**Практическая работа по Docker**

1. Установить Docker в своей ОС (*Windows, Linux, Mac*).

2. Работа с образами (images)

1. Используя команду *git pull* скачать с Docker hub образ wordpress.
2. Используя команду *git pull* скачать с Docker hub образ python последней версии (latest) и версии 3.7 на alpine.
3. Посмотреть список локальных (созданных или загруженных) образов с помощью команды *docker images.*
4. Удалить один из образов python (команда: *docker rmi <image>).*

3. Работа с контейнерами (container)

1. Запустить контейнер hello-world, который не был скачан локально (команда *docker run).*
2. Запустить контейнер из скаченного образа python (который не был удален) в интерактивном виде. Команда: *docker run -it <image\_name>.*
3. Посмотреть список запущенных и всех контейнеров с помощью команды *docker ps -a.*
4. Запустить новый контейнер python в фоновом режиме и задать имя контейнеру. Команда: *docker run -d --name <continer\_name> <image\_name>.*
5. Остановить все запущенные контейнеры python использую имя или идентификатор контейнера. Команда: *docker stop <continer\_name> или <contianer\_id>.*
6. Запустить контейнер python использую команду start использую имя или идентификатор: *docker start <continer\_name> или <contianer\_id>.*
7. Запустить контейнер worldpress в фоновом режиме на порте 8080, команда: *docker run --name some-wordpress -p 8080:80 -d wordpress.* Проверить работу контейнера через любой браузер ([http://localhost:8080](http://localhost:8080/)).
8. Запустить контейнер tomcat в интерактивном виде и последующим удалением после остановки. Команда: *docker run -it --rm -p 8888:8080 tomcat.* Проверить работу контейнера через любой браузер ([http://localhost:8888](http://localhost:8888/)).
9. Запустить контейнер python в интерактивном виде и с подключением локальной папки: *docker run -it -v <local\_path>:<docker\_path> python*
10. Остновить и удалить все контейнеры. Команды:
    1. *docker stop $(docker ps -a -q)*
    2. *docker rm $(docker ps -a -q)*