

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«МИРЭА - Российский технологический университет»**

**РТУ МИРЭА**

Институт информационных технологий

Кафедра инструментального и прикладного программного обеспечения

**ОТЧЕТ**

**ПОПРАКТИЧЕСКОЙРАБОТЕ№ 1**

«Установка приложения по работе с контейнерами Docker»

**по дисциплине**

**«**НАСТРОЙКА И АДМИНИСТРИРОВАНИЕ СЕРВИСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**»**

Выполнил студент группы ИКБО-20-21 Сидоров С.Д.

Принял преподаватель   Хозяинов А.В.

|  |  |  |
| --- | --- | --- |
| Практическая работа выполнена | « » \_\_\_\_\_\_ 2023 г. |  |
|  |  |  |
| «Зачтено» | « » \_\_\_\_\_\_ 2023 г. | |

Москва 2023

СОДЕРЖАНИЕ

[ЦЕЛЬ РАБОТЫ 3](#_Toc135677946)

[ТЕОРЕТИЧЕСКОЕ ВВЕДЕНИЕ 4](#_Toc135677947)

[ВЫПОЛНЕНИЕ РАБОТЫ 5](#_Toc135677948)

[Задание 1 5](#_Toc135677949)

[Задание 2 5](#_Toc135677950)

[Задание 3 5](#_Toc135677951)

[Задание 4 5](#_Toc135677952)

[Задание 5 5](#_Toc135677953)

[Задание 6 6](#_Toc135677954)

[Задание 7 6](#_Toc135677955)

[Задание 8 6](#_Toc135677956)

[Задание 9 6](#_Toc135677957)

[Задание 10 7](#_Toc135677958)

[Задание 11 9](#_Toc135677959)

[ВЫВОД 10](#_Toc135677960)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ 11](#_Toc135677961)

# ЦЕЛЬ РАБОТЫ

Цель данной работы заключается в получении навыков по развертыванию приложений с использованием Docker, популярной платформы для контейнеризации приложений.

# 

# ТЕОРЕТИЧЕСКОЕ ВВЕДЕНИЕ

Containers (контейнеры) -создаются на основе образа и запускают само  
приложение. Создание контейнера осуществляется командой docker run, и  
используется образ redis,. Список запущенных контейнеров можно увидеть  
с помощью команды docker ps.

Docker Daemon (демон Docker) - Фоновый сервис, запущенный на хост-  
машине, который отвечает за создание, запуск и уничтожение Docker-  
контейнеров. Демон — это процесс, который запущен на операционной  
системе, с которой взаимодействует клиент.

Docker Client (клиент Docker) - Утилита командной строки, которая  
позволяет пользователю взаимодействовать с демоном. Существуют другие  
формы клиента, например, Kitematic, с графическим интерфейсом.

Docker Hub – Репозиторий Docker-образов. Сборник всех доступных  
образов. Если нужно, то можно содержать собственный регистр и  
использовать его для получения образов  
Информация о приложении контейнере СУБД Redis.

Redis — NoSQL база данных типа ключ-значение. Redis хранит данные  
в оперативной памяти, что является ключевой особенностью этого  
хранилища. Поэтому он очень быстрый, но не самый надежный.  
Периодически Redis сбрасывает все данные на диск, но, если сервер упадет  
в момент между добавлением новой информации и сохранением на диск,  
данные будут потеряны. По этим причинам Redis часто используют не как  
основное хранилище, а в качестве кэша, системы управления сессиями или  
для решения другой задачи, где не страшно потерять данные. Сегодня мы  
познакомимся с основными возможностями этой базы данных.

# ВЫПОЛНЕНИЕ РАБОТЫ

Задание 1

Обратиться по адресу https://hub.docker.com/ и ознакомиться с приложениями доступными для работы с Docker.

Задание 2

Установить контейнер СУБД redis. Используя CLI контейнера, запустить клиент redis-cli.

Задание 3

Ознакомиться с документацией по командам redis. Документация на СУБД https://redis.io/commands.

Задание 4

В redis выполнить команды SET https://redis.io/commands/set и GET https://redis.io/commands/get. Создать 5 ключей со значениями с помощью SET и прочитать ключи со значениями с помощью GET.

Листинг 1 – Работа с командами SET и GET

|  |
| --- |
| 127.0.0.1:6379> SET elem\_1 1  OK  127.0.0.1:6379> SET elem\_2 2  OK  127.0.0.1:6379> SET elem\_3 3  OK  127.0.0.1:6379> SET elem\_4 4  OK  127.0.0.1:6379> SET elem\_5 5  OK  127.0.0.1:6379> GET elem\_1  "1"  127.0.0.1:6379> GET elem\_2  "2"  127.0.0.1:6379> GET elem\_3  "3"  127.0.0.1:6379> GET elem\_4  "4"  127.0.0.1:6379> GET elem\_5  "5" |

Задание 5

Получение значения по ключу и его замена на новое.

Листинг 2 – Работа со значением ключа

|  |
| --- |
| 127.0.0.1:6379> GETSET elem\_1 6  "1"  127.0.0.1:6379> GET elem\_1  "6" |

Задание 6

Добавление строки к уже существующему значению.

Листинг 3 – Работа со значением ключа

|  |
| --- |
| 127.0.0.1:6379> APPEND elem\_1 54321  (integer) 6  127.0.0.1:6379> get elem\_1  "654321" |

Задание 7

Добавление числа и изменение его значения.

Листинг 4 – Работа с числовым значением ключа

|  |
| --- |
| 127.0.0.1:6379> set el\_8 1  OK  127.0.0.1:6379> INCR el\_8  (integer) 2 |

Задание 8

Создание ключа со значением типа хеш-таблица.

