**Title:** Redis XINFO

**Excerpt:** Redis *XINFO* command is a container command that supports several sub-commands such as XINFO STREAM, XINFO GROUPS, and XINFO CONSUMERS. Each of these sub-commands can be used to query information on a specific stream stored at a specified key and its consumer groups with consumers.

**Permalink:** redis-xinfo

**Category:** Redis

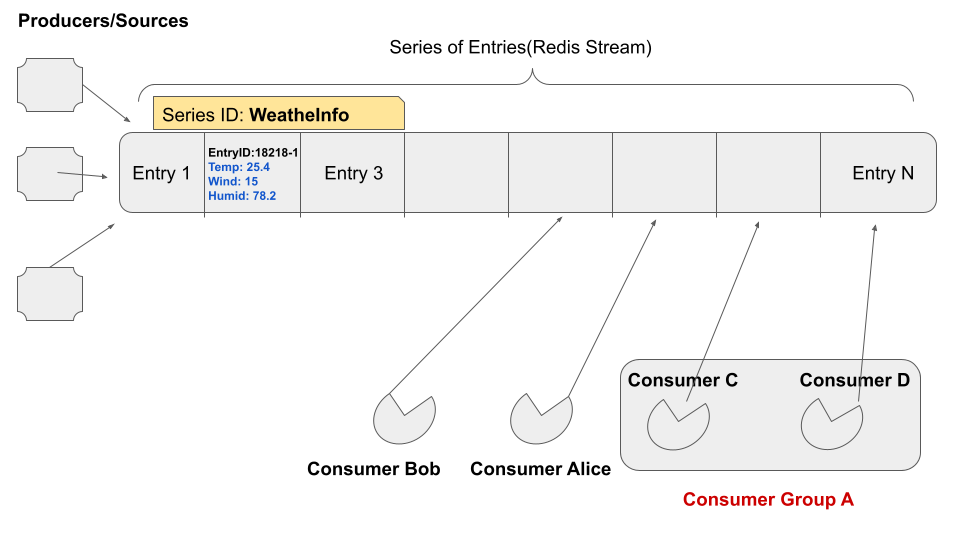
# 

# 

# 

# Redis streams

Redis streams can hold a series of events generated via multiple producers or sources. Usually, a Redis stream has a key that determines each stream uniquely within the data store. A stream carries several entries where each entry is a key-value pair. Redis streams have consumer groups attached to them. Each consumer group might have multiple consumers reading from the stream. The below figure summarizes the high-level picture of a Redis stream and related entities.



There is a lot of information related to the Redis stream, consumer groups and consumers. The above illustration summarizes that information up to some extent. Redis allows its users to query all these details using the XINFO command.

# Information on Redis streams - XINFO command

The XINFO command is a high-level container command to query all the details related to the stream, its consumer groups, and consumers. This command supports several sub-commands to be called as shown in the following.

## Syntax

| XINFO <sub-**command**> |
| --- |

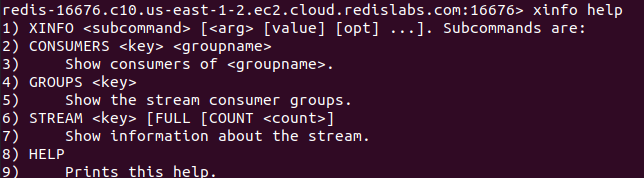
# XINFO subcommands

There are four main subcommands available to use with the XINFO command. The most basic command is the ‘help’ command which lists down all the other subcommands available. Furthermore, it describes the purpose of each subcommand with its syntax.

## Syntax

| XINFO HELP |
| --- |

Output:



## XINFO STREAM

The STREAM sub command will display the details related to the stream stored at the specified key.

### Syntax

| XINFO STREAM <key> [FULL [COUNT count]] |
| --- |

This command will display several details related to the specified stream such as length, groups attached, last-generated-id, first-entry, last-entry, etc.

### Example - Information on location rating information with Redis streams

Let’s add a new stream of id ***locationRatingStream.*** Furthermore, some entries will be added as shown in the following.

| xadd locationRatingStream \* stars 5 comment great xadd locationRatingStream \* stars 3 comment good xadd locationRatingStream \* stars 1 comment worst |
| --- |

Let’s add two consumer groups to the ***locationRatingStream*** as shown in the following.

| xgroup create locationRatingStream canadagroup $ xgroup create locationRatingStream brazilgroup $ |
| --- |

Finally, we will be reading from the consumer group via a consumer ***Alex.***

| xreadgroup group canadagroup Alex streams locationRatingStream > |
| --- |

This would create a new consumer ***Alex*** as well.

