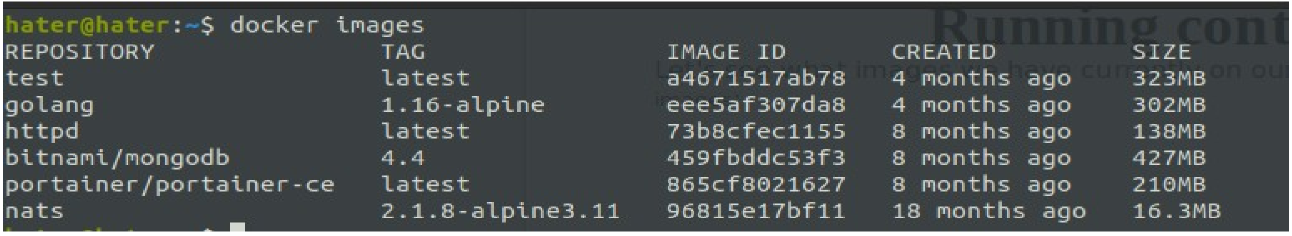
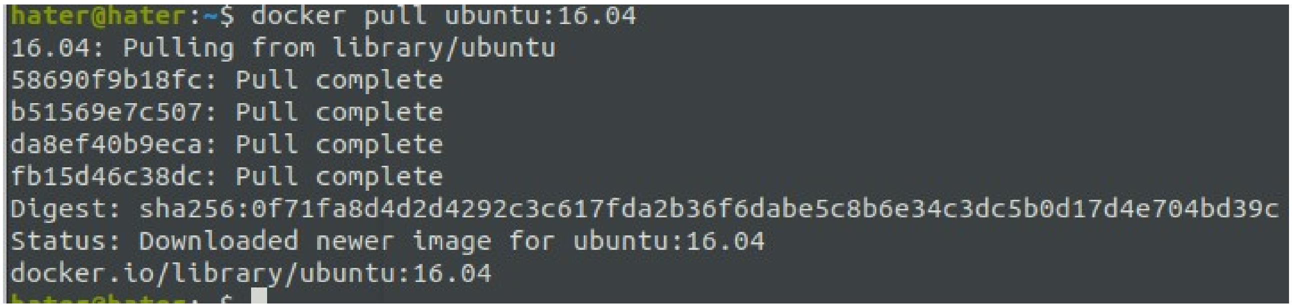
Running containers

Let's see what images we have currently on our machine, by running docker images:



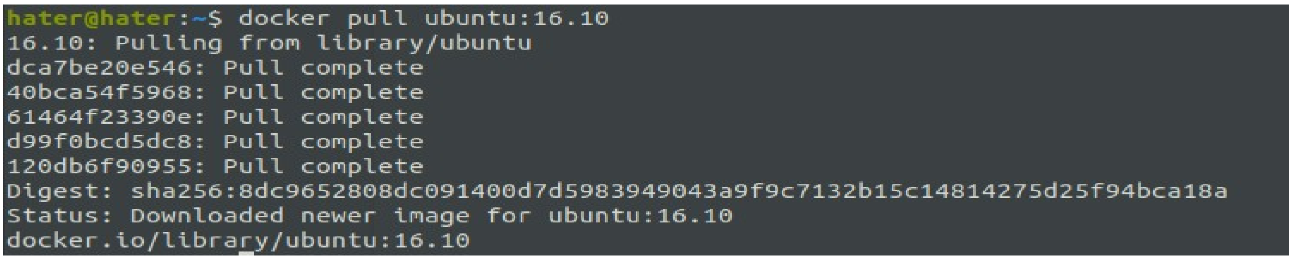
```
hater@hater:~$ docker images
REPOSITORY                TAG                 IMAGE ID       CREATED          SIZE
test                      latest              a4671517ab78   4 months ago     323MB
golang                    1.16-alpine         eee5af307da8   4 months ago     302MB
httpd                     latest              73b8cfec1155   8 months ago     138MB
bitnami/mongodb           4.4                 459fbddc53f3   8 months ago     427MB
portainer/portainer-ce    latest              865cf8021627   8 months ago     210MB
nats                      2.1.8-alpine3.11    96815e17bf11   18 months ago    16.3MB
```

Run docker pull ubuntu:16.04 to pull an image of Ubuntu 16.04 from DockerHub.



