

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2
«Основы работы с Docker»

Выполнила:
Кузнецова Алена Валерьевна
3 курс, группа ИВТ-б-о-21-1,
09.03.01 «Информатика
и вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем»,
очная форма обучения

(подпись)

Ставрополь, 2023 г.

Цель занятия: научиться использовать основные команды Docker для управления контейнерами и понимать их назначение.

Выполнение работы:

Задача 1: Основы Docker

Загрузите образ Ubuntu с Docker Hub.

```
alena@LAPTOP-ARKAM4ED:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
aece8493d397: Pull complete
Digest: sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e496e5c1f05f
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

Рисунок 1 – Загрузка образа ubuntu

Создайте и запустите контейнер на основе этого образа.

Войдите в созданный контейнер и выполните команду `ls`, чтобы просмотреть файлы

внутри контейнера.

```
alena@LAPTOP-ARKAM4ED:~$ docker run -it ubuntu
root@e86ba3ad8b4a:/# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys
```

Рисунок 2 – Запуск контейнера и выполнение команды `ls`

Задача 2: Управление контейнерами и образами

Загрузите образ Nginx с Docker Hub.

```
alena@LAPTOP-ARKAM4ED:~$ docker pull nginx:latest
latest: Pulling from library/nginx
1f7ce2fa46ab: Pull complete
9b16c94bb686: Pull complete
9a59d19f9c5b: Pull complete
9ea27b074f71: Pull complete
c6edf33e2524: Pull complete
84b1ff10387b: Pull complete
517357831967: Pull complete
Digest: sha256:725cd8bd9fe9bb36a661b87b718e5d4ca41c579d1d30560ad1799829dbbafaaf
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
```

Рисунок 3 – Загрузка образа Nginx

Создайте контейнер на основе этого образа и пробросьте порт 8080 контейнера на порт 80 хоста.

```
alena@LAPTOP-ARKAM4ED:~$ docker run -p 8080:80 -d nginx
613a9cc874702146c5b521952318ed8648056d4aebd134d45a42333807675fa1
```

Рисунок 4 – Создание контейнера

Посмотрите список активных контейнеров и убедитесь, что ваш контейнер работает.

```
alena@LAPTOP-ARKAM4ED:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED
613a9cc87470   nginx    "/docker-entrypoint..." 40 seconds ago
```

Рисунок 5 – Просмотр всех контейнеров

Остановите и удалите контейнер.

```
alena@LAPTOP-ARKAM4ED:~$ docker stop 613a9cc87470
613a9cc87470
alena@LAPTOP-ARKAM4ED:~$ docker rm 613a9cc87470
613a9cc87470
```

Рисунок 6 – Удаление контейнера

Задача 3: Мониторинг и управление контейнерами

Запустите контейнер с именем "my_container".

```
alena@LAPTOP-ARKAM4ED:~$ docker run --name my_container -d nginx
92b914abeadc123259e8b472ca30eee7a76db76332b0f6f71f90c14d5bd4d7a6
```

Рисунок 7 – Запуск контейнера

Используя команду `docker ps`, убедитесь, что контейнер запущен.

```
alena@LAPTOP-ARKAM4ED:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS
92b914abeadc   nginx    "/docker-entrypoint..." 21 seconds ago  Up 20 seconds
```

Рисунок 8 – Команда ps

Остановите контейнер.

```
alena@LAPTOP-ARKAM4ED:~$ docker stop my_container
my_container
```

Рисунок 9 – Остановка контейнера

Проверьте его статус снова и убедитесь, что он остановлен.

```
alena@LAPTOP-ARKAM4ED:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS
92b914abeadc   nginx    "/docker-entrypoint..." About a minute ago  Exited (0) 28 seconds ago
```

Рисунок 10 – Просмотр активных контейнеров

Удалите контейнер.

```
alena@LAPTOP-ARKAM4ED:~$ docker rm my_container
my_container
```

Рисунок 11 – Удаление контейнера

Задача 4: Удаление образов и оптимизация дискового пространства

Загрузите образы Ubuntu и Alpine с Docker Hub.

```
alena@LAPTOP-ARKAM4ED:~$ docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
96526aa774ef: Pull complete
Digest: sha256:eece025e432126ce23f223450a0326fbebde39cdf496a85d8c016293fc851978
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
```

Рисунок 12 – Загрузка образа

Создайте контейнеры на основе обоих образов.

