Государственное автономное образовательное учреждение

высшего образования города Москвы

«Московский городской педагогический университет»

Институт цифрового образования

Департамент информатики, управления и технологий

**Лабораторная работа №2.1**

**Тема: «Изучение методов хранения данных на основе NoSQL»**

Дисциплина «Инструменты для хранения и обработки больших данных»

Направление подготовки 38.03.05 – бизнес-информатика

Профиль подготовки «Аналитика данных и эффективное управление»

(очная форма обучения)

Выполнила:

Студентка группы АДЭУ-221

Сабитова Алина Булатовна

Преподаватель:

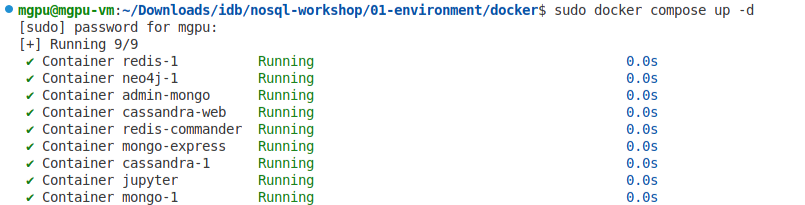
Босенко Тимур Муртазович

Москва

2025

**Цель работы:** изучение трех различных типов NoSQL баз данных: MongoDB, Neo4j, Redis, освоение основных команд для работы с ними и последующее применение полученных знаний для решения практических задач.

Запуск всех контейнеров докера:

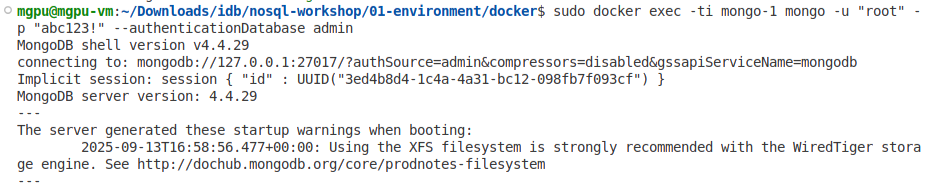


**MongoDB**

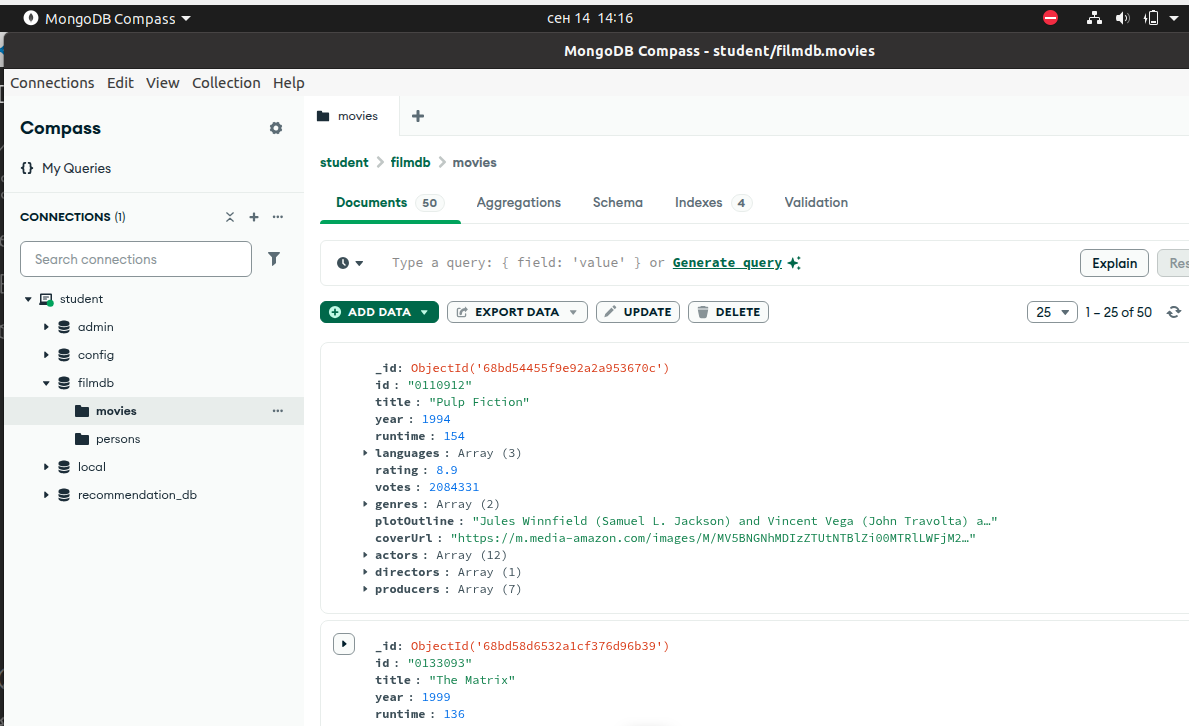
Задание: найти 5 самых старых фильмов в коллекции и обновить у них поле is\_vintage на true.

