**Лабораторная работа № 1**

**ЗНАКОМСТВО С MONGODB**

**Цель работы:** изучить основы работы с коллекциями в документоориентированной СУБД *MongoDB*, научиться проектировать запросы в команде «find».

**Теоретические сведения:**

MongoDB — документоориентированная система управления базами данных, не требующая описания схемы таблиц. Считается одним из классических примеров NoSQL-систем, использует JSON-подобные документы и схему базы данных. Написана на языке C++. Применяется в веб-разработке, в частности, в рамках JavaScript-ориентированного стека MEAN.

Система поддерживает ad-hoc-запросы: они могут возвращать конкретные поля документов и пользовательские JavaScript-функции. Поддерживается поиск по регулярным выражениям. Также можно настроить запрос на возвращение случайного набора результатов.

Имеется поддержка индексов.

Система может работать с набором реплик, то есть содержать две или более копии данных на различных узлах. Каждый экземпляр набора реплик может в любой момент выступать в роли основной или вспомогательной реплики. Все операции записи и чтения по умолчанию осуществляются с основной репликой. Вспомогательные реплики поддерживают в актуальном состоянии копии данных. В случае, когда основная реплика дает сбой, набор реплик проводит выбор, которая из реплик должна стать основной. Второстепенные реплики могут дополнительно являться источником для операций чтения.

Система масштабируется горизонтально, используя технику сегментирования (англ. sharding) объектов баз данных — распределение их частей по различным узлам кластера. Администратор выбирает ключ сегментирования, который определяет, по какому критерию данные будут разнесены по узлам (в зависимости от значений хэша ключа сегментирования). Благодаря тому, что каждый узел кластера может принимать запросы, обеспечивается балансировка нагрузки.

**Задание к лабораторной работе:**

1. *Создать новый документ в коллекции;*
2. *Спроектировать запрос по поиску документа в коллекции;*
3. *Спроектировать запрос, используя «$or»;*
4. *Спроектировать запрос, используя «$sort»;*

**Выполнение работы:**

1. Создаем документ в коллекции, нажав на кнопку «Insert Document» и определив некоторые атрибуты (Рисунок 1).