```
alena@LAPTOP-ARKAM4ED:~$ docker run --name cont_ubu -d ubuntu
c9f247f81a159e2cedf99c087ec31dd26e188e402b455908bddb2e98cc9c1001
alena@LAPTOP-ARKAM4ED:~$ docker run --name cont_alpine -d alpine
d710338d1eae14b135ef52053d58fe81e053612a7c71927a39364c40227c7ae0
```

Рисунок 13 – Создание контейнеров

Убедитесь, что контейнеры запущены и работают.

```
alena@LAPTOP-ARKAM4ED:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
d710338d1eae   alpine         "/bin/sh"               48 seconds ago Exited (0) 48 seconds ago
c9f247f81a15   ubuntu        "/bin/bash"             About a minute ago Exited (0) About a minute ago
```

Рисунок 14 – Просмотр активных контейнеров

Удалите образ Ubuntu.

```
alena@LAPTOP-ARKAM4ED:~$ docker rmi -f ubuntu
Untagged: ubuntu:latest
Untagged: ubuntu@sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e496e5c1f05f
Deleted: sha256:e4c58958181a5925816faa528ce959e487632f4cfd192f8132f71b32df2744b4
```

Рисунок 15 – Удаление образа

Проверьте, что образ Ubuntu больше не существует, но образ Alpine остался на системе.

```
alena@LAPTOP-ARKAM4ED:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
d710338d1eae   alpine         "/bin/sh"               2 minutes ago Exited (0) 2 minutes ago
c9f247f81a15   e4c58958181a   "/bin/bash"             2 minutes ago Exited (0) 2 minutes ago
```

Рисунок 16 – Просмотр активных контейнеров

Задача 5: Взаимодействие с контейнером

Запустите контейнер с именем "my_container" в фоновом режиме.

```
alena@LAPTOP-ARKAM4ED:~$ docker run --name my_container -d nginx
1392296d0fc9430c605a1b9a4b884ca017645829b46be854b1405cba3558c798
```

Рисунок 17 – Запуск контейнера

Используя команду `docker exec`, выполните команду `ls -l /app` внутри контейнера.

```
alena@LAPTOP-ARKAM4ED:~$ docker exec -it my_container bash
root@1392296d0fc9:/# ls -l
total 64
lrwxrwxrwx   1 root root    7 Nov 20 00:00 bin -> usr/bin
drwxr-xr-x   2 root root 4096 Sep 29 20:04 boot
drwxr-xr-x   5 root root  340 Nov 21 14:23 dev
drwxr-xr-x   1 root root 4096 Nov 21 09:05 docker-entrypoint.d
-rwxrwxr-x   1 root root 1620 Nov 21 09:05 docker-entrypoint.sh
drwxr-xr-x   1 root root 4096 Nov 21 14:23 etc
drwxr-xr-x   2 root root 4096 Sep 29 20:04 home
lrwxrwxrwx   1 root root    7 Nov 20 00:00 lib -> usr/lib
lrwxrwxrwx   1 root root    9 Nov 20 00:00 lib32 -> usr/lib32
lrwxrwxrwx   1 root root    9 Nov 20 00:00 lib64 -> usr/lib64
lrwxrwxrwx   1 root root   10 Nov 20 00:00 libx32 -> usr/libx32
drwxr-xr-x   2 root root 4096 Nov 20 00:00 media
```

Рисунок 18 – Команда `ls -l`

Выполните команду `ps aux` внутри контейнера, чтобы увидеть список запущенных процессов.

Остановите и удалите контейнер.

```
alena@LAPTOP-ARKAM4ED:~$ docker stop my_container
my_container
alena@LAPTOP-ARKAM4ED:~$ docker rm my_container
my_container
```

Рисунок 19 – Удаление контейнера

Вывод: мы научились использовать основные команды Docker для управления контейнерами и понимаем их назначение.

Контрольные вопросы:

1. Что делает команда `docker pull`?

Команда `docker pull` в Docker используется для загрузки образа контейнера с Docker Hub или другого репозитория.

2. Какой синтаксис используется для загрузки образа с Docker Hub с помощью docker pull?

```
docker pull <имя_образа>:<тег>
```

3. Как можно просмотреть список всех доступных образов на системе с помощью docker images?

Эта команда выведет список всех образов, которые находятся на вашей системе, включая их имена, теги, размер и ID.

4. Какой ключ используется для просмотра образов в формате таблицы с docker images?

