

Docker

CLI

Rostagni Csaba

2025. szeptember 6.

Tartalom

1 Docker CLI parancsok

Tartalom

1 Docker CLI parancsok

- create
- ps
- inspect
- pull
- start
- run
- stop
- kill
- rm

docker create

```
docker container create [OPTIONS] IMAGE [COMMAND] [ARG...]
```

- Letölti az image fájlt, ha az nem létezik
- Létrehoz egy konténert a megadott imageből, de nem indítja el
- Valójában a `docker create` csak egy alias (lásd dokumentáció)

Linkek:

- `docker create` – docs.docker.com

docker create első futtatás

Terminal

```
docker create --name ubuntu1 ubuntu:24.04
```

- A `--name ubuntu1` egy elhagyható opció
- A felhasznált image "teljes neve": `ubuntu:24.04`

Eredmény

```
Unable to find image 'ubuntu:24.04' locally
24.04: Pulling from library/ubuntu
31e907dcc94a: Pull complete
Digest: sha256:8a37d68f4f73ebf3d4efafbcbf66379bf3728902a8038616808f04e34a9ab63ee
Status: Downloaded newer image for ubuntu:24.04
7537b0f26dbeb4325774903cc941bcc4d8c531b6ec3512db49d1f02daa60eedf
```

- Mivel az image nincs a gépen, így azt letölti
- Alapértelmezetten a Docker Hub-on keresi

Linkek:

- Ubuntu – hub.docker.com

docker create későbbi futtatás

```
docker create --name ubuntu2 ubuntu:24.04
```

Terminal

```
f73e788e8c5e7add0a8f23abdda7e11ebafa5e1fa7990c23bbcf6a2a68187829
```

Eredmény

- Mivel az image már megtalálható a gépen így nem tölti le, hanem rögtön létrehozza a konténert

Tartalom

1 Docker CLI parancsok

- create
- **ps**
- inspect
- pull
- start
- run
- stop
- kill
- rm

docker create példa eredménye

```
docker ps -a
```

Terminal

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f73e788e8c5e	ubuntu:24.04	"bash"	About a minute ago	Created		ubuntu2
3b88ac096a0d	ubuntu:24.04	"bash"	5 minutes ago	Created		ubuntu1

Eredmény

- Létrejött a két konténer
- A **STATUS** mind a két esetben **Created**

```
docker ps -a --no-trunc
```

Terminal

- A teljes ID kinyeréséhez a **--no-trunc** opció szükséges
 - f73e788e8c5e7add0a8f23abdda7e11ebafa5e1fa7990c23bbcf6a2a68187829
 - 3b88ac096a0d0e4f3ceb6507abf968043f6868751af47c10e9ffe6616ece39bf

Linkek:

- `docker ps` – docs.docker.com

Tartalom

1 Docker CLI parancsok

- create
- ps
- **inspect**
- pull
- start
- run
- stop
- kill
- rm

Mi a konténer azonosítója?

```
docker ps -a
```

Terminal

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f73e788e8c5e	ubuntu:24.04	"bash"	About a minute ago	Created		ubuntu2
3b88ac096a0d	ubuntu:24.04	"bash"	5 minutes ago	Created		ubuntu1

Eredmény

```
docker inspect --format="{.Id}" ubuntu2
```

Terminal

```
f73e788e8c5e7add0a8f23abdda7e11ebafa5e1fa7990c23bbcf6a2a68187829
```

Eredmény

- A `--format` szabja meg mely adatok kerüljenek megjelenítésre
- A példában csak az `Id` kerül megjelenítésre

Linkek:

- `docker inspect` – docs.docker.com
- `formatting (GO)` – docs.docker.com

Tartalom

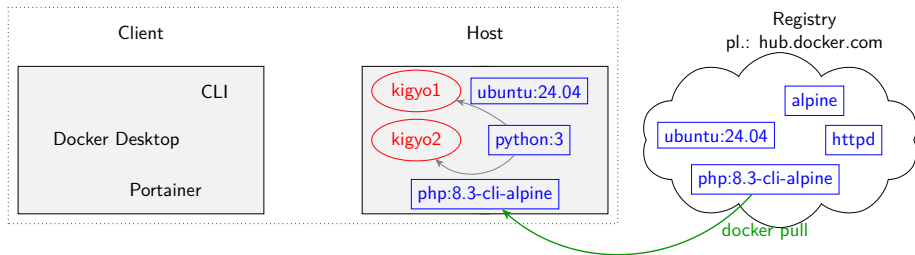
1 Docker CLI parancsok

- create
- ps
- inspect
- **pull**
- start
- run
- stop
- kill
- rm

docker pull

```
docker image pull [OPTIONS] NAME[:TAG|@DIGEST]
```

- A `docker pull` letölti a registryből az image-t



Linkek:

- `docker pull` – docs.docker.com

Tartalom

1 Docker CLI parancsok

- create
- ps
- inspect
- pull
- **start**
- run
- stop
- kill
- rm

docker start

```
docker container start [OPTIONS] CONTAINER [CONTAINER...]
```

- Elindít egy létező konténert, ami **Created** vagy **Stopped** állapotban van

Linkek:

- [docker start – docs.docker.com](https://docs.docker.com)

docker start példa

Terminal

```
docker start ubuntu1
```

Eredmény

```
ubuntu1
```

- Az indítás után, ha minden rendben ment a konténer nevét írja ki

Terminal

```
docker ps -a --format="table {{.ID}}\t {{.Image}}\t{{.Status}}\t{{.Names}}"
```

Eredmény

CONTAINER ID	IMAGE	STATUS	NAMES
f73e788e8c5e	ubuntu:24.04	Created	ubuntu2
3b88ac096a0d	ubuntu:24.04	Exited (0) 2 minutes ago	ubuntu1

- Amint elindult ki is lépett `0` állapotkóddal, azaz rendben lefutott
- `Exited` állapotba került
- Miért? A konténernek nem volt mit csinálnia, így gyorsan végzett a feladatával

Tartalom

1 Docker CLI parancsok

- create
- ps
- inspect
- pull
- start
- **run**
- stop
- kill
- rm

docker run

```
docker container run [OPTIONS] IMAGE [COMMAND] [ARG...]
```

- A `docker create` és a `docker start` parancsokat együttesen helyettesíti

Linkek:

- [docker run – docs.docker.com](https://docs.docker.com/run)

docker run példa

```
docker run [OPTIONS] IMAGE [COMMAND] [ARG...]
```

```
docker run --rm --name szamologep ubuntu:24.04 expr 3 + 7
```

Terminal

- OPTIONS

- `--rm`: a konténer törlésre kerül, miután véget ért a futása
- `--name`: a konténer neve, jelen esetben `szamologep` lesz

- IMAGE

- `ubuntu:24.04`

- COMMAND

- A konténeren belül az `expr` parancsot futtatja

- ARGS: `3`, `+` és `7`

Eredmény

10

docker run interaktív példa

Terminal

```
docker run -i -t --rm --name ubuntu3 ubuntu:24.04 /bin/bash
```

• OPTIONS

- `-i`, vagy `--interactive`: Interaktív mód, azaz a konténer megkapja a terminálban kiadott parancsokat
- `-t`, vagy `--tty`: Lefoglal egy pszeudó TTY-t (Teletypewritert)
- A `--rm` hatására eltávolítja kilépéskor a konténert
- `--name`: a konténer neve, jelen esetben `ubuntu3` lesz

• IMAGE – `ubuntu:24.04`

• COMMAND – `/bin/bash`

Infó

Az `-i` és `-t` együttesen használandó!

Eredmény

```
root@bae4c0ab048f:/#
```

- Az ubuntu promptja megjelenik, ami tartalmazza a konténer ID-t

docker run interaktív példa

```
expr 2 + 2
```

```
bae4c0ab048f
```

```
Eredmény
```

```
4
```

```
exit
```

```
bae4c0ab048f
```

- Az `exit` utasítással lehet kilépni a bashből, így a konténer élete is véget ér
- A `--rm` miatt a kilépés után meg is szűnik a konténer

docker run interaktív példa 2

```
docker run -it --rm ubuntu
```

Terminal

• OPTIONS

- A `-it` az a `-i` és `-t` összevonása, de lehetne `-ti` is
- hatására a konténer standard kimenetre írt szövege megjelenik, továbbá a terminálba írt utasításokat is megkapja a konténer
- A `--rm` hatására eltávolítja kilépéskor a konténert

• IMAGE

- ubuntu

```
root@f7fdede22552:/#
```

Eredmény

- A `--name` elhagyható, ha nem szükséges hivatkozni rá
- A tag elhagyásával az `ubuntu:latest` image lesz az alap
- Alapértelmezetten a `bash` indul el, de ez image függő
- Ez egy másik container, így a hash is megváltozott (f7fdede22552)

Tartalom

1 Docker CLI parancsok

- create
- ps
- inspect
- pull
- start
- run
- **stop**
- kill
- rm

docker stop

```
docker container stop [OPTIONS] CONTAINER [CONTAINER...]
```

- A konténer fő folyamatát állítja le **SIGTERM**, majd a türelmi idő lejártá után **SIGKILL** segítségével
- A konténer állapota **Stopped** lesz
- Egyszerre több konténert is le lehet állítani

