EXPERIMENT – 13

WORKING ON DOCKER

1. Use $docker images command to check the docker images available in your machine.

```
$ docker images
REPOSITORY            TAG         IMAGE ID          CREATED          SIZE
ubuntu                latest      16508e5c265d      2 years ago      84.1MB
redis                 latest      4e8db158f18d      2 years ago      83.4MB
weaveworks/scope      1.9.1       4b07159e407b      2 years ago      68MB
alpine                latest      11cd0b38bc3c      2 years ago      4.41MB
```

2. Use $docker search <imagename> command to search for the images available for download.

$docker search redis – For searching Redis Images

```
$ docker search redis
NAME                          DESCRIPTION                                 STARS        OFFICIAL
  AUTOMATED
redis                         Redis is an open source key-value store that…  8740       [OK]

bitnami/redis                 Bitnami Redis Docker Image                  167
  [OK]
sameersbn/redis                                                           82
  [OK]
grokzen/redis-cluster         Redis cluster 3.0, 3.2, 4.0, 5.0, 6.0       72

rediscommander/redis-commander    Alpine image for redis-commander - Redis man…  47
  [OK]
kubeguide/redis-master        redis-master with "Hello World!"            33

redislabs/redisearch          Redis With the RedisSearch module pre-loaded…  29

redislabs/redis               Clustered in-memory database engine compatib…  27

oliver006/redis_exporter        Prometheus Exporter for Redis Metrics. Supp…  22

redislabs/rejson              RedisJSON - Enhanced JSON data type processi…  22

arm32v7/redis                 Redis is an open source key-value store that…  21
```

3. Use $docker pull <imagename> to download a Docker Image

$docker pull arm32v7/redis

```
$ docker pull arm64v8/redis
Using default tag: latest
latest: Pulling from arm64v8/redis
4e6164a63b7b: Pull complete
66a6505e9bfd: Pull complete
fb4766a9fc53: Pull complete
56b3977748bf: Pull complete
a858349766d4: Pull complete
b5df6354e952: Pull complete
Digest: sha256:2f6d224aec3045a18ba7daa5920fe10bf0ae973e8309d57d922892322a950f04
Status: Downloaded newer image for arm64v8/redis:latest
```

arm32v7/redis image has been downloaded

```
$ docker images
REPOSITORY          TAG        IMAGE ID        CREATED         SIZE
arm64v8/redis       latest     c123aa3694ab    13 days ago     98.5MB
ubuntu              latest     16508e5c265d    2 years ago     84.1MB
redis               latest     4e8db158f18d    2 years ago     83.4MB
weaveworks/scope    1.9.1      4b07159e407b    2 years ago     68MB
alpine              latest     11cd0b38bc3c    2 years ago     4.41MB
```

4. Use $docker run -d <imagename> to run the docker image and -d is used to keep it running in the background. Same images can also run multiple times.

$docker run -d redis

```
$ docker run -d redis
bd0eb5c13cffc76706e8e7b3cd841d5a9a741a3a3ee5c1b22927981f1ba538f7
```

5. Use $docker ps to see all the images that are only running currently in your machine.

```
$ docker ps
CONTAINER ID       IMAGE          COMMAND             CREATED          STATUS          PORTS          NA
MES
bd0eb5c13cff       redis          "docker-entrypoint.s…"   18 seconds ago    Up 16 seconds    6379/tcp       re
laxed_bhaskara
```

6. Use $docker ps -a to see all the images that are running and terminated.

```
$ docker ps -a
CONTAINER ID       IMAGE          COMMAND             CREATED          STATUS          PORTS          NA
MES
0d417fbbe29e       redis          "docker-entrypoint.s…"   11 seconds ago    Up 5 seconds     6379/tcp       he
uristic_roentgen
bd0eb5c13cff       redis          "docker-entrypoint.s…"   7 minutes ago     Up 7 minutes     6379/tcp       re
laxed_bhaskara
```

