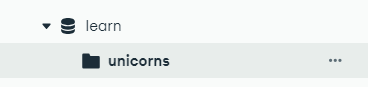
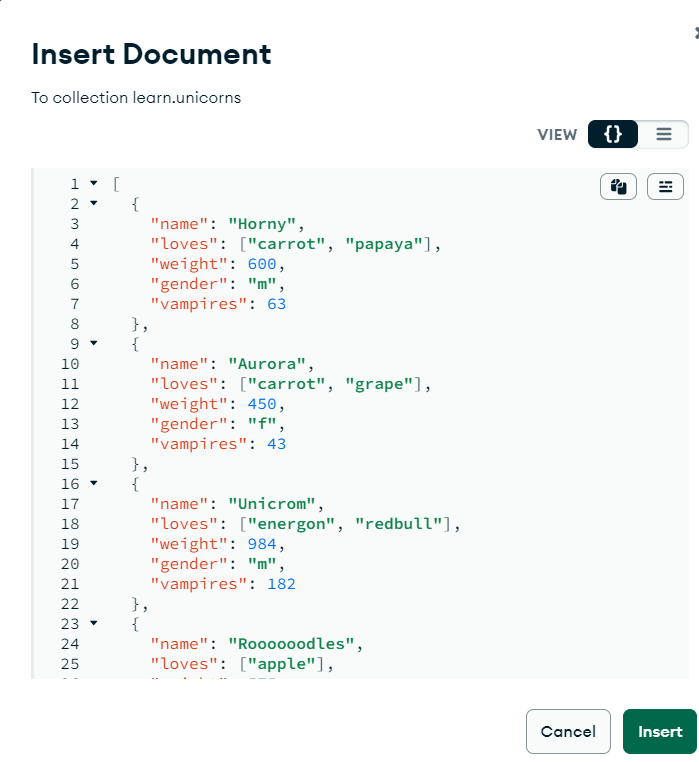
Отчет лр №6

## 2 Вставка документов в коллекцию

Практическое задание 2.1.1:

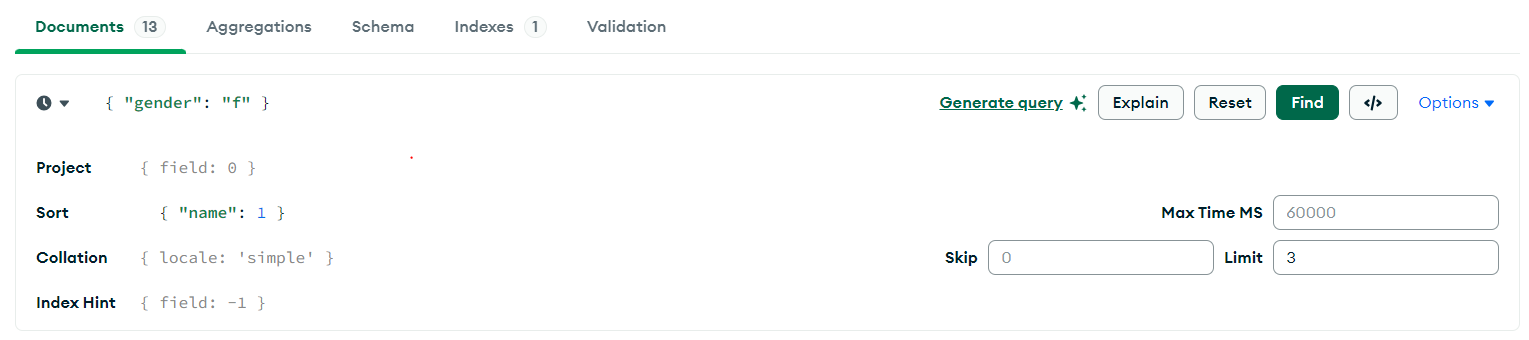
1. Создание базы данных learn и коллекции unicorns  
   
2. Заполнение коллекции данными  
   
3. Альтернативный способ заполнения коллекции (через документ)

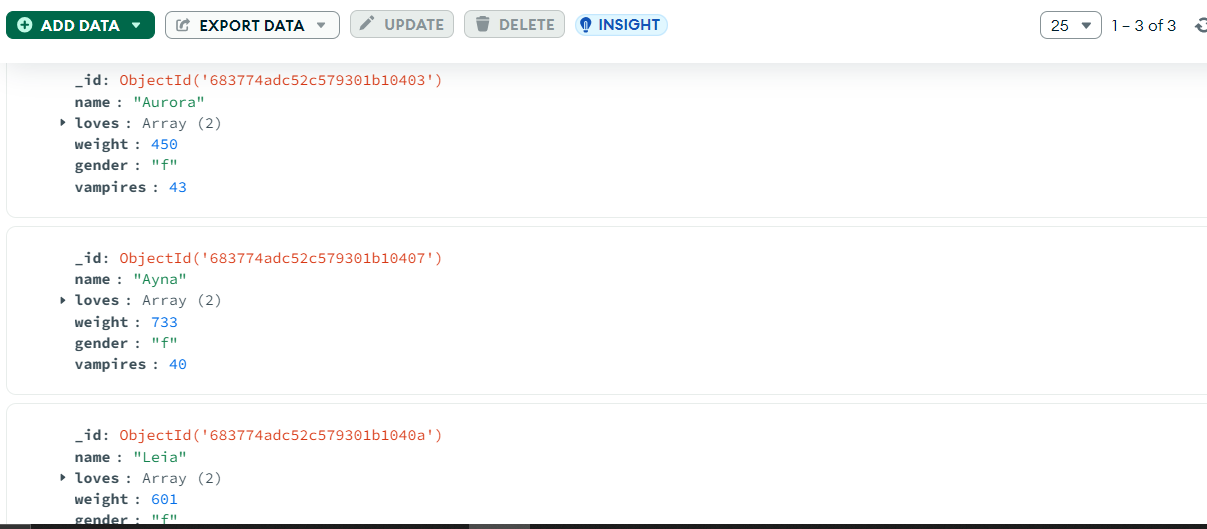
Так как я работаю через MongoDB Compass я выполнил этот пункт аналогично с пунктом 2.

