

# Домашняя работа №4

**Задание:** необходимо создать Dockerfile, основанный на любом образе (вы в праве выбрать самостоятельно).

В него необходимо поместить приложение, написанное на любом известном вам языке программирования (Python, Java, C, C#, C++).

При запуске контейнера должно запускаться самостоятельно написанное приложение.

---

1. Создаем папку:

```
mkdir hw  
cd hw
```

2. Создаем приложение на python:

```
nano main.py
```

Содержимое:

```
print("Hello world from docker :)")
```

3. Создаем dockerfile:

```
nano dockerfile
```

Содержимое:

```
FROM ubuntu:22.10  
RUN apt-get update && \  
    apt-get install python3 -y && \  
    apt-get install python3-pip -y  
COPY main.py main/main.py  
WORKDIR /main  
CMD ["python3", "main.py"]
```

4. Создаем контейнер:

```
docker build -t pytest .
```

5. Проверим список образов и убедимся, что наш образ присутствует:

```
docker images
```

6. Запускаем:

```
docker run pytest
```

7. Вывод в консоль:

```
Hello world from docker :)
```

```
root@user-VB: ~/hw
root@user-VB:~# mkdir hw
root@user-VB:~# cd hw
root@user-VB:~/hw# nano main.py
root@user-VB:~/hw# nano dockerfile
root@user-VB:~/hw# docker build -t pytest .
[+] Building 0.7s (9/9) FINISHED
=> [internal] load build definition from dockerfile                                0.1s
=> => transferring dockerfile: 212B                                              0.0s
=> [internal] load .dockerignore                                                 0.1s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/ubuntu:22.10                 0.0s
=> [1/4] FROM docker.io/library/ubuntu:22.10                                   0.0s
=> [internal] load build context                                                0.0s
=> => transferring context: 70B                                                 0.0s
=> CACHED [2/4] RUN apt-get update && apt-get install python3 -y && apt-get install python3- 0.0s
=> [3/4] COPY main.py main/main.py                                             0.1s
=> [4/4] WORKDIR /main                                                         0.2s
=> exporting to image                                                           0.2s
=> => exporting layers                                                         0.2s
=> => writing image sha256:8bffbddc2e81b0fe61e64f2e14410bbd18977009602b009db35fb334fbce4517 0.0s
=> => naming to docker.io/library/pytest                                       0.0s
root@user-VB:~/hw# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
pytest        latest   8bffbddc2e81   9 seconds ago  502MB
<none>        <none>   fbe529089b62   13 minutes ago 502MB
ubuntu        22.10    f15f05d87425   5 weeks ago   70.3MB
root@user-VB:~/hw# docker run pytest
Hello world from docker :)
root@user-VB:~/hw#
```