

Alex Ojemann  
Homework 7 - Redis

Pull Docker Image of Redis

```
(base) cu-biot-3-10:~ admin$ docker pull redis
Using default tag: latest
latest: Pulling from library/redis
1fe172e4850f: Pull complete
6fbcd347bf99: Pull complete
993114c67627: Pull complete
90ee703e9ece: Pull complete
80d8b53e83a7: Pull complete
a09780893e15: Pull complete
Digest: sha256:96c3e4dfe047ba9225a7d36fc92b5a5cff9e047daf41a1e0122e2bd8174c839e
Status: Downloaded newer image for redis:latest
docker.io/library/redis:latest
```

I will use the same network "database-network" that I already created for my Cassandra assignment.

Start Redis Server Instance and Connect to Server From CLI Container:

```
(base) cu-biot-3-10:~ admin$ docker run --name redis-server --network database-network -d redis:latest
741ddb7e90fa6776cf77c8b07b609414005c277be5a57b3e3911de7dd5aad221
(base) cu-biot-3-10:~ admin$ docker run -it --network database-network --rm redis redis-cli -h redis-server
```

Get and Set:

```
redis-server:6379> SET "Batman" "Joker"
OK
redis-server:6379> GET "Batman"
"Joker"
redis-server:6379> exit
```

FLASK REST SERVER:

BUILD FLASK REST SERVER DOCKER IMAGE:

```
(base) cu-biot-3-10:~ admin$ cd Desktop
(base) cu-biot-3-10:Desktop admin$ cd FlaskRest-Server
(base) cu-biot-3-10:FlaskRest-Server admin$ pwd
/Users/admin/Desktop/FlaskRest-Server
```

```
(base) cu-biot-3-10:FlaskRest-Server admin$ docker build . -f DockerFile -t rest_server:latest
[+] Building 11.7s (11/11) FINISHED
=> [internal] load build definition from DockerFile
=> => transferring dockerfile: 797B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3.8-alpine
=> [1/6] FROM docker.io/library/python:3.8-alpine@sha256:45168bf789428a963f86a03da1039b1845d8bcb81d0d0ca031e4d6fc24394e72
=> => resolve docker.io/library/python:3.8-alpine@sha256:45168bf789428a963f86a03da1039b1845d8bcb81d0d0ca031e4d6fc24394e72
=> => sha256:6cb9c3e38f460480c4268813fabc5eff836c6429af0e1f5ab707d05993375ac7 11.81MB / 11.81MB
=> => sha256:45168bf789428a963f86a03da1039b1845d8bcb81d0d0ca031e4d6fc24394e72 1.65kB / 1.65kB
=> => sha256:88607509b9ff50b81b0109dc02d9549a5af47a5a61b28cd0adee2e1e5e4a489b 1.37kB / 1.37kB
=> => sha256:f185fb08e43cd610f87b51d17133fd60f99685c0f75a57c3177991557012b8e9 7.07kB / 7.07kB
=> => sha256:df9b9388f04ad6279a7410b85cedfdb2208c0a003da7ab5613af71079148139 2.81MB / 2.81MB
=> => sha256:a1ef3e6b7a02c0ce1d7d3ee96f273c2c5074b3f127082b4cd3f629f82097e1a8 667.03kB / 667.03kB
=> => extracting sha256:df9b9388f04ad6279a7410b85cedfdb2208c0a003da7ab5613af71079148139
=> => sha256:01f10cc3d53d4d3f81ae4a90c16016570e12b6a21e82db77b64a4a470824d509 231B / 231B
=> => sha256:8722ec4741b65d430f45effce7f072a13b52eabfd6de57d622083188d3983adb 2.87MB / 2.87MB
=> => extracting sha256:a1ef3e6b7a02c0ce1d7d3ee96f273c2c5074b3f127082b4cd3f629f82097e1a8
=> => extracting sha256:6cb9c3e38f460480c4268813fabc5eff836c6429af0e1f5ab707d05993375ac7
=> => extracting sha256:01f10cc3d53d4d3f81ae4a90c16016570e12b6a21e82db77b64a4a470824d509
=> => extracting sha256:8722ec4741b65d430f45effce7f072a13b52eabfd6de57d622083188d3983adb
=> [internal] load build context
=> => transferring context: 3.72kB
=> [2/6] COPY ./requirements.txt /redis/requirements.txt
=> [3/6] COPY ./Rest-Redis-Server.py /redis/Rest-Redis-Server.py
=> [4/6] WORKDIR /redis
=> [5/6] RUN pip install -r requirements.txt
=> [6/6] COPY . /app
=> exporting to image
=> => exporting layers
=> => writing image sha256:6d95a55ef7ca1f4593cc80a96c5300a36f63d1418f6b888cc127dd7f2d2aa53
=> => naming to docker.io/library/rest_server:latest
```

## CREATE AN INSTANCE FROM THE IMAGE CREATED:

```
(base) cu-biot-3-10:FlaskRest-Server admin$ docker run --network database-network -p 6999:6999 -d rest_server:latest
8361da3cf6f536e24167b04ad0306c03b8c060cb3c2a1a7177aaa1d591936f83
```

## CHECK IF THE REST SERVER IS UP AND RUNNING:

```
(base) cu-biot-3-10:FlaskRest-Server admin$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
8361da3cf6f5	rest_server:latest	"python3 Rest-Redis-..."	9 seconds ago	Up 5 seconds	0.0.0.0:6999->6999/tcp, :::6999->6999/tcp	mystifying_blackwell

```
(base) cu-biot-3-10:FlaskRest-Server admin$ docker logs 8361da3cf6f5
* Serving Flask app 'Rest-Redis-Server' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on all addresses (0.0.0.0)
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://127.0.0.1:6999
* Running on http://172.27.0.4:6999 (Press CTRL+C to quit)
```

## Rest APIs: Get and Set:

```
(base) cu-biot-3-10:FlaskRest-Server admin$ curl http://localhost:6999/redis/get/Batman
{"key":"Batman","value":"Joker"}
(base) cu-biot-3-10:FlaskRest-Server admin$ curl -i -H "Content-Type: application/json" -X POST -d '{"key":"Batman","value":"Joker"}'
curl: no URL specified!
curl: try 'curl --help' or 'curl --manual' for more information
(base) cu-biot-3-10:FlaskRest-Server admin$ curl -i -H "Content-Type: application/json" -X POST -d '{"key":"Batman","value":"Joker"}' http://localhost:6999/redis/set
HTTP/1.1 201 CREATED
Server: Werkzeug/2.1.2 Python/3.8.13
Date: Mon, 02 May 2022 01:00:45 GMT
Content-Type: application/json
Content-Length: 19
Connection: close

{"data":"Success"}
```