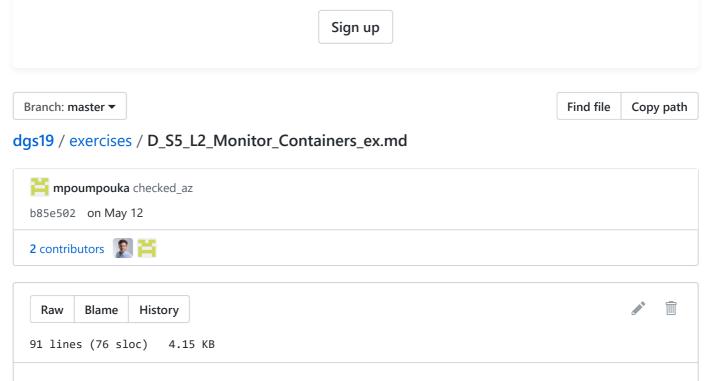
Dismiss

Join GitHub today

GitHub is home to over 36 million developers working together to host and review code, manage projects, and build software together.



class: center, middle

Section 5 - Containers lifecycle

2 - Monitor & Inspect Containers

Exercises

Exercise 1

- 1. Create two containers, mysql and nginx as follow:
- # docker container run -d --name mysql -e MYSQL_RANDOM_ROOT_PASSWORD=true
 mysql
- # docker container run -d --name nginx nginx sha1sum /dev/zero =>

- 2. Note that in case of nginx, we used /dev/zero => (this is a small hack to make the main process use $\sim 100\%$ of the CPU).
- 3. Display the running processes of each container.
- 4. Display the most CPU intensive processes running on the host system.
- 5. Display a single view (Disable streaming) of containers resource usage statistics.
- 6. Find the "IPAddress" of the nginx container from the output of the "docker container inspect" command.

Exercise 1 Solution

1. Create two containers, mysql and nginx as follow:

```
# docker container run -d --name mysql -e MYSQL_RANDOM_ROOT_PASSWORD=true
mysql
67efa407d4f03743118d5b5c43302e5f34e7aac42e052712a75caffbe37a439d
# docker container run -d --name nginx nginx sha1sum /dev/zero
ddbcc964f9229cf518dd85da43b4c2b1fc102d1ee561c3fc326689b078c70ab8
```

2. Display the running processes of each container:

```
# docker container top mysql
                                                              C
UID
                    PID
                                         PPID
STIME
                    TTY
                                         TIME
                                                              CMD
polkitd
                    5080
                                         5067
10:47
                                                              mysqld
                    ?
                                         00:00:00
# docker container top nginx
                                                              C
UID
                    PID
                                         PPID
                                                              CMD
STIME
                    TTY
                                         TIME
                    5211
                                         5198
                                                              91
root
10:47
                    ?
                                                              sha1sum /dev/zero
                                         00:00:58
```

Press <Ctrl-c> to exit from the real time view of the statistics.

3. Display the most CPU intensive processes running on the host system:

```
# top
 PID USER
             PR NI
                      VIRT
                             RES
                                   SHR S %CPU %MEM
                                                     TIME+ COMMAND
5211 root
             20
                  0
                      4192
                             344
                                   284 R 99.9 0.0 8:35.63 sha1sum
             20 0 125632
                            4124
                                   2588 S 0.0 0.1 0:01.15 systemd
   1 root
   2 root
             20
                        0
                               0
                                     0 S 0.0 0.0
                                                   0:00.00 kthreadd
```

Press <Ctrl-C> to exit from the real time view of the statistics. Note that the first PID, reported from the "top" command which is also the most CPU intensive process, is the same as the one reported from the "docker container top nginx" command.

4. Display a single view of containers resource usage statistics: With the use of the CLI help documentation we can see that the "--no-stream" option can be used to disable the streaming of statistics

docker stats --help

0.00%

9.60%

67efa407d4f0

```
Usage: docker stats [OPTIONS] [CONTAINER...]
Display a live stream of container(s) resource usage statistics
Options:
 -a, --all
                      Show all containers (default shows just running)
     --format string Pretty-print images using a Go template
     --no-stream Disable streaming stats and only pull the first result
     --no-trunc Do not truncate output
# docker stats --no-stream
CONTAINER ID
                                     CPU %
                                                       MEM USAGE / LIMIT
                  NAME
MEM %
                 NET I/O
                                     BLOCK I/O
                                                       PIDS
ddbcc964f922
                nginx
                                    98.57%
                                                       108KiB / 3.701GiB
```

0B / 0B

0.25%

0B / 661MB

364MiB / 3.701GiB

37

5. Find the "IPAddress" of the nginx container from the output of the "docker container inspect" command:

729B / 0B

729B / 0B

mysql

Note that we use the "--format" option to format the output using the given Go template.

Ref: https://docs.docker.com/engine/reference/commandline/inspect/

```
# docker container inspect --format='{{range .NetworkSettings.Networks}}
{{.IPAddress}}{{end}}' nginx
172.17.0.2
```