

**Национальный исследовательский технологический университет
«МИСиС»
Институт Информационных технологий и компьютерных наук (ИТКН)**

Курс «Системная и программная инженерия»

**Лабораторная работа № 1
по теме
«Основы Docker»**

Выполнил:
Смирнов А.А.

Проверил:
Козлов М.Е.

Москва, 2023

Цель: познакомиться с основами docker-контейнеров и инструментом docker-compose.

Часть 1: docker-контейнер

Я создал контейнер, который будет использовать образ «busybox», и запустил его (рисунок 1.1).

```
C:\My information\docker>docker pull busybox
Using default tag: latest
latest: Pulling from library/busybox
205dae5015e7: Pull complete
Digest: sha256:7b3ccabffc97de872a30dfd234fd972a66d247c8cfc69b0550f276481852627c
Status: Downloaded newer image for busybox:latest
docker.io/library/busybox:latest

C:\My information\docker>docker run -it busybox sh
/ #
/ # ls
bin    dev    etc    home   lib    lib64  proc   root   sys    tmp    usr    var
```

Рисунок 1.1 – Загрузка образа busybox.

В данном контейнере я вошел в директорию /home и создал файл «SmirnovAA_bivt-20-1.txt» (рисунок 1.2).

```
/ # cd /home
/home # ls
/home # ls -a
.      ..
/home # touch SmirnovAA_bitv-20-1.txt
/home # ls
SmirnovAA_bitv-20-1.txt
```

Рисунок 1.2 – Создание текстового файла.

Часть 2: docker compose

Для выполнения данного задания я решил сделать образ приложения RabbitMQ. Я создал файл «docker-compose.yml», через блокнот изменил его согласно рисунку 2.1.

```
version: "2.1"
services:
  smirnovAA_bivt-20-1:
    image: rabbitmq:3.10.7-management
    restart: always
    ports:
      - 15672:15672
```

Рисунок 2.1 – docker-compose.yml.

Далее я запустил выбранный контейнер (рисунок 2.2). На рисунке 2.3 можно увидеть, что программа успешно запустилась на выбранном порте.

```
C:\My information\docker\docker-compose images\rabbitmq>docker-compose up -d
[+] Running 2/2
 - Network rabbitmq_default          Created                                0.0s
 - Container rabbitmq-smirnovAA_bivt-20-1-1 Started                        0.5s

C:\My information\docker\docker-compose images\rabbitmq>docker ps
CONTAINER ID   IMAGE                                COMMAND                                  CREATED      STATUS      PORTS
574d0d84813f   rabbitmq:3.10.7-management         "docker-entrypoint.s..."             7 seconds ago Up 6 seconds 4369/tcp, 5671-5672/
tcp, 15671/tcp, 15691-15692/tcp, 25672/tcp, 0.0.0.0:15672->15672/tcp rabbitmq-smirnovAA_bivt-20-1-1
```

Рисунок 2.2 – Запуск контейнера через .yaml файл.

The screenshot shows the RabbitMQ web interface at localhost:15672. The interface includes a navigation bar with tabs: Overview, Connections, Channels, Exchanges, Queues, and Admin. The 'Overview' tab is active, displaying various statistics and a table of nodes.

Overview Statistics:

- Queued messages: last minute
- Currently idle
- Message rates: last minute
- Currently idle
- Global counts

Global counts:

- Connections: 0
- Channels: 0
- Exchanges: 7
- Queues: 0
- Consumers: 0

Nodes Table:

Name	File descriptors	Socket descriptors	Erlang processes	Memory	Disk space	Uptime	Info	Reset stats
rabbit@574d0d84813f	36 1048576 available	0 943629 available	392 1048576 available	150 MiB 1.4 GiB high watermark	954 GiB 48 MiB low watermark	1m 58s	basic disc 2 rss	This node All nodes

Churn statistics

Ports and contexts

Рисунок 2.3 – Web-приложение RabbitMQ на localhost.

Вывод: я познакомился с основами docker-контейнеров и инструментом docker-compose. Также создал контейнер для образа «busybox» и запустил контейнер RabbitMQ через «docker-compose.yaml» файл.