Шаги выполнения:

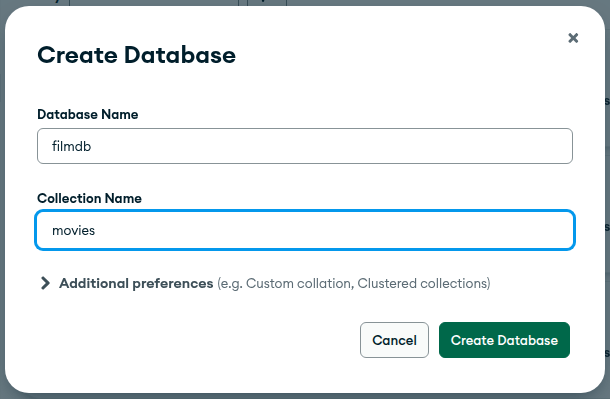
1. Подключение к контейнеру Mongo и запуск оболочки внутри него:

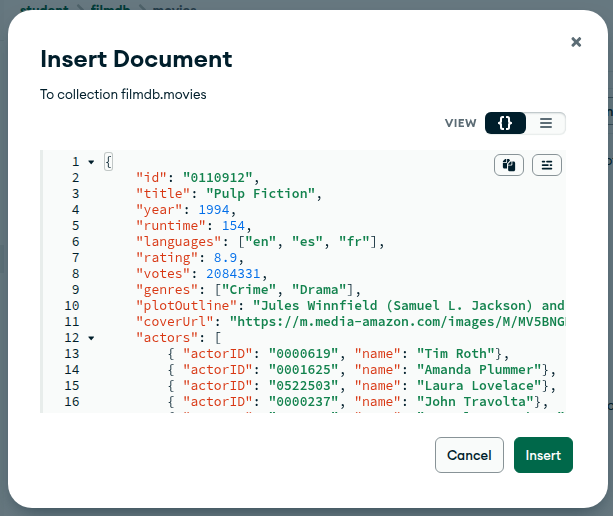


2. Подключение в MongoDB Compass:

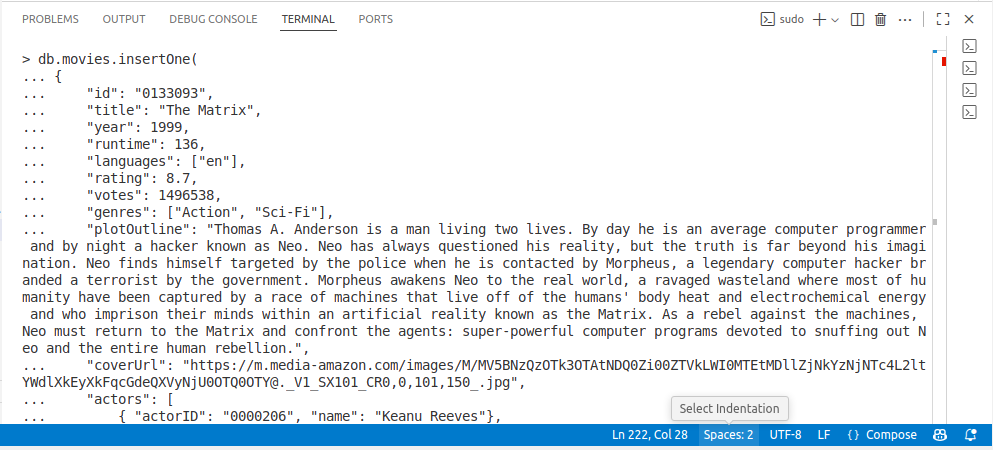


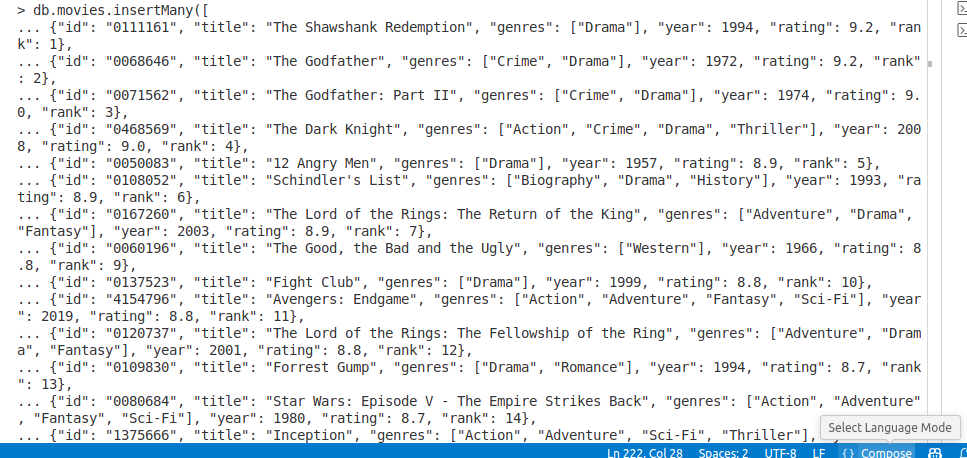
3. Создание базы данных filmdb и коллекции movies:





4. Добавим еще 49 фильм через терминал для того, чтобы работать со списком 50 лучших фильмов из IMDB:





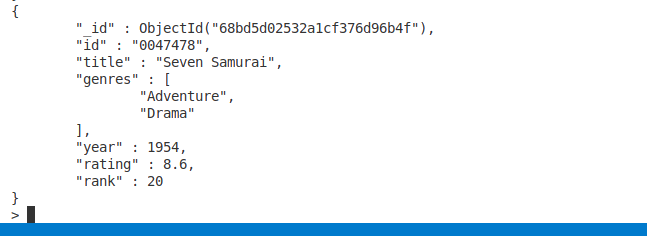
Убедимся, что их действительно 50:



5. Приступим к выполнению индивидуального задания. Найдем 5 самых старых фильмов:

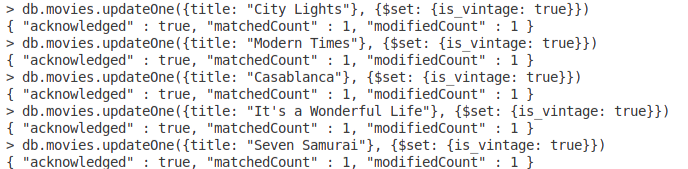






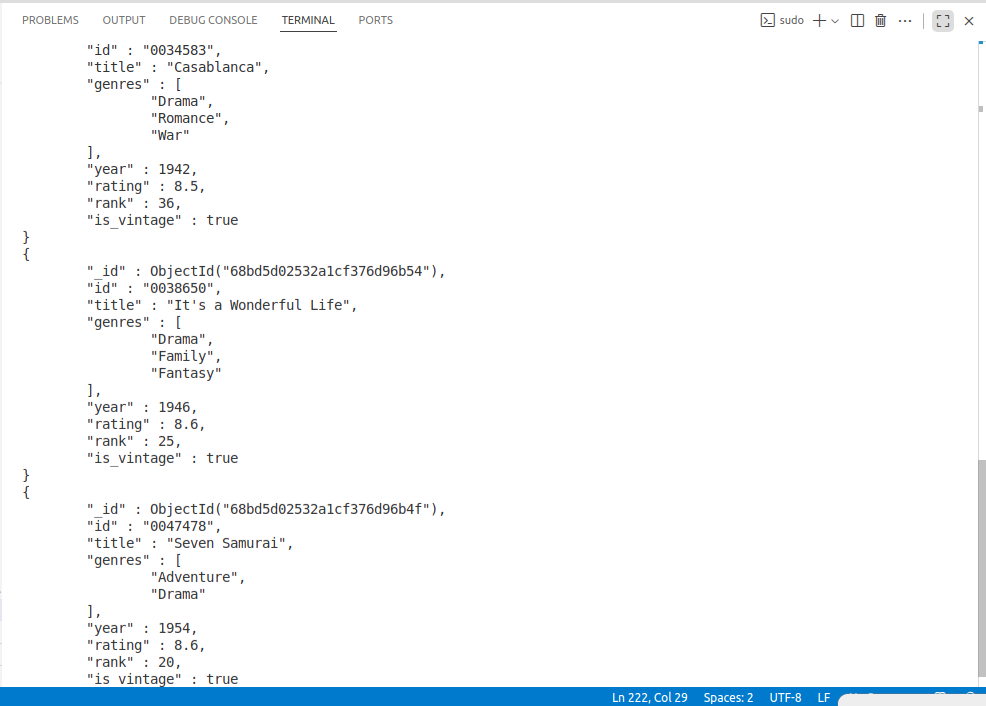
Это фильмы «City Lights», «Modern Times», «Casablanca», «It’s a wonderful life» и «Seven Samurai».

6. Для каждого из них поменяем значение поля is\_vintage на true с помощью команды updateOne()



7. С помощью команды find() проверим, что поле is\_vintage действительно приобрело нужное нам значение:





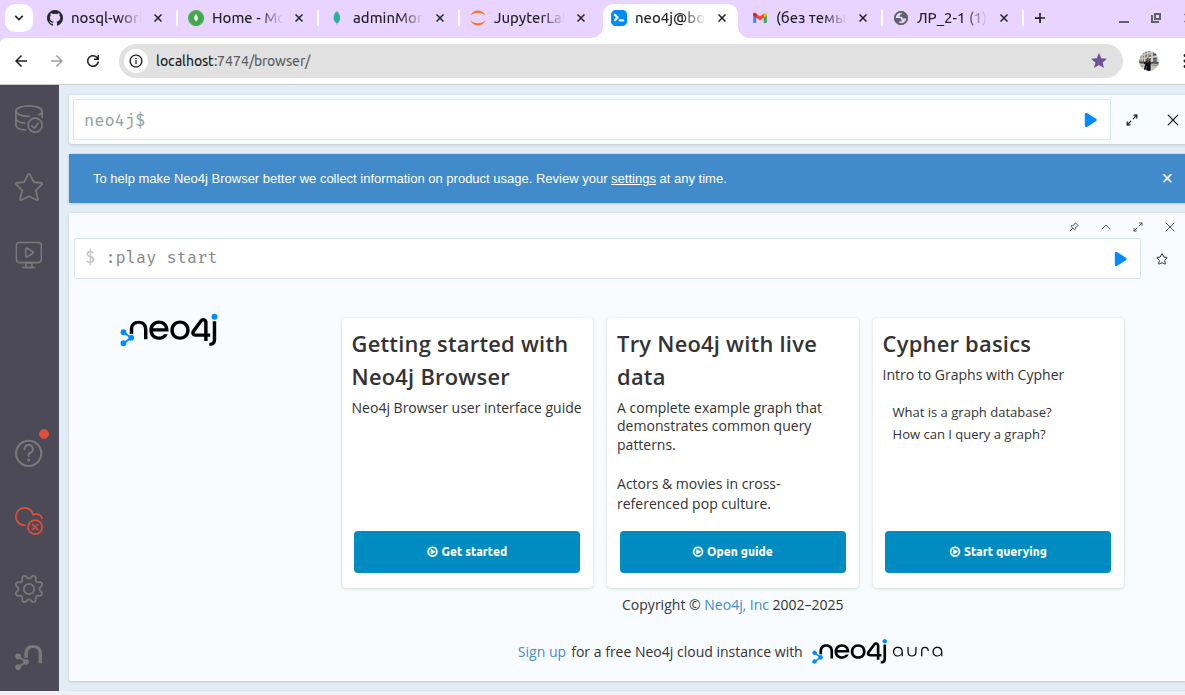
Видим, что у всех фильмов появилось поле is\_vintage со значением true.

