

Yılmaz ÜSTÜNTAŞ Docker Ödevi

docker images : kurulu imajlarla ilgili bilgileri sunar.

docker image rm iamege name : belirli imajı siler.

```
C:\Windows\system32\cmd.exe: x + v
C:\Users\Yilmaz>docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
alpine latest 324bc02ae123 2 days ago 7.8MB
nginx latest a72860cb95fd 4 weeks ago 188MB

C:\Users\Yilmaz>docker image rm alpine
Untagged: alpine:latest
Untagged: alpine@sha256:0a4eaa0eefcf5f8c050e5bba433f58c052be7587ee8af3e8b3910ef9ab5f9e9f5
Deleted: sha256:324bc02ae1231fd9255658c128086395d3fa0aedd5a41ab6b034fd649d1a9260
Deleted: sha256:78561cef0761903dd2f7d09856150a6d4fb48967a8f113f3e33d79effbf59a07

C:\Users\Yilmaz>|
```

docker : docker komutlarını kullanım şeklini gösterir.

```
C:\Windows\system32\cmd.exe: x + v
C:\Users\Yilmaz>docker
Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Common Commands:
run      Create and run a new container from an image
exec     Execute a command in a running container
ps       List containers
rm       Remove one or more containers
pull     Pull an image from a registry
push     Push an image to a registry
images   List images
login    Log in to a registry
logout   Log out from a registry
search   Search Docker Hub for images
version  Show the Docker version information
info     Display system-wide information

Management Commands:
builder  Manage builds
buildx   Docker Buildx
compose  Docker Compose
container Manage containers
context  Manage contexts
debug    Get a shell into any image or container
exec     Execute a command in a running container
events   Docker Events
logs     Docker Log
network  Manage Docker networks
plugin   Manage Docker plugins
storage  Manage Docker storage
system   Manage Docker
trust    Manage Docker trust
volume   Manage Docker volumes

Source Commands:
save     Save images

Commands:
attach   Attach local standard input, output, and error streams to a running container
commit   Create a new image from a container's changes
cp       Copy files/folders between a container and the local filesystem
create   Create a new container
diff     Inspect changes to files or directories on a container's filesystem
events   Get real time events from the server
export   Export a container's filesystem as a tar archive
history  Show the history of an image
import   Import the contents from a tarball to create a filesystem image
inspect  Return low-level information on Docker objects
kill     Kill one or more running containers
load     Load an image from a tar archive or STDIN
logs     Fetch the logs of a container
pause    Pause all processes within one or more containers
port     List port mappings or a specific mapping for the container
prune    Remove all stopped containers
restart  Restart one or more containers
rm       Remove one or more containers
rmi       Remove one or more images
save     Save one or more images to a tar archive (streamed to STDOUT by default)
start    Start one or more stopped containers
stats    Display a live stream of container(s) resource usage statistics
stop     Stop one or more running containers
tag       Create a new TAG for an existing image
top       Display the running processes of a container
unpause  Unpause all processes within one or more containers
update   Update configuration of one or more containers
wait     Block until one or more containers stop, then print their exit codes

Global Options:
--config string Location of client config file (default
                  %UserProfile%\docker)
--context string Name of the context to use to connect to the
```

docker search image name : docker hub'daki imagerleri aramak için kullanılır.

```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker search alpine

NAME                DESCRIPTION                STARS     OFFICIAL
alpine              A minimal Docker image based on Alpine Linux. 109941    [OK]
alpine/docker-cli   Simple and lightweight Alpine Linux image wi... 11
alpine/alpine-gitlab-ci Build Alpine Linux packages with Gitlab CI 3
alpine/gitlab-runner-helper Helper image container gitlab-runner-helper ... 7
alpine/rsyncd        2
alpine/unbound       13
alpine/alpine-drone-ci Build Alpine Linux packages with drone CI 0
alpine/docker-alpine 0
alpine/ansible       21
alpine/gitlab-runner Alpine Linux gitlab-runner (supports more ar... 7
grafana/alpine       7
alpine/docker-compose docker-compose image based on Alpine Linux 2
alpine/apkbuild-lint-tools Tools for linting APKBUILD files in a CI env... 0
bellsoft/iberica-openjdk-alpine Liberica is a 100% open-source Java implemen... 58
alpine/darkhttpd      2
alpine/golang         3
alpine/alpine-docker-gitlab Gitlab running on Alpine Linux 0
alpine/build-base     0
alpine/alpine-www     0
bellsoft/iberica-openjdk-alpine Liberica is a 100% open-source Java implemen... 22
alpine/docker-abuild 0
bellsoft/iberica-openjdk-alpine-musl Liberica is a 100% open-source Java implemen... 24
balenalib/amd64-alpine-node This image is part of the balena.io base ima... 2
alpine/mqtt-exec      0
alpine/netbox         0
C:\Users\Yilmaz>
```