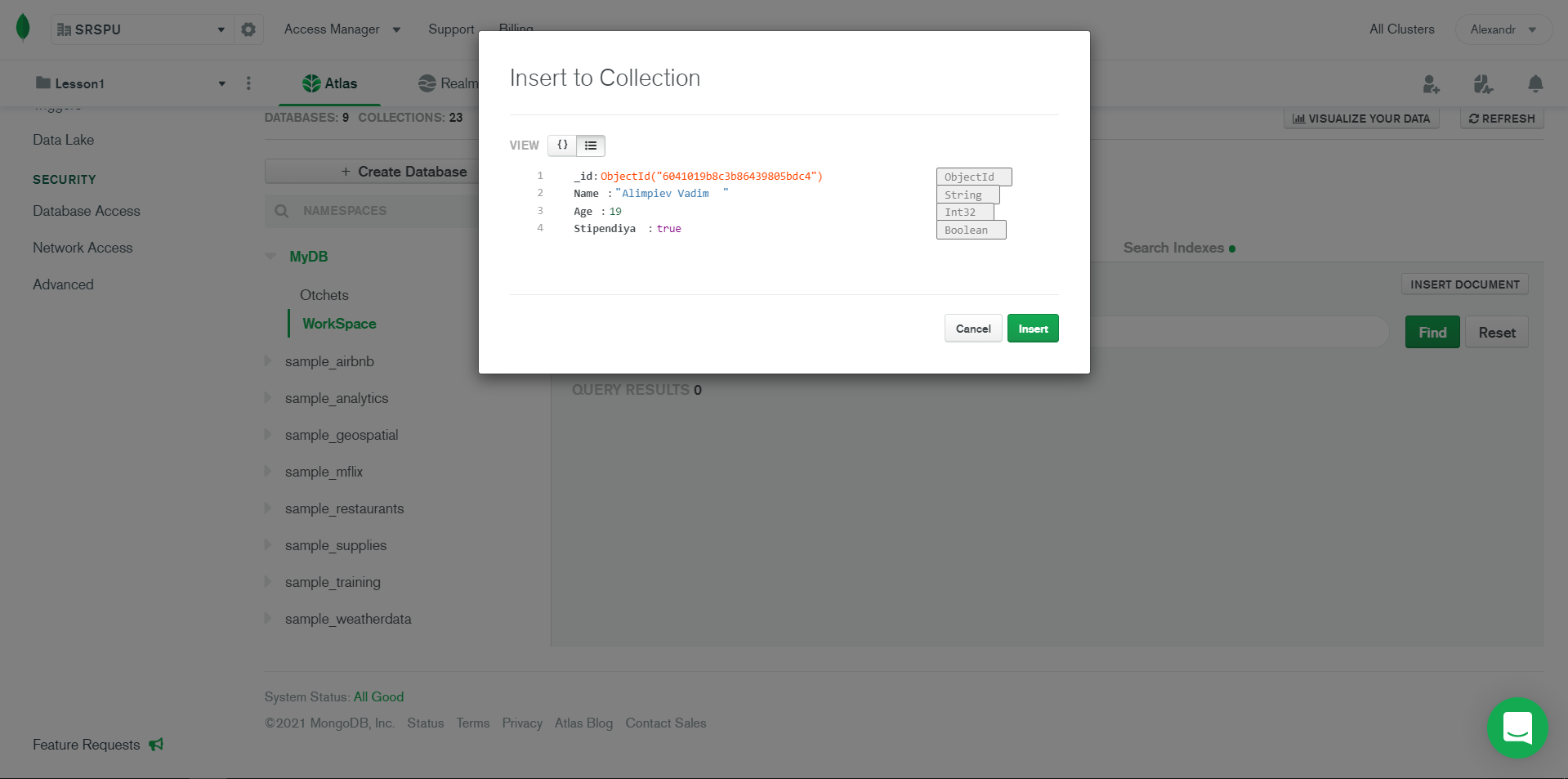


Рисунок 1 - Создание документа

Добавим еще несколько документов, чтобы продолжить работу в нашей БД.

1. Теперь напишем запрос, который найдет документы с необходимым нам значением атрибута {name:” имя”}(Рисунок 2).

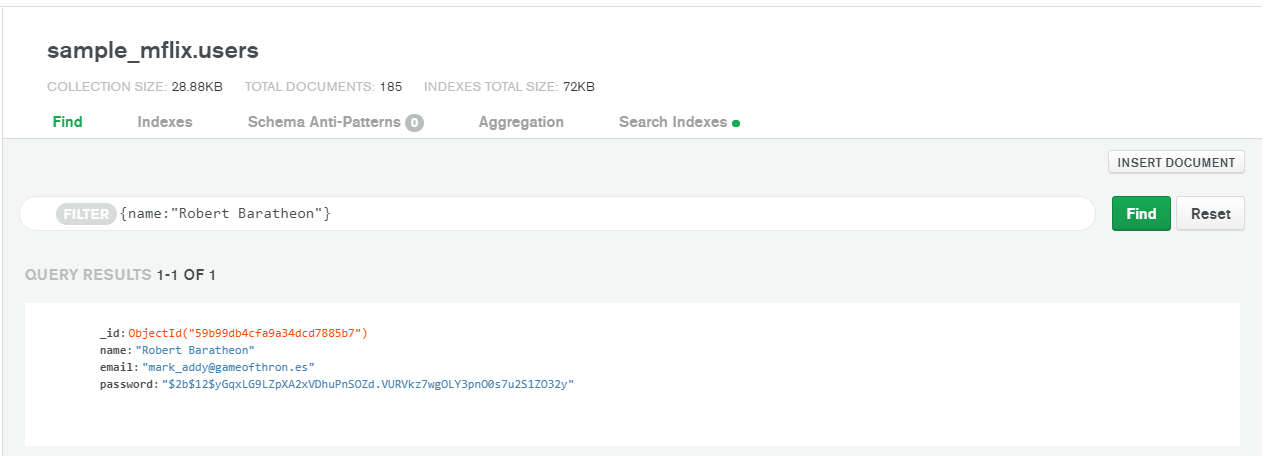


Рисунок 2 - Выполнение запроса

1. Теперь напишем запрос, включающий в себя операцию «или», то есть, используем в запросе «$or» как строится запрос {$or:[{account\_id:номер\_id},{ account\_id:номер\_id }]}(Рисунок 3).



