

Day - 1

2. Installation of NoSQL Database - Redis on Linux

Update your package list: `sudo apt-get update`

Install Redis: `sudo apt-get install redis-server`

Start Redis: `sudo systemctl start redis`

Enable Redis to start on boot: `sudo systemctl enable redis`

Verify the installation: `redis-cli ping` (**You should see PONG as the response.**)

3. Configuration of Redis Database. (CONFIG, GET).

```
127.0.0.1:6379> ping
```

```
PONG
```

```
127.0.0.1:6379> config get *
```

```
1) "dbfilename"
```

```
2) "dump.rdb"
```

```
3) "requirepass"
```

```
4) ""
```

```
5) "masterauth"
```

```
6) ""
```

```
7) "cluster-announce-ip"
```

```
8) ""
```

```
9) "unixsocket"
```

4. Modifying or Editing the Configurations (CONFIG, SET)

```
127.0.0.1:6379> get maxmemory
```

```
(nil)
```

```
127.0.0.1:6379> set maxmemory 512mb
```

```
OK
```

```
127.0.0.1:6379> get maxmemory
```

```
"512mb"
```

5. Redis-CLI Connection Commands (ECHO,PING,AUTH,SELECT,QUIT)

```
127.0.0.1:6379>127.0.0.1:6379> echo "Siva Sivani"
```

```
"Siva Sivani"
```

```
127.0.0.1:6379> ping
```

```
PONG
```

```
127.0.0.1:6379> config set requirepass ssdc
```

```
OK
```

```
127.0.0.1:6379> auth pass
```

```
(error) ERR invalid password
```

```
127.0.0.1:6379> auth ssdc
```

```
OK
```

```
127.0.0.1:6379> config set requirepass ""
```

```
OK
127.0.0.1:6379> ping
PONG
127.0.0.1:6379> 127.0.0.1:6379> select 2
OK
127.0.0.1:6379[2]> select 0
OK
```

6.Practice Different Data types in Redis Database using Redis-CLI.

Strings:

```
127.0.0.1:6379> set clg ssdc
```

OK

```
127.0.0.1:6379> get clg
```

"ssdc"

Hashes:

```
127.0.0.1:6379> hset clgcourses bscds 100 bscele 50
```

(integer) 2

```
127.0.0.1:6379> hmset clgcourses bcomca 180
```

OK

```
127.0.0.1:6379> hset clgcourses bba 180
```

(integer) 1

```
127.0.0.1:6379> hgetall clgcourses
```

1) "bscds"

2) "100"

3) "bscele"

4) "50"

5) "bcomca"

6) "180"

List:

```
127.0.0.1:6379> lpush order laptop
```

(integer) 1

```
127.0.0.1:6379> lpush order keyboard
```

(integer) 2

```
127.0.0.1:6379> lrange order 0 -1
```

```
1) "keyboard"
```

```
2) "laptop"
```

```
127.0.0.1:6379> rpush order mouse
```

```
(integer) 3
```

```
127.0.0.1:6379> lrange order 0 -1
```

```
1) "keyboard"
```

```
2) "laptop"
```

```
3) "mouse"
```

Sets:

```
127.0.0.1:6379> sadd languages c
```

```
(integer) 1
```

```
127.0.0.1:6379> sadd languages c+
```

```
(integer) 1
```

```
127.0.0.1:6379> sadd languages java
```

```
(integer) 1
```

```
127.0.0.1:6379> sadd languages python
```

```
(integer) 1
```

```
127.0.0.1:6379> smembers languages
```

```
1) "python"
```

```
2) "java"
```

```
3) "c+"
```

```
4) "c"
```

Sorted Sets:

```
127.0.0.1:6379> zadd rating 1 AI&ML
```

```
(integer) 1
```

```
127.0.0.1:6379> zadd rating 1 GenerativeAI
```

```
(integer) 1
```

```
127.0.0.1:6379> zadd rating 1 PromptEngineering
```

```
(integer) 1
```

```
127.0.0.1:6379> zrangebyscore rating 0 100
```

```
1) "AI&ML"
```

```
2) "GenerativeAI"
```

```
3) "PromtEngineering"
```

```
127.0.0.1:6379>
```

Day - 2

In Redis-cli we can use pre- defined databses. There are 16 databses in redis-cli. These are names as 1 to 15.

```
127.0.0.1:6379> config get databases
```

```
1) "databases"
```

```
2) "16"
```

```
127.0.0.1:6379> select 10
```

```
OK
```

```
127.0.0.1:6379[10]> select 0
```

```
OK
```

```
127.0.0.1:6379> hmset std:1 name raju dob 10-6-2000 course 3-ds address hyd mobi  
le 00000000000
```

```
OK
```

```
127.0.0.1:6379> hmset std:2 name rakesh dob 1-9-2001 course 3bba address wrgl 00  
00000000
```

```
(error) ERR wrong number of arguments for HMSET
```

```
127.0.0.1:6379> hmset std:2 name rakesh dob 1-9-2001 course 3bba address wrgl mo  
bile 00000000000
```

```
OK
```

```
127.0.0.1:6379> hgetall std:1 std:2
```

```
(error) ERR wrong number of arguments for 'hgetall' command
```

```
127.0.0.1:6379> hgetall std:1
```

1) "name"