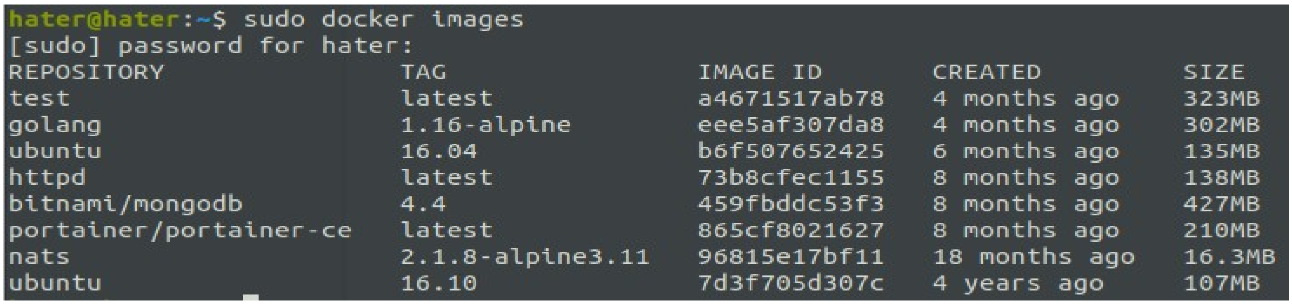
```
hater@hater:~$ docker pull ubuntu:16.04
16.04: Pulling from library/ubuntu
58690f9b18fc: Pull complete
b51569e7c507: Pull complete
da8ef40b9eca: Pull complete
fb15d46c38dc: Pull complete
Digest: sha256:0f71fa8d4d2d4292c3c617fda2b36f6dabe5c8b6e34c3dc5b0d17d4e704bd39c
Status: Downloaded newer image for ubuntu:16.04
docker.io/library/ubuntu:16.04
```

Run docker pull ubuntu:16.10 to pull an image of Ubuntu 16.10.

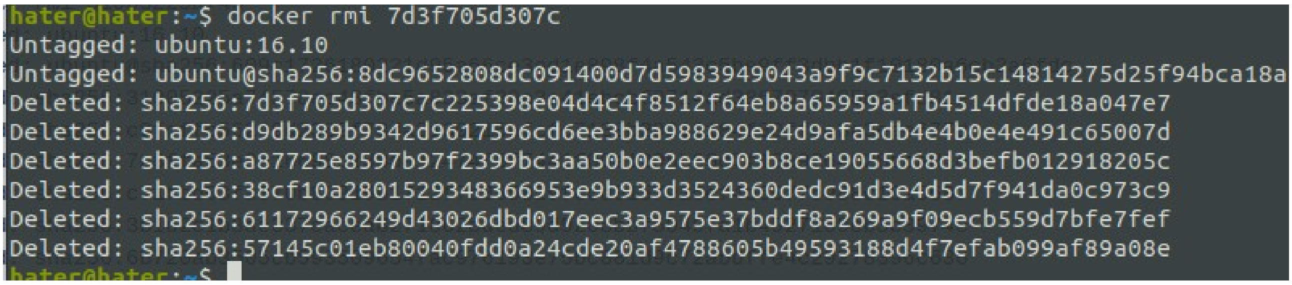


```
hater@hater:~$ docker pull ubuntu:16.10
16.10: Pulling from library/ubuntu
dca7be20e546: Pull complete
40bca54f5968: Pull complete
61464f23390e: Pull complete
d99f0bcd5dc8: Pull complete
120db6f90955: Pull complete
Digest: sha256:8dc9652808dc091400d7d5983949043a9f9c7132b15c14814275d25f94bca18a
Status: Downloaded newer image for ubuntu:16.10
docker.io/library/ubuntu:16.10
```

Run docker rmi <IMAGE ID> to remove the Ubuntu 16.10 image we won't be using.

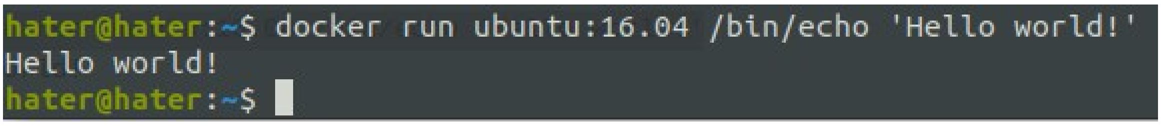


```
hater@hater:~$ sudo docker images
[sudo] password for hater:
REPOSITORY                TAG                 IMAGE ID       CREATED          SIZE
test                      latest              a4671517ab78   4 months ago     323MB
golang                    1.16-alpine         eee5af307da8   4 months ago     302MB
ubuntu                    16.04               b6f507652425   6 months ago     135MB
httpd                     latest              73b8cfec1155   8 months ago     138MB
bitnami/mongodb           4.4                 459fbddc53f3   8 months ago     427MB
portainer/portainer-ce    latest              865cf8021627   8 months ago     210MB
nats                      2.1.8-alpine3.11    96815e17bf11   18 months ago    16.3MB
ubuntu                    16.10               7d3f705d307c   4 years ago      107MB
```



