**Title:** Redis ZREVRANGE

**Excerpt:** The ZREVRANGE command is used to query the members in reversed order from a Redis sorted set stored at a given key. It specifies the range by index values. The index can be both positive and negative where the negative values denote offsets starting from the last element of the sorted set.

**Permalink:** redis-zrevrange

**Category:** Redis

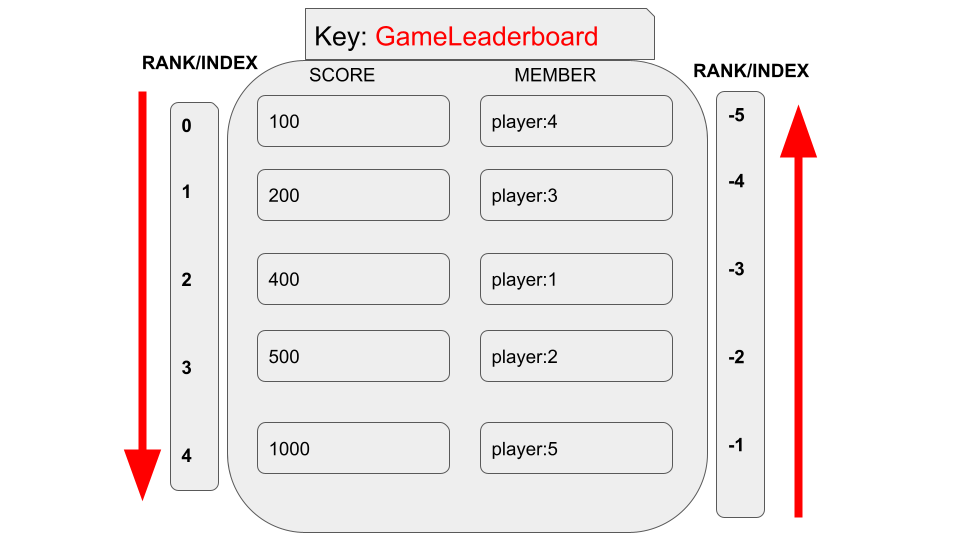
# 

# 

# 

# Introduction to Redis Sorted Set Data Structure

The sorted set data structure is a special data type introduced by Redis where the stored elements are ordered in ascending order. The members stored inside a sorted set are more like field-value pairs of a Redis hash. Each member is associated with a score and rank where the querying can be done by both properties. The rank or index is assigned to each member starting from zero, the next element is one, and so on. Also, the index can be a negative value. The -1 rank indicates the last member of the sorted set. The element next to the last element is denoted by the -2 index and so on.



The ZREVRANGE and ZRANGE commands are capable of retrieving members between a specified range of indexes of a sorted set stored at a given key. These commands accept both the positive and negative index values when querying the members of a given range. In this guide, we will be focusing on the ZREVRANGE command and its practical usage.

# Redis ZREVRANGE Command

The ZREVRANGE command is used to query the members between specified minimum and maximum index values of a sorted set stored at a given key. The members are returned in descending order based on their score value. Whenever the multiple members have the same score, the elements will be ordered lexicographically.

## Syntax:

| ZREVRANGE sorted\_set\_key start\_index end\_index [WITHSCORES] |
| --- |

***sorted\_set\_key:*** This is the unique ID that identifies the sorted set in the Redis data store

***start\_index:*** The starting index of the range specified

***end\_index:*** The ending index of the range specified

***[WITHSCORES]:*** This is an optional argument when specified members will be returned with their scores.

The command will return an array of members stored between the specified index range. If the ***WITHSCORES*** argument has been specified, the scores will be included in the output.

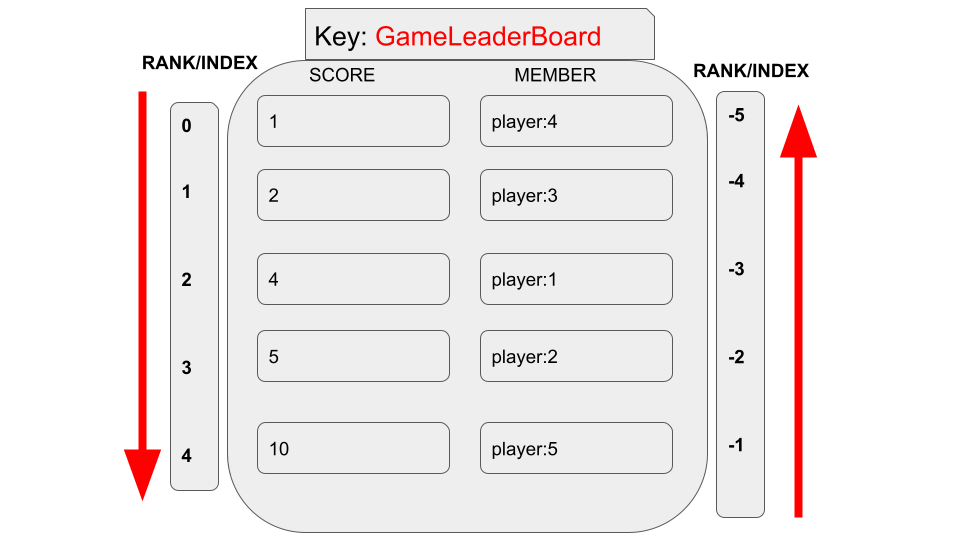
## Example - Return the Players Who have Completed the Most Number of Levels in an online game

