

ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ  
Государственное автономное образовательное учреждение  
Высшего образования города Москвы  
«Московский городской педагогический университет»  
(ГАОУ ВО МГПУ)

Институт цифрового образования

Лабораторная работа №2.1  
«Изучение методов хранения данных на основе NoSQL»  
По дисциплине «Инструменты для хранения и обработки больших  
данных»

Выполнила: Татаринова Екатерина Михайловна

Группа: АДЭУ-221

Преподаватель: доцент Босенко Т.М.

Москва 2025

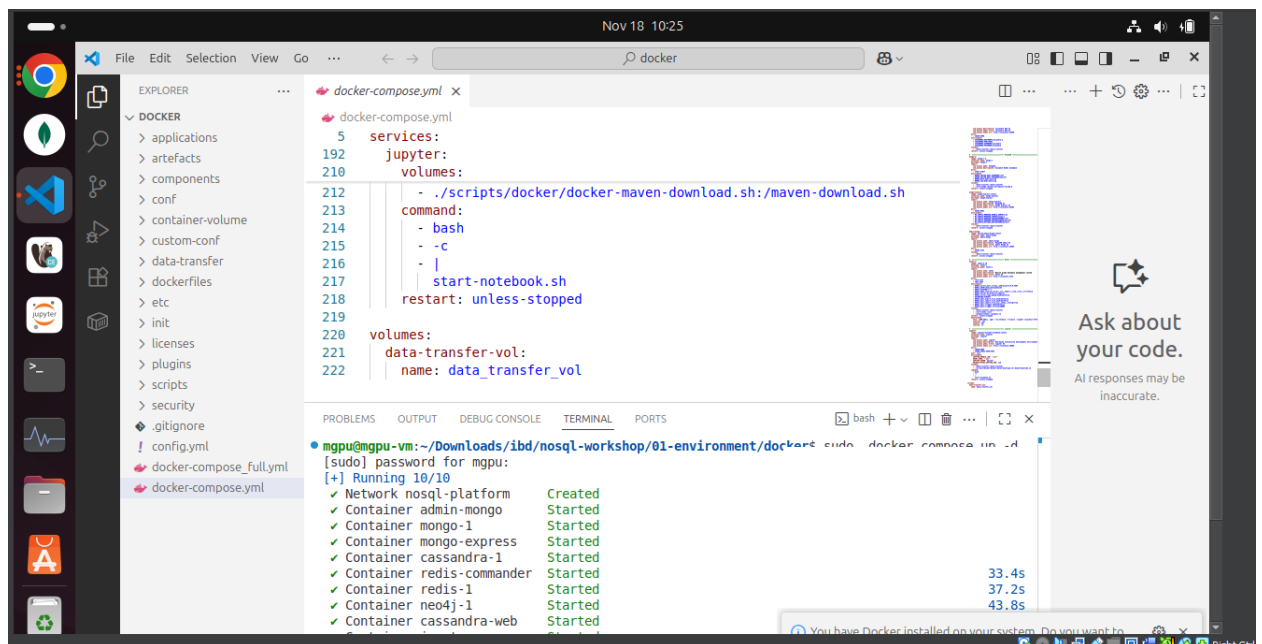
**Цель работы:** изучение и практическое применение трех различных типов NoSQL баз данных: документо-ориентированной (MongoDB), графовой (Neo4j) и ключ-значение (Redis). Студенты должны научиться создавать, заполнять и анализировать структуры данных в каждой из систем, а также выполнять запросы для получения необходимой информации, развивая навыки работы с нереляционными моделями данных.

**Краткое описание процесса выполнения:**

1. Подключение к базе данных MongoDB, создание коллекций, документов, заполнение документов, работа с документами, их изменение и удаление.
2. Подключение к Redis Commander, рассмотрение разных структур данных (string, set, hash и т.д.)
3. Подключение к Neo4j Browser, создание узлов, отношений, атрибутов