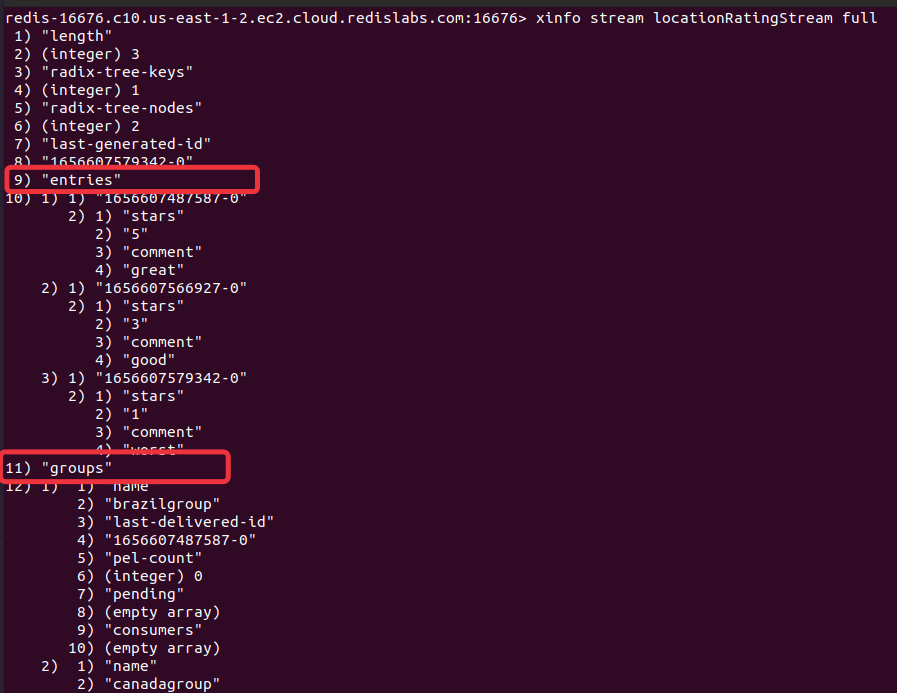
Now we can try out the ***XINFO STREAM*** sub command as shown in the following.

| xinfo **stream** locationRatingStream |
| --- |

Output:



We can use the ***FULL*** parameter to see all the entries and groups stored in the stream. Furthermore, this would show two new properties *entries* and *groups* as shown in the following.



The ***COUNT*** is another optional parameter that we can specify with the ***FULL*** parameter. It will limit the number of rows returned per stream and PEL entry arrays.

## XINFO GROUPS

The XINFO GROUPS sub command displays all the consumer groups attached to a stream stored at a specified key. This command displays several details such as the name of the group, number of consumers per group, length of the PEL, last-delivered-id, lag, etc.

### Syntax

| XINFO GROUPS <**stream\_key**> |
| --- |

Let’s use the previously created stream ***locationRatingStream*** to check the information of its attached consumer groups.

| xinfo groups locationRatingStream |
| --- |

Output:



As expected, the XINFO GROUPS command lists two consumer groups related to the stream stored at key ***locationRatingStream.*** Furthermore, it lists the number of consumers and the length of the PEL for both groups.

## XINFO CONSUMERS

Each consumer group contains one or more consumers. Hence, the XINFO CONSUMERS sub-command shows information related to all the consumers in a consumer group where the group belongs to a stream identified by the specified key. This command lists the following information about consumers.

**name**: The name of the consumer.

**pending**: The number of messages that are not acknowledged by the consumer.

**idle**: The number of milliseconds passed since the last active interaction with the server by the consumer.

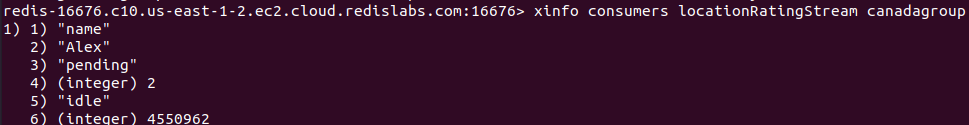
### Syntax

| XINFO CONSUMERS <**stream\_key**> <**consumer\_group**> |
| --- |

Let’s use the previously created stream ***locationRatingStream*** and the consumer group ***canadagroup*** to get the details about its consumers.

| xinfo consumers locationRatingStream canadagroup |
| --- |

Output:



As expected, the consumer group ***canadagroup*** has one consumer called ***Alex*** and it has two messages yet to be acknowledged. In addition, the consumer had no interaction with the Redis server for 4550962 milliseconds.

# Conclusion

Redis streams can hold a sequence of events generated by multiple sources. In addition, each stream can have several consumer groups and consumers reading from it. There is a lot of valuable information available about those entities. Hence, the Redis provides a container command XINFO to use with different sub-commands for various purposes. The XINFO HELP command lists all the sub-commands available such as XINFO STREAM, XINFO GROUPS, and XINFO CONSUMERS. Each of these sub-commands provides a large amount of information about a specific stream, its consumer groups, and related consumers.