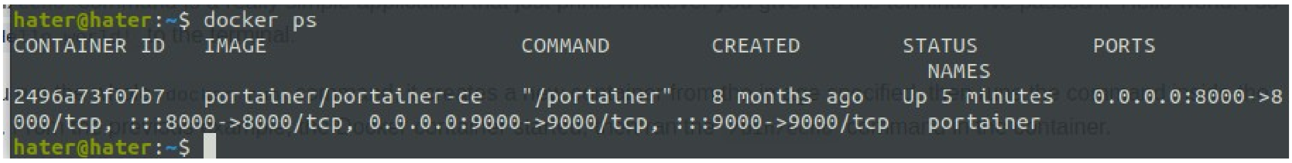
```
hater@hater:~$ docker rmi 7d3f705d307c
Untagged: ubuntu:16.10
Untagged: ubuntu@sha256:8dc9652808dc091400d7d5983949043a9f9c7132b15c14814275d25f94bca18a
Deleted: sha256:7d3f705d307c7c225398e04d4c4f8512f64eb8a65959a1fb4514dfde18a047e7
Deleted: sha256:d9db289b9342d9617596cd6ee3bba988629e24d9afa5db4e4b0e4e491c65007d
Deleted: sha256:a87725e8597b97f2399bc3aa50b0e2eec903b8ce19055668d3befb012918205c
Deleted: sha256:38cf10a2801529348366953e9b933d3524360dedc91d3e4d5d7f941da0c973c9
Deleted: sha256:61172966249d43026dbd017eec3a9575e37bddf8a269a9f09ecb559d7bfe7fef
Deleted: sha256:57145c01eb80040fdd0a24cde20af4788605b49593188d4f7efab099af89a08e
hater@hater:~$
```

Let's do a very simple example. Run docker run ubuntu:16.04 /bin/echo 'Hello world!'



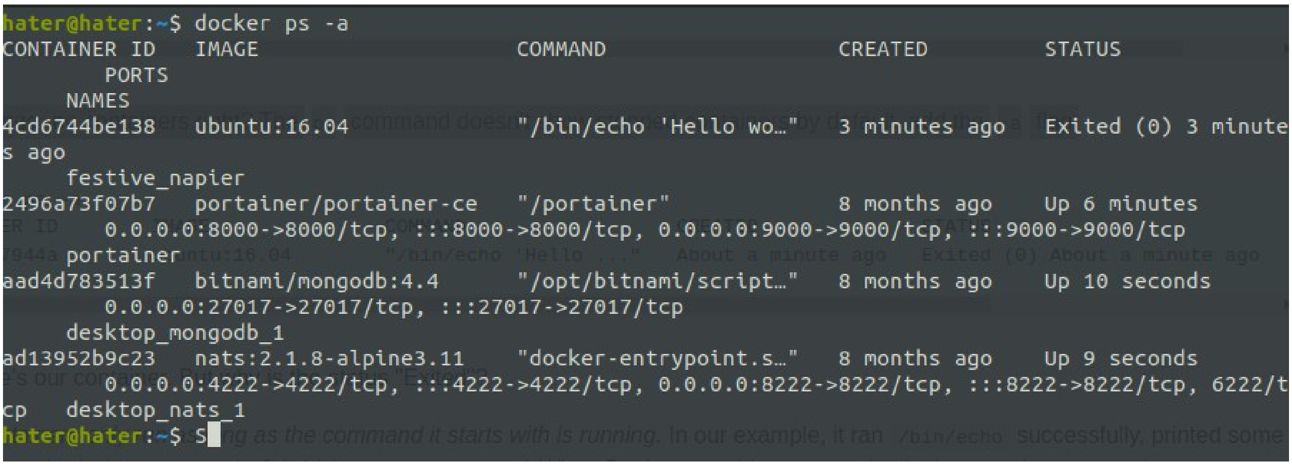
```
hater@hater:~$ docker run ubuntu:16.04 /bin/echo 'Hello world!'
Hello world!
hater@hater:~$ 
```