Let’s assume an online community game where each player is ranked based on the number of completed levels. At the end of each week, the top three players get bonus gifts. Hence, we need to query the top 3 players who have completed the most number of levels in the game. We can assign the number of completed levels as the score of a member stored in a Redis sorted set.

First we will add a few players to a sorted set called *GameLeaderBoard.*

| **zadd** **GameLeaderBoard** 4 **player**:1 **zadd** **GameLeaderBoard** 2 **player**:3 **zadd** **GameLeaderBoard** 5 **player**:2 **zadd** **GameLeaderBoard** 1 **player**:4 **zadd** **GameLeaderBoard** 10 **player**:5 |
| --- |

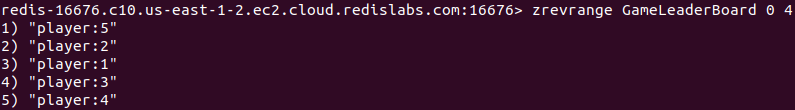
In the above example, *player:1* has completed 4 levels and *player:5* has completed 10 levels, and so on. The above information regarding a player has been mapped to a sorted set as shown in the following.



Let’s use the ZREVRANGE command to query all the members whose completed level count is higher and goes to the top of the list.

| zrevrange GameLeaderBoard 0 4 |
| --- |

As in the above illustration, the sorted set indexes spread from 0 to 4. Hence, we specified the start and end indexes as 0 and 4 respectively which should ideally return all the members belonging to the sorted set identified by the key *GameLeaderBoard.* As you can see it has ordered the members in a way that the top most players have the most number of completed levels in the game.

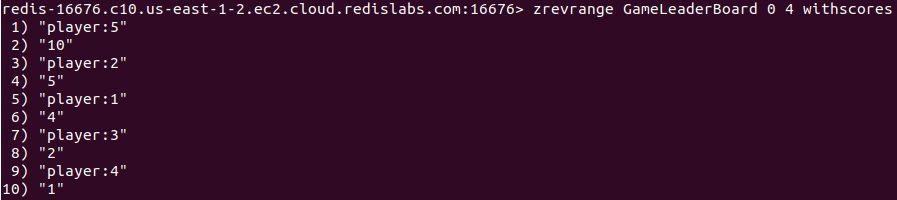


### Query Members with their Scores

The ZRANGE command can be used with the WITHSCORES optional argument to display the score values of each player in the above example.

| zrevrange GameLeaderBoard 0 4 withscores |
| --- |

Output:

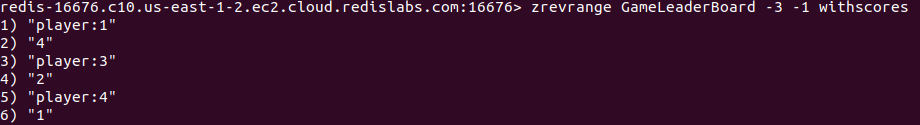


### Query Members Starting from the End of the Sorted Set

In all the above examples, we have used the 0-based index that is counting from the start of the sorted set. We can specify the index range starting from the end of the sorted set as well. Let’s say we need to query the players whose scores are the lowest using the ZREVRANGE command.

| zrevrange GameLeaderBoard -3 -1 withscores |
| --- |

In this case, the end index has been specified as -1 which means the last element. The start index is -3. Hence, it will query members up to the 3rd member from the end of the sorted set. This is very clear if you use the above figure which shows how the indexes can be used in both ways.



## The ZRANGE Command Instead of ZREVRANGE

The ZREVRANGE command has been deprecated from the Redis version 6.2.0. Hence, it is recommended to use the ZRANGE command with the optional REV argument which behaves the same as the ZREVRANGE command.

# Conclusion

To summarize, the ZREVRANGE command is used to query members between a specified minimum and maximum indexes from a Redis sorted set stored at a given key. It retrieves the members stored between the specified index values in reverse order. Apart from the reversed ordering of members, this command behaves similar to the ZRANGE command. As stated above, it has a logarithmic time complexity which makes the command pretty fast to use in real-time applications efficiently. Hence, the ZREVRANGE command is ideal to use whenever you need to retrieve elements in descending order from a sorted set.