**Title:** Redis ZRANGE

***Excerpt:*** The ZRANGE command is used to query a range of members in a sorted set stored at a specified key. The range interval can be specified by score, index, or lexicographical values. In addition, the set elements can be ordered in descending order by passing the REV argument. Several other optional arguments such as BYSCORE, BYLEX, and LIMIT are available that extend the capabilities of the ZRANGE command.

**Permalink:** redis-zrange

**Category:** Redis

# 

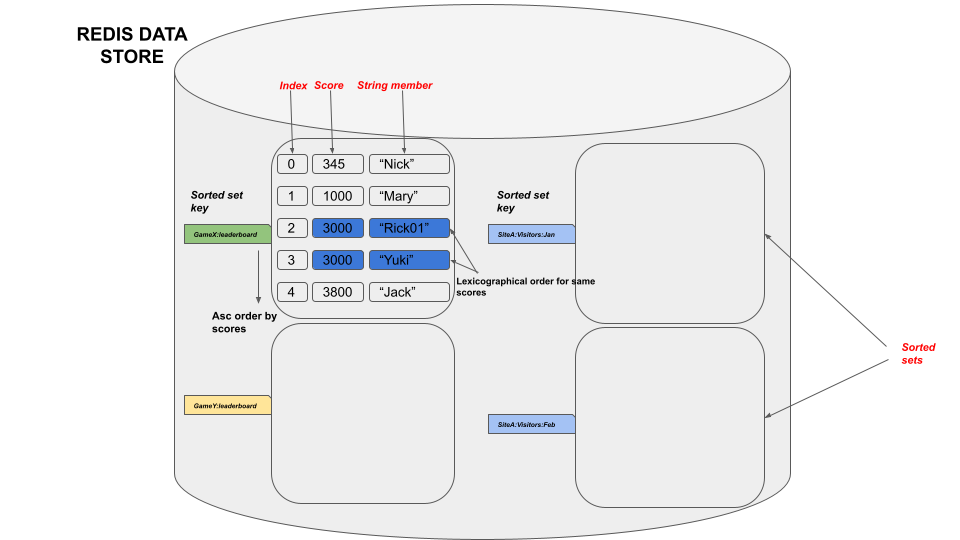
# 

# 

# 

# Redis Sorted Set Fundamentals

Redis sorted sets contain unique, non-repeating string type members in an ordered manner. By default, the sorted set members are ordered in ascending order by their score values. In addition, sorted sets maintain a zero-based index where members are ranked as 0,1,..., and so on. Whenever the scores are similar among two or more members, the ordering will be done by lexicographical order. Since the sorted set elements are already ordered, members can be queried by rank or score in a fast way. Furthermore, Redis sorted sets are based on a dual-ported data structure which makes add, read, remove and update operations execute in O(log(N)) time complexity.



In this guide, we will be focusing on the ZRANGE command that can be used to query a range of members by their scores, indexes, or lexicographically.

# The ZRANGE Command

The ZRANGE command is used to retrieve a range of members of a sorted set stored at a given key. This command allows you to query a range of elements based on different properties such as rank, score, or lexicographical order. By default, the ZRANGE command uses an index-based range. The basic syntax of the ZRANGE command is as follows.

## Syntax:

| ZRANGE sorted\_set\_key start end [BYSCORE | BYLEX] [REV] [LIMIT offset count] [WITHSCORES] |
| --- |

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

***start***: The starting value of the range query. This can be a score, rank, or lexicographical value.

***end***: The closing value of the range query. This can be a score, rank, or lexicographical value.

The above arguments are mandatory for this command and some optional arguments can be specified as follows.

***BYSCORE***: The range(start and end arguments) is specified based on the scores.

***BYLEX***: The range is specified based on the lexicographical values.

***REV***: This argument will reverse the order of returned members where the 0th index will hold the member with the highest score.

***LIMIT***: This argument will limit the number of returned members from a given offset.

***WITHSCORES***: By default, the ZRANGE command only returns members in a given range interval. When this argument is specified the output contains the score values associated with each member as well.

As mentioned, the ZRANGE command returns the list of members in a given range of a sorted set stored at a given key. If the WITHSCORES argument is specified, the associated score will be displayed. From Redis version 6.2.0, the ZREVRANGE, ZRANGEBYSCORE, ZREVRANGEBYSCORE, ZRANGEBYLEX, ZREVRANGEBYLEX commands can be replaced by the ZRANGE command.

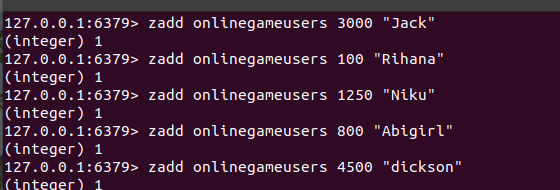
## Use Case 01 - Retrieving the top 3 users with the lowest experience points on a game leaderboard

Index based range

Let’s assume an online game where each user is rewarded with experience points based on the number of completed missions. Redis sorted set data structure is an ideal candidate to hold this information as shown in the following.

