Лабораторная работа №2

Dockerfile

Для выполнения данной лабораторной работы я создала у себя на развернутом на Vscale сервере папку, в которой с помощью терминала создала Dockerfile.

```
root@for-pit-course:/# mkdir site2
root@for-pit-course:/# touch Dockerfile
root@for-pit-course:/# ls
Dockerfile  dev   lib    libx32      mnt   root  site   srv  usr
bin         etc   lib32  lost+found  opt   run   site2  sys  var
boot        home  lib64  media       proc  sbin  snap   tmp
root@for-pit-course:/# rm Dockerfile
root@for-pit-course:/# ls
bin   etc   lib32   lost+found  opt   run   site2  sys  var
boot  home  lib64   media       proc  sbin  snap   tmp
dev   lib   libx32  mnt         root  site  srv    usr
root@for-pit-course:/# cd site2
root@for-pit-course:/site2# tuoch Dockerfile

Command 'tuoch' not found, did you mean:

  command 'touch' from deb coreutils (8.30-3ubuntu2)

Try: apt install <deb name>

root@for-pit-course:/site2# touch Dockerfile
root@for-pit-course:/site2# ls
Dockerfile
```

Затем с помощью редактора nano отредактировала данный Dockerfile и вывела его содержимое в терминал.

```
root@for-pit-course:/demosite# nano Dockerfile
root@for-pit-course:/demosite# tail Dockerfile
# Version 0.1pre-alpha
FROM ubuntu
RUN apt-get update
RUN apt-get install -y nginx
RUN echo "Hello... It's me..." > /var/www/html/index.nginx-debian.html

EXPOSE 8080
```

После я запустила build Docker-файла.

```
root@for-pit-course:/demosite# docker build -t demosite .
Sending build context to Docker daemon  2.048kB
Step 1/5 : FROM ubuntu
 ---> 9873176a8ff5
Step 2/5 : RUN apt-get update
 ---> Using cache
 ---> 943be4447f4e
Step 3/5 : RUN apt-get install -y nginx
 ---> Using cache
 ---> 0966b09384f3
Step 4/5 : RUN echo "Hello... It's me..." > /var/www/html/index.nginx-debian.htm
l
 ---> Running in 91233cc761b0
Removing intermediate container 91233cc761b0
 ---> 844d6410daf3
Step 5/5 : EXPOSE 8080
 ---> Running in 584419f6544f
Removing intermediate container 584419f6544f
 ---> 7d2c2bb83418
Successfully built 7d2c2bb83418
Successfully tagged demosite:latest
root@for-pit-course:/demosite#
```

На скрине представлена довольно маленькая версия того, что было на самом деле, так как перед тем, как процесс билда завершился успешно, было несколько не успешных попыток из-за ошибок в Docker-файле.

Далее было опробовано два метода сборки Докера, однако ни одна из них не смогла запустить сервер.

```
root@for-pit-course:/demosite# docker build -t demosite .
Sending build context to Docker daemon  2.048kB
Step 1/5 : FROM ubuntu
 ---> 9873176a8ff5
Step 2/5 : RUN apt-get update
 ---> Using cache
 ---> 943be4447f4e
Step 3/5 : RUN apt-get install -y nginx
 ---> Using cache
 ---> 0966b09384f3
Step 4/5 : RUN echo "Hello... It's me..." > /var/www/html/index.nginx-debian.htm
l
 ---> Running in 91233cc761b0
Removing intermediate container 91233cc761b0
 ---> 844d6410daf3
Step 5/5 : EXPOSE 8080
 ---> Running in 584419f6544f
Removing intermediate container 584419f6544f
 ---> 7d2c2bb83418
Successfully built 7d2c2bb83418
Successfully tagged demosite:latest
root@for-pit-course:/demosite# docker run -p 8080:8080 demosite
root@for-pit-course:/demosite# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@for-pit-course:/demosite# docker run -p 8080:8080 -d demosite
bdca0cba6772e666ffd1db63bd2821202a953c9949b0ffe9d356b21ba240367e
root@for-pit-course:/demosite# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

```
root@for-pit-course:/demosite# docker build -t demosite .
Sending build context to Docker daemon  2.048kB
Step 1/5 : FROM ubuntu
 ---> 9873176a8ff5
Step 2/5 : RUN apt-get update
 ---> Using cache
 ---> 943be4447f4e
Step 3/5 : RUN apt-get install -y nginx
 ---> Using cache
 ---> 0966b09384f3
Step 4/5 : RUN echo "Hello... It's me..." > /usr/share/nginx/html/index.html
 ---> Running in 8584194c4094
Removing intermediate container 8584194c4094
 ---> 4220b3393304
Step 5/5 : EXPOSE 8080
 ---> Running in 848a7025be78
Removing intermediate container 848a7025be78
 ---> 444d057ebfe8
Successfully built 444d057ebfe8
Successfully tagged demosite:latest
root@for-pit-course:/demosite# docker run -p 8080:8080 -d demosite
8742bb938bcc53b9b3445d8fce9de02d2a0471e6a6a6f74d97b6195108e9b739
root@for-pit-course:/demosite# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@for-pit-course:/demosite#
```

Далее я исправила докер файл, как было показано в видео.

```
  GNU nano 4.8                          Dockerfile
# Version 0.1pre-alpha
FROM ubuntu
RUN apt-get update
RUN apt-get install -y nginx
RUN echo "Hello... It's me..." > /var/www/html/index.nginx-debian.html

CMD ["nginx", "-g", "daemon off;"]

EXPOSE 8080
```

После исправления Docker-файла мы можем видеть, что сервер запущен:

```
root@for-pit-course:/demosite# docker build -t demosite .
Sending build context to Docker daemon  2.048kB
Step 1/6 : FROM ubuntu
 ---> 9873176a8ff5
Step 2/6 : RUN apt-get update
 ---> Using cache
 ---> 943be4447f4e
Step 3/6 : RUN apt-get install -y nginx
 ---> Using cache
 ---> 0966b09384f3
Step 4/6 : RUN echo "Hello... It's me..." > /var/www/html/index.nginx-debian.htm
l
 ---> Using cache
 ---> 844d6410daf3
Step 5/6 : CMD ["nginx", "-g", "daemon off;"]
 ---> Running in 511c452ea894
Removing intermediate container 511c452ea894
 ---> 9e5dd8f455ec
Step 6/6 : EXPOSE 8080
 ---> Running in f459769f50da
Removing intermediate container f459769f50da
 ---> a2c842e29c5c
Successfully built a2c842e29c5c
Successfully tagged demosite:latest
root@for-pit-course:/demosite# docker run -p 8080:8080 -d demosite
b181beda159be7af4b1439b8fe7658ed8dddc3189d997450fdc92bcf0a273005
root@for-pit-course:/demosite# docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED         STATUS
  PORTS                                       NAMES
b181beda159b   demosite   "nginx -g 'daemon of…"   9 seconds ago   Up 8 seconds
  0.0.0.0:8080->8080/tcp, :::8080->8080/tcp   admiring_maxwell
root@for-pit-course:/demosite#
```

Затем я зарегистрировалась на Docker Hub и подключилась через консоль к своему аккаунту.

```
root@for-pit-course:/demosite# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.
com to create one.
Username: iam100
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@for-pit-course:/demosite#
```

И затем перебилдила свой докерфайл, чтобы сделать его с нужным тегом и запушила его в Docker Hub.

По следующей ССЫЛКЕ можно найти мой Docker Hub.