docker pull image name : docker hub tan image indirmek için kullanılır.

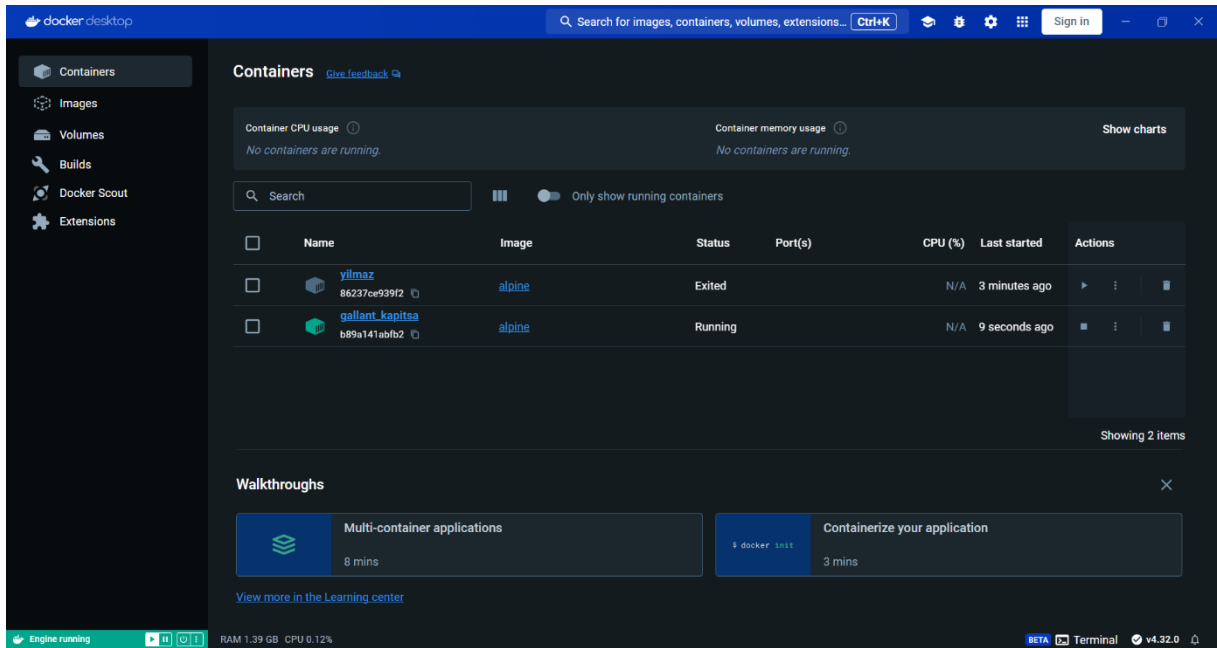
```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
Digest: sha256:0a4eaa0eefcf5f8c050e5bba433f58c052be7587ee8af3e8b3910ef9ab5f5be9f5
Status: Image is up to date for alpine:latest
docker.io/library/alpine:latest

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview alpine
C:\Users\Yilmaz>
```

docker run options image commands: image kullanarak konteyner oluřturur.

docker run --rm image sleep 60 : image iři bitince kendini siler.

```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker run --rm alpine sleep 60
```



docker run --name image : konteynera özel isim koyar.

