Redis Command Reference Guide

1. Strings/Numbers

Command	Syntax	Example	Output
SET	SET key value	SET myKey "Hello"	`127.0.0.1:6379> SET myKey "Hello"` `OK`
Description: Set key to hold the string value. If key already holds a value, it is overwritten, regardless of its type.Time Complexity: O(1)			
GET	GET key	GET myKey	`127.0.0.1:6379> GET myKey` `"Hello"`
Description: Get the string value of key. If the key does not exist the special value nil is returned.Time Complexity: O(1)			

MGET	MGET key [key ...]	MGET myKey nonExistentKey	`127.0.0.1:6379> MGET myKey nonExistentKey`
			`1) "Hello"`
			`2) (nil)`

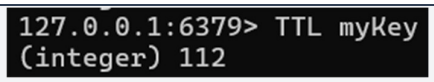
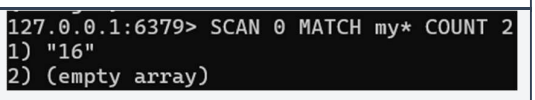
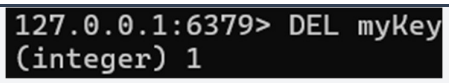
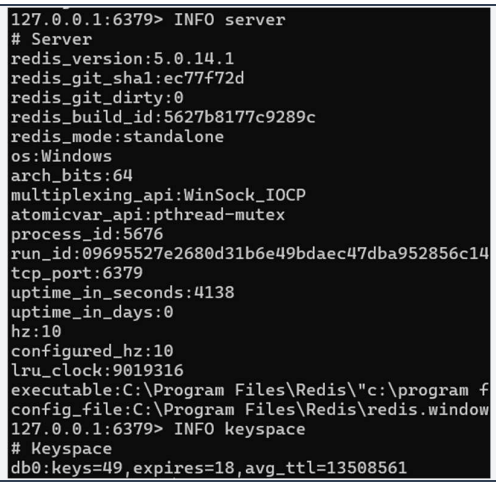
Description: Returns the values of all specified keys. For every key that does not hold a string value or does not exist, the special value nil is returned.Time Complexity: O(N)			
INCR	INCR key	INCR myCounter	`127.0.0.1:6379> INCR myCounter` `(integer) 1`
Description: Increments the number stored at key by one. If the key does not exist, it is set to 0 before performing the operation.Time Complexity: O(1)			

2. Generic

Command	Syntax	Example	Output

KEYS	KEYS pattern	KEYS my*	`127.0.0.1:6379> KEYS my*`
			`1) "myCounter"`
			`2) "myKey"`

Description: Returns all keys matching pattern.Time Complexity: O(N)			
EXISTS	EXISTS key [key ...]	EXISTS myKey	`127.0.0.1:6379> EXISTS myKey` `(integer) 1`
Description: Checks if one or more keys exist.Time Complexity: O(N)			
EXPIRE	EXPIRE key seconds	EXPIRE myKey 120	`127.0.0.1:6379> EXPIRE myKey 120` `(integer) 1`

Description: Set a timeout on a key.After the timeout has expired, the key will automatically be deleted.Time Complexity:O(1)			
TTL	TTL key	TTL myKey	
Description: Returns the remaining time to live of a key that has a timeout.Time Complexity: O(1)			
PERSIST	PERSIST key	PERSIST myKey	
Description: Removes the expiration from a key.Time Complexity:O(1)			
SCAN	SCAN cursor [MATCH pattern] [COUNT count]	SCAN 0 MATCH my* COUNT 2	
Description: Iterates the set of keys in the currently selected Redis database.Time Complexity: O(1) for every call. O(N) for a complete iteration.			
DEL	DEL key [key ...]	DEL myKey	
Description: Removes the specified keys.Time Complexity: O(N)			
INFO	INFO [section]	INFO server INFO keyspace	
Description:Returns information and statistics about the server, with the different sections like - server, clients, memory, persistence, stats, replication, cpu, commandstats, latencystats, sentinel, cluster, modules, keyspace, errorstats.Time Complexity: O(1)			

