ERR value is not an integer or out of range
127.0.0.1:6379>
```

Рисунок 3 – створення рядка.

- с. Виконайте INCR mycounter (двічі) та перевірте його значення.

```
127.0.0.1:6379> set counter 1
OK
127.0.0.1:6379> get counter
"1"
127.0.0.1:6379> incr counter
(integer) 2
127.0.0.1:6379> get counter
"2"
```

Рисунок 4 – використання INCR.

3. Структури даних

а. 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> lrange 0 -1
(error) ERR wrong number of arguments for 'lrange' command
127.0.0.1:6379> lrange tasks 0 -1
1) "task2"
2) "task1"
127.0.0.1:6379> lpop tasks
"task2"
127.0.0.1:6379> lrange tasks 0 -1
1) "task1"
127.0.0.1:6379>
```

Рисунок 5 – робота зі списками.

б. Множина (Set):

```
127.0.0.1:6379> sadd fruits "apple"
(integer) 1
127.0.0.1:6379> sadd fruits "banana"
(integer) 1
127.0.0.1:6379> sadd fruits "banana"
(integer) 0
127.0.0.1:6379> sadd fruits "apple"
(integer) 0
127.0.0.1:6379> SMEMBERS fruits
1) "apple"
2) "banana"
127.0.0.1:6379> SISMEMBER fruits "apple"
(integer) 1
127.0.0.1:6379> 
```

Рисунок 6 – робота з множинами.

с. Хеш (Hash):

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

HGET profile name, HGETALL profile

```

127.0.0.1:6379> hset profile name "Nazar"
(integer) 1
127.0.0.1:6379> hset profile age 20
(integer) 1
127.0.0.1:6379> hget profile name
"Nazar"
127.0.0.1:6379> hgetall profile
1) "name"
2) "Nazar"
3) "age"
4) "20"
127.0.0.1:6379> 

```

Рисунок 6 – робота з хешами.

d. 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 score 92 "Student2"
(integer) 1
127.0.0.1:6379> zadd score 99 "Student3"
(integer) 1
127.0.0.1:6379> ZREVRANGE score 0 -1 WITHSCORES
1) "Student3"
2) "99"
3) "Student2"
4) "92"
5) "Student1"
6) "85"
127.0.0.1:6379> ZREVRANGE score 0 -1
1) "Student3"
2) "Student2"
3) "Student1"
127.0.0.1:6379> zrank score "Student1"
(integer) 0

```

Рисунок 7 – робота з сортованими сетами.

4. Працюємо з TTL

a. SET temp:data "Hello" EX 10

b. GET temp:data, TTL temp:data

```

127.0.0.1:6379> set temp "Hello" EX 20
OK
127.0.0.1:6379> ttl temp
(integer) 15
127.0.0.1:6379> get temp
"Hello"
127.0.0.1:6379> get temp
(nil)
127.0.0.1:6379> 

```

Рисунок 8 – робота з TTL.

5. Міні-програма з Redis (додатково).

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

```
import redis

r = redis.Redis(host='localhost', port=6379, db=0)

r.incr("visits")
r.lpush("tasks", "new_task")

tasks = r.lrange("tasks", 0, -1)

print("Tasks:", [task.decode() for task in tasks])
```

Рисунок 9 – Python код для роботи з Redis.

```
Tasks: ['new_task', 'task1']
PS D:\Навчання\db>
```

```
(redis)
127.0.0.1:6379> get visits
"1"
```

Рисунок 10 – Результат роботи коду.

Висновок: Під час виконання лабораторної роботи я закріпив розуміння роботи Redis та навчився використовувати його основні можливості.