The screenshot shows the Docker Desktop application. The top bar is blue with the Docker logo and a search bar. The left sidebar contains navigation icons for Containers, Images, Volumes, Builds, Docker Scout, and Extensions. The main panel is titled 'Containers' and shows a summary of container usage. Below this, there is a table of containers. The table has columns for Name, Image, Status, Port(s), CPU (%), Last started, and Actions. One container is listed: 'yilmaz' (ID: 86237ce939f2) running on the 'alpine' image, with a status of 'Exited'. Below the table, there are 'Walkthroughs' for 'Multi-container applications' and 'Containerize your application'. At the bottom, a terminal window shows the command 'docker run --name yilmaz alpine' being executed.

Container CPU usage ⓘ
No containers are running.

Container memory usage ⓘ
No containers are running.

Show charts

Search

Only show running containers

<input type="checkbox"/>	Name	Image	Status	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	yilmaz 86237ce939f2	alpine	Exited		N/A	10 seconds ago	▶ ⋮ ■

Showing 1 item

Walkthroughs

Multi-container applications
8 mins

Containerize your application
3 mins

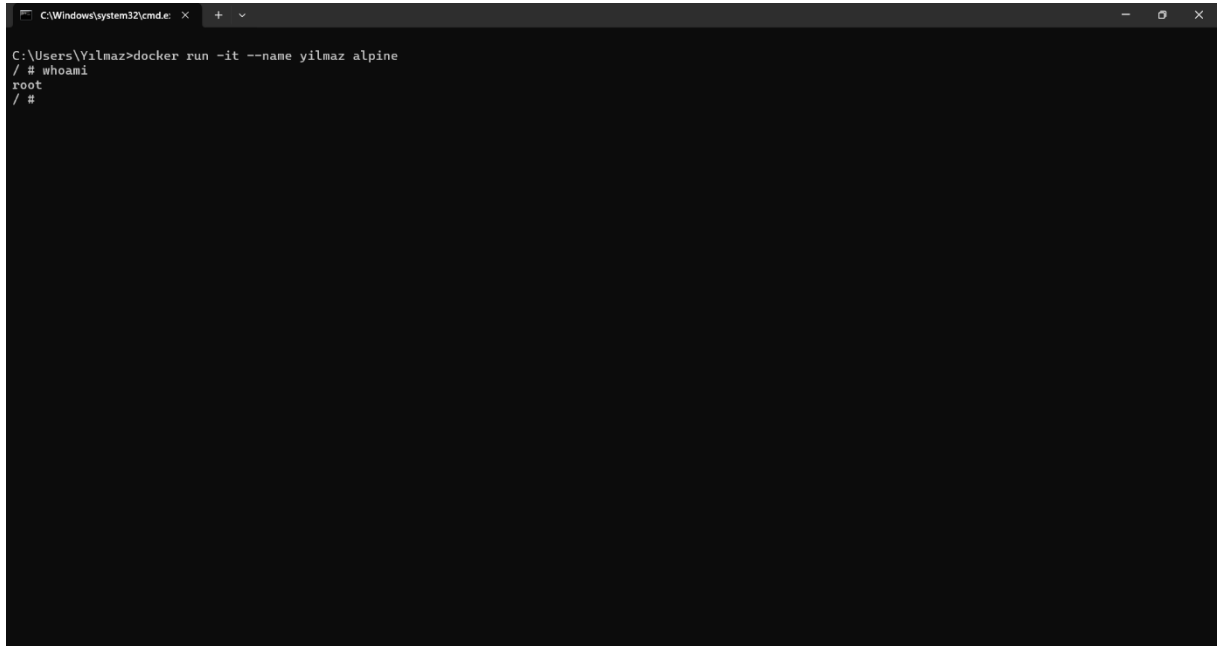
View more in the Learning center

Engine running RAM 1.38 GB CPU 0.06%

Terminal v4.32.0

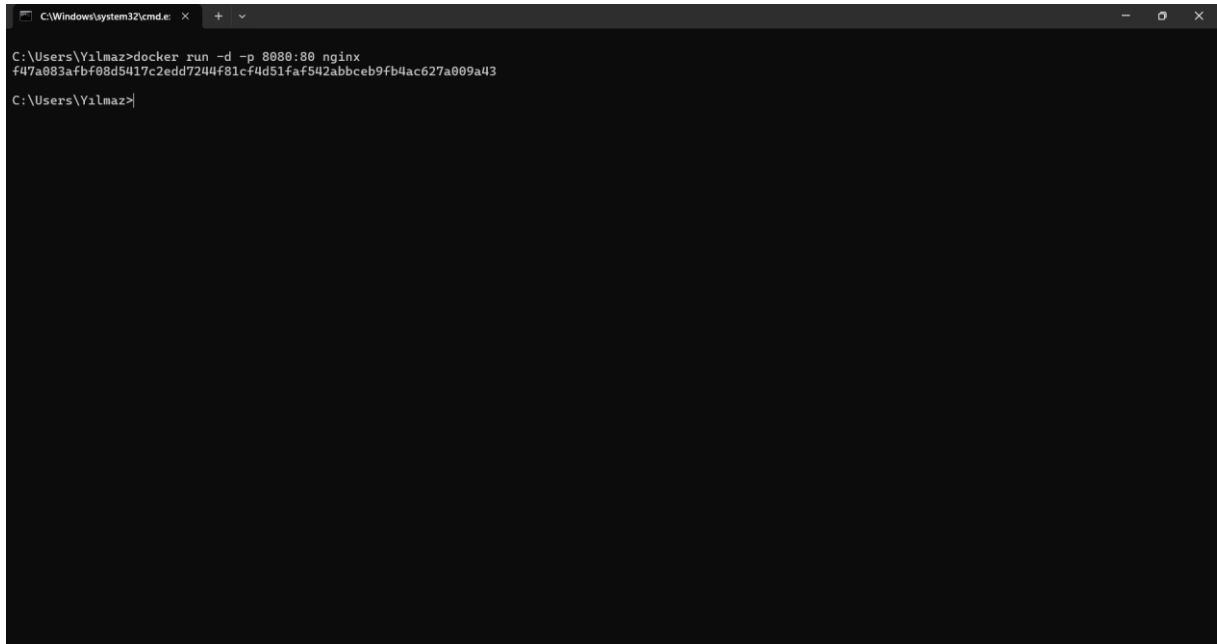
```
C:\Users\Yilmaz>docker run --rm alpine
C:\Users\Yilmaz>docker run --name yilmaz alpine
C:\Users\Yilmaz>
```

docker run -it --name yilmaz alpine : konteynera özel isim koyar ve terminal almamızı sağlar.

