Befehle

- Docker bash / shell einloggen
- Alle Docker instanzen stoppen und/oder löschen

Docker bash / shell einloggen

Beschreibung:

Manchmal möchte man auch gerne direkt im container was nachschauen.
Hier der Befehle um sich in einen docker container ein zu loggen.

Befehle:

Erstmal wissen wie der container heißt.
Entweder die container ID oder den Namen nehhmen. Ich bevorzuge Namem.

```
docker ps
```

Ausgabe

```
root@unvr:~# docker ps
CONTAINER ID   IMAGE                          COMMAND                  CREATED      STATUS       PORTS     NAMES
375668882f07   markdegroot/unifi-protect-arm64   "/lib/systemd/systemd"   3 days ago   Up 16 hours
unifi-protect
root@unvr:~#



ENtweder container ID: 375668882f07
oder Name:  unifi-protect
```

Nun kann entweder bash oder shell genutzt werden, je nachdem was der Container beinhaltet

```
mit bash
docker exec -it <container-id> bash

mit shell
docker exec -it <container-id> sh
```

Beispiel ID

mit bash

```
docker exec -it 375668882f07 bash
```

mit shell

```
docker exec -it 375668882f07 sh
```

Beispiel Name

mit bash

```
docker exec -it unifi-protect bash
```

mit shell

```
docker exec -it unifi-protect sh
```

Alle Docker instanzen stoppen und/oder löschen

Lösung:

Alle laufenden Container sofort stoppen

```
docker stop $(docker ps -aq)
```

Alle Container löschen

```
docker rm $(docker ps -a -q)
```

Image(s) löschen

```
docker rmi $(docker images -q)
```

Erklärung der docker ps Parameter:

- `docker ps -q`: Listet die IDs aller laufenden Container auf.
- `docker stop $(docker ps -q)`: Stoppt alle laufenden Container.
- `docker ps -a -q`: Listet die IDs aller Container auf (laufende und gestoppte).
- `docker rm $(docker ps -a -q)`: Löscht alle Container.