

Запускаем первый контейнер

1. Выполните следующую команду в консоли:

```
docker run -t -i debian:jessie bash
```

Обратите внимание, что все параметры для запуска контейнера (-t -i) указываются до образа.

Все что находится после образа является командами и параметрами внутри контейнера.

После выполнения данной команды, Вы попадете в консоль контейнера:

```
docker run -t -i debian:jessie bash
```

```
Unable to find image 'debian:jessie' locally  
jessie: Pulling from library/debian  
bf295113f40d: Pull complete  
Digest: sha256:201b887113d1190ed7811152af4c14f46a7ce8612dfdaee540fc353f5135d25a  
Status: Downloaded newer image for debian:jessie  
root@fe8a995c3727:/#
```

2. Выполните внутри контейнера команду:

```
cat /etc/os-release
```

Скопируйте и отправьте ее результат в качестве ответа.

P.S. Для выхода из контейнера наберите команду `exit` или нажмите `Ctrl+D`.

Запускаем второй контейнер

1. Выполните следующую команду в консоли:

```
docker run -d -p 8080:80 nginx:1.13
```

После выполнения данной команды, убедитесь, что контейнер запущен:

```
docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
3c281f47fb9a	nginx:1.13	"nginx -g 'daemon of...'"	5 seconds ago
Up 4 seconds	0.0.0.0:8080->80/tcp	mystifying_chaplygin	

2. С локальной машины попробуйте обратиться на:localhost:8080

```
curl localhost:8080
```

3. Посмотрите в логи контейнера:

Обратите внимание, что вместо placeholder'a <ИМЯ КОНТЕЙНЕРА> Вам нужно подставить реальное имя вашего контейнера из команды `docker ps`

```
docker logs <ИМЯ КОНТЕЙНЕРА>
```

Скопируйте и отправьте одну строку из лога контейнера как решение для данного шага.

4. После выполнения задания, удалите контейнер:

```
docker rm -f <ИМЯ КОНТЕЙНЕРА>
```

Запускаем третий контейнер

1. Сохраните на машине с Docker файл `default.conf` с содержимым

```
server {
    listen      80 default_server;
    server_name _;

    location / {
        return 200 '$hostname\n';
    }
}
```

2. Запустите контейнер с сохраненным файлом:

```
docker run -d -p 8080:80 -v "$PWD/default.conf":/etc/nginx/conf.d/default.conf
nginx:1.13
```

После выполнения данной команды, убедитесь, что контейнер запущен:

```
docker ps
```

Вывод будет примерно таким:

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
e0911beb600a	nginx:1.13	"nginx -g 'daemon of...'"	3 seconds ago
Up 1 second	0.0.0.0:8080->80/tcp	recursing_driscoll	

3. С локальной машины попробуйте обратиться на:localhost:8080

```
curl localhost:8080
```

Скопируйте и отправьте ответ этой команды в качестве решения данного шага.

4. После выполнения задания, удалите контейнер:

```
docker rm -f <ИМЯ КОНТЕЙНЕРА>
```

Самостоятельная работа

Запустите контейнер из образа `slurmio/hello-nginx:latest`, перебросьте локальный порт `8080` в `80` порт контейнера, при запуске задайте переменной окружения `USERNAME` в контейнере значение с вашим именем.

Чтобы при старте контейнера задавать значения переменных окружения, добавьте к команде `docker run` ключ `-e`.

Пример:

```
docker run -e 'VARIABLE=value' nginx:1.12
```

Обратитесь на переброшенный порт, скопируйте ответ контейнера и отправьте его как решение для этого шага.

Не забудьте удалить контейнер после окончания работы, чтобы не держать локальный порт `8080`, он нам еще понадобится.