Задание:

Выполнить подключение в базе данных Redis. Ввести информацию о 10-ти фильмах. Название, время, год выхода в прокат, жанр, рейтинг (0-10), страна, бюджет, возрастное ограничение.   
1) Вывести отсортированный список фильмов по годам (возрастанию) и бюджету (убывание).

2) Вывести самый дорогой фильм.

3) Вывести фильмы, чей рейтинг больше 8.7.

4) Вывести названия фильмов, чей бюджет больше среднего бюджета среди всех фильмов.

**Установка Redis на Linux**

*apt install lsb-release*

*apt update*

*apt install redis-tools*

*apt install redis-server*

*apt install redis*

Проверка

*redis-cli ping*

Ответ

*PONG*

Запуск

*redis-cli*

Ответ (Открыто подключение к базе данных)

*127.0.0.1:6379>*

Пример сохранения записи в базу:

*set test "It's working!"*

Пример получения записи из базы:

*get test*

**Реляционная база – таблицы.**

Redis – строки, списки, хеши, множества, упорядоченный множества.

В Redis база данных идентифицируется целым числом, которое по умолчанию равняется 0, если необходимо переключиться на другую базу данных, то можно это сделать командой SELECT.

**SELECT 0**

Redis должен ответить сообщением **OK.**

Каждая из пяти используемых структур обязательно имеет ключ и значение.

Должен быть уникальным в базе. Это идентификатор, который позволяет получить доступ к ассоциированному с ним значению.

*USER:1:name “Дима”*

Формировать ключи в виде следующей структуры: **<тип объекта : идентификатор объекта : атрибут>**

**cartoon:1:name**

Значением в хранилище “ключ-значение” может быть что угодно. Например, длинный или короткий текст, число, программный код, изображение и т. п.

**SET <ключ> <значение>**

**GET <ключ>**

**DEL <ключ>**

**TYPE <ключ>**

**RENAME <ключ> <ключ>**

Кроме того, в списке основных команд есть команда KEYS. Она позволяет выбирать значения ключей в соответствии с заданным шаблоном.

**KEYS cartoon:\***

Работа со строками

**STRLEN <ключ>** - используется для вычисления длины строкового значения

**GETRANGE <ключ>** **<начало> <конец>** – возвращает подстроку

**APPEND <ключ> <значение>** - добавляет значение к концу существующей строки

Работа со строками-числами

**INCR <ключ>**

**DECR** **<ключ>**

**INCRBY <ключ> <значение>**

**DECRBY <ключ> <значение>**

Операции над списками

**RPUSH <ключ> <значение>**

**LPUSH <ключ> <значение>**

**LRANGE <ключ> <начало> <конец>**

**LLEN <ключ>**

**LPOP <ключ>**

**Примеры работы с Redis:**

[**https://github.com/redis/redis-py**](https://github.com/redis/redis-py)