**Title:** Redis ZCOUNT

***Excerpt:*** The ZCOUNT command is used to count the number of members in a specified range of scores for a sorted set stored at a given key. It operates with great performance with O(log(N)) time complexity. It takes a minimum and maximum score values to define a range and these values are inclusive by default.

**Permalink:** redis-zcount

**Category:** Redis

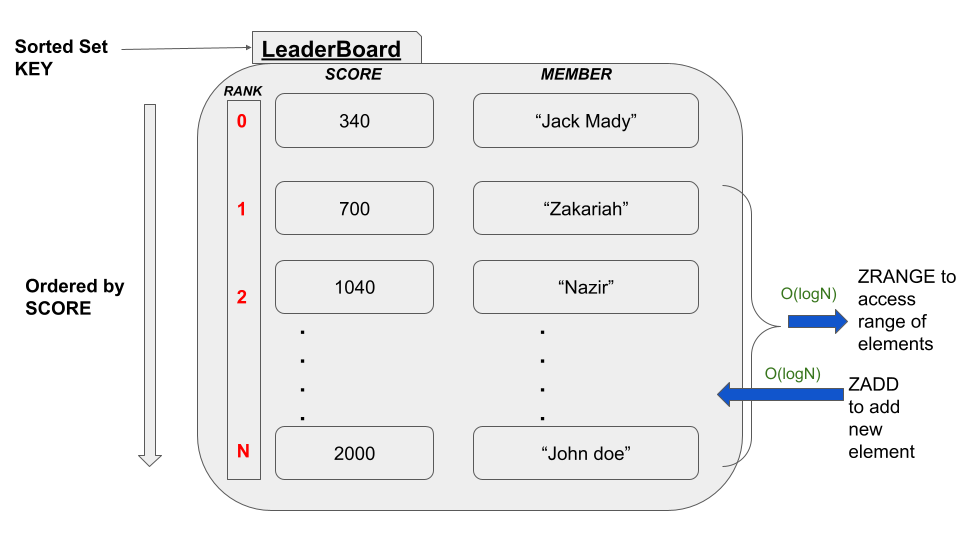
# 

# 

# 

# Redis Sorted Set Score and Rank

Redis Sorted Sets are the ordered variant of the typical set data structure. Each sorted set element is associated with two special properties: rank and score. The score value has been used to order the sorted set elements in ascending order. Furthermore, the scores can be repeated while the members must be unique for a given set. Since the sorted set elements are ordered, the add, update and remove operations are faster. In addition, this data structure allows querying elements in a range by score or rank value with great performance.



As shown in the above illustration, most of the major commands that operate on Redis sets have the time complexity of O(log(N)) which is faster.

# The ZCOUNT Command

The ZCOUNT command is used to query a range of set elements between the specified scores. The returned elements are sorted from the lowest to highest score. Whenever multiple elements hold the same scores, those are ordered by lexicographical order. This command also has the O(log(N))time complexity because it uses the rank property when querying a range of elements. Hence, no linear relationship to the number of elements when measuring the execution time.

The following is the syntax of the ZCOUNT command.

## Syntax:

| ZCOUNT set\_key minimum\_score maximum\_score |
| --- |

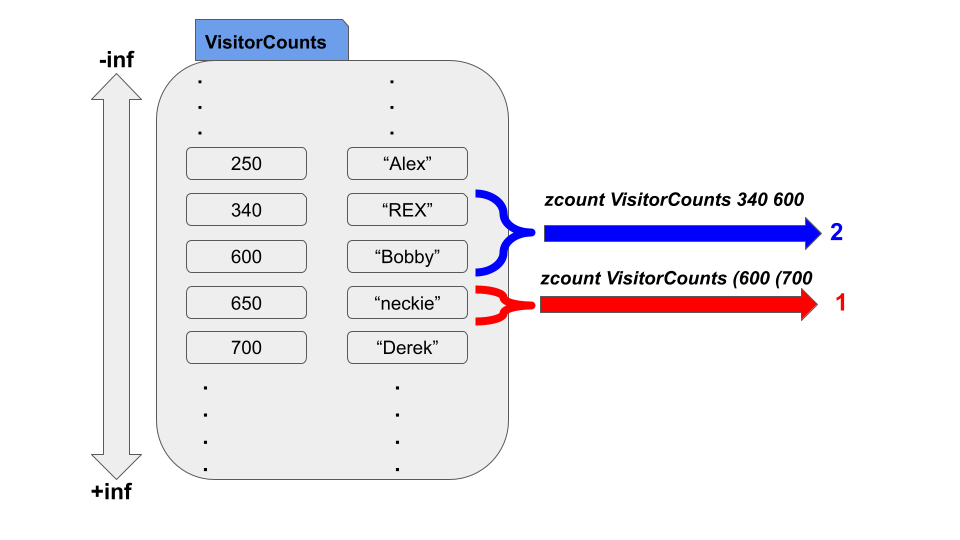
***set\_key:*** The key of the Redis sorted set.

