

#020 Docker cleaning up

Introduction

this is part 19 from the journey it's a long journey(360 day) so go please check previous parts , and if you need to walk in the journey with me please make sure to follow because I may post more than once in 1 Day but surely I will post daily at least one 😊.

And I will cover lot of tools as we move on.

Cleaning

```
(base) in ~
> docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
88588cd07e2c	app_017_1	"sh --rm --name app_..."	2 days ago	Exited (2) 2 days ago		jolly_rhodes
14171bd7b9c5	app_017_2	"--volumes-from obje..."	2 days ago	Created		suspicious_sand
erson						
04431f374602	app_017_2	"--volumes-from obje..."	2 days ago	Created		boring_lehmann
177765e28a5c	app_017_2	"-itd --volumes-from..."	2 days ago	Created		gifted_visvesva
raya						
60b181237c16	app_017_2	"-itd --volumes-from..."	2 days ago	Created		quirky_jackson
295e0347fd80	app_017_2	"--rm -itd --volumes..."	2 days ago	Created		sad_euclid
66cba51159ca	app_017_2	"--rm -itd --volumes..."	2 days ago	Created		boring_murdock
2a6edb5ebd85	app_017_2	"--volumes-from app_..."	2 days ago	Created		gifted_yonath
f6109e0b8181	app_017_2	"--volumes-from app_..."	2 days ago	Created		agitated_swartz
2a871ffc7b3f	app_017_2	"--volumes-from app_..."	2 days ago	Created		priceless_vaugh
an						
7c7e6b29cb71	app_017_2	"--volumes-from app_..."	2 days ago	Created		tender_solomon
db4827a96393	app_017_2	"sh"	2 days ago	Exited (137) 2 days ago		keen_bohr
23e1798e7399	app_017_2	"sh --volumes-from a..."	2 days ago	Exited (2) 2 days ago		competent_herma
nn						
9fccf85ddb3b	app_017_2	"sh --volumes-from a..."	2 days ago	Created		mystifying_hell
man						
6de49f40d68c	app_017_2	"sh"	2 days ago	Exited (0) 2 days ago		admiring_merkle
ea33cce60264	app_017_2	"sh"	2 days ago	Exited (130) 2 days ago		gallant_shannon
79a0c936101f	app_017_1	"sh"	2 days ago	Exited (0) 2 days ago		objective_keple

```
r
(base) in ~
>
```

Let's see how many stopped containers we have by using

```
docker ps -a
```

We have a lot!

what about images?

```
(base) in ~
> docker image ls
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
app_018             latest      fad47b82469b     2 days ago      83MB
app_017_2           latest      10b87eb2faaa     2 days ago      83MB
app_017_1           latest      9aa2d3ffcb7b     2 days ago      83MB
<none>              <none>      756f949384db     2 days ago      970MB
celeryworker        latest      00e5c2965990     2 days ago      1.39GB
frontend            latest      378ac8d93d53     2 days ago      114MB
<none>              <none>      cdbd5546e4db     2 days ago      1.87GB
c2-client           latest      225532576e5f     4 days ago      83MB
c1-server           latest      2bffee066e35     4 days ago      83MB
<none>              <none>      a919f9419fea     4 days ago      83MB
<none>              <none>      c207c8702592     4 days ago      83MB
<none>              <none>      d6aa34a1ca12     4 days ago      83MB
rabbitmq            3          7e50c60f7e3e     4 days ago      156MB
app_014             latest      af274e2c7588     5 days ago      83MB
app_013             latest      db29148a2bf0     6 days ago      83MB
omarelkhatib/app_009_1 latest      c3a5aa66659c     7 days ago      83MB
<none>              <none>      9bf1c30c534b     8 days ago      77.7MB
<none>              <none>      da4ea3800722     8 days ago      77.7MB
<none>              <none>      650a0161cf8d     8 days ago      77.7MB
<none>              <none>      c8a238d4f3eb     8 days ago      83MB
<none>              <none>      63b849d01559     8 days ago      77.7MB
python              3.8-alpine c206683c6cb1     8 days ago      77.7MB
```

```
docker image ls
```

we see a lot of basically those none are failed building images and they are not being used by anything , they called dangling images.

to see how much docker using space we run

```
(base) in ~
> docker system df
TYPE                TOTAL                ACTIVE              SIZE                RECLAIMABLE
Images              37                   2                   6.1GB                6.1GB (99%)
Containers          17                   0                   103B                 103B (100%)
Local Volumes       6                    2                   90.95MB              90.95MB (99%)
Build Cache         0                    0                   0B                   0B
(base) in ~
>
```

```
docker system df
```

we can see it's taking 6.1GB lot of space

let's clean all images that are not used by containers and dangling , we run