Рисунок 3 - Выполнение запроса с "$or"

Как видно на рисунке, выведены документы, удовлетворяющие хотя бы одному из требований в запросе.

1. Вывести запрос, использую сортировку, можно следующим образом: Переходим на вкладку «Aggregation». Теперь используем этап «$sort» из «Aggregation Pipeline Stages». Результат выполнения и команда изображены на Рисунке 4.

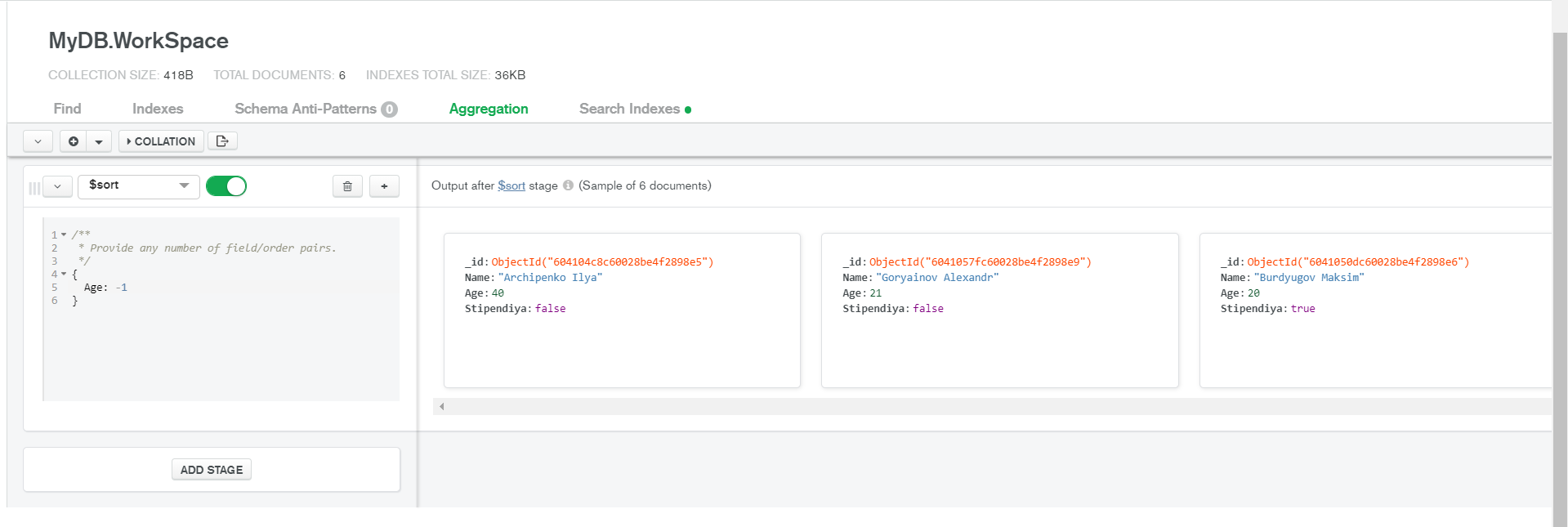


Рисунок 4 – Результат обработки этапа «$sort»

Таким образом, выведен запрос, в котором документы отсортированы в порядке уменьшения атрибута «Age».

***Вывод:***

В ходе выполнения лабораторной работы мною были получен опыт в основах работы с коллекциями в *MongoDB*, научился проектироватьзапросы в команде(поиск по ключевому слову ,поиск по логическому или “$or”) в поле «find».