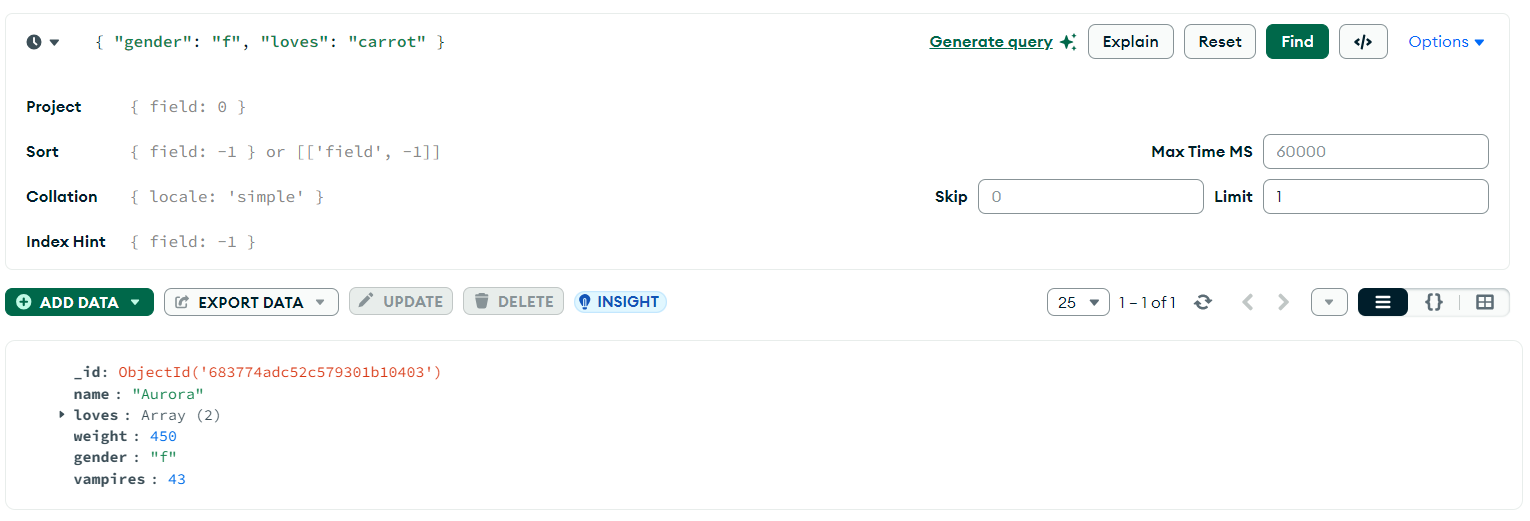
1. Документы успешно были добавлены в коллекцию  
   