Let's check what containers we have after running this. Run docker ps



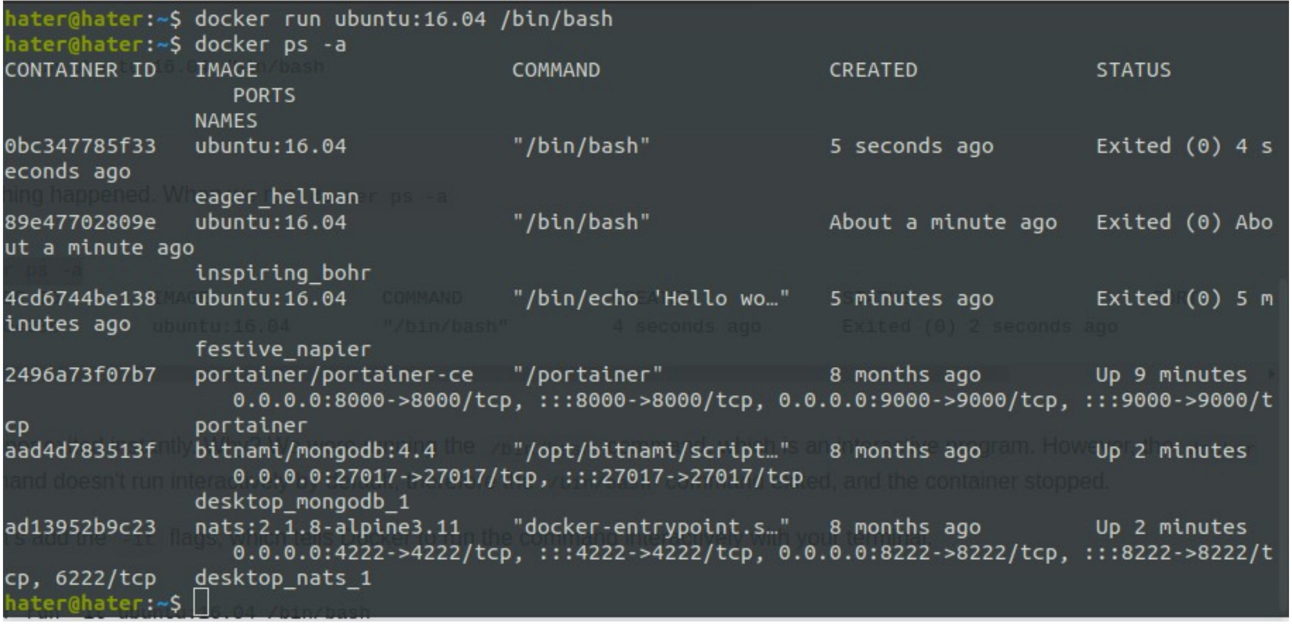
```
hater@hater:~$ docker ps
CONTAINER ID   IMAGE                        COMMAND        CREATED        STATUS         PORTS
                                                                           NAMES
2496a73f07b7   portainer/portainer-ce   "/portainer"   8 months ago   Up 5 minutes   0.0.0.0:8000->8
000/tcp, :::8000->8000/tcp, 0.0.0.0:9000->9000/tcp, :::9000->9000/tcp   portainer
hater@hater:~$ 
```

That's strange: no containers right? The ps command doesn't show stopped containers by default, add the -a flag.



