|  |
| --- |
| Close-up image showing the leaf-sides of two oversized books side-by-side on a bookshelf, with additional books in soft focus background |
| NoSQL Project  Redis Cache – Advanced Databases |
| |  |  |  | | --- | --- | --- | | Reonne Benoy | 5/13/20 | Athlone institute of Technology | |

Table of Contents

[Introduction 2](#_Toc40297495)

[Installation 2](#_Toc40297496)

[Running the Server and Client using Command Prompt 3](#_Toc40297497)

[Questions and Solutions 6](#_Toc40297498)

[Redis using Docker Toolbox 10](#_Toc40297499)

[Other datatypes - Import and Use JSON files in Redis 10](#_Toc40297500)

[Applications 11](#_Toc40297501)

[Conclusion 11](#_Toc40297502)

[References 12](#_Toc40297503)

[Appendix 12](#_Toc40297504)

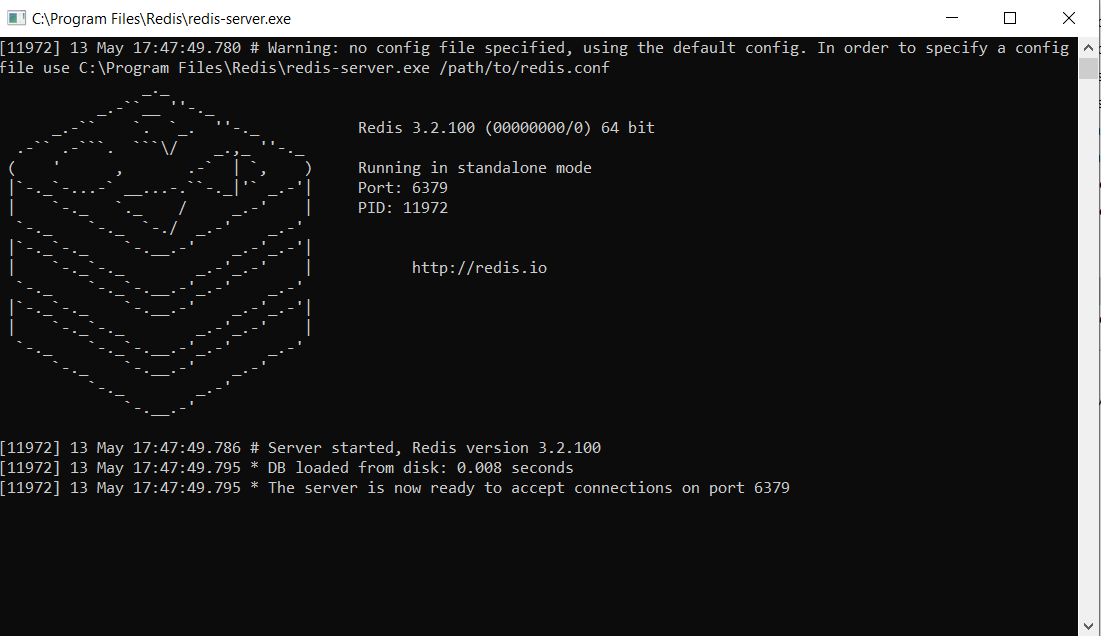
Redis

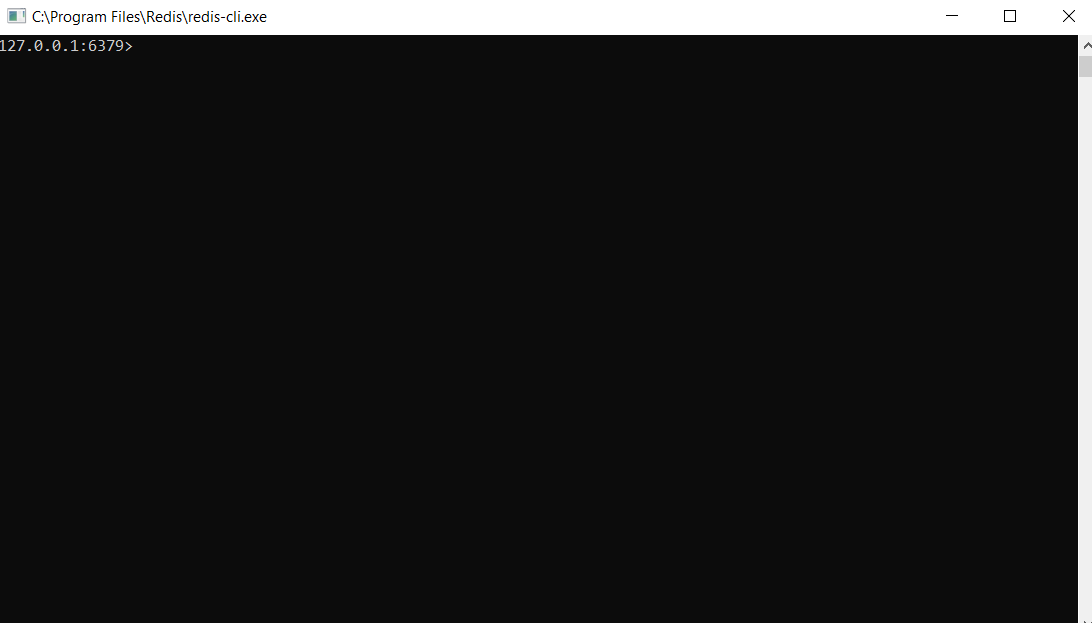