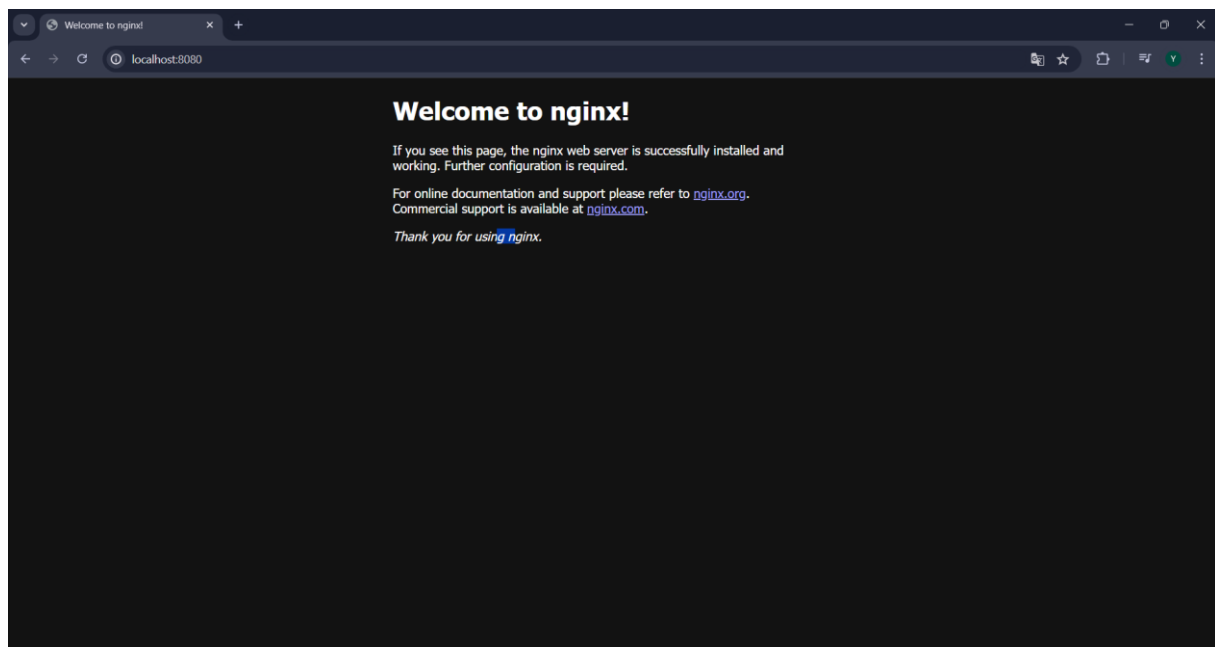
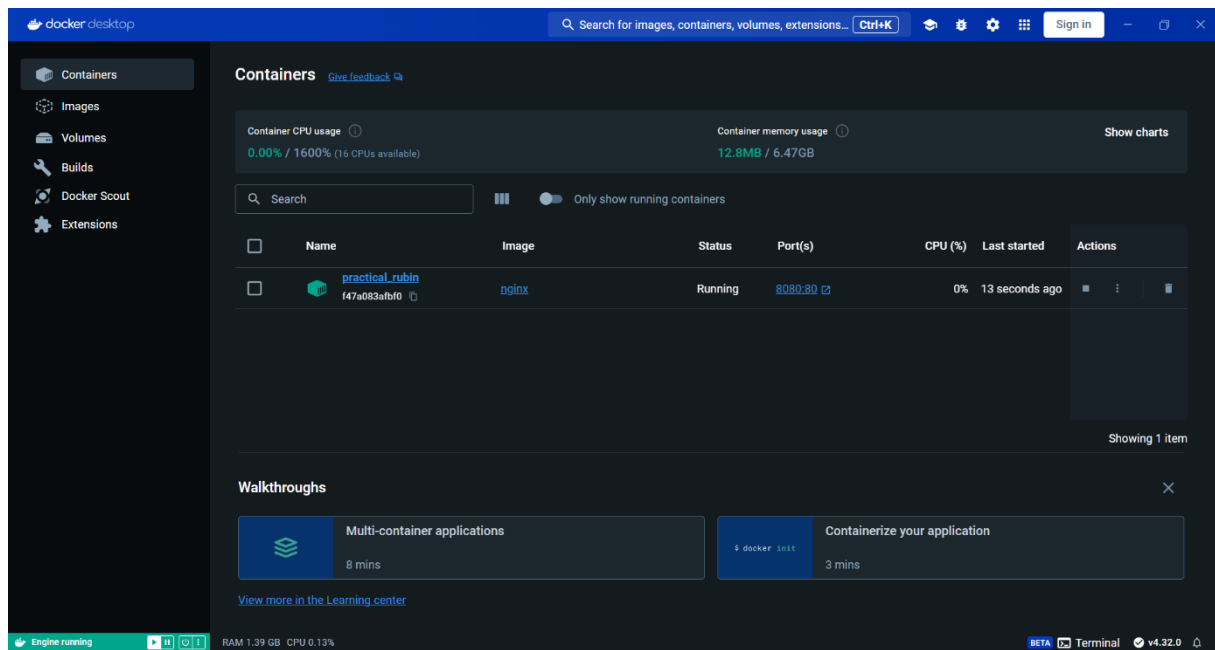


```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker run -it --name yilmaz alpine
/# whoami
root
/#
```

docker run -d -p 8080:80 nginx : -d ifadesi arkaplanda çalıştırmayı ifade eder -p ifadesi ise belirlediğimiz portu ana makineye yönlendirir.



```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker run -d -p 8080:80 nginx
f47a083afb08d5417c2edd7244f81cf4d51faf542abbceb9fb4ac627a009a43
C:\Users\Yilmaz>
```