Листинг 5 – Работа с ключом со значением типа хеш-таблица

|  |
| --- |
| 127.0.0.1:6379> HSET el\_1 age 23  (integer) 1  127.0.0.1:6379> HSET el\_1 weight 60  (integer) 1  127.0.0.1:6379> HGET el\_1 age  "23"  127.0.0.1:6379> HGET el\_1 weight  "60" |

Задание 9

Работа со множествами. Задействовать команды SADD, SDIFF, SMOVE, SPOP, SUNION, SREM.

Листинг 6 – Работа с множествами

|  |
| --- |
| 127.0.0.1:6379> SADD MySet 1  (integer) 1 |

Продолжение листинга 6

|  |
| --- |
| 127.0.0.1:6379> SADD MySet 1  (integer) 0  127.0.0.1:6379> SADD MySet 2  (integer) 1  127.0.0.1:6379> SADD MySet 3  (integer) 1  127.0.0.1:6379> SPOP MySet  "2"  127.0.0.1:6379> SADD MySet1 1  (integer) 1  127.0.0.1:6379> SADD MySet1 12  (integer) 1  127.0.0.1:6379> SADD MySet1 123  (integer) 1  127.0.0.1:6379> SUNION MySet MySet1  1) "1"  2) "3"  3) "12"  4) "123"  127.0.0.1:6379> SREM MySet 1  (integer) 1  127.0.0.1:6379> SDIFF MySet MySet1  1) "3"  127.0.0.1:6379> SMOVE MySet MySet1 3  (integer) 1 |

Задание 10

Работа с упорядоченными наборами. Задействовать команды ZADD, ZCOUNT, ZDIFF, ZPOPMAX, ZPOPMIN, ZUNION, ZMSCORE, ZLEXCOUNT

Листинг 7 – Работа с упорядоченными наборами

|  |
| --- |
| 127.0.0.1:6379> ZADD myzset 1 "one"  (integer) 1  127.0.0.1:6379> ZADD myzset 1 2  (integer) 1  127.0.0.1:6379> ZADD myzset 1 3  (integer) 1  127.0.0.1:6379> ZCOUNT myzset 1 5  (integer) 3  127.0.0.1:6379> ZCOUNT myzset (1 3  (integer) 0  127.0.0.1:6379> ZADD myzset 4 5  (integer) 1  127.0.0.1:6379> ZADD myzset 4 1  (integer) 1  127.0.0.1:6379> ZCOUNT myzset -inf +inf  (integer) 6  127.0.0.1:6379> ZRANGE myzset 0 -1 WITHSCORES  1) "2" |

Продолжение листинга 7

|  |
| --- |
| 2) "1"  3) "4"  4) "1"  5) "one"  6) "1"  7) "3"  8) "2"  9) "1"  10) "4"  11) "5"  12) "4"  127.0.0.1:6379> ZADD myzset1 1 2  (integer) 1  127.0.0.1:6379> ZADD myzset1 2 3  (integer) 1  127.0.0.1:6379> ZADD myzset1 3 4  (integer) 1  127.0.0.1:6379> ZADD myzset2 1 2  (integer) 1  127.0.0.1:6379> ZADD myzset2 2 3  (integer) 1  127.0.0.1:6379> ZADD myzset2 3 4  (integer) 1  127.0.0.1:6379> ZADD myzset2 4 5  (integer) 1  127.0.0.1:6379> ZDIFF 2 myzset1 myzset2 WITHSCORES  (empty array)  127.0.0.1:6379> ZDIFF 2 myzset2 myzset1 WITHSCORES  1) "5"  2) "4"  127.0.0.1:6379> ZPOPMAX myzset2  1) "5"  2) "4"  127.0.0.1:6379> ZPOPMIN myzset2  1) "2"  2) "1"  127.0.0.1:6379> ZUNION 2 myzset1 myzset2  1) "2"  2) "3"  3) "4"  127.0.0.1:6379> ZMSCORE myzset2 2 3  1) (nil)  2) "2"  127.0.0.1:6379> ZADD myzset3 0 a 0 b 0 c 0 d 0 e  (integer) 5  127.0.0.1:6379> ZADD myzset3 0 f 0 g  (integer) 2  127.0.0.1:6379> ZLEXCOUNT myzset3 - +  (integer) 7  127.0.0.1:6379> ZLEXCOUNT myzset3 [b [c  (integer) 2 |

Задание 11

Из документации выбрать любые не использовавшиеся ранее 5 команд и задействовать их в работе.

Листинг 8 – Дополнительные команды для работы

|  |
| --- |
| 127.0.0.1:6379> EXISTS elem\_1  (integer) 1  127.0.0.1:6379> DEL elem\_1  (integer) 1  127.0.0.1:6379> SET element 123 12 seconds  (error) ERR syntax error  127.0.0.1:6379> SET element 123 12  (error) ERR syntax error  127.0.0.1:6379> SET element 123  OK  127.0.0.1:6379> TTL element  (integer) -1  127.0.0.1:6379> EXPIRE element 120  (integer) 1  127.0.0.1:6379> TTL element  (integer) 118  127.0.0.1:6379> PERSIST element  (integer) 1 |

# ВЫВОД

В ходе данной работы мною были получены навыки по развертыванию приложения в Docker и навыки работы с NoSQL базой данных Redis. Были изучены базовые команды с данной нереляционной базой данных.

# СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Сейерс, Э. Х. Docker на практике / Э. Х. Сейерс, А. Милл ; перевод с английского Д. А. Беликов. — Москва : ДМК Пресс, 2020. — 516 с.
2. Миронов, А. Н. Системное программное обеспечение: учебное пособие / А. Н. Миронов, Ю. А. Воронцов, Е. К. Михайлова. — Москва: РТУ МИРЭА, 2022. — 216 с.