# Introduction

Redis is an open source, in-memory data structure store. It is used as a database, cache and message broker. The data structures supported by Redis are strings, hashes, lists, sets, bitmaps, geospatial indexes. It enables statelessness for an for an applications’ processes while reducing duplication of data or requests to external data sources.

# Installation

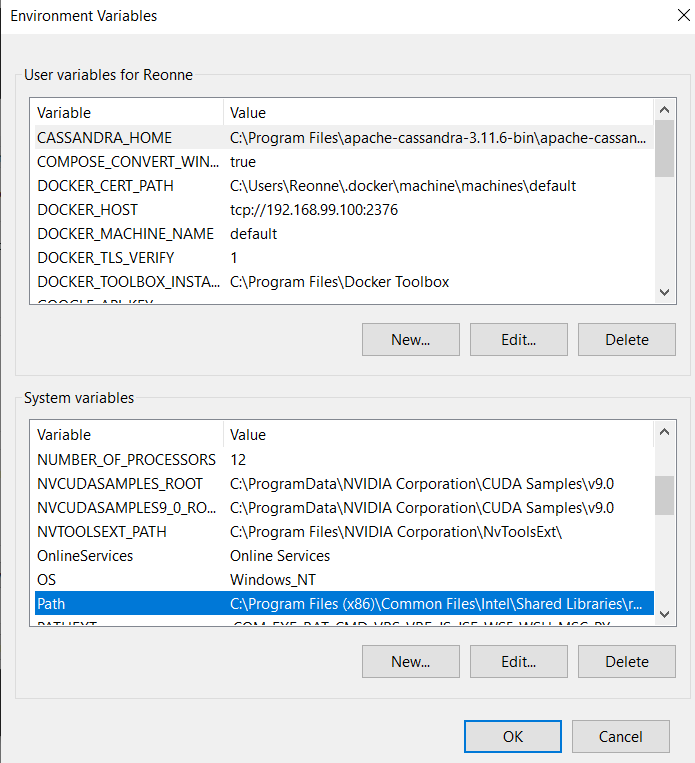
1. Download and install Redis. Use the github link for easy download: <https://github.com/microsoftarchive/redis/releases/tag/win-3.2.100>
2. Extract the zip folder  to the preferred location.
3. From the extracted files, we see both the redis-server and redis-cli. As the name suggests, redis-server is the server that needs to be established first before we can use a client.
4. Run redis-server first, then run client



