**Title:** Redis ZRANK

***Excerpt:*** Redis sorted sets can contain unique members in an ordered manner. By default, the sorted sets are sorted in ascending order by their score values. In addition, each member is assigned to a 0-based index which is called a member’s rank. The ZRANK command is used to get the rank of a given member.

**Permalink:** redis-zrank

**Category:** Redis

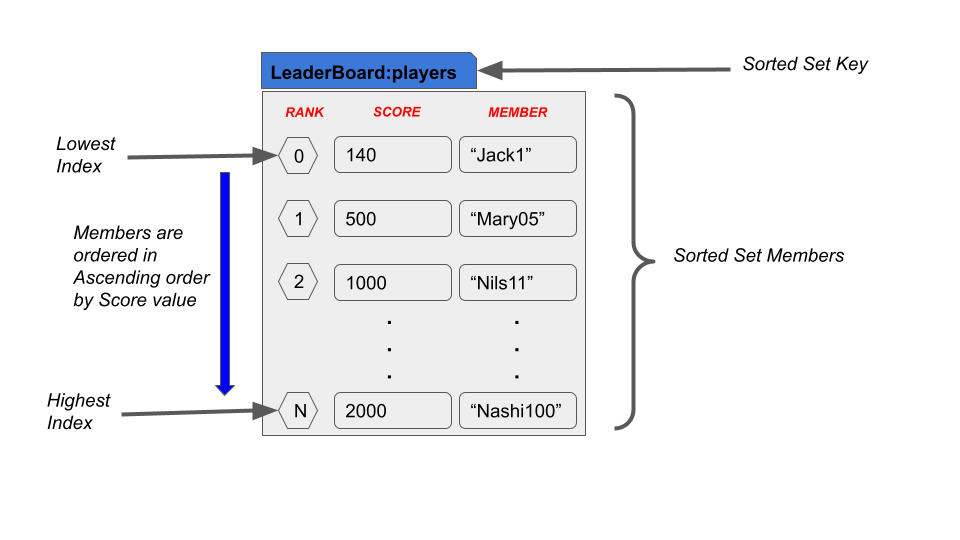
# 

# 

# 

# Properties of Sorted Set Members

Redis sorted sets can hold a collection of strings where each is called a set member. Each member is associated with a rank and score value. Rank is the index of a particular member in the sorted set stored at a given key. The score value is used to order the set members from the lowest to highest score value. Hence, a member with the lowest score value will be assigned to an index of 0. These properties make sorted sets to do a lot more tasks with high performance.



Various commands are available to operate on sorted sets. In some cases, you need to know the index or rank of a member in the Redis sorted set stored at a given key. The ZRANK is the ideal command to use and it will be discussed in the following section.

# The ZRANK Command

Now you have an idea of how the index value has been assigned to each member in a sorted set. The ZRANK command is used to obtain the rank of a specified member with a linear time complexity. The following is the syntax for the ZRANK command.

| ZRANK sorted\_set\_key member |
| --- |

***sorted\_set\_key*** : The key of the sorted set to which the member belongs to.

***member*** : The name of the member that you need to obtain the rank.

This command returns the rank or index of the specified member as an integer value if the member exists in the sorted set. Whenever the specified member or sorted set key doesn’t exist, the ZRANK command returns a string value *nil.*

## Use Case 01 - Get the Game Leaderboard Position of a Player

Let’s assume that we got an online game leaderboard implemented using Redis sorted set. Now we are going to display each player’s rank when they log into their user profile.

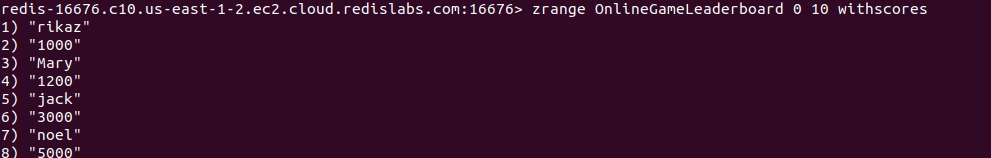
First, let’s create a dummy leaderboard *OnlineGameLeaderboard*, and add players to it using the ZADD command.

| zadd OnlineGameLeaderboard 3000 "jack" 1200 "Mary" 1000 "rikaz" 5000 "noel" |
| --- |

Now we will be calling the ZRANGE command to check whether the members have been added properly.

| zrange OnlineGameLeaderboard 0 10 withscores |
| --- |

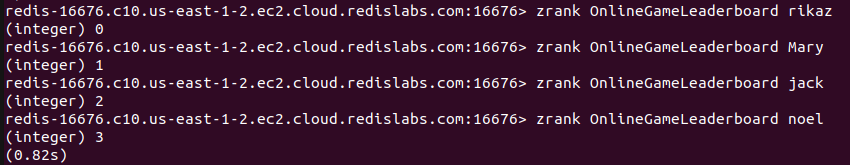
Output:



As expected, the members have been sorted in ascending order by their score values. Let’s say we need to obtain each of these members’ ranks. The ZRANK can be used as the following.

| zrank OnlineGameLeaderboard rikaz zrank OnlineGameLeaderboard Mary zrank OnlineGameLeaderboard jack zrank OnlineGameLeaderboard noel |
| --- |

Output:



As expected, the rank is returned for each member where the member “rikaz” has the lowest rank of 0. In addition, the member “noel” got the highest rank of 3.

Let’s check how the command behaves if you specify a non-existing member in the command.

| zrank OnlineGameLeaderboard Martin |
| --- |



The member that we have specified is not in the sorted set. Hence, it returned the string *nil.*

Next we will check the command for the non-existing key.

| zrank NonExistingKey Martin |
| --- |



The return value is again nil*.*

The ZRANK command doesn’t support multiple members as arguments. Hence if you provide multiple members it will throw an error. The ZRANK command is recommended to use when you need to obtain the index value of a sorted set member.

# Conclusion

In short, Redis sorted sets can hold string values that are called members. Each of these members is associated with two properties: score and rank. As discussed, the rank is based on 0 and incremented by one per member. Usually, the Redis sorted set members are ordered by their scores in ascending order. Hence, the member who has the lowest score will be assigned to the 0th index. The ZRANK command allows you to obtain this index value for a given member in the sorted set stored at a given key. This command operates with linear time complexity.