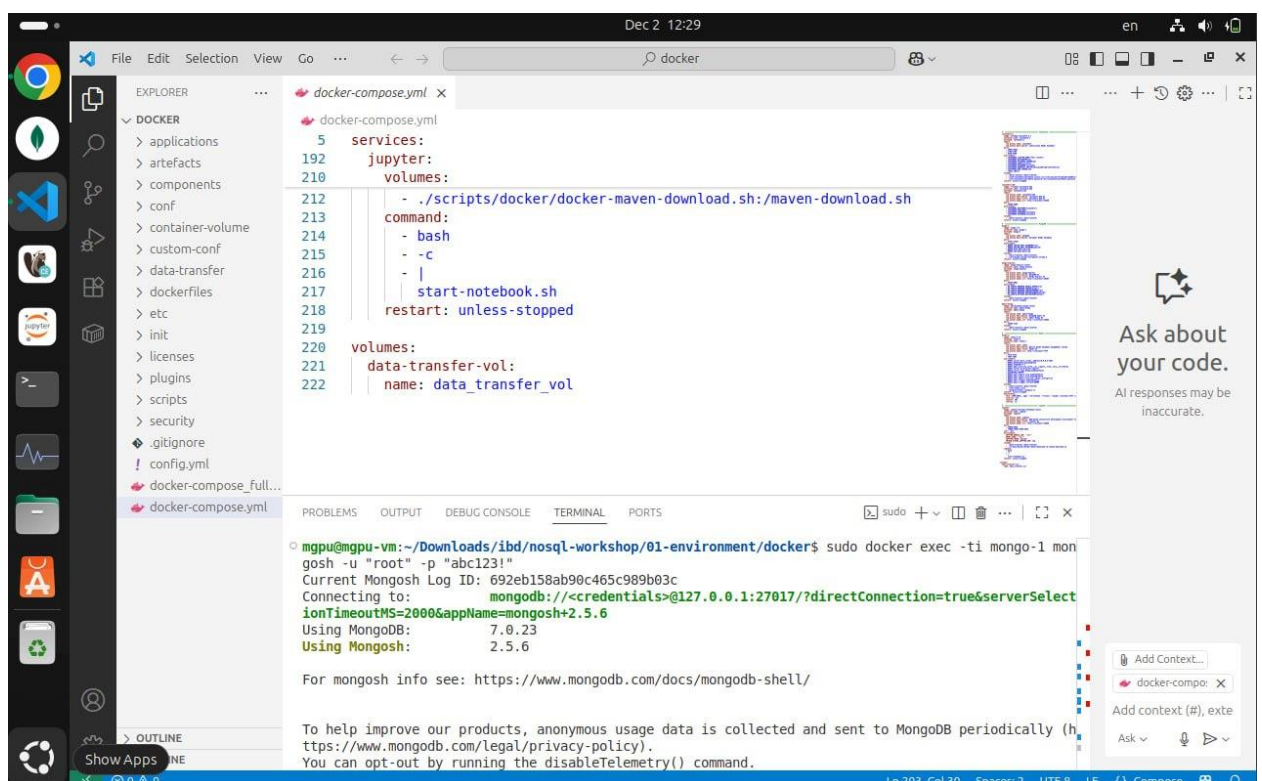
## Вариант 15.

