



YAVUZLAR

Web Security & Development Team

İBRAHİM SERVET ÇETİNKAYA 3. GÜN ÖDEVİM

Öğrendiğim Komutlar ;

Komut ; “ docker search image:tag ”

Docker aracımız ile kendisinin bulundurduğu Dockerhub içerisindeki uygulamaları ismini verdiğimiz dosyayı arıyor ve ben komutta latest yazdım, bunun anlamı dosyanın son güncel sürümünü karşımıza getirmesi için ama : dan sonra tag parametreleri ekliyebiliyorduk.

```
root@Hegir:/home/servet# docker search image:latest
NAME                                DESCRIPTION                                STARS    OFFICIAL    AUTOMATED
ananya9434/h2db                    h2db image:latest                        0
robarr/alpine-squid-proxy           Alpine image:latest running squid-5.0.6-r0.a... 0
sumanreddy568/core-automation       adc core-automation test image:latest        0
villagoettingen/renderstreaming     image:latest                              0
root@Hegir:/home/servet#
```

Komut ; “docker pull mysql”

Docker aracımız ile dosyamızı aradıktan sonra ismini pull aracıyla çağırarak indirme sağlayabiliyoruz. (İndirmeyi ben yarıda kestim ctrl c ile)

```
servet@Hegir:~$ sudo docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
d9a40b27c30f: Downloading [====>] 4.975MB/48.99MB
fe4b01031aab: Download complete
aa72c34c4347: Download complete
473ade985fa2: Downloading [=====>] 4.827MB/6.712MB
cc168a9482de: Download complete
3ca3786815dd: Download complete
3e3fac98ea83: Downloading [====>] 3.422MB/47.7MB
10e5505c3ae4: Waiting
a79ade39aab9: Waiting
ae34d51c6da2: Waiting
^C
```

Komut ; “docker network create name”

Hocam burada bir ağ oluşturuyoruz ve bize sonucunda benzersiz olması amacıyla bir id gibi bir kimlik no veriyor.

```
servet@Hegir:~$ sudo docker network create yavuzlar_adayi_servet
9084fac3f935b5371e4cf0eba68043e01e6a57c41ccaa9bb70043c048fabd124
```

Komut ; “ docker network ls ”

Burada hem ağıımızı oluşturmuş muyuz diye hemde ağları listemek için kullanılan bir komut

```
servet@Hegir:~$ sudo docker network ls
NETWORK ID          NAME                DRIVER             SCOPE
068498bbfc96        bridge             bridge             local
0da8b67d5b50        host               host               local
47efc9feffb2        none              null               local
9084fac3f935        yavuzlar_adayi_servet  bridge            local
servet@Hegir:~$
```

Komut ; “ network rm agin_idisi ”

Burada rm komutu ile istedigimiz ağın id numarasını girerek ağı silebiliriz.

```
servet@Hegir:~$ sudo docker network rm 9084fac3f935
9084fac3f935
servet@Hegir:~$ sudo docker network ls
NETWORK ID          NAME                DRIVER             SCOPE
068498bbfc96        bridge             bridge             local
0da8b67d5b50        host               host               local
47efc9feffb2        none              null               local
servet@Hegir:~$
```

Komut ; “ docker images ”

İndirdigimiz dosyaları listeler

```
servet@Hegir:~$ sudo docker images
REPOSITORY          TAG                IMAGE ID           CREATED            SIZE
alpine              latest            324bc02ae123      44 hours ago      7.8MB
nginx               latest            a72860cb95fd      4 weeks ago       188MB
hello-world         latest            d2c94e258dcb      14 months ago     13.3kB
```

Komut;” docker rmi -f hello-world[:latest] ”

Burada istedigimiz image’yi image id sini yazarak silebiliriz. Hocam bu silme alternatifini biz dersde görmedik “**docker images rm image_id**” bende hata verdiğinden dolayı böyle bir komut buldum , bu da güçlü silme yöntemiymiş, böyle öğrendim.

```

servet@Hegir:~$ sudo docker rmi -f hello-world:latest
Untagged: hello-world:latest
Untagged: hello-world@sha256:1408fec50309afee38f3535383f5b09419e6dc0925bc69891e79d84cc4cdcec6
Deleted: sha256:d2c94e258dcb3c5ac2798d32e1249e42ef01cba4841c2234249495f87264ac5a
servet@Hegir:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
alpine        latest    324bc02ae123   45 hours ago   7.8MB
nginx         latest    a72860cb95fd   4 weeks ago    188MB
servet@Hegir:~$

```

Komut” `docker run -d --name mynginx -p 8080:80 nginx ”`

Hocam burada konteynır oluşturunuz bunu run sayesinde yapıyoruz -d yi arka planda çalışması için ekliyoruz bu kısma farklı opsiyonlarda eklenbilir. Daha sonra konteynıra isim koyup çalışacak port ve yazılımı yazıyoruz.

```

servet@Hegir:~$ sudo docker run -d --name mynginx -p 8080:80 nginx
9425747db1c853c7a2fdd9446100c87dab8738a6511030299944f0263b137e65
servet@Hegir:~$

```

Komut “ `docker ps ”`

Çalışan aktif konteynırları listeleyebilmemize yarar.

```

servet@Hegir:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
9425747db1c8   nginx     "/docker-entrypoint..." 4 minutes ago  Up 4 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp  mynginx
servet@Hegir:~$

```

ÇOKLU

Komut “ `docker start id_name ”`

Konteynırı başlatır.

Komut “ `docker stop id_name ”`

Konteynırı durdurur.

Komut “ `docker restart id_name ”`

Konteynırı yeniden başlatır

```

servet@Hegir:~$ sudo docker stop 9425747db1c8
9425747db1c8
servet@Hegir:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
9425747db1c8   nginx     "/docker-entrypoint..." 7 minutes ago  Up 14 seconds  0.0.0.0:8080->80/tcp, :::8080->80/tcp  mynginx

```

```

servet@Hegir:~$ sudo docker start 9425747db1c8
9425747db1c8

```

```

servet@Hegir:~$ sudo docker restart 9425747db1c8
9425747db1c8
servet@Hegir:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
9425747db1c8   nginx     "/docker-entrypoint..." 7 minutes ago  Up 14 seconds  0.0.0.0:8080->80/tcp, :::8080->80/tcp  mynginx

```