docker image inspect image name : image hakkında detaylı bilgileri gösterir.

```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker image inspect alpine
[
  {
    "Id": "sha256:320bc82ae121f6925568c128886396d3faaed5a1ab0834f6d49d1a9260",
    "RepoTags": [
      "alpine:latest"
    ],
    "RepoDigests": [
      "alpine@sha256:b6aaaf0ecf5f8c85d0bba413f8dc852b07587eabaf3e83910ef9ab5fb09f5"
    ],
    "Parent": "",
    "Comment": "",
    "Created": "2020-07-22T22:26:43.778747613Z",
    "DockerVersion": "23.0.11",
    "Author": "",
    "Config": {
      "Hostname": "",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStdout": false,
      "AttachStderr": false,
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": [
        "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
      ],
      "Cmd": [
        "/bin/sh"
      ],
      "Image": "sha256:3e25d883ea289c810d30867ada83c56bd09c50f93f6b228ba2fe74f6819ad2",
      "Labels": null,
      "WorkingDir": "",
      "Entrypoint": null,
      "OnBuild": null,
      "Labels": null
    },
    "Architecture": "amd64",
    "Os": "linux",
    "Size": 7797760,
    "VirtualSize": {
      "Data": {
        "MergeDir": "/var/lib/docker/overlay2/5595d8ce8f031a0b2ee08ba1d3c2688195a01e483f5324e13928f1a405b65/merged",
        "UpperDir": "/var/lib/docker/overlay2/5595d8ce8f031a0b2ee08ba1d3c2688195a01e483f5324e13928f1a405b65/diff",
        "WorkDir": "/var/lib/docker/overlay2/5595d8ce8f031a0b2ee08ba1d3c2688195a01e483f5324e13928f1a405b65/work"
      },
      "Size": "overlay2"
    },
    "RootFS": {
      "Type": "layers",
      "Layers": [
        "sha256:78561c0f9761983d27f08985615badd0f048967abf113f3e33d799efbf09a87"
      ]
    },
    "Metadata": {
      "LastTagTime": "2021-01-01T00:00:00Z"
    }
  }
]
```

docker network create yilmaz : yilmaz isminde özel ağ oluşturur.

docker network ls : ağları görüntüler.

docker network rm network_id : girilen network id ye sahip ağ siler.

```
C:\Windows\system32\cmd.exe
C:\Users\Yilmaz>docker network create yilmaz
c654ab495e5a047c2758d7f03c153fc927e6212f39fedbc50520606777c46379

C:\Users\Yilmaz>docker network ls
NETWORK ID        NAME      DRIVER  SCOPE
b93241767fe6     bridge   bridge  local
c0b08c1d3d74     host     host    local
49e59d947531     none     null     local
c654ab495e5a     yilmaz   bridge  local

C:\Users\Yilmaz>docker network rm c654ab495e5a
c654ab495e5a

C:\Users\Yilmaz>docker network ls
NETWORK ID        NAME      DRIVER  SCOPE
b93241767fe6     bridge   bridge  local
c0b08c1d3d74     host     host    local
49e59d947531     none     null     local

C:\Users\Yilmaz>
```

docker run -v host_path:container_path image name : ana makinadaki dizini konteyneradaki dizine bağlar.

```
C:\Users\Yilmaz>docker run -v C:\Users\Yilmaz\Desktop:app nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/07/24 23:11:40 [notice] 1#1: using the "epoll" event method
2024/07/24 23:11:40 [notice] 1#1: nginx/1.27.0
2024/07/24 23:11:40 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/07/24 23:11:40 [notice] 1#1: OS: Linux 5.15.153.1-microsoft-standard-WSL2
2024/07/24 23:11:40 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/07/24 23:11:40 [notice] 1#1: start worker processes
2024/07/24 23:11:40 [notice] 1#1: start worker process 29
2024/07/24 23:11:40 [notice] 1#1: start worker process 30
2024/07/24 23:11:40 [notice] 1#1: start worker process 31
2024/07/24 23:11:40 [notice] 1#1: start worker process 32
2024/07/24 23:11:40 [notice] 1#1: start worker process 33
2024/07/24 23:11:40 [notice] 1#1: start worker process 34
2024/07/24 23:11:40 [notice] 1#1: start worker process 35
2024/07/24 23:11:40 [notice] 1#1: start worker process 36
2024/07/24 23:11:40 [notice] 1#1: start worker process 37
2024/07/24 23:11:40 [notice] 1#1: start worker process 38
2024/07/24 23:11:40 [notice] 1#1: start worker process 39
2024/07/24 23:11:40 [notice] 1#1: start worker process 40
2024/07/24 23:11:40 [notice] 1#1: start worker process 41
2024/07/24 23:11:40 [notice] 1#1: start worker process 42
2024/07/24 23:11:40 [notice] 1#1: start worker process 43
2024/07/24 23:11:40 [notice] 1#1: start worker process 44
```