Linkek:

- [docker stop – docs.docker.com](https://docs.docker.com/engine/reference/commandline/stop/)
- [sigterm vs. sigkill – linuxhandbook.com](https://linuxhandbook.com/sigterm-vs-sigkill/)

docker stop példa: leállítás név alapján

```
docker run --name kigyo -it python:3
```

Terminal

- Elindít egy interaktív módban futó konténert `kigyo` néven

```
>>> exit()
```

7daed29f5e07

- Az `exit()` függvénnyel lép ki a python 3 interaktív felülete

```
docker stop kigyo
```

Terminal

- Az utasítást egy másik terminál ablakban lehet kiadni!
- A konténer a nevére hivatkozva leállítható

docker stop példa: leállítás ID alapján

```
docker run --name kigyo2 -it python:3
```

Terminal

- Elindít egy interaktív módban futó konténert `kigyo2` néven

```
docker inspect --format="{.Id}" kigyo2
```

Terminal

- A `kigyo2` nevű konténer teljes azonosítójának lekérdezése

```
7daed29f5e078113696cf3bff335bbb122e5f1e2ad2d3b77c4ad0392a9531b1e
```

Eredmény

```
docker stop 7daed29f5e078113696cf3bff335bbb122e5f1e2ad2d3b77c4ad0392a9531b1e
```

Terminal

- A konténer ID megadásával is leállítható

docker stop példa: leállítás ID részlet alapján

```
docker run -d httpd
```

Terminal

- Elindít egy, a konzolról lecsatlakoztatott konténert, ami a háttérben tovább fut
- A `-d` vagy `--detach` hatására a háttérben fut a folyamat

```
8eb7130627a5937e3ada73a4524f9e4e24718eaa3c9baadc2eea9aff74d6e778
```

Eredmény

- A konténer indításakor megjeleníti az ID-t

```
docker stop 8eb
```

Terminal

- Elegendő az első néhány karaktert megadni az azonosítóból, amennyiben az egyedi

```
8eb
```

Eredmény

docker stop példa: azonos kezdetű azonosítók

```
docker stop 3b
```

Hiba!

Error response from daemon:
Cannot kill container: 3b:
Multiple IDs found with provided prefix: 3b

Eredmény

- A rendszeren több olyan konténer is található, melyeknek az azonosítójának az első két karaktere: **3b**

docker stop példa: több image leállítása

Terminal

```
docker run --rm -d nginx;  
docker run --rm -d --name web2 httpd;
```

Eredmény

```
52e6413b43e924b2abe90fbff36c4aff0f2755ddfb9aa41479bc874e63b8a870  
b4bfdee10b47dbbcbcb70807d848263336e35af95ec58677d61d32478aa1cb7dc
```

Terminal

```
docker stop 52e6413b web2
```

Eredmény

```
52e6413b  
web2
```

- Az `nginx` konténer az azonosítójának a részletével lett leállítva
- A `httpd` az neve által lett leállítva
- Amikor leáll ugyanezeket az értékeket írja ki

Tartalom

1 Docker CLI parancsok

- create
- ps
- inspect
- pull
- start
- run
- stop
- **kill**
- rm

docker kill

```
docker container kill [OPTIONS] CONTAINER [CONTAINER...]
```

- A konténer fő folyamatát "öli meg" **SIGKILL** segítségével
- A konténer állapota **Stopped** lesz
- Egyszerre több konténerre is kiadható

Linkek:

- [docker kill – docs.docker.com](https://docs.docker.com/engine/reference/commandline/kill/)
- [sigterm vs. sigkill – linuxhandbook.com](https://linuxhandbook.com/sigterm-vs-sigkill/)

Tartalom

1 Docker CLI parancsok

- create
- ps
- inspect
- pull
- start
- run
- stop
- kill
- **rm**

docker rm

```
docker container rm [OPTIONS] CONTAINER [CONTAINER...]
```

- Alap esetben leállított konténereket távolítja el
- A `-f` vagy `--force` opcióval futó konténereket is eltávolítja
- A `-v` vagy `--volume` opcióval a névtelen felcsatolt köteteket is törli
- Egyszerre több konténerre is kiadható

Linkek:

- `docker rm` – docs.docker.com

docker rm példa

```
docker run -d --name web nginx;
```

Terminal

```
docker rm web;
```

Terminal

```
Error response from daemon:  
You cannot remove a running container ....  
Stop the container before attempting removal or force remove
```

Eredmény

```
docker rm -f web;
```

Terminal

- A `-f` opció kikényszeríti a konténer törlését