* Start server: redis-server (sudo service redis-server start)
* Restart redis service: sudo service redis-server restart
* Stop redis service: sudo service redis-server stop
* Access redis client: redis-cli

Test connection:

127.0.0.1:6379> ping

PONG

* Quit redis client: quit
* Clear screen: clear

**Data types:**

1. String

+ SET key value: Set key, nếu key đã tồn tại sẽ ghi đè value

+ GET key: Lấy key

+ DEL key: Xóa key

+ EXISTS key: kiểm tra key có tồn tại hay không? (return 0,1)

+ KEYS pattern: search các key match với pattern (KEYS \* sẽ lấy toàn bộ key hiện có)  
+ FLUSHALL: Xóa toàn bộ key

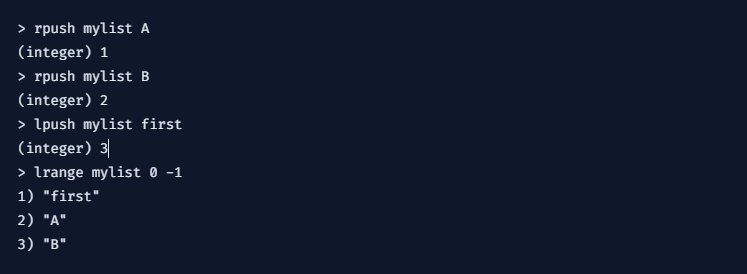
1. List

Redis List are implemented via Linked List.

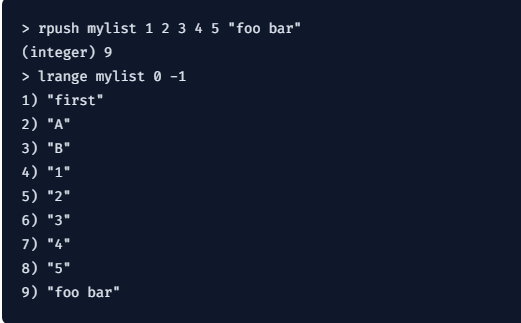
+ LPUSH list value: thêm value vào bên trái list

+ RPUSH list value: thêm value vào bên phải list

+ LRANGE list 0 -1: Lấy ra tất cả các phần tử của list (-1 nghĩa là phần tử cuối cùng của list, -2 là áp chót)



Chúng ta có thể push nhiều phần tử vào 1 list trong 1 câu lệnh:



+ RPOP list: Xóa phần tử ở bên phải cùng và trả về phần tử đó

+ LPOP list: Xóa phần tử ở bên trái cùng và trả về phần tử đó



Nếu không có phần tử nào thì sẽ trả về null:



1. Set

+ SADD set value: Thêm value vào set

+ SMEMBERS set: Trả về set

+ SISMEMBER set value: Kiểm tra value có tồn tại trong set hay không

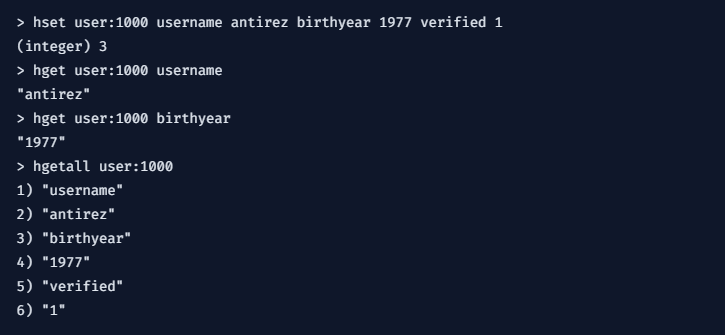
+ SREM set value: Xóa value ra khỏi set

1. Hash

Redis hash lưu trữ hash table của các cặp key-value, trong đó key được sắp xếp ngẫu nhiên, không

theo thứ tự nào cả. Redis hỗ trợ các thao tác thêm, đọc, xóa từng phần tử, cũng như đọc tất cả giá

trị.



**Expiration**

+ TTL key: trả về thời gian sống còn lại của key (time to live), -1 nghĩa là k ko bị expiration, -2 nghĩa key đã hết hạn (không có key).

+ EXPIRE key second: set thời gian sống của key là second giây.

+ SETEX key second value: tạo key và set thời gian sống cho nó