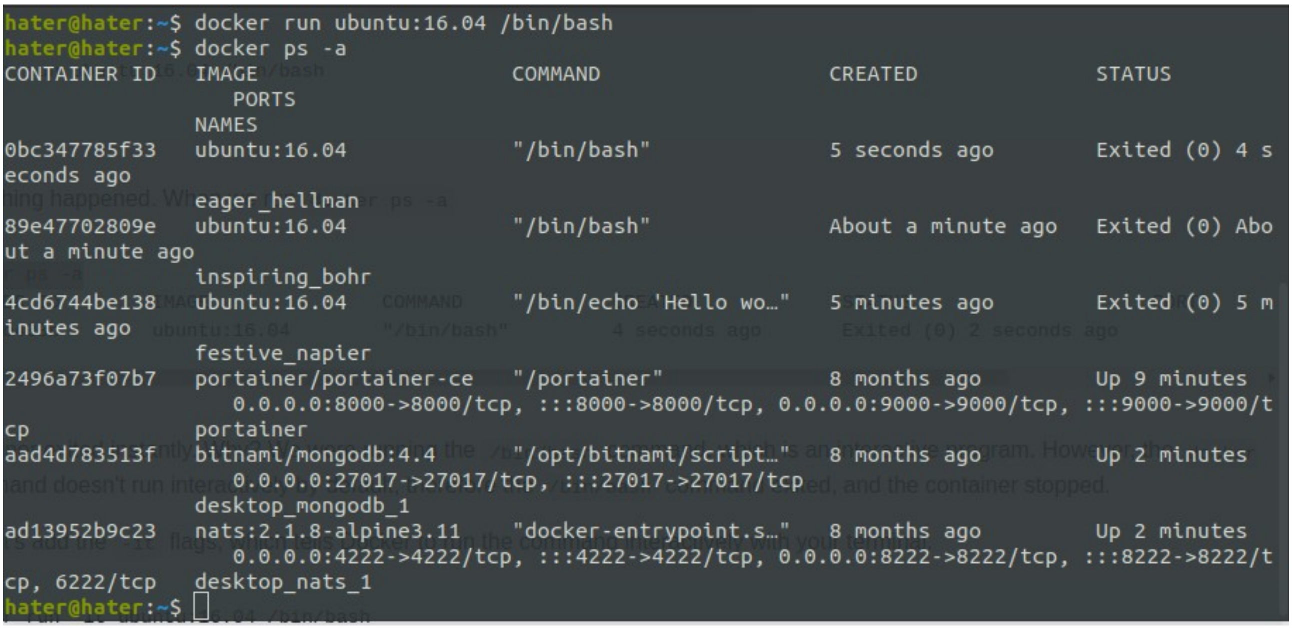
```
hater@hater:~$ docker ps -a
CONTAINER ID   IMAGE                        COMMAND                  CREATED          STATUS
        PORTS
     NAMES
4cd6744be138   ubuntu:16.04                 "/bin/echo 'Hello wo…"   3 minutes ago    Exited (0) 3 minute
s ago
     festive_napier
2496a73f07b7   portainer/portainer-ce   "/portainer"             8 months ago     Up 6 minutes
        0.0.0.0:8000->8000/tcp, :::8000->8000/tcp, 0.0.0.0:9000->9000/tcp, :::9000->9000/tcp
     portainer
aad4d783513f   bitnami/mongodb:4.4      "/opt/bitnami/script…"   8 months ago     Up 10 seconds
        0.0.0.0:27017->27017/tcp, :::27017->27017/tcp
     desktop_mongodb_1
ad13952b9c23   nats:2.1.8-alpine3.11    "docker-entrypoint.s…"   8 months ago     Up 9 seconds
        0.0.0.0:4222->4222/tcp, :::4222->4222/tcp, 0.0.0.0:8222->8222/tcp, :::8222->8222/tcp, 6222/t
cp   desktop_nats_1
hater@hater:~$ S
```

Let's do something a bit more interactive. Run docker run ubuntu:16.04 /bin/bash



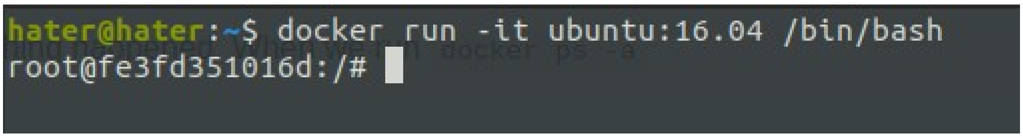
```
hater@hater:~$ docker run ubuntu:16.04 /bin/bash
hater@hater:~$ docker ps -a
CONTAINER ID   IMAGE                      COMMAND                  CREATED              STATUS
                  PORTS
               NAMES
0bc347785f33   ubuntu:16.04               "/bin/bash"              5 seconds ago        Exited (0) 4 s
econds ago
               eager_hellman
89e47702809e   ubuntu:16.04               "/bin/bash"              About a minute ago   Exited (0) Abo
ut a minute ago
               inspiring_bohr
4cd6744be138   ubuntu:16.04               "/bin/echo 'Hello wo…"   5 minutes ago        Exited (0) 5 m
inutes ago
               festive_napier
2496a73f07b7   portainer/portainer-ce     "/portainer"             8 months ago         Up 9 minutes
                  0.0.0.0:8000->8000/tcp, :::8000->8000/tcp, 0.0.0.0:9000->9000/tcp, :::9000->9000/t
cp             portainer
aad4d783513f   bitnami/mongodb:4.4        "/opt/bitnami/script…"   8 months ago         Up 2 minutes
                  0.0.0.0:27017->27017/tcp, :::27017->27017/tcp
               desktop_mongodb_1
ad13952b9c23   nats:2.1.8-alpine3.11      "docker-entrypoint.s…"   8 months ago         Up 2 minutes
                  0.0.0.0:4222->4222/tcp, :::4222->4222/tcp, 0.0.0.0:8222->8222/tcp, :::8222->8222/t
cp, 6222/tcp   desktop_nats_1
hater@hater:~$ 
```