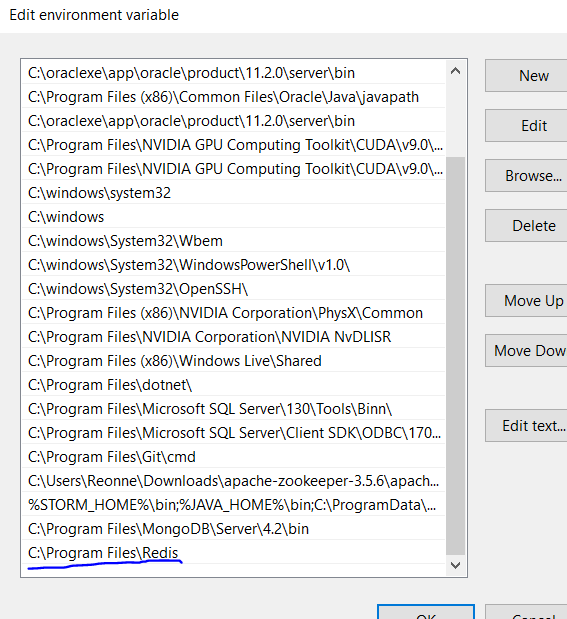


## Running the Server and Client using Command Prompt

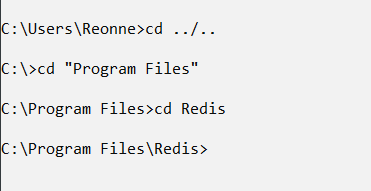
1. Set path for the environment variable.
2. Select Path to add path to the list.



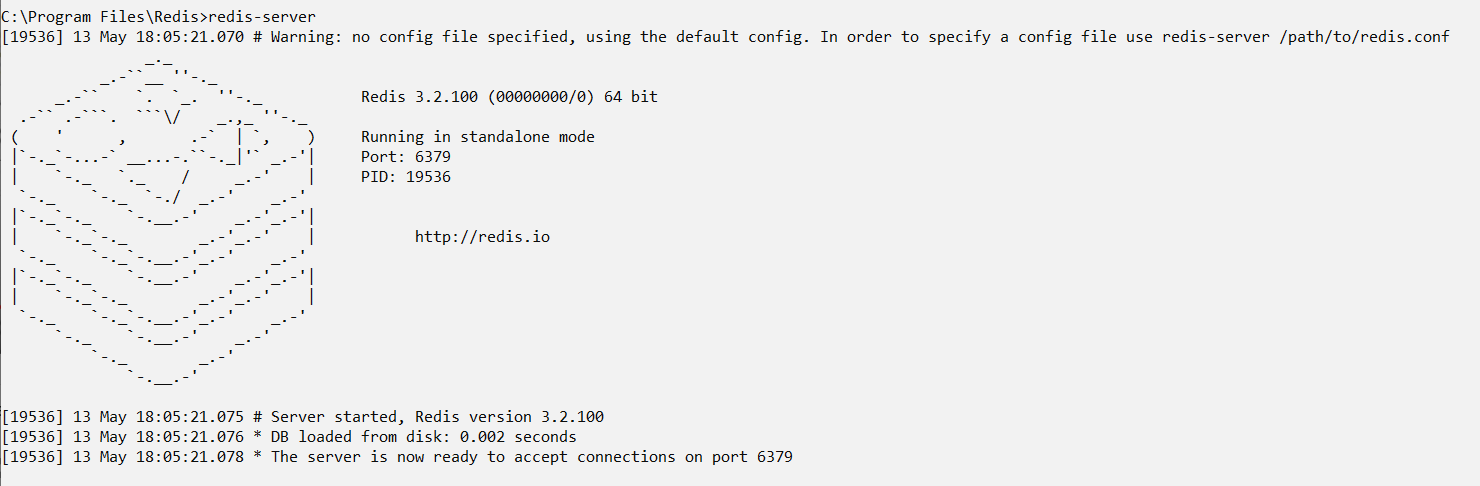
1. After selecting Path, add a new path to the list.



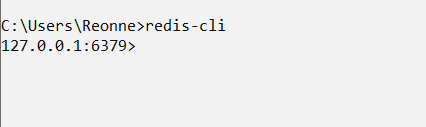
1. Click on OK.
2. Now open Command Prompt.
3. Open the path so that the programs can be run.



1. Run the redis-server.



1. Run the redis-cli in a separate terminal.



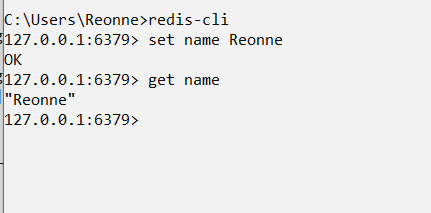
1. Now we have successfully established the client and server connection using redis cache.