**Neo4j**

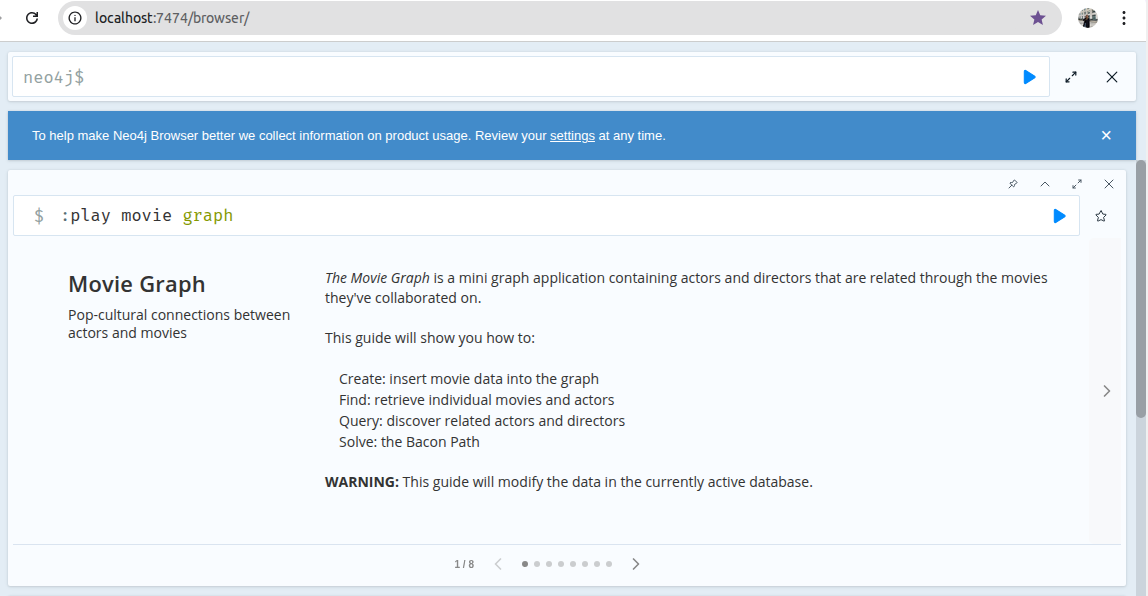
Задание: найти всех актеров, которые не снимались в фильмах с "Tom Hanks".

Шаги выполнения:

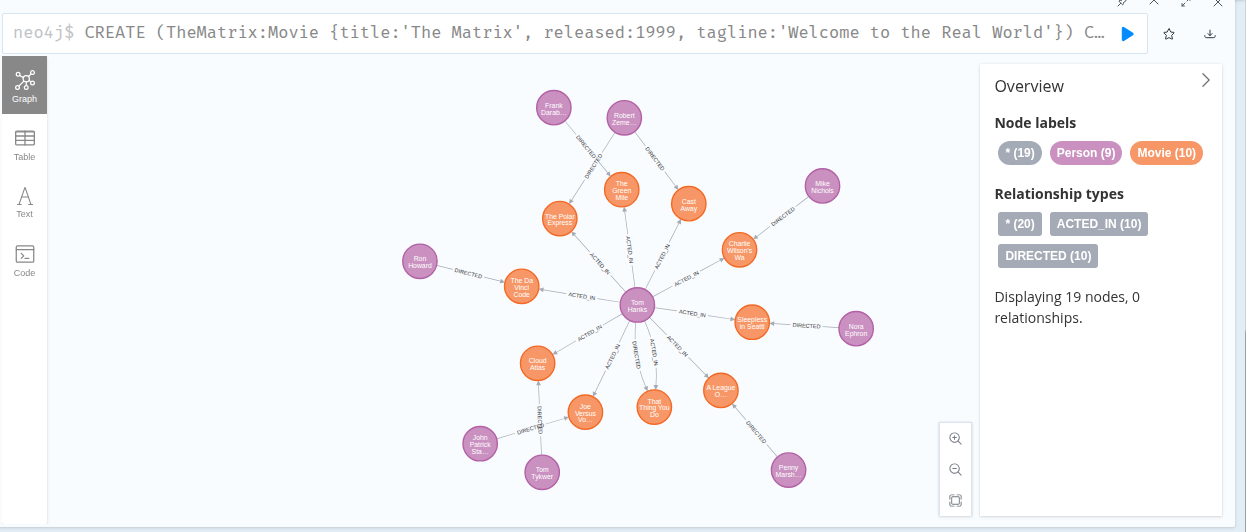
1. Подключимся к Neo4j Browser:



2. Начнем работу с учебным пособием Movie Graph:



3. Загрузим граф фильмов:



4. Найдем актеров, которые никогда не снимались с Томом Хэнксом.

Запрос:

MATCH (p:Person)-[:ACTED\_IN]->(m:Movie)

WHERE NOT EXISTS {

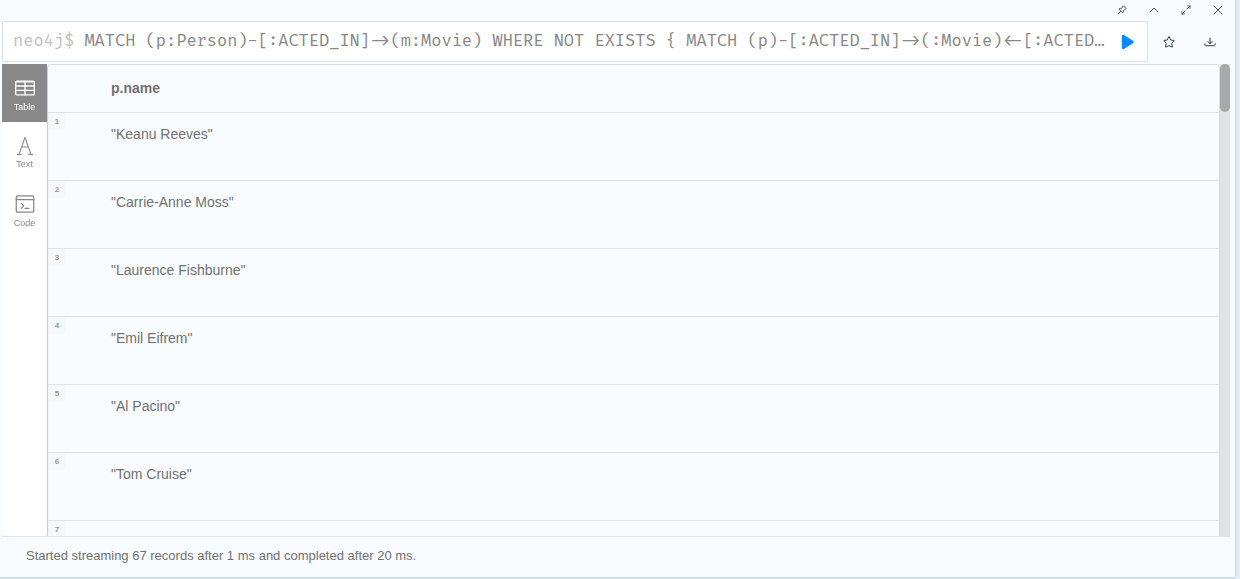
MATCH (p)-[:ACTED\_IN]->(:Movie)<-[:ACTED\_IN]-(tom:Person {name:"Tom Hanks"})

}

AND p.name <> "Tom Hanks"

RETURN DISTINCT p.name

Результат выполнения:



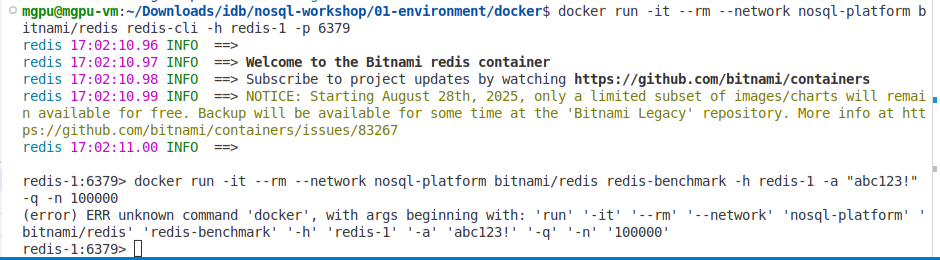
Видим, что 67 актеров никогда не снимались с Томом Хэнксом.