3. Hashes

Command	Syntax	Example	Output
HSET	HSET key field value [field value ...]	HSET h_employee_profile:101 name "Nicol" age 33	127.0.0.1:6379> HSET h_employee_profile:101 name "Nicol" age 33 (integer) 2
Description: Sets the specified fields to their respective values in the hash stored at key.Time Complexity: O(N)			
HGET	HGET key field	HGET h_employee_profile:101 name	127.0.0.1:6379> HGET h_employee_profile:101 name "Nicol"
Description: Returns the value associated with field in the hash stored at key.Time Complexity: O(1)			
HGETALL	HGETALL key	HGETALL h_employee_profile:101	127.0.0.1:6379> HGETALL h_employee_profile:101 1) "name" 2) "Nicol" 3) "age" 4) "33"
Description: Returns all fields and values of the hash stored at key.Time Complexity: O(N)			
HMGET	HMGET key field1 [field2]	HMGET h_employee_profile:101 name age	127.0.0.1:6379> HMGET h_employee_profile:101 name age 1) "Nicol" 2) "33" 127.0.0.1:6379>
Description: Returns the values associated with the specified fields in the hash stored at key.Time Complexity: O(N)			

4. Sets

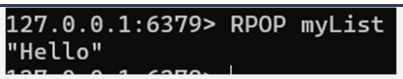
Command	Syntax	Example	Output
SADD	SADD key member [member ...]	SADD mySet "Hello"	127.0.0.1:6379> SADD mySet "Hello" (integer) 1
Description: Adds the specified members to the set stored at key. Time Complexity: O(N)			
SMEMBERS	SMEMBERS key	SMEMBERS mySet	127.0.0.1:6379> SMEMBERS mySet 1) "Hello"
Description: Returns all the members of the set value stored at key.Time Complexity: O(N)			

SCARD	SCARD key	SCARD mySet	127.0.0.1:6379> SCARD mySet (integer) 1
Description: Returns the set cardinality (number of elements) of the set stored at key.Time Complexity: O(1)			
SISMEMBER	SISMEMBER key member	SISMEMBER mySet "Hello"	127.0.0.1:6379> SISMEMBER mySet "Hello" (integer) 1
Description: Returns if member is a member of the set stored at key.Time Complexity: O(1)			
SDIFF	SDIFF key1 [key2]	SDIFF mySet myOtherSet	127.0.0.1:6379> SDIFF mySet myOtherSet 1) "Hello"
Description: Returns the members of the set resulting from the difference between the first set and all the successive sets.Time Complexity: O(N)			
SDIFFSTORE	SDIFFSTORE destination key1 [key2]	SDIFFSTORE myNewSet mySet myOtherSet	127.0.0.1:6379> SDIFFSTORE myNewSet mySet myOtherSet (integer) 1
Description: This command is equal to SDIFF, but instead of returning the resulting set, it is stored in destination.Time Complexity: O(N)			
SREM	SREM key member [member ...]	SREM mySet "Hello"	127.0.0.1:6379> SREM mySet "Hello" (integer) 1
Description: Removes the specified members from the set stored at key.			

5.Sorted sets

Command	Syntax	Example	Output
ZADD	ZADD key score member [score member ...]	ZADD myZSet 1 "one" 2 "two"	127.0.0.1:6379> ZADD myZSet 1 "one" 2 "two" (integer) 2
Description: Adds all the specified members with the specified scores to the sorted set stored at key. Time Complexity: O(log(N))			
ZRANGE	ZRANGE key start stop [WITHSCORES]	ZRANGE myZSet 0 -1	127.0.0.1:6379> ZRANGE myZSet 0 -1 1) "one" 2) "two"
Description: Returns the specified range of elements in the sorted set stored at key.Time Complexity: O(log(N)+M) where M is the number of elements returned			

6. Lists

Command	Syntax	Example	Output
LPUSH	LPUSH key value [value ...]	LPUSH myList "World"	
Description: Inserts the specified values at the head of the list stored at key. Time Complexity: O(N)			
RPUSH	RPUSH key value [value ...]	RPUSH myList "Hello"	
Description: Inserts the specified values at the tail of the list stored at key.Time Complexity: O(N)			
LRANGE	LRANGE key start stop	LRANGE myList 0 -1	
Description: Returns the specified elements of the list stored at key.Time Complexity: O(S+N) where S is the distance of start and N is the number of elements in the specified range.			
LLEN	LLEN key	LLEN myList	
Description: Returns the length of the list stored at key.Time Complexity: O(1)			
LPOP	LPOP key [count]	LPOP myList	
Description: Removes and returns the first element of the list stored at key.Time Complexity: O(N)			
RPOP	RPOP key [count]	RPOP myList	
Description: Removes and returns the last element of the list stored at key.Time Complexity: O(N)			