***minimum\_score:*** The lowest score value of the range specified.

***maximum\_score:*** The highest score value of the range specified.

The min and max ends of the range can be specified in different ways. Whenever you have no clue about the lowest and highest possible scores in the sorted set, the -inf and +inf can be used. It will ideally fetch all the elements in the sorted set.

In addition, the specified minimum and maximum values are inclusive of the range. If you want any of these values to be exclusive, then the `(` character can be used as in the *ZRANGEBYSCORE* command.



This command returns an integer value which is the number of elements in the specified range.

## Use Case - Count the Players with a Gold Count between given range

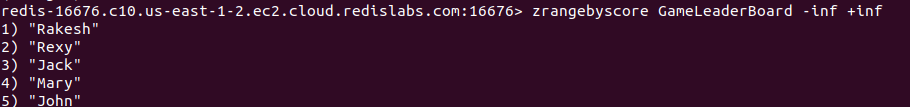
Redis sorted set data structure is an ideal candidate for storing leaderboard data. Let’s assume a scenario where an online game offers an amount of gold for its players when the missions are completed. Based on each player’s gold amount, a leaderboard needs to be implemented. We can easily use Redis sorted sets to implement such a leaderboard. The gold amount can be mapped as the score of each member.

Let’s create a sorted set *GameLeaderBoard* and add some players with gold amounts as shown in the following. The ZADD command is used to create and add players to the sorted set stored at the key *GameLeaderBoard.*

| zadd GameLeaderBoard 1000 "Jack" 450 "Rexy" 3000 "John" 1600 "Mary" 450 "Rakesh" |
| --- |

Let’s use the ZRANGEBYSCORE command to check whether the members have been added and sorted properly.

| zrangebyscore GameLeaderBoard -inf +inf |
| --- |



As expected the members have been stored and sorted by scores. Since “Rexy” and “Rakesh” got the same scores, they have been ordered lexicographically with “Rakesh” being the top member of the returned list.

Let’s count the number of elements in the sorted set using the ZCOUNT command.

| zcount GameLeaderBoard -inf +inf |
| --- |



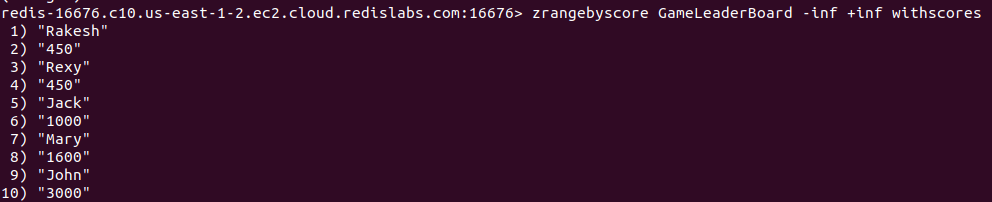
Since we got five set members, the returned value is 5 because the range is from the -infinity to +infinity which covers the whole set.

Let’s specify a range starting from 1000 to 3000.

| zcount GameLeaderBoard 1000 3000 |
| --- |

Let’s first inspect our set with the ZRANGEBYSCORE command as shown in the following.

| zrangebyscore GameLeaderBoard -inf +inf withscores |
| --- |



There are three members within the range of 1000 to 3000. Since the 1000 and 3000 are inclusive by default, the above ZCOUNT command should return 3.



Let’s use the ‘(’ character to exclude 1000 and 3000 scores with the same example shown above.

| zcount GameLeaderBoard (1000 (3000 |
| --- |



Since the 1000 and 3000 scores are excluded, the only left member is “Mary” with a 1600 score. Hence, the returned count is 1.

Whenever you need to count the members between a range of scores, it is recommended to use the ZCOUNT command which is immensely faster.

# Conclusion

To summarize, the ZCOUNT command is used to count the number of elements in a given range of score values. It has an O(log(N)) time complexity. As discussed, it can be used with minimum and maximum values to define a range of scores as in the ZRANGEBYSCORE command. The min and max values are inclusive by default. As shown previously, the ‘(’ can be used to exclude score values. Overall, the ZCOUNT command is simple to use and operates with great performance.