



SAKARYA
ÜNİVERSİTESİ

SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ

SİSTEM YÖNETİCİLİĞİ
BİR PROGRAMIN DOCKERİZE EDİLMESİ
2.ÖDEV

ENGİN YENİCE
S201200019

İçindekiler

Docker Nedir?.....	1
Dockerfile Nedir?.....	1
Dockerfile Dosyasının Çalıştırma Adımları.....	1
Dockerfile Dosyası	1
Dockerfile Dosyasının Build Edilmesi.....	1
Oluşturulan Image'in Görüntülenmesi.....	2
Oluşturulan Image'in Çalıştırılması.....	2
Çalıştırılan Container'ın Bilgilerinin Görüntülenmesi	3
Çalışır Durumdaki Container'ın Görüntüleri.....	4

Docker Nedir?

Docker, "konteynerleştirme" olarak da bilinen işletim sistemi seviyesinde sanallaştırma sağlayan bir bilgisayar programıdır. İlk sürümü 2013'te yayınlanmıştır. Docker, "konteyner" adı verilen yazılım paketlerini çalıştırmak için kullanılmaktadır.

Dockerfile Nedir?

Dockerfile, Docker Image'ı oluşturmak için kullanılan içeriğinde tanımlar bulunan bir dosyadır. Dosya adı kesinlikle "Dockerfile" şeklinde olmalıdır. "dockerfile", "DockerFile" gibi verilen isimlendirmeler yanlıştır. Dockerfile dosyasının herhangi bir uzantısı bulunmamaktadır.

Dockerfile Dosyasının Çalıştırma Adımları

Dockerfile Dosyası

```
Dockerfile M X
Dockerfile
1 FROM php:7.0-apache
2 #DockerHub üzerinde bulunan php:7.0-apache sürümü belirlendi.
3 #https://hub.docker.com/_/php üzerinden
4 COPY ./source/index.php /var/www/html/index.php
5 # source klasörü içerisinde bulunan index.php dosyasını container içerisinde ki /var/www/html/ klasörü içerisine kopyalandı.
```

FROM etiketi ile DockerHub üzerinden indireceğimiz paketin ne olduğunu belirtiyoruz.

- Burada php:7.0-apache paketinin indirileceğini belirtmiş olduk.

COPY komutu ile kendi bilgisayarımızdan container içerisine kopyalayacağımız dosyaları belirtiyoruz.

- Kendi bilgisayarımızda bulunan ./source/index.php dosyasının container içerisinde /var/www/html/index.php adresine kopyalanacağını belirtmiş olduk.

Dockerfile Dosyasının Build Edilmesi

Dockerfile dosyamızın bulunduğu dizine komut yönetici(cmd,terminal,vb.) ile gidiyoruz. Ardından build etme komudumuzu girip build etme işlemine başlıyoruz.

```
docker build -t enginyenice/sistem-yoneticiligi-odev:0.0.1 .
```

Girdiğimiz build etme komutunu detaylı bir şekilde inceleyecek olursak eğer:

- **docker:** Gerçekleştirilecek işlemin Docker cli yardımı ile yapılacağını belirtiyoruz.

- **build:** Yeni bir image oluşturacağımızı belirtiyoruz.
- **-t, --tag list:** Oluşturulacak image e özel bir isim vereceğimizi belirtiyoruz. (opsiyonel bir işlemdir. İsim verilmemesi durumunda rastgele bir isimle oluşturulabilir.)
- **enginyenice/sistem-yoneticiligi-odev:0.0.1:** Oluşturulacak image e vereceğimiz isimdir.
- **.** : Dockerfile dosyasının nerede bulunduğunu belirtiyoruz. Nokta (.) bulunduğunuz dizinde olduğunu belirtmek için kullanılmıştır.

```
PS C:\Users\Engin Yenice\Desktop\Yaz-Okulu\Sistem Yöneticiliği\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma> docker build -t enginyenice/sistem-yoneticiligi-odev:0.0.1 .
[+] Building 6.2s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 353B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/php:7.0-apache
=> [auth] library/php:pull token for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 60B
=> CACHED [1/2] FROM docker.io/library/php:7.0-apache@sha256:1d34b2e491a82ba7a8d26478132015e197a5ffea37f8a93b42621d1cfe042cc
=> [2/2] COPY ./source/index.php /var/www/html/index.php
=> exporting to image
=> => exporting layers
=> writing image sha256:87d6c6c0e9d35c7cafe3fbb2c420c329f450870c4890abf92b54fbfce3fe687f
=> naming to docker.io/enginyenice/sistem-yoneticiligi-odev:0.0.1
```

Oluşturulan Image'in Görüntülenmesi

Oluşturduğumuz tüm imajeleri

Docker images

Komutunu kullanarak görüntüleyebiliriz.

