

Practical – 02

Aim: - Interacting With Redis

- a) Install Redis on your lab server or local machine.
- b) Store and retrieve data in Redis using various data structures like strings, lists, and sets.
- c) Implement basic Redis commands for data manipulation and retrieval

Description: - Redis is an open-source in-memory storage, used as a distributed, in-memory key–value database, cache and message broker, with optional durability. Because it holds all data in memory and because of its design, Redis offers low-latency reads and writes, making it particularly suitable for use cases that require a cache.

Execution:

1. Installing Redis

- `sudo apt-get update`
- `sudo apt-get install redis-server`

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

```
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libatomic1 amd64 12.3.0-1ubuntu1~22.04 [10.4 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 libjemalloc2 amd64 5.2.1-4ubuntu1 [240 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 liblua5.1-0 amd64 5.1.5-8.1build4 [99.9 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 liblzf1 amd64 3.6-3 [7,444 B]
Get:5 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 lua-bitop amd64 1.0.2-5 [6,680 B]
Get:6 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 lua-cjson amd64 2.1.0+dfsg-2.1 [17.4 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 redis-tools amd64 5:6.0.16-1ubuntu1 [856 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 redis-server amd64 5:6.0.16-1ubuntu1 [45.9 kB]
Fetched 1.283 kB in 45s (28.4 kB/s)
```

2) Connecting Redis

- Redis-cli

```
127.0.0.1:6379> ping hi
"hi"
127.0.0.1:6379> █
```

3) Performing CRUD operations using Strings, List and Sets.

- Creating Strings

```
127.0.0.1:6379> SET mykey "Redis connected !"
OK
127.0.0.1:6379> GET mykey
"Redis connected !"
```

- Working with Lists

```
127.0.0.1:6379> LPUSH mylist element1 element2 element3 element3
(integer) 4
127.0.0.1:6379> LRANGE mylist 0 -1
1) "element3"
2) "element3"
3) "element2"
4) "element1"
```

- Working with Sets

```
127.0.0.1:6379> SADD myset 1 2 3 4 4 5 6
(integer) 6
```

```
127.0.0.1:6379> SMEMBERS myset
1) "1"
2) "2"
3) "3"
4) "4"
5) "5"
6) "6"
```

- Basic Redis

Commands# Check

```
127.0.0.1:6379> EXISTS mykey
(integer) 1
```

if
a key exists

Delete a key

```
127.0.0.1:6379> DEL mykey
(integer) 1
```

Increment a key

```
127.0.0.1:6379> SET mynumber 10
OK
127.0.0.1:6379> INCR mynumber
(integer) 11
127.0.0.1:6379> GET mynumber
"11"
```

Decrement a key

```
127.0.0.1:6379> DECR mynumber
(integer) 10
127.0.0.1:6379> GET mynumber
"10"
```

Expire a key after a specific time (in second)

```
127.0.0.1:6379> EXPIRE mynumber 60
(integer) 1
127.0.0.1:6379> TTL mynumber
(integer) 50
127.0.0.1:6379> TTL mynumber
(integer) 48
```