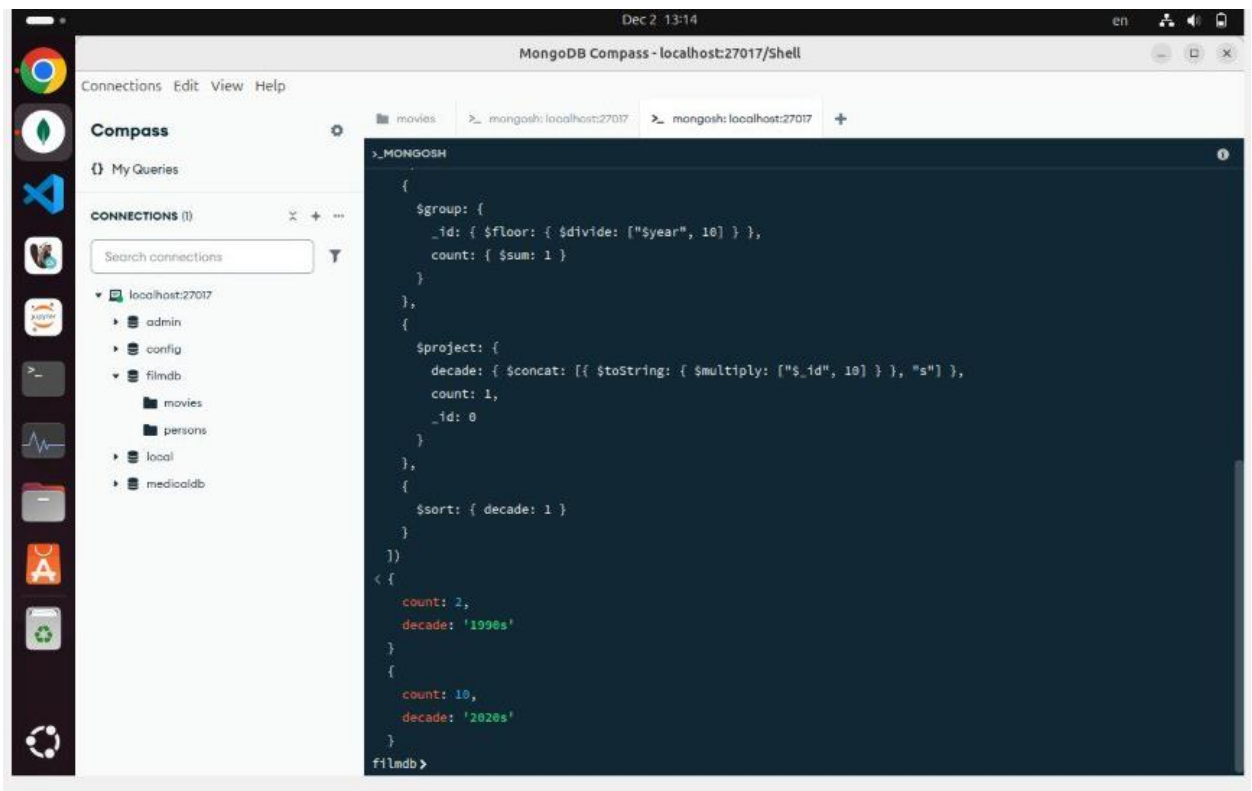
### Подготовка окружения.



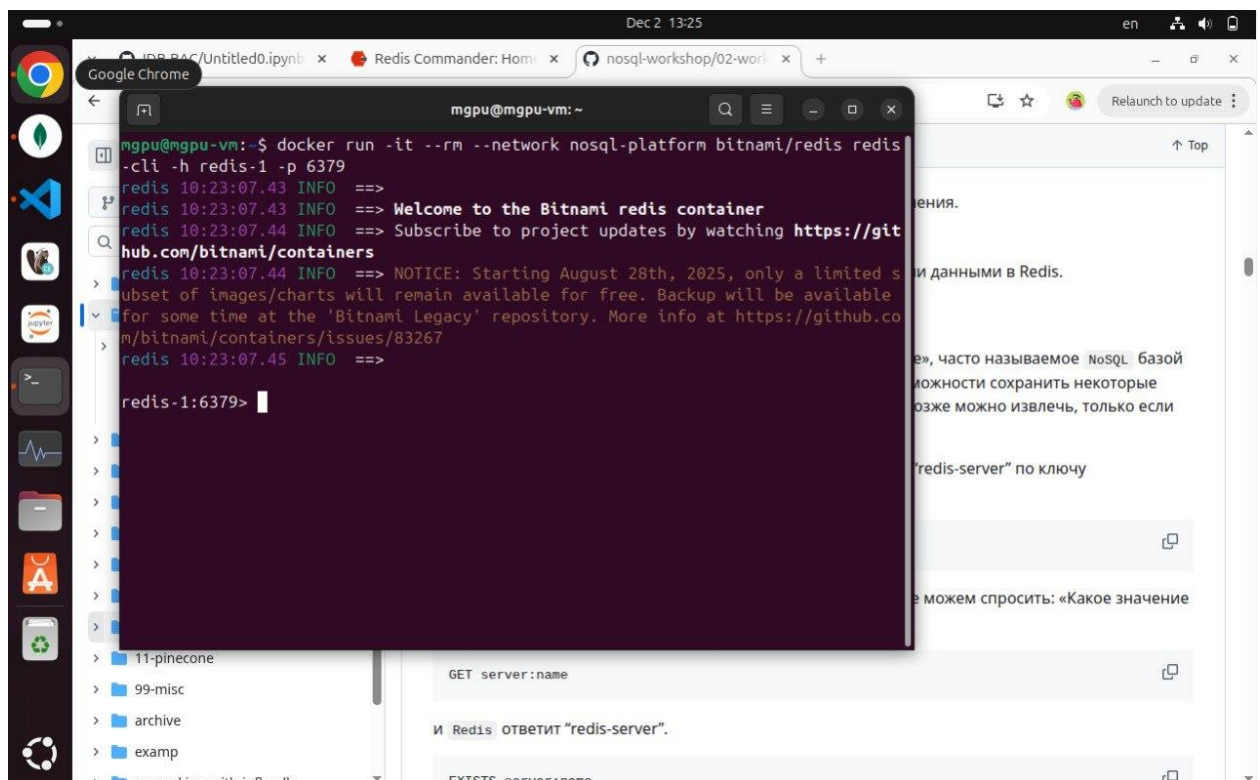
## Задание для самостоятельной работы MongoDB