## 2.2 Выборка данных из БД

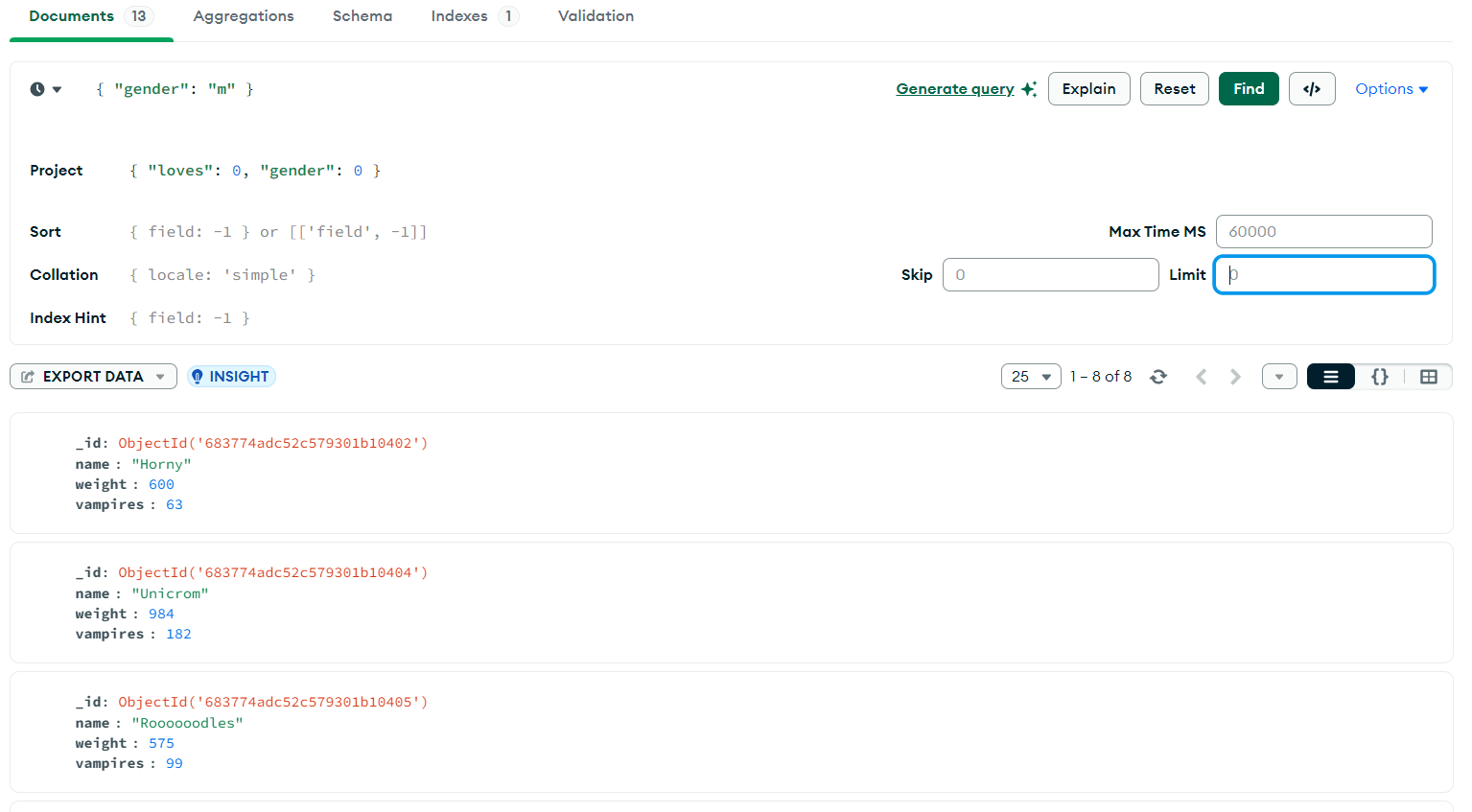
Практическое задание 2.2.1:

1. Вывод списка самок единорогов, ограниченный тремя записями и отсортированный по имени.  
     
   Вывод:   
   

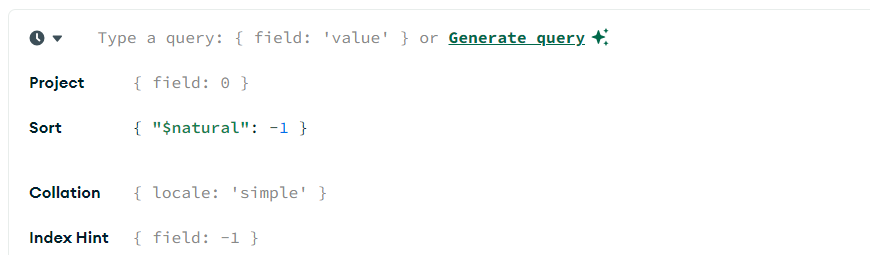
Результат вывода:  


1. Поиск все самок, которые любят carrot, вывод только первой особи  
   

Практическое задание 2.2.2:

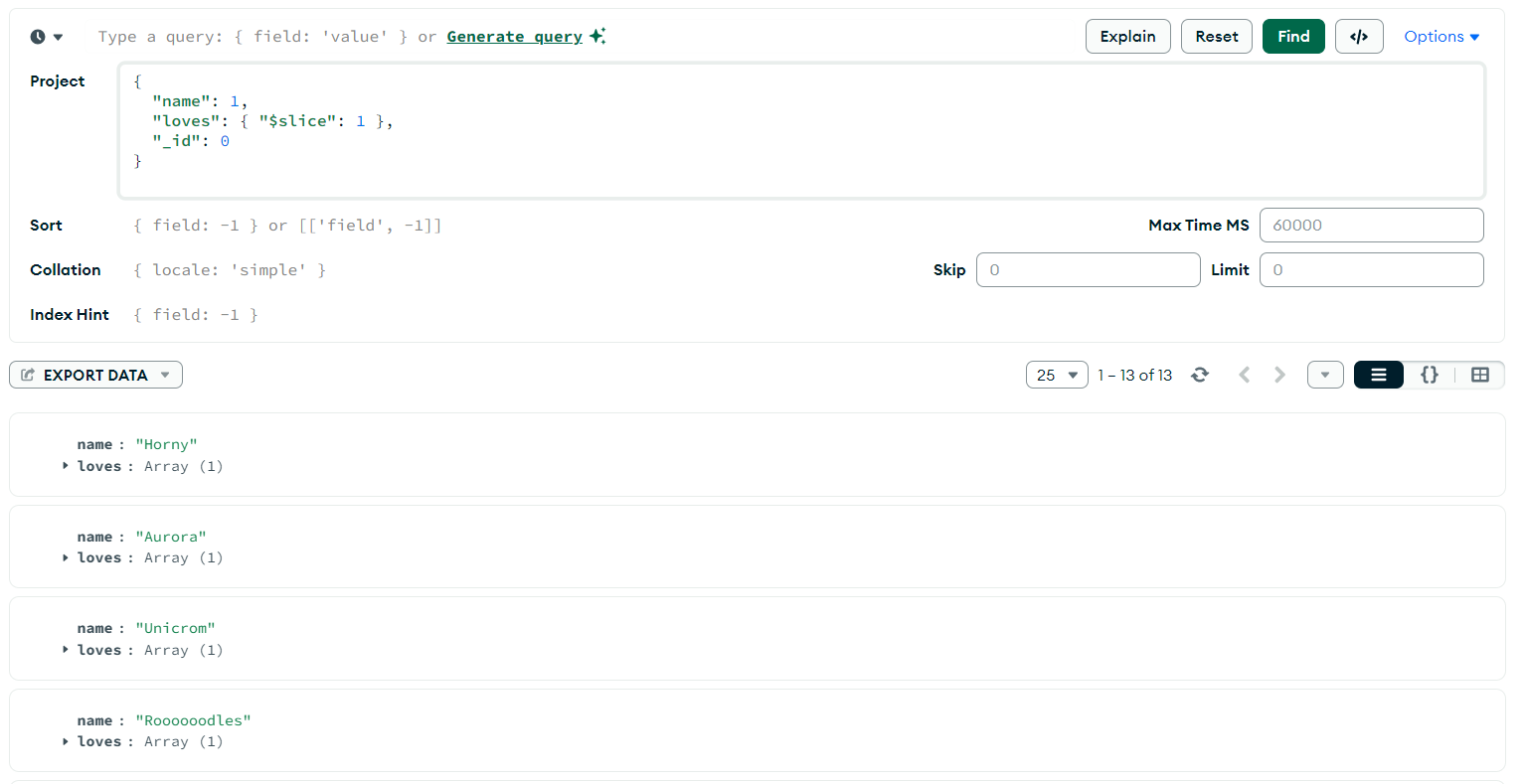
1. Вывод всех самцов, исключив предпочтение и пол  
   

Практическое задание 2.2.3:

1. Вывод всех единорогов в обратном порядке добавления  
   

Практическое задание 2.1.4:

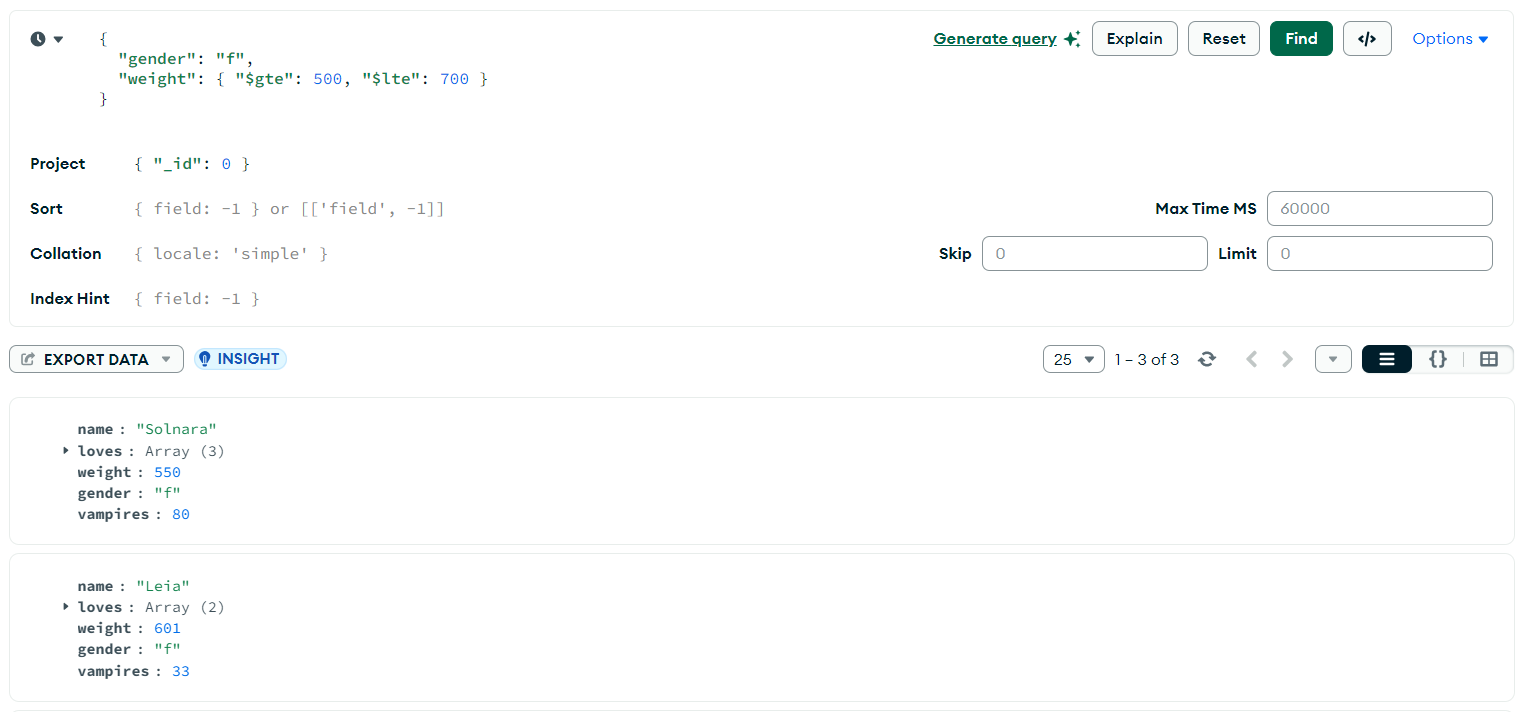
1. Вывод списка единорогов, с названием первого предпочтения и исключение идентификатора