- **Repository:** Bilgisayarda kayıtlı olan imageslerin isimlerini göstermektedir.
- **Tag:** Bilgisayarda kayıtlı olan imagesin versiyonunu belirtmektedir.
- **Image Id:** Docker tarafından otomatik olarak atanan benzersiz kimlik numarasıdır.
- **Created:** Ne zaman oluşturulduğu ile ilgili bilgi vermektedir.
- **Size:** Image 'in hafızada kapladığı boyutu belirtmektedir.

```
PS C:\Users\Engin Yenice\Desktop\Yaz-Okulu\Sistem Yöneticiliği\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma> docker images
REPOSITORY          TAG          IMAGE ID      CREATED        SIZE
enginyenice/sistem-yoneticiligi-odev  0.0.1       87d6c6c0e9d3  12 minutes ago 368MB
PS C:\Users\Engin Yenice\Desktop\Yaz-Okulu\Sistem Yöneticiliği\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma> |
```

Oluşturulan Image'in Çalıştırılması

Oluşturduğumuz image'i çalıştırmak için komut yönetici(cmd,terminal,vb.) ile gidiyoruz. Ardından gerekli komutu yazarak çalıştırma işlemini gerçekleştiriyoruz.

```
docker run -d --rm -p 8080:80 --name sy-running-app enginyenice/sistem-  
yoneticiligi-odev:0.0.1
```

Girdiğimiz çalıştırma(run) komutunu detaylı bir şekilde inceleyecek olursak eğer:

- **docker:** Gerçekleştirilecek işlemin Docker cli yardımı ile yapılacağını belirtiyoruz.
- **run:** Bir başlatma (çalıştırma) işlemi gerçekleştireceğimizi belirtiyoruz.
- **-d, --detach:** Çalıştırılacak olan Container'ın arka planda çalışması gerektiğini belirtiyoruz. (Opsiyoneldir, kullanılmaması durumunda terminal üzerinde çalıştırılacaktır.)
- **--rm:** Aynı isimde çalışan başka bir container varsa eğer onu silmesi gerektiğini belirtiyoruz. (Opsiyoneldir. Yazılmaması durumunda arka planda aynı isimde çalışan bir container varsa hata verecektir ve çalışan Container'ın silmenizi isteyecektir.)
- **-p, --publish list:** Container içerisinde çalışan bir portu dış dünyaya açmamıza yararmaktadır. Kullanımı ise, dışarıdan erişilecek port numarası : içeride çalışan port numarası olarak kullanılabilir. (Opsiyoneldir. Açılmaması durumunda dışarıdan içeride çalışan porta erişim sağlanmayacaktır.)
- **--name:** Çalıştırılacak olan Container'a özel bir isim vermek için kullanılmaktadır. (Opsiyoneldir. Kullanılmaması durumunda rastgele bir isim ile oluşturulacaktır.)
- **sy-running-app:** Çalıştırılacak Container'a verdiğimiz özel isimdir.
- **enginyenice/sistem-yoneticiligi-odev:0.0.1:** Çalıştırılacak olan Container'ın hangi image'i kullanacağını belirtiyoruz.

```
PS C:\Users\Engin_Yenice\Desktop\Yaz-Okulu\Sistem_Yoneticiligi\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma> docker run -d --rm -p 8080:80 --name sy-running-app enginyenice/sistem-yoneticiligi-odev:0.0.1  
2194bf86f1ed75b4d22e051f72df90535fab48bfeadac2bf19d7b941ede7d63  
PS C:\Users\Engin_Yenice\Desktop\Yaz-Okulu\Sistem_Yoneticiligi\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma>
```

Çalıştırılan Container'ın Bilgilerinin Görüntülenmesi

Çalıştırılmak üzere oluşturulan Container ile ilgili bilgilere ulaşmak için,

```
docker ps
```

Komutunu kullanabiliriz. Komutu çalıştırdıktan sonra Container'lar ile ilgili bazı bilgileri bizimle paylaşmaktadır. Bu bilgiler:

- **Container ID:** Docker tarafından container'a verilen benzersiz kimlik numarasının bir kısmıdır.

- **Image:** Hangi image'i kullandığını göstermektedir.
- **Command:** Container oluşturulurken çalıştırılması gereken temel komutlardır.
- **Created:** Oluşturulduğu zaman.
- **Status:** Çalışma durumu
- **Ports:** Çalıştığı portları ve iç ve dış portlarını göstermektedir.
- **Names:** Container'ın ismidir.

```
PS C:\Users\Engin_Yenice\Desktop\Yaz-Okulu\Sistem Yöneticiliği\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma> docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
2194bf86f1ed   enginyenice/sistem-yoneticiligi-odev:0.1   "docker-php-entrypoi..." 13 minutes ago   Up 13 minutes   0.0.0.0:8080->80/tcp, :::8080->80/tcp   sy-running-app
PS C:\Users\Engin_Yenice\Desktop\Yaz-Okulu\Sistem Yöneticiliği\Odev Dosyaları\Dockerfile oluşturma ve çalıştırma>
```

Çalışır Durumdaki Container'ın Görüntüleri

Tarayıcı üzerinden <http://localhost:8080/> adresine gidilerek. Container'ın çalışır durumda ki web sitesine ulaşabilirsiniz.