№1. Используя агрегацию, посчитать количество фильмов, выпущенных в каждое десятилетие.



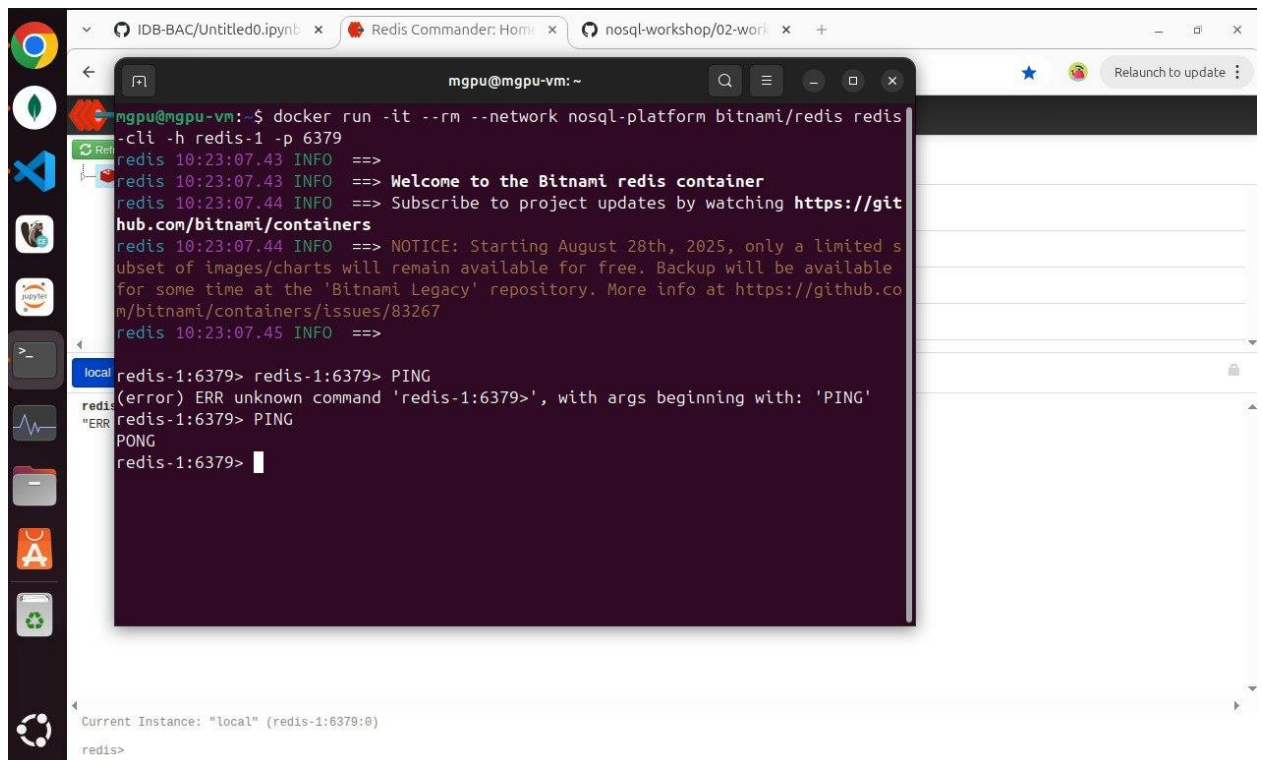


## Задание для самостоятельной работы Redis

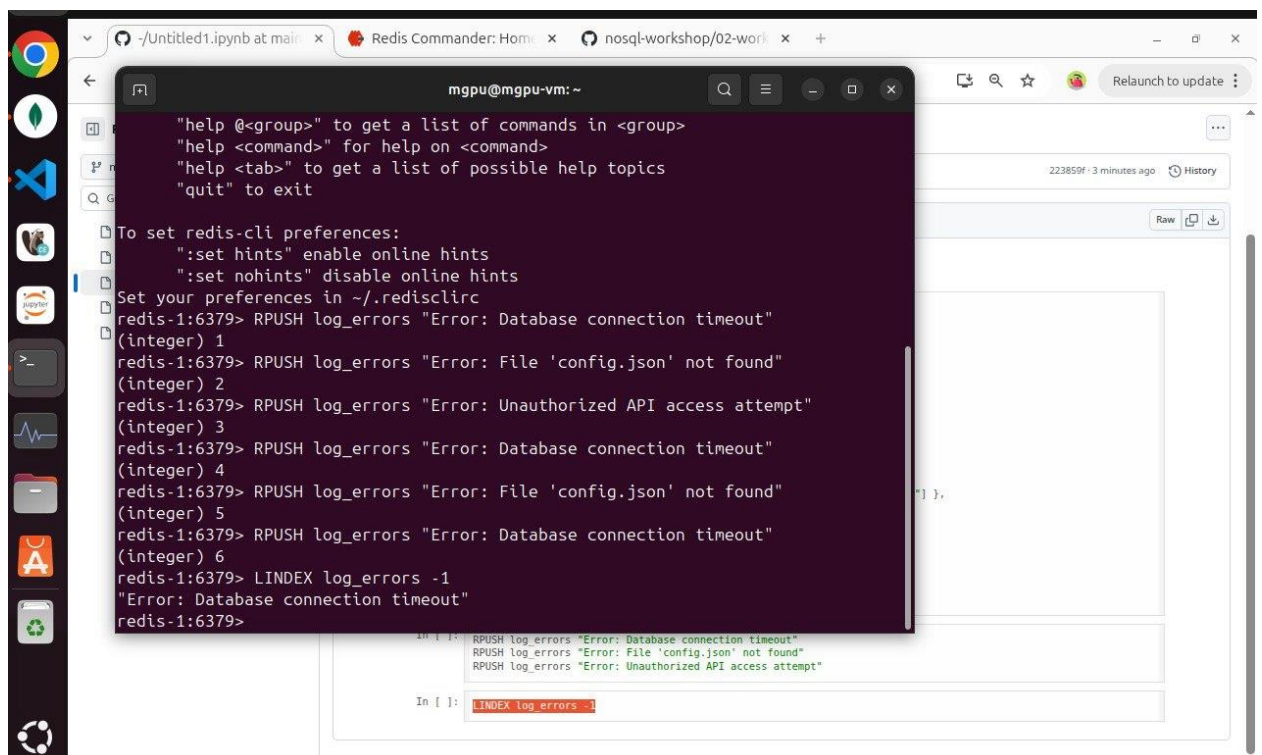


## №2 Задание 3 (Redis)

Создать список `log_errors` и добавить в него 3 сообщения об ошибках. Получить последнее добавленное сообщение, не удаляя его из списка (`LINDEX`)



```
mgpu@mgpu-vm:~$ docker run -it --rm --network nosql-platform bitnami/redis redis
cli -h redis-1 -p 6379
redis 10:23:07.43 INFO ==>
redis 10:23:07.43 INFO ==> Welcome to the Bitnami redis container
redis 10:23:07.44 INFO ==> Subscribe to project updates by watching https://github.com/bitnami/containers
redis 10:23:07.44 INFO ==> NOTICE: Starting August 28th, 2025, only a limited subset of images/charts will remain available for free. Backup will be available for some time at the 'Bitnami Legacy' repository. More info at https://github.com/bitnami/containers/issues/83267
redis 10:23:07.45 INFO ==>
redis-1:6379> redis-1:6379> PING
(error) ERR unknown command 'redis-1:6379', with args beginning with: 'PING'
redis-1:6379> PING
PONG
redis-1:6379>
```



```
mgpu@mgpu-vm:~$
"help @<group>" to get a list of commands in <group>
"help <command>" for help on <command>
"help <tab>" to get a list of possible help topics
"quit" to exit

To set redis-cli preferences:
  ":set hints" enable online hints
  ":set nohints" disable online hints
Set your preferences in ~/.rediscli.rc
redis-1:6379> RPUSH log_errors "Error: Database connection timeout"