# Questions and Solutions

1. **Enter a name and get the response from the client-server structure of Redis.**

**Solution:**

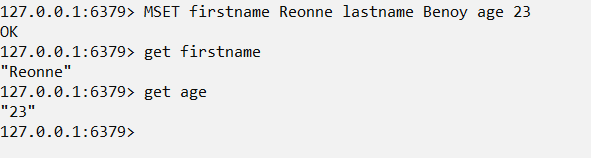
* *set name Reonne*
* *get name*



1. **Store the full name and age and retrieve it.**

**Solution:**

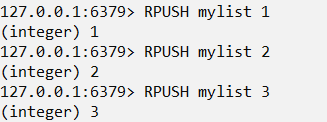
* *MSET firstname Reonne lastname Benoy age 23*
* *get firstname*
* *get age*



1. **Create a list of integers.**

Solution:

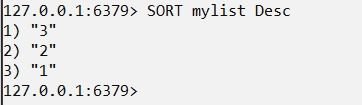
* *LPUSH mylist 1*
* *LPUSH mylist 2*
* *LPUSH mylist 3*



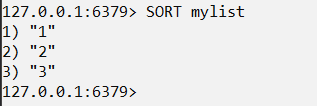
1. **Sort the created list in ascending and descending order.**

Solution:

* *SORT mylist Desc*



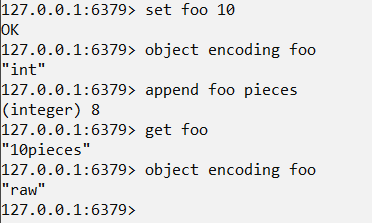
* *SORT mylist*



1. **Change the datatype of the keys previously entered.**

Solution:

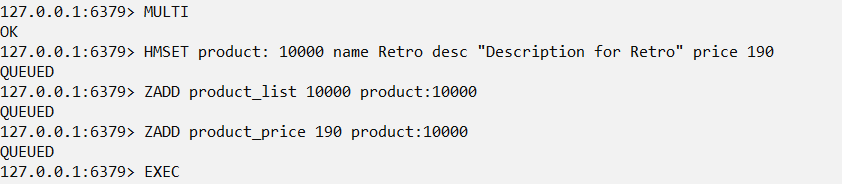
* *set foo 10*
* *object encoding foo*
* *append foo pieces*
* *get foo*
* *object encoding foo*



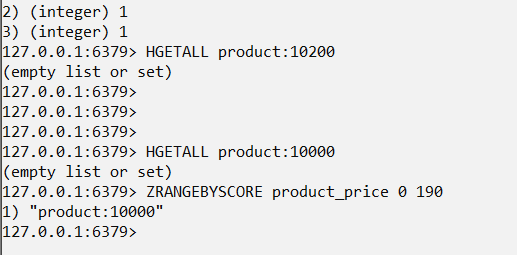
1. **Create an SQL type data set using the keys.**

Solution:

* *MULTI*
* *HMSET product: 10000 name Retro desc “Description for Retro” price 190*
* *ZADD product\_list 10000 product:10000*
* *ZADD product\_price 190 product:10000*
* *EXEC*

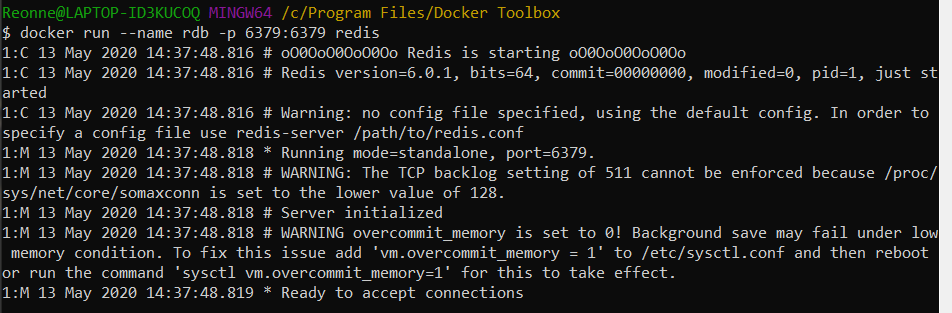


* HGETALL product:10000
* ZRANGEBYSCORE product\_price 0 190

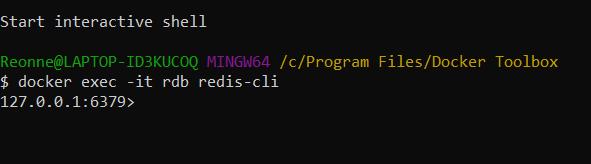


# Redis using Docker Toolbox

1. Install and open Docker Toolbox Terminal.
2. Run the command: docker run –name rdb -p 6379:6379 redis, where 6379 is the port number.