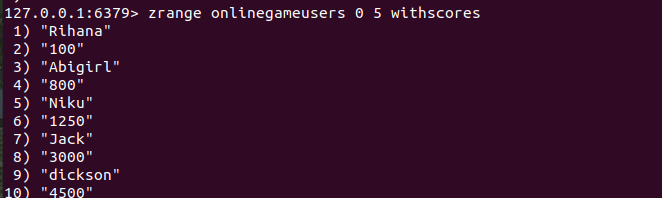
| zadd onlinegameusers 3000 "Jack" zadd onlinegameusers 100 "Rihana" zadd onlinegameusers 1250 "Niku" zadd onlinegameusers 800 "Abigirl" zadd onlinegameusers 4500 "dickson" |
| --- |

As you could see, the game experience points have been stored as the score of each member.



Let’s inspect the created sorted set stored at key ‘onlinegameusers’. Since the default range query is based on indexes, we will be specifying the minimum and maximum bounds as 0 and 5.

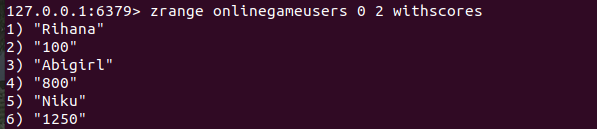
| zrange onlinegameusers 0 5 withscores |
| --- |



As expected, the members are sorted by scores in ascending order. Let’s query the top three users with the lowest experience points. Since the set is sorted in ascending order, it is fast and trivial to get the top three members with the lowest points as follows.

| zrange onlinegameusers 0 2 withscores |
| --- |

As discussed earlier, the sorted sets have zero-based indexes. Hence, the first three elements can be specified as an index range starting from 0 to 2. The output should display the top 3 elements with the lowest scores as follows.



## Use Case 02 - Retrieving users with more than 3000 experience points on a game leaderboard

In this scenario, we need to query all the users who got more than 1000 experience points. Since we are talking about score-based range, let’s use the *BYSCORE* optional argument to specify the range in score values.

| zrange onlinegameusers (3000 +inf byscore withscores |
| --- |

The range boundaries have been specified using scores. Since we need scores more than 3000, the score of 3000 has to be excluded. The ZRANGE command supports excluding scores by prefixing it with the ‘(’ character.

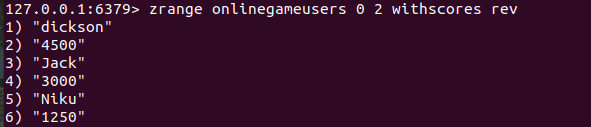


As expected, we got only one member whose experience point(score) is more than 3000.

## Use Case 03: Getting the top 3 users whose having the highest experience points

By default, the sorted set orders its elements in ascending order. Hence, the member with the lowest score is positioned at the 0th index. To get the top 3 users with the highest scores, we have to pass the REV argument to the ZRANGE command as follows. It will order the sorted set in descending order and query the index range starting from 0 to 2.

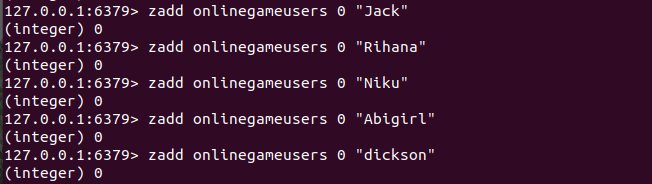
| zrange onlinegameusers 0 2 withscores rev |
| --- |



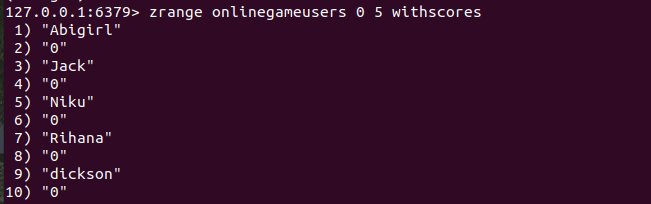
## Use Case 04 - Retrieving users whose name starts with “L” or following letter

The ZRANGE command allows you to retrieve a range of members based on lexicographical values if the scores are similar among the members. Let’s reset all the scores to 0 as follows.

| zadd onlinegameusers 0 "Jack" zadd onlinegameusers 0 "Rihana" zadd onlinegameusers 0 "Niku" zadd onlinegameusers 0 "Abigirl" zadd onlinegameusers 0 "dickson" |
| --- |



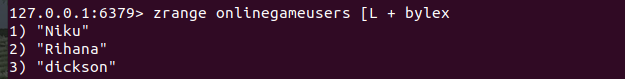
As expected, the experience points(scores) have been reset to 0 as shown in the following.



Let’s use the *BYLEX* argument to retrieve members whose name starts with the letter ‘L’ or following letter.

| zrange onlinegameusers [L + bylex |
| --- |

The range boundaries have been specified using string values(lexicographical). The output should be all the members whose name starts with the letter ‘L’ or the following letter. Ideally, the “Niku”, “Rihana” and “dickson” should be the output. The lowercase letter “d” comes after the uppercase “L” according to their ASCII values.



# Conclusion

In summary, the ZRANGE command is used to fetch a range of members stored in a sorted set stored at a given key. As mentioned, this command allows you to query a range of set elements by the scores, ranks, or lexicographical values. Also, the ZRANGE command can be used with the REV argument to reverse the sort order. Overall, several capabilities are packed into this command and it has become a replacement for several commands such as ZRANGEBYLEX, ZRANGEBYSCORE, etc.