(integer) 1
redis-1:6379> RPUSH log_errors "Error: File 'config.json' not found"
(integer) 2
redis-1:6379> RPUSH log_errors "Error: Unauthorized API access attempt"
(integer) 3
redis-1:6379> RPUSH log_errors "Error: Database connection timeout"
(integer) 4
redis-1:6379> RPUSH log_errors "Error: File 'config.json' not found"
(integer) 5
redis-1:6379> RPUSH log_errors "Error: Database connection timeout"
(integer) 6
redis-1:6379> LINDEX log_errors -1
"Error: Database connection timeout"
redis-1:6379>
```

**Задание для самостоятельной работы Neo4j**

## №3 Задание 2 (Neo4j)

Найти всех актеров, которые работали с режиссерами Вачовски (Lille или Lana)

The screenshot shows the Neo4j Browser interface. On the left, the 'Database Information' panel is visible, showing the database 'neo4j' and node labels 'Movie' and 'Person'. The main panel displays a Cypher query:

```
1 MATCH (actor:Person)-[:ACTED_IN]-(movie:Movie)←[:DIRECTED]-(wachowski:Person)
2 WHERE wachowski.name IN ['Lilly Wachowski', 'Lana Wachowski']
3 RETURN DISTINCT actor.name AS actor
4 ORDER BY actor.name
5
6
```

The results are shown in a table view with the column 'actor'.

actor
"Ben Miles"
"Carrie-Anne Moss"
"Christina Ricci"
"Emil Eifrem"
"Emile Hirsch"
"Halle Berry"

The screenshot shows the Neo4j Browser interface. On the left, the 'Database Information' panel is visible, showing the database 'neo4j' and node labels 'Movie' and 'Person'. The main panel displays a Cypher query:

```
neo4j$ MATCH (actor:Person)-[:ACTED_IN]-(movie:Movie)←[:DIRECTED]-(wachowski:Person)
