

Dockerfile – Moj prvi image



Konačno smo dogurali do onog zanimljivog dela. Do pravljenja naše verzije nekog imidža. Ako vam je kojim slučajem izvetro šta je imidž, ne krivim vas, mnogo sam veliki razmak napravio između tekstova. Podsetite se klikom na [OVAJ](#) link

Pa da zaronimo. Dockerfile je srce svakog imidža. Svi do poslednjeg image koji ste preuzeli sa nekog registra napravljeni su na ovaj način.

Napomena: Dockerfile, mora baš tako i da se imenuje veliko D je obavezno. Bez ekstenzije je.

Udjite u test direktorijum, otvorite terminal. I kreirajte Dockerfile

```
touch Dockerfile
```

Editujte ga editorom koji vam odgovara. Ja ću koristiti nano.

```
nano Dockerfile
```

Hajde da napravimo imidž koji će biti zasnovan na php 7.2 fpm. I da mu ubacimo samo najpotrebnije stvari. Naravno i da ga ažuriramo.

```
FROM php: 7.2-fpm

RUN apt-get update && apt-get install -y

CMD ["php-fpm"]

EXPOSE 9000
```

Da raščlanimo:

`FROM php: 7.2-fpm` – Ovim dockeru kažemo da preuzme imidž iz repozitorijuma php sa tagom 7.2-fpm. Ovako možete modifikovati bilo koji image koji imate u lokalnim repozitorijumima ili ste od negde preuzeli. Time će biti kreiran prvi sloj(layer) našeg imidža. A Dockerfile php:7.2-fpm možete pogledati [ovde](#).

`RUN apt-get update && apt-get install -y` – Ovom komandom stvaramo novi sloj. Operativni sistem na kome je zasnovan php 7.2 fpm(debian:buster-slim) ažuriraće se i postaviti sloj broj 2 unutar ovog imidža.

RUN komandu, koristite isključivo za proces izvršavanja instalacija i modifikacija koji će se upisati u vašu verziju imidža.

`CMD ["php-fpm"]` – Po stvaranju kontejnera od ovog imidža, izvršiće se ova komanda. I tako će docker kontejner ostati da radi, neće pasti. U stvari mi samo pokrećemo php-fpm.

CMD – Ako imate više CMD direktiva u okviru jednog dockerfajla, izvršiće se sve u build procesu, sem poslednje. Ona će ostati da se izvrši pri podizanju kontejnera. Ovo je bitno znati, da se ne oslanjate na njih za više stvari koje treba uraditi na inicijalizaciji kontejnera. Za te stvari, koristimo ENTRYPOINT.

`EXPOSE 9000` – To znači da vam port 9000 neće biti otvoren kada pokrenete ovaj kontejner ka host mašini, već samo ka dockerovoj mreži.

Konačno buildanje

```
sudo docker build -t ime_imidza .
```

Komanda se pokreće iz direktorijuma gde je smešten Dockerfile. -t označava tag i obavezna je

tačka na kraju, kako bi dockeru rekli da smo unutar direktorijuma gde nam je Dockerfile.

```
sistemas@sistemas-VirtualBox:~/Desktop/test$ sudo docker build -t moj_image .
Sending build context to Docker daemon 3.584kB
Step 1/4 : FROM php:7.2-fpm
7.2-fpm: Pulling from library/php
1ab2bdf9778: Already exists
1448c64389e0: Pull complete
4b8a4e62b444: Pull complete
9eb9d1e8e241: Pull complete
1ea1192f8448: Pull complete
ec5a150f7008: Pull complete
f39d8e1e9c06: Pull complete
ec9ce5c6ea3e: Pull complete
dab566ec6646: Pull complete
d978f8b39145: Pull complete
4a364e7ada90: Pull complete
Digest: sha256:43868cbbdb154b8984a563201bab31b0165f95a5816a47b5d6206ecd1aa27a84
Status: Downloaded newer image for php:7.2-fpm
---> 8bbd5f15e775
Step 2/4 : RUN apt-get update && apt-get install -y
---> Running in 59b1306ee08e
Get:1 http://security-cdn.debian.org/debian-security buster/updates InRelease [39.1 kB]
Get:2 http://cdn-fastly.deb.debian.org/debian buster InRelease [122 kB]
Get:3 http://cdn-fastly.deb.debian.org/debian buster-updates InRelease [49.3 kB]
Get:4 http://security-cdn.debian.org/debian-security buster/updates/main amd64 Packages [82.1 kB]
Get:5 http://cdn-fastly.deb.debian.org/debian buster/main amd64 Packages [7899 kB]
Get:6 http://cdn-fastly.deb.debian.org/debian buster-updates/main amd64 Packages [884 B]
Fetched 8191 kB in 11s (725 kB/s)
Reading package lists...
Reading package lists...
Building dependency tree...
Reading state information...
The following package was automatically installed and is no longer required:
  lsb-base
Use 'apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 21 not upgraded.
Removing intermediate container 59b1306ee08e
---> f07fe1b702d7
Step 3/4 : CMD ["php-fpm"]
---> Running in 46cbf13b0f46
Removing intermediate container 46cbf13b0f46
---> 3e3e3f28b9c5
Step 4/4 : EXPOSE 9000
---> Running in 5a264f6e06dc
Removing intermediate container 5a264f6e06dc
---> 79bf6710f563
Successfully built 79bf6710f563
Successfully tagged moj_image:latest
```

Kao što vidite, imali smo 4 komande u našem Dockerfajlu. Sve 4 su kreirale svoje kontejnere u okviru kojih su se izvršile i postale lajersi tj slojevi. I na kraju smo stvorili novi image. Sada pustite mašti na volju, vidite koliko se lako prave imidži. A jednom napravljeni imidž, možete na milion mesta koristiti.

I na kraju da vam napomenem. Prijavite se na moju mailing listu, ili na telegram kanal.

<https://t.me/LudiSistemas> . U slučaju da se desi neki problem sa fejsbuk stranicom, da ostanemo u kontaktu.

NAPOMENA: Ovo i sva uputstva koja slede nisu oficijalna i moja su interpretacija originalne dokumentacije i uslova pod kojim se izvode. Tako da mogu i ne moraju da budu ispravna.

Revision #1

Created 10 July 2021 14:24:50 by Admin

Updated 10 July 2021 14:25:10 by Admin