**Redis**

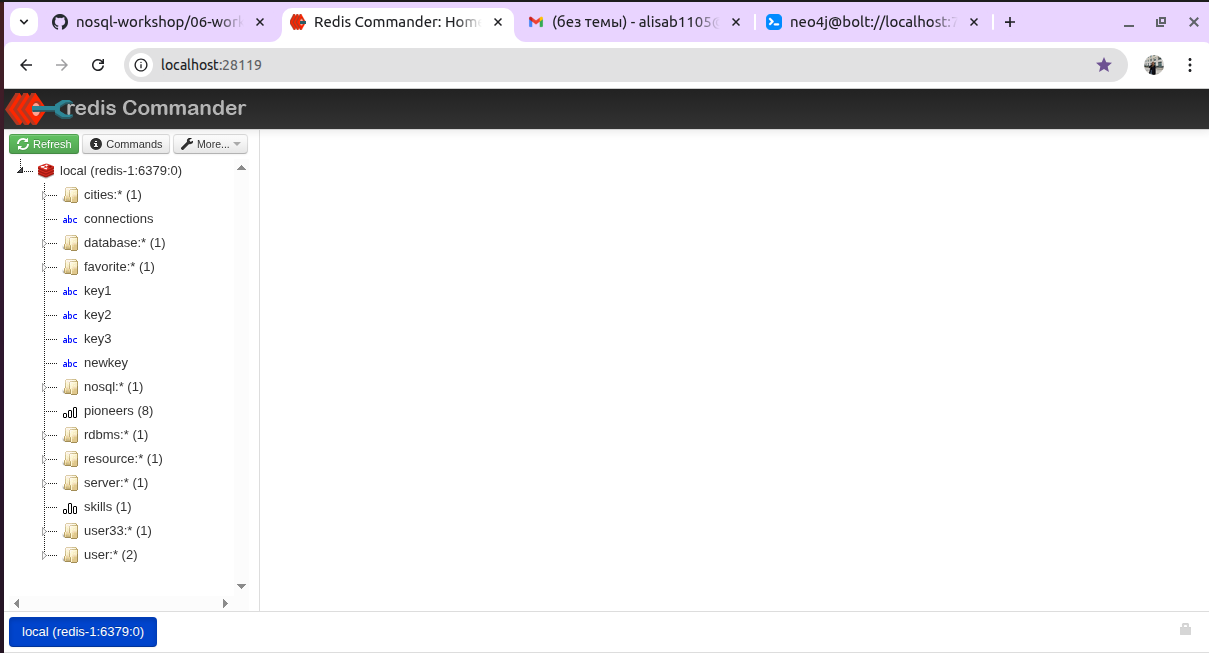
Задание: смоделировать права доступа: создать множество user:33:permissions и добавить в него права "read", "write". Проверить, есть ли у пользователя право "delete" (SISMEMBER).

Шаги выполнения:

1. Запустим Redis CLI в docker-контейнере:



2. Создадим подключение в Redis Commander:



3. Создадим множество user:33:permisions и добавим в него значение «read»:



4. Теперь добавим значение «write»:



5. Проверим, добавились ли в множество указанные нами права:



6. Проверим, есть ли у пользователя право «delete»:

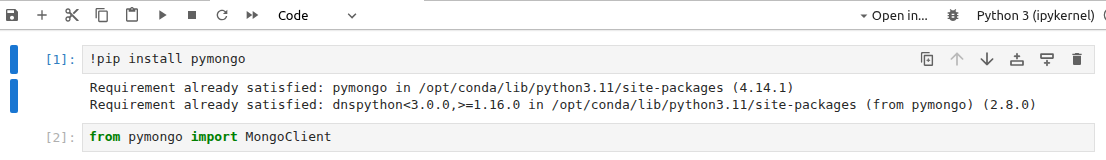


Видим результат – 0, значит такое значение в множестве отсутствует.