2) "raju"

3) "dob"

4) "10-6-2000"

5) "course"

6) "3-ds"

7) "address"

8) "hyd"

9) "mobile"

10) "0000000000"

127.0.0.1:6379> hget std:2

(error) ERR wrong number of arguments for 'hget' command

127.0.0.1:6379> hget std:2 naame

(nil)

127.0.0.1:6379> hget std:2 name

"rakesh"

127.0.0.1:6379> hset std:1 name balu

(integer) 0

127.0.0.1:6379> hget std:1 name

"balu"

127.0.0.1:6379> hdel std:1

(error) ERR wrong number of arguments for 'hdel' command

127.0.0.1:6379> hdel std:1 name

(integer) 1

127.0.0.1:6379> hgetall std:1

1) "dob"

2) "10-6-2000"

3) "course"

4) "3-ds"

5) "address"

6) "hyd"

7) "mobile"

8) "0000000000"

3. CRUD (Create, Read, Update, and Delete) operations on the Redis Database using Redis-CLI for Employee Database.

Write same as Above for Employee Details

Create:

Write same as Above for Product Details

(Name, Qualification, Experience, Department, Mobile, Salary)

Read:

a) Display details of employee 2

b) Display name, department, Salary of employee 1

c) Display name , experience of employee

Update:

a) update the salary of employee 2 by incrementing 5000.

b) Change the department of employee 3

Delete:

a) Delete the employee 2

4. CRUD (Create, Read, Update, and Delete) operations on the Redis Database using Redis-CLI for Product Database.

Create:

Write same as Above for Product Details

(Name, Company, Quantity, Price, Mfd-Date,Exp-Date)

Read:

- d) Display Product Details of product 5
- e) Display name, price, exp-date of product 3

Update:

- c) update the price of product 1
- d) update the exp-date of product 3

Delete:

- a) Delete the product 3

5. CRUD (Create, Read, Update, and Delete) operations on the Redis Database using GUI Interface RedisInsight for Student Database.

The RedisInsight graphic user interface helps you visually browse and interact with Redis data. Browse, filter, and visualize Redis keys, perform CRUD operations, or delete keys in bulk.

1. Redis Insight window. Click on ADD **REDIS DATABASE** Button.
2. Select **Add Database** Manually option, then enter the details of host- localhost, port:6379 and Database Alias: **New Database Name**.
Then click on Add Redis Database.
3. Redis Database created with specified Alias Name.
4. Click on Database Name to perform CRUD operations on RADIS database.

6. CRUD (Create, Read, Update, and Delete) operations on the Redis Database using GUI Interface RedisInsight for Employee Database.

Create:

1. Click on “ **+Key** ” Button. New key window opened in the same window.
2. Select which type of key going to create (**Set, String, Hash, List, Sortedlist**) from the option “ Key Type ”
3. Then Type “ **Key Name** ” and value for the Key.
4. Then click on “ **Add Key** ”

The key is successfully added to database.

Read: To read the data click on " **Key Name** " then the values are displayed right side of the same window.

a) Read the details of Student 1

Step 1: Click on the **Student1 Key** which is listed in Keys.

Right side window the student 1 details will be displayed.

Update:

1. click on " **Key Name** " then the values are displayed right side of the same window.

2. Which field / key value want to change simply click edit icon of that field value and edit.

a) Change the student1 mobile number.

Step 1: Select the Student1 Key, which is to be update. The student details are displayed right of the window.

Step 2: Click on edit icon of the value field of **mobile field**. Delete and Retype new mobile number.

Delete: We can delete the total Key or some fields of Key using delete icon.

Q: Delete the Address field of student 1.

Ans: 1. Select the **Student1** key from **Keys list**

2. Click on delete icon of the Address field. Popup window opened.

3. Then click on Remove button in popup window.

4. The address field will be deleted.

Q: Delete the record of student1

Ans: 1. Select the Student1 key from Keys list

2. Click on delete icon available in Key properties. Popup window opened.

3. Then click on Delete button in popup window.

4. The student1 will be deleted.

7. CRUD (Create, Read, Update, and Delete) operations on the Redis Database using GUI

Interface RedisInsight for Product Database.

Ans: Write the above steps for Product Table

8. Connecting Python Application program with Redis Database.

Step 1: Run a Redis server

Step 2: Install the Redis client library using pip command

```
C:\Program Files\Python312\Scripts>pip install redis
Defaulting to user installation because normal site-packages is not writeable
Collecting redis
  Downloading redis-5.0.8-py3-none-any.whl.metadata (9.2 kB)
  Downloading redis-5.0.8-py3-none-any.whl (255 kB)
----- 255.6/255.6 kB 231.0 kB/s eta 0:00:00
Installing collected packages: redis
Successfully installed redis-5.0.8

[notice] A new release of pip is available: 24.0 -> 24.2
[notice] To update, run: C:\Program Files\Python312\python.exe -m pip install --
C:\Program Files\Python312\Scripts>
```

pip install redis

Step 3: import redis module

import redis

Step 4: Establish connection to localhost

Redis_obj = redis.Redis(host= 'localhost', port= '6379')

Step 5: Write the code using differ data type keys.

Ex: redis_obj.set ('mykey', 'Hello from Python!')

value = redis_obj.get('mykey')

print(value)