

DOCKER KOMUTLARI

pull → Dockerhub'dan image indirmemize yarıyor.

```
C:\Users\ozcan>docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
Digest: sha256:0a4eaa0eecf5f8c050e5bba433f58c052be7587ee8af3e8b3910ef9ab5fbe9f5
Status: Image is up to date for alpine:latest
docker.io/library/alpine:latest

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview alpine
```

push → Karşıya image yüklememizi sağlar.

```
C:\Users\ozcan>docker push alpine
Using default tag: latest
The push refers to repository [docker.io/library/alpine]
78561cef0761: Layer already exists
errors:
denied: requested access to the resource is denied
unauthorized: authentication required
```

logs → Kontainer loglarını görüntülememizi sağlar.

```
C:\Users\ozcan>docker logs yavuz
selamlar!
selamlar!
selamlar!
selamlar!
selamlar!
```

images → Image'ları liseliyor.

```
C:\Users\ozcan>docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
alpine        latest    324bc02ae123   45 hours ago   7.8MB
```

exec ls → Çalışan bir containerda komut çalıştırmamıza yarıyor.

```
C:\Users\ozcan>docker exec nice_hopper ls
bin
dev
etc
home
lib
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
```

run → Image dosyasını çalıştırıp yeni bir container oluşturur.

```
C:\Users\ozcan>docker run alpine
```

→ **name** → Yeni oluşan container ismini değiştirmemize yarıyor.

```
C:\Users\ozcan>docker run --name yavuzlar alpine
```

sleep 10 → Uyumasını sağlıyor.

```
C:\Users\ozcan>docker run alpine sleep 10
```

-rm → Konteynerin çalıştırıldıktan sonra silinmesini sağlıyor

```
C:\Users\ozcan>docker run --rm --name zayotem alpine
```

ps → Çalışan konteynerleri listeliyor.

```
C:\Users\ozcan>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
65a06aa810da	alpine	"sleep 20"	6 hours ago	Up 4 seconds		hardcore_euclid

-p → Konteynerin portu yönlendirmek için kullanılıyor.

start → Konteyneri çalıştırır.

```
C:\Users\ozcan>docker start yavuzlar
yavuzlar
```

ps -a → Tüm konteynerleri listeler.

```
C:\Users\ozcan>docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
d45b1cd4f1c0	alpine	"rm sleep 10"	13 minutes ago	Exited (1) 13 minutes ago		yavuzs
47e9b800f18b	alpine	"sleep 10"	16 minutes ago	Exited (0) 16 minutes ago		blissful_cohen
0d4f12d89f67	alpine	"/bin/sh"	18 minutes ago	Exited (0) About a minute ago		yavuzlar
d5edce257969	alpine	"/bin/sh"	19 minutes ago	Exited (0) 19 minutes ago		serene_kepler
34d36dfff0ee	alpine	"sleep 100"	6 hours ago	Exited (137) 21 minutes ago		nice_hopper
dce928f89189	alpine	"yavuzlar"	6 hours ago	Created		sleepy_gould
fad6d0515b35	alpine	"sleep 100"	6 hours ago	Exited (0) 6 hours ago		kind_allen
d766a3d31cd1	alpine	"sleep 100"	6 hours ago	Exited (137) 6 hours ago		ecstatic_shaw
65a06aa810da	alpine	"sleep 20"	6 hours ago	Exited (0) 3 minutes ago		hardcore_euclid
8e93c099c5e7	alpine	"cl"	6 hours ago	Created	0.0.0.0:80->80/tcp	recurring_payne
c18044391dd2	alpine	"/bin/sh"	6 hours ago	Exited (0) 6 hours ago		peaceful_hoover
a16700ee01ca	alpine	"sleep 10"	6 hours ago	Exited (0) 6 hours ago		pensive_sutherland
082065a51ea9	alpine	"echo selamlar!"	6 hours ago	Exited (0) About a minute ago		yavuz
db69eb28ae5c	alpine	"echo Selamlar!"	6 hours ago	Exited (0) 56 seconds ago		laughing_carson

-it → Terminali kullanabilmemizi sağlar.

```
C:\Users\ozcan>docker run -it alpine
/ # ls
bin    dev    etc    home  lib    media mnt    opt    proc   root   run    sbin   srv    sys    tmp    usr    var
/ # |
```

-d → Konteynerin arka planda çalışmasını sağlar.

```
C:\Users\ozcan>docker run -d alpine
2849da5b769f7130e04885616514758dea2135f23c293a9cdd417db41fe4e0db
```

network create → Kendi ağıımızı oluşturmamızı sağlıyor.

```
C:\Users\ozcan>docker network create yavuz
affcd6863c81adeb30916146f4a62361f1a48f1c50d5476ae186e4fe8ab3981a
```

network ls → Ağları listeliyor.

```
C:\Users\ozcan>docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
24eefa35dbc0	bridge	bridge	local
fb50e1dcea70	host	host	local
408b2e691f70	none	null	local
affcd6863c81	yavuz	bridge	local

inspect → Konteyner veya image hakkında bilgi verir.