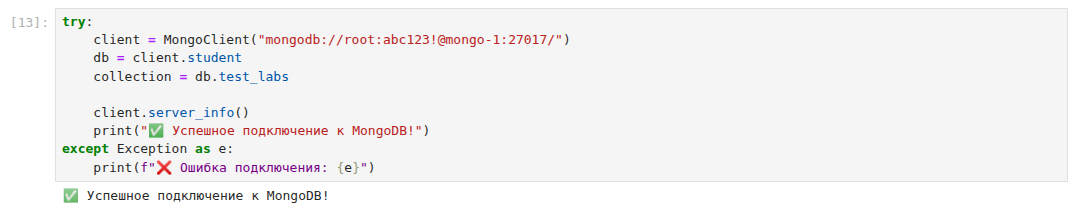
**Работа в JupyterLab**

Рассмотрим взаимодействие с MongoDB через язык Python.

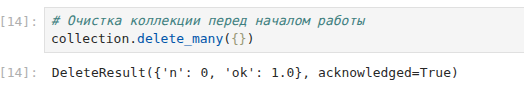
1. Установка pymongo:



2. Подключение к базе данных:



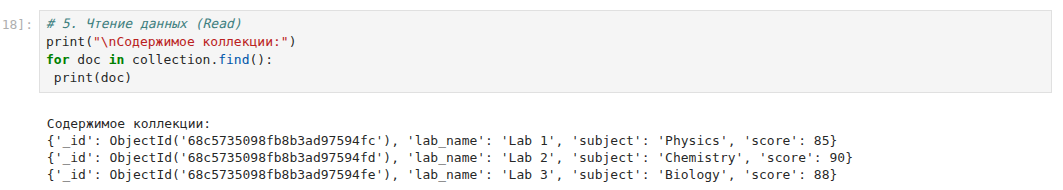
3. Очистка коллекции перед началом работы:



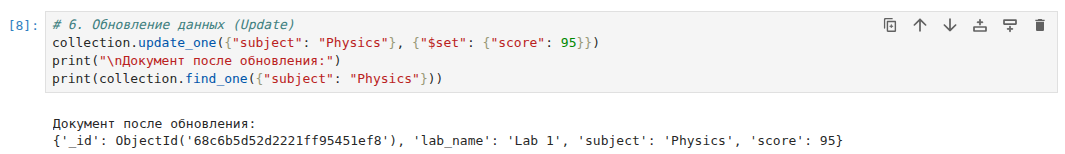
4. Вставка данных в коллекцию:



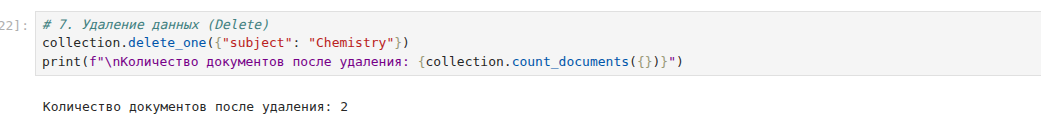
5. Просмотр добавленных документов:



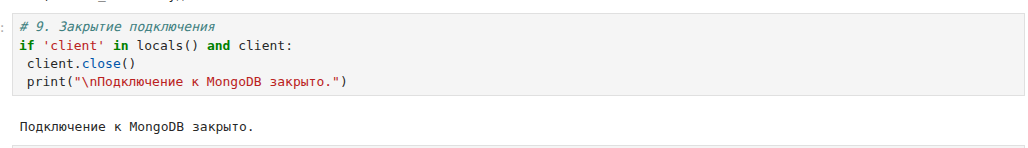
6. Обновление данных:



7. Удаление данных:



8. Закрытие подключения:



Выводы: в ходе выполнения лабораторной работы были изучены основные команды для работы с тремя различными типами NoSQL баз данных. В MongoDB были освоены: создание баз данных и коллекций, поиск, подсчет и сортировка документов, селекторы, обновление документов и пр. В Neo4j удалось ознакомиться с созданием узлов и отношений, научиться выполнять запросы и строить графы. В Redis были рассмотрены команды для работы со списками, множествами, упорядоченными множествами и многие другие. Кроме того, была изучена работа с MongoDB через Python в Jupyter notebook. В результате были получены следующие практические навыки: настройка и подключение к различным NoSQL базам данных, опыт написания запросов и понимание принципов работы с данными в каждой из них.