Notice nothing happened. When we run docker ps -a



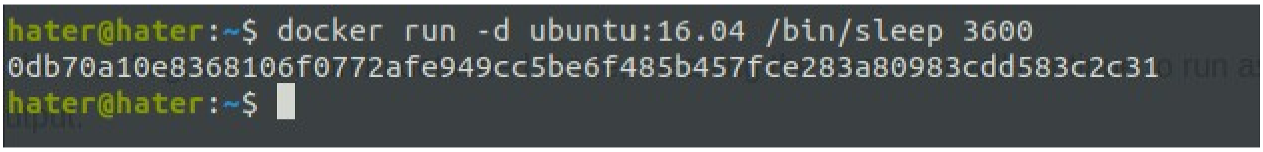
```
hater@hater:~$ docker run ubuntu:16.04 /bin/bash
hater@hater:~$ docker ps -a
CONTAINER ID   IMAGE                 COMMAND                  CREATED              STATUS
                  PORTS
               NAMES
0bc347785f33   ubuntu:16.04          "/bin/bash"              5 seconds ago        Exited (0) 4 s
econds ago
               eager_hellman
89e47702809e   ubuntu:16.04          "/bin/bash"              About a minute ago   Exited (0) Abo
ut a minute ago
               inspiring_bohr
4cd6744be138   ubuntu:16.04          "/bin/echo 'Hello wo…"   5 minutes ago        Exited (0) 5 m
inutes ago
               festive_napier
2496a73f07b7   portainer/portainer-ce   "/portainer"          8 months ago         Up 9 minutes
                  0.0.0.0:8000->8000/tcp, :::8000->8000/tcp, 0.0.0.0:9000->9000/tcp, :::9000->9000/t
cp             portainer
aad4d783513f   bitnami/mongodb:4.4   "/opt/bitnami/script…"   8 months ago         Up 2 minutes
                  0.0.0.0:27017->27017/tcp, :::27017->27017/tcp
               desktop_mongodb_1
ad13952b9c23   nats:2.1.8-alpine3.11   "docker-entrypoint.s…"   8 months ago       Up 2 minutes
                  0.0.0.0:4222->4222/tcp, :::4222->4222/tcp, 0.0.0.0:8222->8222/tcp, :::8222->8222/t
cp, 6222/tcp   desktop_nats_1
hater@hater:~$
```

Instead, let's add the -it flags, which tells Docker to run the command interactively with your terminal.



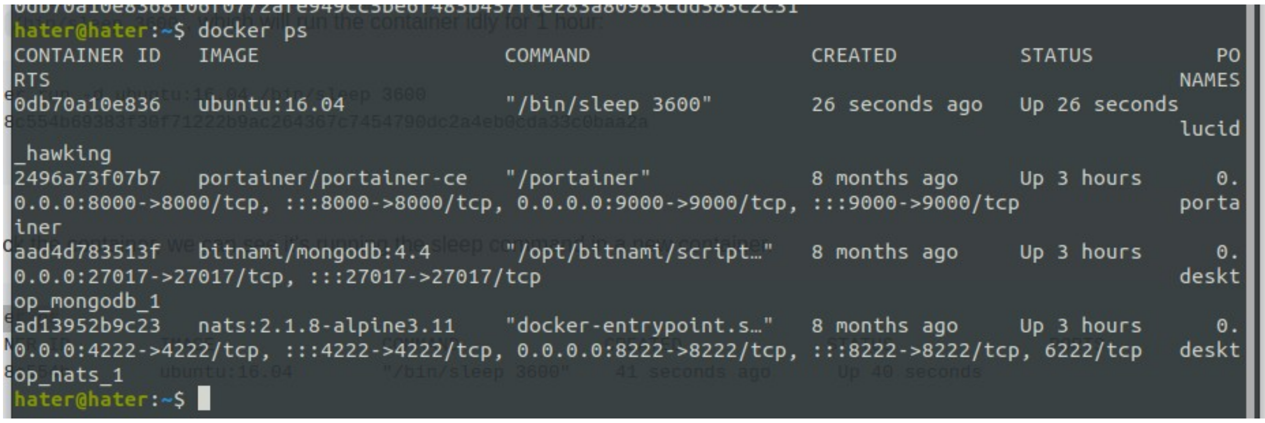
```
hater@hater:~$ docker run -it ubuntu:16.04 /bin/bash
root@fe3fd351016d:/#
```