docker ps : çalışan konteynerları gösterir.

```
C:\Users\Yilmaz>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS      NAMES
c6f9d5f8506f   nginx    "/docker-entrypoint...." 21 seconds ago Up 21 seconds 80/tcp     yilmaz3

C:\Users\Yilmaz>
```


docker exec yilmaz ls -l : yilmaz isminde ki konteynerda ls -l çalıştırır.

docker stop yilmaz : yilmaz ismindeki konteynerı durdurur.

docker start yilmaz : yilmaz ismindeki konteynerı başlatır.

docker restart yilmaz : yilmaz ismindeki konteynerı yeniden başlatır.

docker rm yilmaz : yilmaz ismindeki konteynerı siler.

```
C:\Windows\system32\cmd.exe X + -
C:\Users\Yilmaz>docker run -d --name yilmaz nginx
5faeaf95f6cdf991587a834610bc7ace75d31f07a9ac6f04e9d5bd7a08f9ba5

C:\Users\Yilmaz>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
5faeaf95f6dc   nginx    "/docker-entrypoint..." 5 seconds ago  Up 5 seconds  80/tcp  yilmaz

C:\Users\Yilmaz>docker exec yilmaz ls -l
total 64
lrwxrwxrwx 1 root root    7 Jul 22 00:00 bin -> usr/bin
drwxr-xr-x 2 root root 4096 Mar 29 17:28 boot
drwxr-xr-x 3 root root 368 Jul 24 23:28 dev
drwxr-xr-x 1 root root 4096 Jul 23 07:14 docker-entrypoint.d
-rwxr-xr-x 1 root root 1620 Jul 23 07:14 docker-entrypoint.sh
drwxr-xr-x 1 root root 4096 Jul 24 23:28 etc
drwxr-xr-x 2 root root 4096 Mar 29 17:28 home
lrwxrwxrwx 1 root root    7 Jul 22 00:00 lib -> usr/lib
lrwxrwxrwx 1 root root    9 Jul 22 00:00 lib64 -> usr/lib64
drwxr-xr-x 2 root root 4096 Jul 22 00:00 media
drwxr-xr-x 2 root root 4096 Jul 22 00:00 mnt
drwxr-xr-x 2 root root 4096 Jul 22 00:00 opt
dr-xr-xr-x 339 root root    0 Jul 24 23:28 proc
drwxr-xr-x 2 root root 4096 Jul 22 00:00 root
drwxr-xr-x 1 root root 4096 Jul 24 23:28 run
lrwxrwxrwx 1 root root    8 Jul 22 00:00 sbin -> usr/sbin
drwxr-xr-x 2 root root 4096 Jul 22 00:00 srv
dr-xr-xr-x 11 root root    0 Jul 24 23:28 sys
drwxrwxrwt 2 root root 4096 Jul 22 00:00 tmp
drwxr-xr-x 1 root root 4096 Jul 22 00:00 usr
drwxr-xr-x 1 root root 4096 Jul 22 00:00 var

C:\Users\Yilmaz>docker exec yilmaz echo yilmaz
yilmaz

C:\Users\Yilmaz>docker stop yilmaz
yilmaz

C:\Users\Yilmaz>docker start yilmaz
yilmaz

C:\Users\Yilmaz>docker restart yilmaz
yilmaz

C:\Users\Yilmaz>docker rm yilmaz
Error response from daemon: cannot remove container "/yilmaz": container is running: stop the container before removing or force remove

C:\Users\Yilmaz>docker stop yilmaz
yilmaz

C:\Users\Yilmaz>docker rm yilmaz
yilmaz

C:\Users\Yilmaz>
```