```
C:\Users\ozcan>docker inspect yavuz
[
  {
    "Id": "082065a51ea937805236493db45308c036822ca4f0c6572512d210bb46b810b0",
    "Created": "2024-07-24T13:31:04.894574043Z",
    "Path": "echo",
    "Args": [
      "selamlar!"
    ],
    "State": {
      "Status": "exited",
      "Running": false,
      "Paused": false,
      "Restarting": false,
      "OOMKilled": false,
      "Dead": false,
      "Pid": 0,
      "ExitCode": 0,
      "Error": "",
      "StartedAt": "2024-07-24T19:47:29.311033213Z",
      "FinishedAt": "2024-07-24T19:47:29.311544557Z"
    }
  },
  ...
]
```

volume create → Volume oluşturur.

```
C:\Users\ozcan>docker volume create
ae541c872ee955aaf0a090cfe55a58286246b4b501c2de155446de1d3e4f0195
```

DOCKERFILE

FROM php:7.4-apache --> Konteynerin php:7.4-apache imageını temel alacağını belirtiyor.

WORKDIR /var/www/html --> Çalışma dizinini belirtiyor.

COPY ./app . --> Ana makineden konteyner içine dosya kopyalamamızı sağlar.

RUN echo "ServerName localhost" >> /etc/apache2/apache2.conf --> Hata vermemesi için gerekli bir konfigürasyon metni.

RUN apt-get update --> Hata almamak için en güncel sürüme yükseltmek için kullanılıyor.

RUN docker-php-ext-install pdo pdo_mysql --> pdo ve pdo_mysql uzantılarını yüklememizi sağlıyor.

EXPOSE 80 --> Docker container'ınızın hangi port üzerinden dış dünyaya erişilebilir olacağını belirtir.

DOCKER COMPOSE

version: '3' --> Versiyon 3 kullanılıyor.

services: --> Servislerin başladığı bölümü başlatıyor.

app: --> Uygulamanın başladığı belirtiliyor. Farklı şeyler de yazılabilir.

context: . --> Mevcut dizinde bulunduğunu belirtiyor.

dockerfile: Dockerfile -- > Kullanılacak Dockerfile'ı belirtir.

depends_on:
- **db** -- > Bağımlılık belirtiyordu. db servisine bağımlı.

ports:
- **"80:80"** -- > Container'ın hangi portlarının host ile eşleştirileceğini belirtir. Host'un 80 portunu container'ın 80 portuna yönlendirir.

networks:
- **net** --> net ağını kullanılacağını belirtir.

db: -- > Veritabanı servisini tanımlıyor.

image: mysql:latest -- > En son MySQL image'ını kullanır.

environment: -- > Ortam değişkenlerini belirtir.

- **MYSQL_DATABASE=yavuzlar** -- > MySQL veritabanı adını yavuzlar olarak ayarlar.

- **MYSQL_ROOT_PASSWORD=1** -- > MySQL root şifresini 1 olarak ayarlar.

volumes: -- > Veritabanı verilerini saklamak ve paylaşmak için kullanılan volümleri belirtir.

- **db_data:/var/lib/mysql** -- > Veritabanı verilerini container dışındaki bir alanda saklar.

- **./yavuzlar_messages.sql:/docker-entrypoint-initdb.d/yavuzlar_messages.sql**

-- > yavuzlar_messages.sql dosyasını MySQL başlangıç dosyası olarak kopyalar.

ports:
- **"8080:3306"** -- > Host'un 8080 portunu container'ın 3306 portuna yönlendirir.

networks:
- **net** -- > net ağını kullanılacağını belirtir.

networks:
net:
driver: bridge -- > Ağ sürücüsünü olarak bridge kullanır.

volumes:
db_data: -- > db_data adlı volüm tanımı yapılıyor.