```
docker images --format "table {{.Repository}}\t{{.Tag}}\t{{.Size}}"
```

5. Как создать и запустить контейнер с использованием docker run?

```
docker run [опции] <имя_образа> [команда] [аргументы]
```

6. Как пробросить порт при запуске контейнера с docker run?

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

7. Как изменить имя контейнера при его создании с помощью docker run?

```
docker run --name my_container -d nginx
```

8. Как создать контейнер в фоновом режиме с docker run?

```
docker run -d nginx
```

9. Какая команда используется для просмотра активных контейнеров на системе?

```
docker ps
```

10. Какие опции могут использоваться с docker ps для отображения остановленных контейнеров?

```
docker ps -a
```

11. Как можно просмотреть список всех контейнеров, включая остановленные, с docker ps?

```
docker ps -a
```

12. Что делает команда docker start?

Команда docker start в Docker используется для запуска остановленных

контейнеров.

13. Какой синтаксис используется для запуска остановленного контейнера с `docker start`?

`docker start [опции] <имя_или_ID_контейнера>`

14. Как запустить контейнер в фоновом режиме с `docker start`?

`docker start -d my_container`

15. Что делает команда `docker stop`?

Команда `docker stop` в Docker используется для остановки работающего контейнера.

16. Как остановить контейнер по его имени с помощью `docker stop`?

`docker stop my_container`

17. Как принудительно остановить контейнер с `docker stop`?

`docker stop -f my_container`

18. Что делает команда `docker rm`?

Команда `docker rm` в Docker используется для удаления контейнера, который был остановлен.

19. Как удалить контейнер по его ID с использованием `docker rm`?

`docker rm 1234567890`

20. Как удалить несколько контейнеров сразу с `docker rm`?

`docker rm container1 container2`

21. Что делает команда `docker rmi`?

Команда `docker rmi` в Docker используется для удаления образов контейнеров с вашей системы.

22. Как удалить Docker-образ по его имени и тегу с помощью `docker rmi`?

`docker rmi ubuntu:20.04`

23. Как удалить несколько Docker-образов сразу с `docker rmi`?

`docker rmi image1 image2`

24. Как выполнить команду внутри работающего контейнера с `docker exec`?

`docker exec [опции] <имя_или_ID_контейнера> <команда> [аргументы]`

25. Как выполнить команду внутри контейнера в интерактивном режиме с `docker exec`?

`docker exec -it my_container /bin/bash`

26. Как выполнить команду с использованием определенного пользователя внутри контейнера с `docker exec`?

`docker exec -u 1000 my_container whoami`

27. Какой ключ используется для запуска команды в фоновом режиме с `docker exec`?

`docker exec -d my_container my_command`

28. Как выполнить команду внутри контейнера с именем вместо ID с `docker exec`?

`docker exec -it $(docker ps -q -f "name=my_container") /bin/bash`

29. Как передать аргументы при выполнении команды с `docker exec` ?

`docker exec [опции] <имя_или_ID_контейнера> <команда> [аргументы]`

30. Как проверить список доступных команд и опций для `docker exec`?

`docker exec --help`

31. Как передать переменную окружения в контейнер при его запуске?

`docker run -e MYSQL_ROOT_PASSWORD=my-secret-pw mysql`

32. Какой ключ используется для запуска контейнера в фоновом режиме с командой `docker run`?

`docker run -d nginx`

33. Как проверить статус выполнения контейнеров на системе с помощью `docker ps`?

`docker ps -s`

34. Как завершить выполнение контейнера без его удаления?

`docker stop my_container`

35. Каким образом можно удалить все остановленные контейнеры с системы?

```
docker rm $(docker ps -aq)
```

36. Что делает опция -a при использовании docker ps?

Добавление опции -a позволяет просматривать все контейнеры, включая те, которые были остановлены.

37. Что означает опция -q при выполнении docker ps?

Добавление опции -q выводит только ID контейнеров.

38. Как принудительно удалить контейнер с флагом -f?

```
docker rm -f my_container
```

39. Какой Docker-образ и какую команду можно использовать для создания контейнера с базой данных PostgreSQL?

```
docker run --name postgres_container postgres
```

40. Какой ключ используется для выполнения команды внутри контейнера в интерактивном режиме?

```
docker exec -it my_container <команда>
```

41. Какой ключ можно использовать для передачи ID пользователя при выполнении команды внутри контейнера?

С опцией -u мы указываем ID пользователя, от имени которого будет выполнена команда.