Let's run /bin/sleep 3600, which will run the container idly for 1 hour:



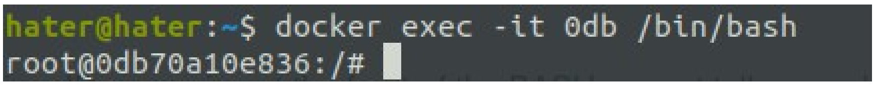
```
hater@hater:~$ docker run -d ubuntu:16.04 /bin/sleep 3600
0db70a10e8368106f0772afe949cc5be6f485b457fce283a80983cdd583c2c31
hater@hater:~$
```

If we check the container, we can see it's running the sleep command in a new container.



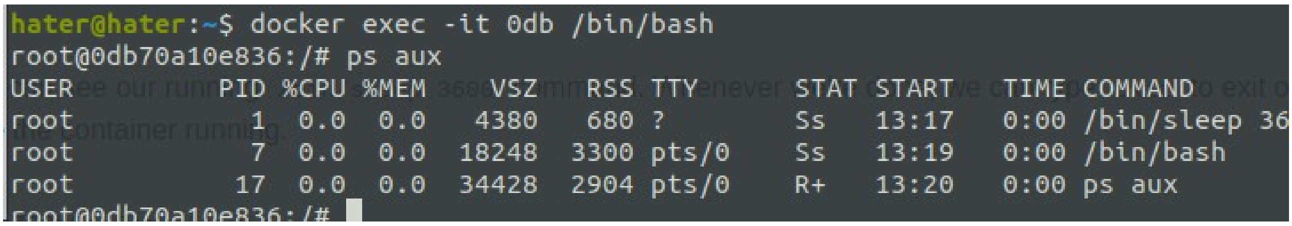
```
0db70a10e8368106f0772afe949cc5be6f485b457fce283a80983cdd583c2c31
hater@hater:~$ docker ps
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS           PO
RTS                                                                                             NAMES
0db70a10e836   ubuntu:16.04             "/bin/sleep 3600"        26 seconds ago   Up 26 seconds
                                                                                                lucid
_hawking
2496a73f07b7   portainer/portainer-ce   "/portainer"             8 months ago     Up 3 hours       0.
0.0.0:8000->8000/tcp, :::8000->8000/tcp, 0.0.0.0:9000->9000/tcp, :::9000->9000/tcp              porta
iner
aad4d783513f   bitnami/mongodb:4.4      "/opt/bitnami/script…"   8 months ago     Up 3 hours       0.
0.0.0:27017->27017/tcp, :::27017->27017/tcp                                                     deskt
op_mongodb_1
ad13952b9c23   nats:2.1.8-alpine3.11    "docker-entrypoint.s…"   8 months ago     Up 3 hours       0.
0.0.0:4222->4222/tcp, :::4222->4222/tcp, 0.0.0.0:8222->8222/tcp, :::8222->8222/tcp, 6222/tcp   deskt
op_nats_1
hater@hater:~$
```

Let's do that now, running the following, passing the first few characters of the container ID:



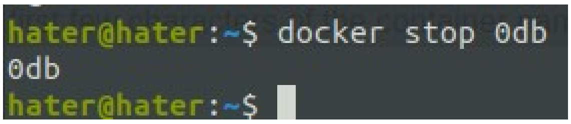
```
hater@hater:~$ docker exec -it 0db /bin/bash
root@0db70a10e836:/# 
```

Let's list the running processes:



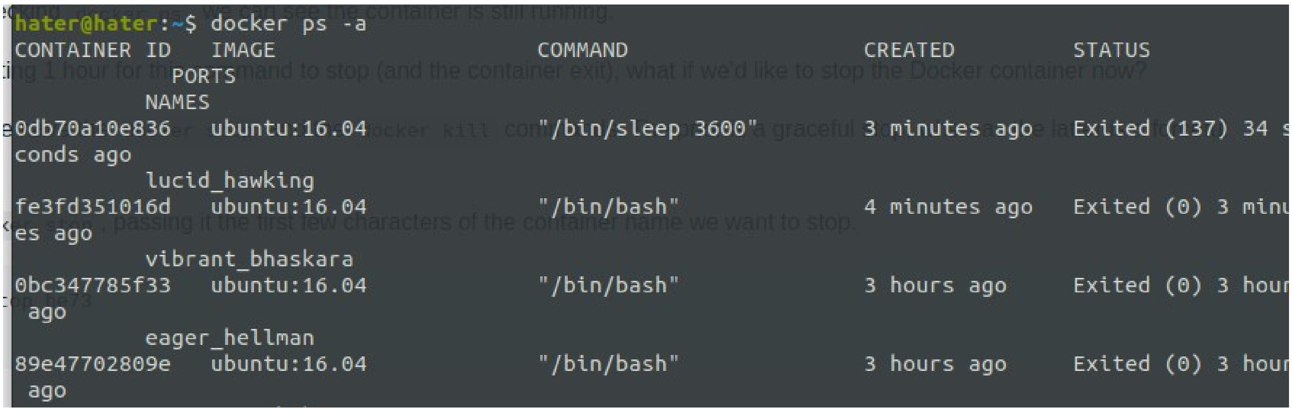
```
hater@hater:~$ docker exec -it 0db /bin/bash
root@0db70a10e836:/# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   4380   680 ?        Ss   13:17   0:00 /bin/sleep 36
root         7  0.0  0.0  18248  3300 pts/0    Ss   13:19   0:00 /bin/bash
root        17  0.0  0.0  34428  2904 pts/0    R+   13:20   0:00 ps aux
root@0db70a10e836:/# 
```

Let's use docker stop, passing it the first few characters of the container name we want to stop.



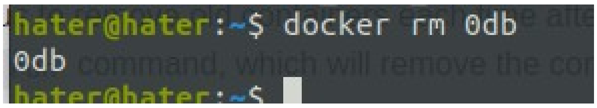
```
hater@hater:~$ docker stop 0db
0db
hater@hater:~$ 
```

Then checking docker ps -a...



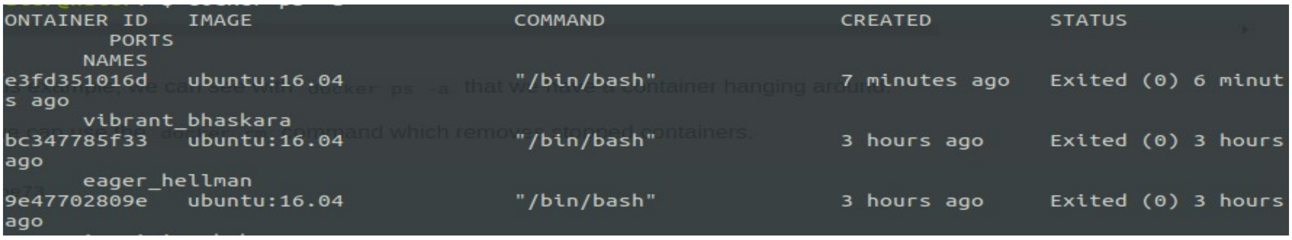
```
hater@hater:~$ docker ps -a
CONTAINER ID   IMAGE                       COMMAND                  CREATED          STATUS
      PORTS
        NAMES
0db70a10e836   ubuntu:16.04               "/bin/sleep 3600"        3 minutes ago    Exited (137) 34 s
conds ago
          lucid_hawking
fe3fd351016d   ubuntu:16.04               "/bin/bash"              4 minutes ago    Exited (0) 3 minu
es ago
          vibrant_bhaskara
0bc347785f33   ubuntu:16.04               "/bin/bash"              3 hours ago      Exited (0) 3 hour
 ago
          eager_hellman
89e47702809e   ubuntu:16.04               "/bin/bash"              3 hours ago      Exited (0) 3 hour
 ago
```

To remove this we can use the docker rm command which removes stopped containers.



```
hater@hater:~$ docker rm 0db
0db
hater@hater:~$ 
```

It can be tedious to remove old containers each time after you run them. To address this, Docker also allows you to specify the --rm flag to the docker run command, which will remove the container after it exits



```
ONTAINER ID   IMAGE                       COMMAND                  CREATED          STATUS
        PORTS
      NAMES
e3fd351016d   ubuntu:16.04               "/bin/bash"              7 minutes ago    Exited (0) 6 minut
s ago
      vibrant_bhaskara
bc347785f33   ubuntu:16.04               "/bin/bash"              3 hours ago      Exited (0) 3 hours
ago
      eager_hellman
9e47702809e   ubuntu:16.04               "/bin/bash"              3 hours ago      Exited (0) 3 hours
ago
```