$docker inspect <container name or container id> provides more details about a running container, such as IP address.

```
$ docker inspect 0d4
[
    {
        "Id": "0d417fbbe29e7829a700f05c91119054d922eaf69fee64b179426393ed0946f3",
        "Created": "2020-11-10T04:54:08.854906511Z",
        "Path": "docker-entrypoint.sh",
        "Args": [
            "redis-server"
        ],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 1930,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2020-11-10T04:54:12.942456743Z",
            "FinishedAt": "0001-01-01T00:00:00Z"
        },
        "Image": "sha256:4e8db158f18dc71307f95260e532df39a9b604b51d4e697468e82845c50cfe28",
        "ResolvConfPath": "/var/lib/docker/containers/0d417fbbe29e7829a700f05c91119054d922eaf69fee64b179426393ed0946f3/resolv.c
onf",
```

7. Use $docker run -d --name redisHostPort -p 6379:6379 redis:latestlatest to define customized docker image names and ports.

```
$ docker run -d --name redisHostPort -p 6379:6379 redis:latest
66fd83861336096d0d7a3f5ae8fb94ebc63e2107be1f205500968c37802a8dfd
```

```
$ docker ps
CONTAINER ID       IMAGE           COMMAND            CREATED          STATUS           PORTS
    NAMES
66fd83861336       redis:latest    "docker-entrypoint.s…"  18 seconds ago   Up 17 seconds    0.0.0.0:6379->6379/tcp
    redisHostPort
```

8. $docker port to see the assigned port

```
$ docker run -d --name redisDynamic -p 6379 redis:latest
61979dbbaafbd9dda06cb35fe6dba764aa69a3c7e3f2bcdee919a1211f6a365f
$ docker port redisDynamic 6379
0.0.0.0:32768
```

$docker ubuntu run ps command launches an Ubuntu container and executes the command *ps* to view all the processes running in a container.

```
$ docker run ubuntu ps
  PID TTY          TIME CMD
    1 ?        00:00:00 ps
$ docker ps -a
CONTAINER ID       IMAGE           COMMAND            CREATED          STATUS                 PORTS
        NAMES
d26e3b379f35       ubuntu          "ps"               13 seconds ago   Exited (0) 11 seconds ago
```

$docker run -it ubuntu command allows to get access to a bash shell inside of a ubuntu container.

```
$ docker run -it ubuntu bash
root@6704739f57d1:/# ls
bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@6704739f57d1:/# exit
```

$docker stop

```
$ docker stop bd
bd
```

```
$ docker ps -a
CONTAINER ID        IMAGE             COMMAND                CREATED          STATUS                      PORTS
        NAMES
6704739f57d1        ubuntu            "bash"                 5 minutes ago    Exited (0) 59 seconds ago
        sharp_hawking
d26e3b379f35        ubuntu            "ps"                   5 minutes ago    Exited (0) 5 minutes ago
        modest_noether
61979dbbaafb        redis:latest      "docker-entrypoint.s…" 13 minutes ago   Up 13 minutes               0.0.0.0:32768-
>6379/tcp   redisDynamic
66fd83861336        redis:latest      "docker-entrypoint.s…" 18 minutes ago   Up 18 minutes               0.0.0.0:6379->
6379/tcp   redisHostPort
0d417fbbe29e        redis             "docker-entrypoint.s…" 25 minutes ago   Up 25 minutes               6379/tcp
        heuristic_roentgen
bd0eb5c13cff        redis             "docker-entrypoint.s…" 32 minutes ago   Exited (0) 12 seconds ago
        relaxed_bhaskara
```

$docker start

```
$ docker start bd
bd
```

```
$ docker ps
CONTAINER ID        IMAGE             COMMAND                CREATED          STATUS           PORTS
    NAMES
61979dbbaafb        redis:latest      "docker-entrypoint.s…" 14 minutes ago   Up 14 minutes    0.0.0.0:32768->6379/tc
p   redisDynamic
66fd83861336        redis:latest      "docker-entrypoint.s…" 18 minutes ago   Up 18 minutes    0.0.0.0:6379->6379/tcp
    redisHostPort
0d417fbbe29e        redis             "docker-entrypoint.s…" 25 minutes ago   Up 25 minutes    6379/tcp
    heuristic_roentgen
bd0eb5c13cff        redis             "docker-entrypoint.s…" 32 minutes ago   Up 3 seconds     6379/tcp
    relaxed_bhaskara
```