```
(base) in ~
> docker system prune
WARNING! This will remove:
- all stopped containers
- all networks not used by at least one container
- all dangling images
- all dangling build cache

Are you sure you want to continue? [y/N]
```

```
docker system prune
```

some useful tags here :

-f used to force , so we don't get a prompt to delete (good for automation tools)

-a used to delete every things including used images.

I am going to use

```
(base) in ~
> docker system prune -f
Deleted Containers:
88588cd07e2cd2062e8b865f6837b695d530d41e437b52f370a4fe1d8e40d596
14171bd7b9c5a41d041a67f993a06851496185305f99953aeabfb44d30398482
04431f3746020edb77c65547c7057e19ade6abc54216d1668183f3c619b79105
177765e28a5ca0a435927852df7380c0a0bc997ecf3ce48f7fe2745d8f4196bc
60b181237c169844d23bee50a4efdae3c191408c9becf68cbf603805bdb96d0f
295e0347fd801fd1f792411da02de06eb994641bc808c8e5ea82e6cb633420ad
66cba51159ca47770b6dfacce1c3f16ddb325c75a8620b570c8c1042ffd2e7c7
2a6edb5ebd85aa8475e9fa4897be2ae1660350a4ab3b45a2a07b1b1d2a290e18
f6109e0b818156b6e4ff0deb6c0dd4c0c903589f1414b379711bb4883e2652d
2a871ffc7b3f08963e2780ec2563c2304d7ca2fbd8162238a36e86cdbc6cf4d1
7c7e6b29cb7197de79c96105a40c84c3605e373a9cd52f27e187972b8dbc9f7a
db4827a96393631d3d5260fa26c833bd29e2c018b03aa450557567d94cb56766
23e1798e7399e18fbb8b9d8b53a083b24a51ae5bb87efd313c5e8e10aa89b5dd
9fccf85ddb3b0429b686a9ed6e3efc8f6a03ebc6618c3dc7080cc13b993a4270
6de49f40d68c4cc8cd15fc905e63bcff971982a6ebbd6259db6f49c656a1f80f
ea33cce602641e7abeb1ad8a58755448d8ab998a16b0e1007d560b5c824dbc24
79a0c936101f09fd20aade43adb2f6bd5314e853269494959b25226ca8822661

Deleted Networks:
valorant-watcher_default
fastapi_dashboard_traefik-public
barja_census_default
fastapi_dashboard_default
network1

Deleted Images:
deleted: sha256:9bf1c30c534b04b27db6b152dad8f4c558f70f2fd5f3477714995c9779a3c90c
deleted: sha256:c12faca1f329a6d483aa709c77af03b3167535753292d58bf2808a2eeadc2914
deleted: sha256:63b849d015592b0a7fae173f231de9cc857a4a45b16a70bffa26ac9f7ab0df2b
deleted: sha256:3d5e86e1948f55d822177ad688483d3a655661b2c1a866aefae27e788dad360e
```

```
docker system prune -f
```

we see it delete lot of things , cool!

```
> docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
(base) in ~
> docker image ls
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
app_018             latest             fad47b82469b       2 days ago         83MB
app_017_2           latest             10b87eb2faaa       2 days ago         83MB
app_017_1           latest             9aa2d3ffcb7b       2 days ago         83MB
celeryworker        latest             00e5c2965990       2 days ago         1.39GB
frontend            latest             378ac8d93d53       2 days ago         114MB
c2-client           latest             225532576e5f       4 days ago         83MB
c1-server           latest             2bffee066e35       4 days ago         83MB
rabbitmq            3                  7e50c60f7e3e       4 days ago         156MB
app_014             latest             af274e2c7588       5 days ago         83MB
app_013             latest             db29148a2bf0       6 days ago         83MB
omarelkhatib/app_009_1 latest             c3a5aa66659c       7 days ago         83MB
python              3.9-rc-alpine     a206803e6cb1       9 days ago         77.7MB
python              3.7               e4e55e98f1e0       12 days ago        919MB
```

```
docker ps -a
docker image ls
```

as we see we have no containers anymore all stopped
and all our images gone!

if you need only to stop containers .

```
(base) in ~  
> docker container stop $(docker container ls -a -q)  
88588cd07e2c  
14171bd7b9c5  
04431f374602  
177765e28a5c  
60b181237c16  
295e0347fd80  
66cba51159ca  
2a6edb5ebd85  
f6109e0b8181  
2a871ffc7b3f  
7c7e6b29cb71  
db4827a96393  
23e1798e7399  
9fccf85ddb3b  
6de49f40d68c  
ea33cce60264  
79a0c936101f  
(base) in ~  
>
```

```
docker container stop $(docker container ls -a -q)
```