WHERE wachowski.name IN ['Lilly Wachowski', 'Lana Wachowski']
RETURN DISTINCT actor.name AS actor
ORDER BY actor.name
```

The results are shown in a graph view, displaying a network of nodes and relationships. The nodes are labeled with names, and the relationships are represented by lines connecting them. The graph shows a central node 'Lana Wachowski' connected to several other nodes, including 'Ben Miles', 'Carrie-Anne Moss', 'Christina Ricci', 'Emil Eifrem', 'Emile Hirsch', and 'Halle Berry'.



## Вывод

В ходе выполнения задания были успешно применены три ключевые технологии из экосистемы NoSQL, каждая из которых решает свои специфические задачи:

1. MongoDB (документная база данных)
  - Показала свою гибкость в работе со структурированными, но схемой-независимыми данными (фильмы с годами выпуска).
  - С помощью агрегационного конвейера удалось эффективно группировать данные по десятилетиям, что демонстрирует мощь MongoDB в аналитических задачах без необходимости писать сложный код на стороне приложения.
2. Redis (ключ-значение хранилище, in-memory)
  - Продемонстрировала скорость и простоту для операций с временными или лог-подобными данными.
  - Использование списка `log_errors` и команды `LINDEX` показало, как Redis может служить буфером для логирования, мониторинга или очередей в реальном времени.
3. Neo4j –
  - возможность находить связи между сущностями (актёры ↔ режиссёры) иллюстрирует силу графовых БД в задачах, где важны отношения, а не только атрибуты объектов.