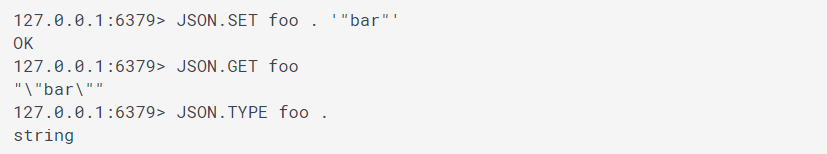


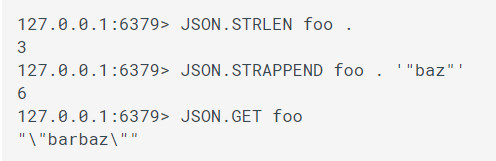
1. Open another Docker Toolbox Terminal.
2. Run the following command: docker exec -it rdb redis-cli, where rdb is the name.



# Other datatypes - Import and Use JSON files in Redis

We need another application for this purpose. We also need to use another port number for running the commands so as to avoid clashing with the local client at port 6379:6379. For this purpose, we need another port of let’s say, 7000:6379 and use the JSON.SET command lines.





# Applications

1. Session Cache: Advantages of Redis over Memcached is that Redis offers persistence.
2. Full Page Cache: Redis provides a very easy FPC platform to operate in.
3. Queues: The in-memory storage engine to do list and set operations makes it a real strong platform to use for a message queue.
4. Leaderboards: Increment and decrement operations are done at efficient capacity since it is in-memory.
5. Data mapping: Redis uses Dataframes to automatically map the tables to Redis data structures. Redis does this by its collaboration with Apache Spark and Spark-Redis library. Using the DataFrame APIs from the Spark-Redis library, one could insert, update and query data using SQL commands and the data is internally mapped to Redis data structures.

# Conclusion

Redis Cache is an in-memory data structure store and we can see the difference in its applications. All the command lines are straight forward and does not involve intricate commands for data access. Redis is fairly easy to install and run and its collaboration with other database softwares and applications can be easily understood. Redis, however, lacks where other databases such as HBase and Cassandra prevails. Since it is, in-memory, the data isn’t stored for further use in the future. Also, utilizing other file formats are a bit tedious in Redis.

# References

*a-3-2-installing-redis-on-window*. (n.d.). Retrieved from redislabs.com: https://redislabs.com/ebook/appendix-a/a-3-installing-on-windows/a-3-2-installing-redis-on-window/

*commands*. (n.d.). Retrieved from https://oss.redislabs.com: https://oss.redislabs.com/redisjson/commands/

*lset*. (n.d.). Retrieved from https://redis.io: https://redis.io/commands/lset

*quickstart*. (n.d.). Retrieved from https://redis.io: https://redis.io/topics/quickstart

*RedisJSON*. (n.d.). Retrieved from https://github.com: https://github.com/RedisJSON/RedisJSON

*redis-server-port-already-in-use*. (n.d.). Retrieved from stackoverflow.com: https://stackoverflow.com/questions/7417232/redis-server-port-already-in-use

*rejson*. (n.d.). Retrieved from https://hub.docker.com: https://hub.docker.com/r/redislabs/rejson/

# Appendix

Youtube video 1: <https://www.youtube.com/watch?v=Ig9C4TW6xJc>

Youtube video 2: <https://www.youtube.com/watch?v=gmIepVEUE2M>

Reference video 1: <https://www.youtube.com/watch?v=AM5Qblt5_SQ>

Reference video 2: <https://www.youtube.com/watch?v=xbs3wUiZHO0&list=PLTgRMOcmRb3Mi17Kinrjft-cGcH_TVBPM&index=1>