## 2.3 Логические операторы

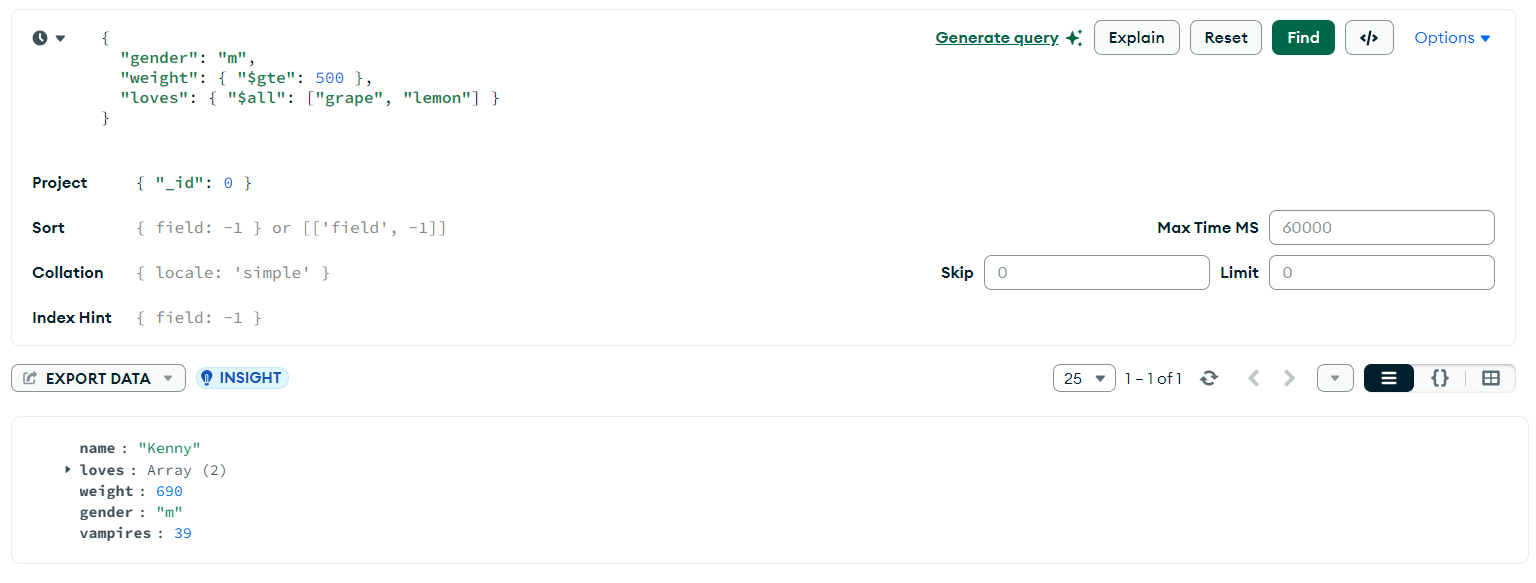
Практическое задание 2.3.1:

1. Вывести список самок единорогов весом от 500 до 700 кг, исключив вывод идентификатора

****

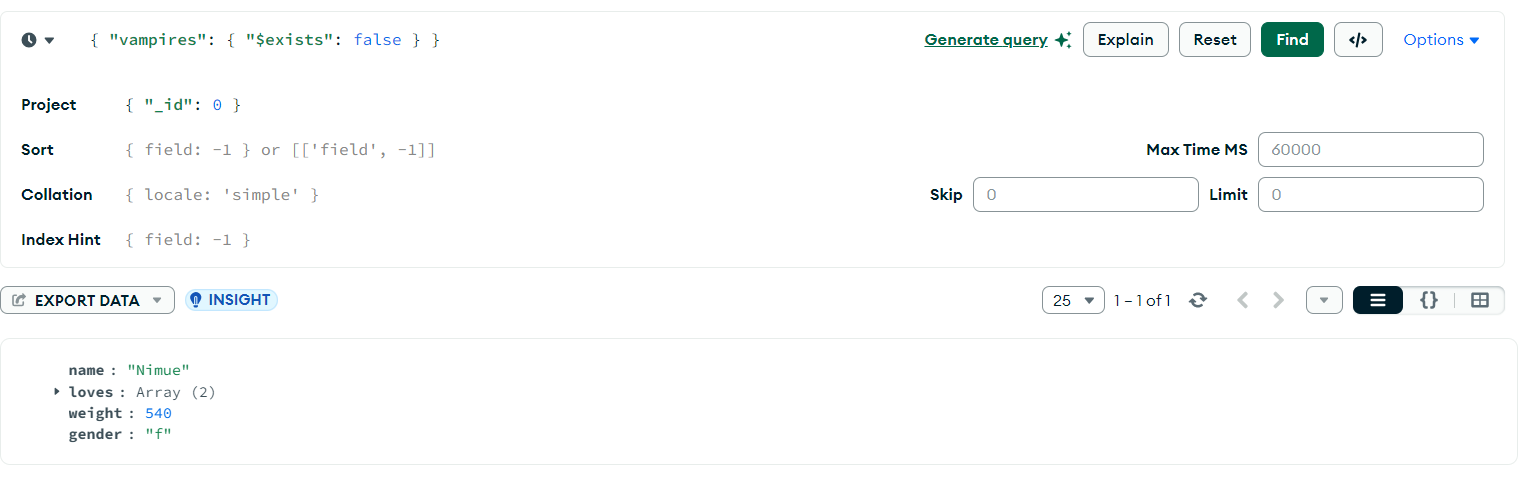
Практическое задание 2.3.2:

1. Вывести список самцов единорогов весом от полутонны и предпочитающих grape и lemon, исключив вывод идентификатора.



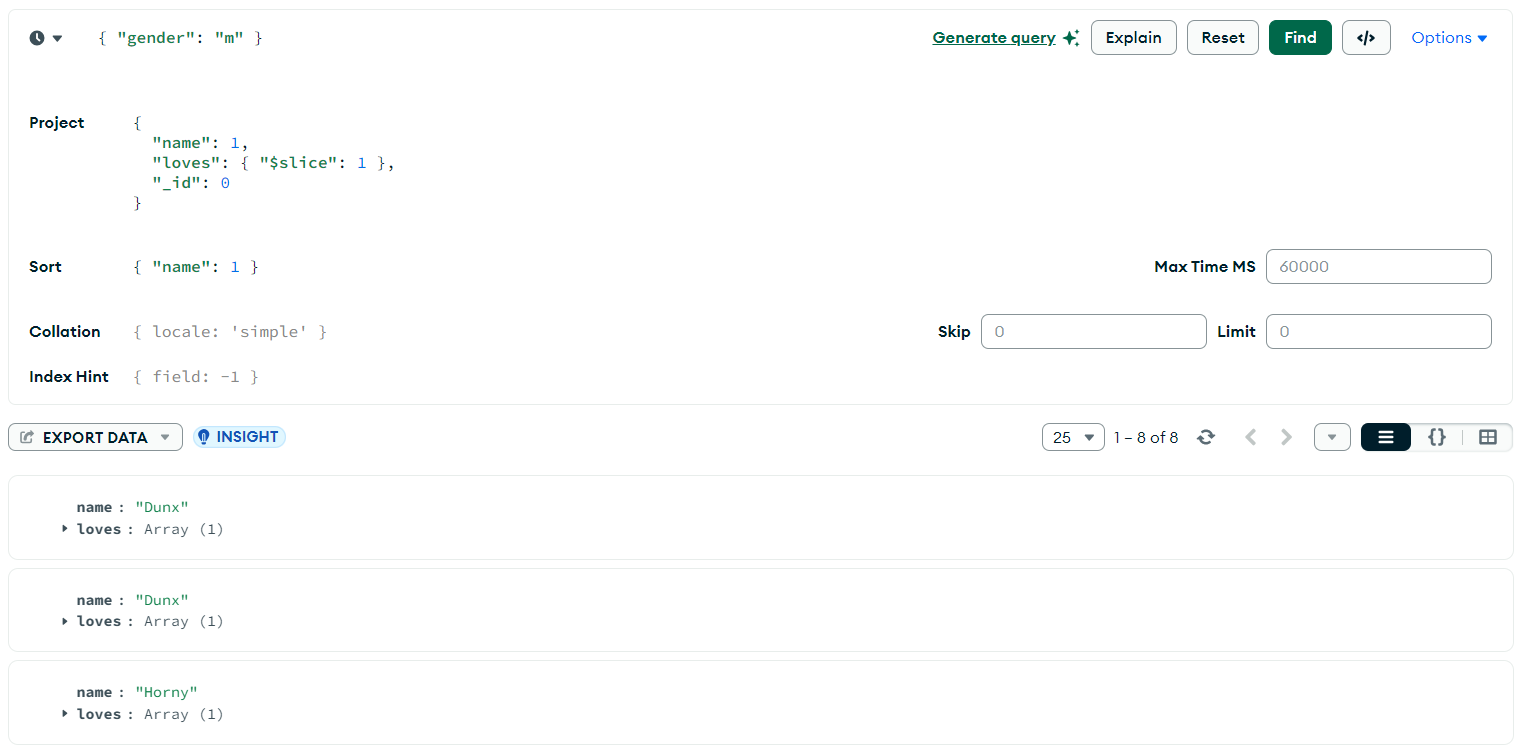
Практическое задание 2.3.3:

1. Найти всех единорогов, не имеющих ключ vampires



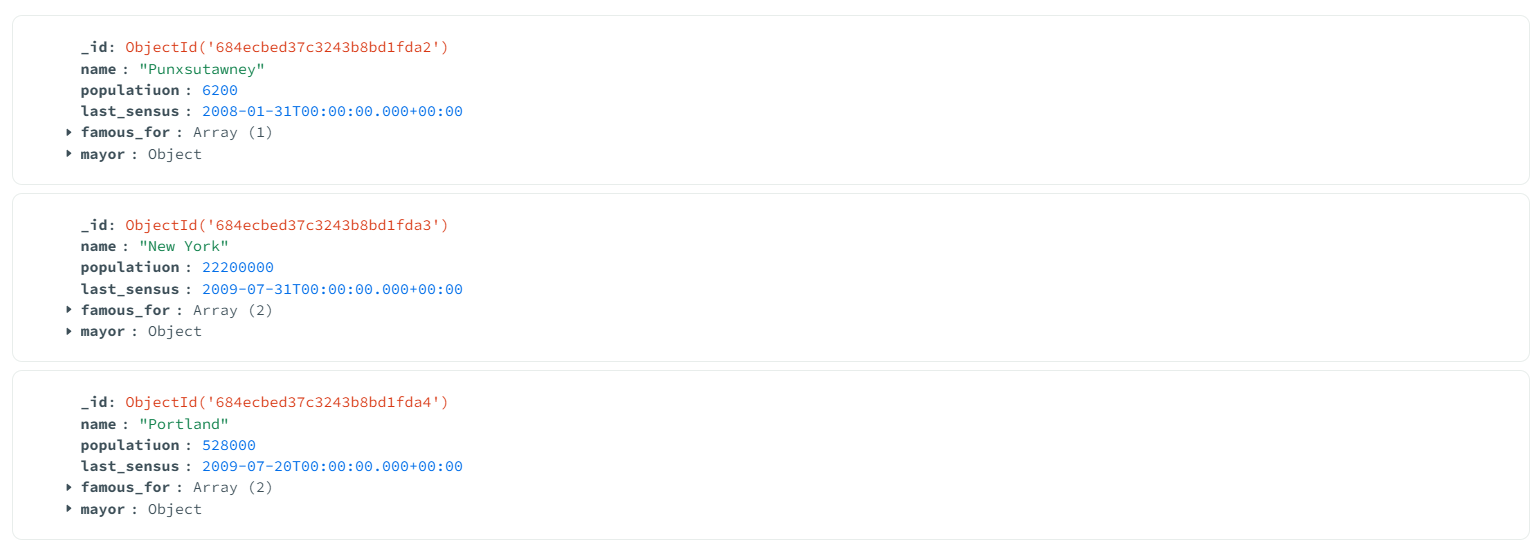
Практическое задание 2.3.4:

1. Вывести список упорядоченный список имен самцов единорогов с информацией об их первом предпочтении.

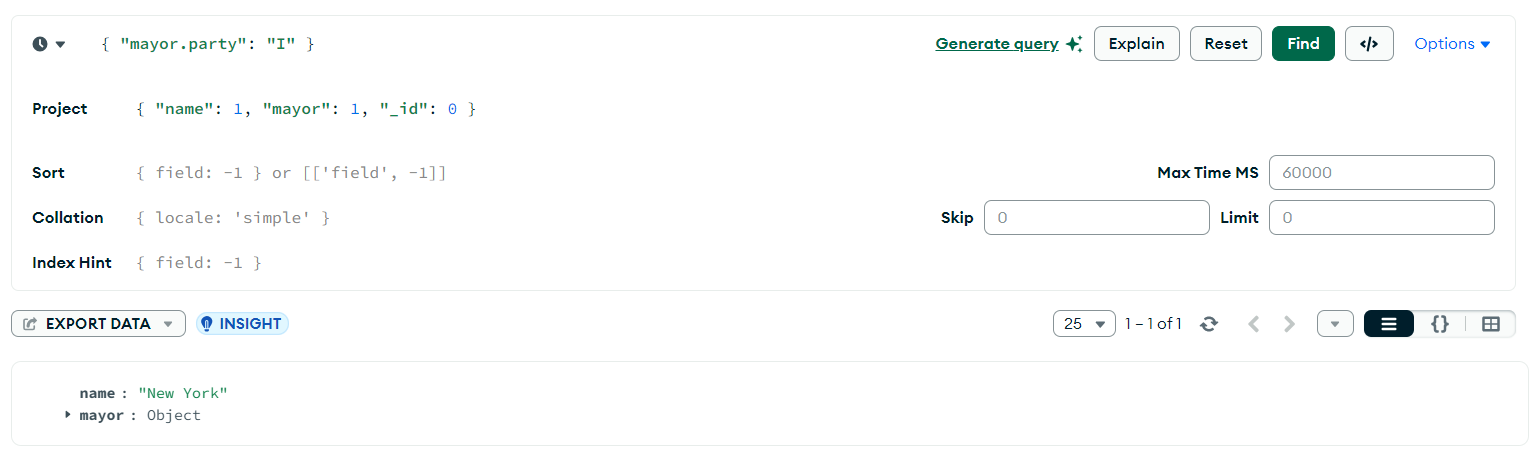


Практическое задание 3.1.1:

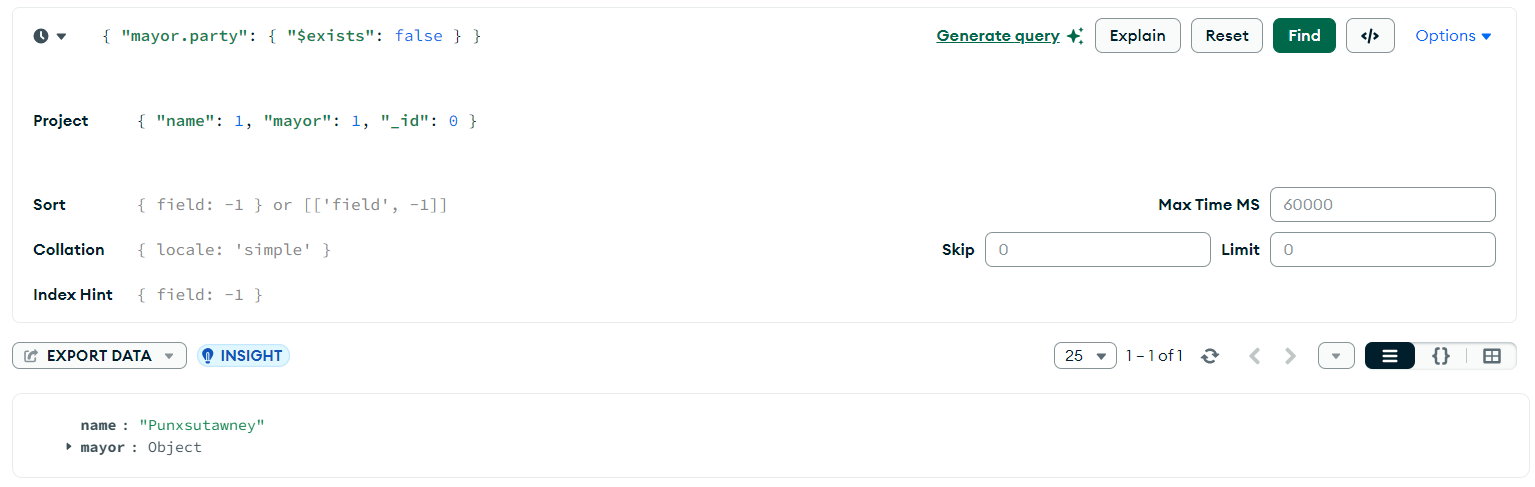
1. Создайте коллекцию towns



1. Сформировать запрос, который возвращает список городов с независимыми мэрами (party="I"). Вывести только название города и информацию о мэре



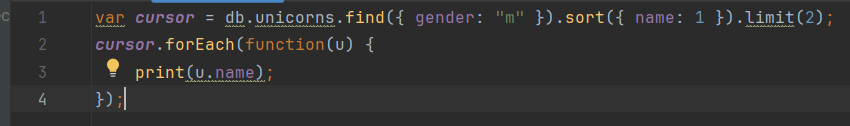
1. Сформировать запрос, который возвращает список беспартийных мэров (party отсутствует). Вывести только название города и информацию о мэре.



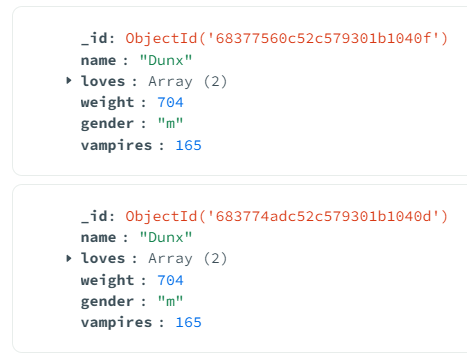
## 3.1 Запрос к вложенным объектам

Практическое задание 3.1.2:

Сформировать функцию для вывода самцов единорога



Скрипт который можно использовать в mongosh

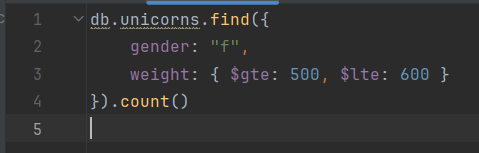


аналогичный вывод в compass.

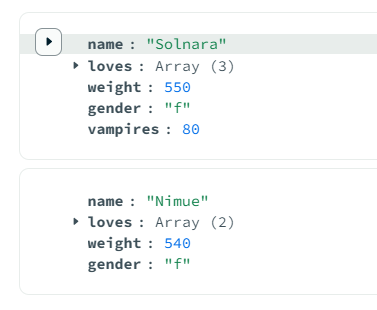
## 3.2 Агрегированные запросы

Практическое задание 3.2.1:

1. Вывести количество самок единорогов весом от полутонны до 600 кг.



Скрипт JS



Аналогичный результат вывода в compass.

Практическое задание 3.2.2:

1. Вывести список предпочтений

db.unicorns.distinct("loves")

Практическое задание 3.2.3

1. Подсчитать количество особей единорогов обоих полов

db.unicorns.aggregate([

{ $group: { \_id: "$gender", count: { $sum: 1 } } }

])

## 3.3 Редактирование данных.

Практическое задание 3.3.1:

1. Выполнить команду

db.unicorns.save({name: 'Barny', loves: ['grape'],

weight: 340, gender: 'm'})



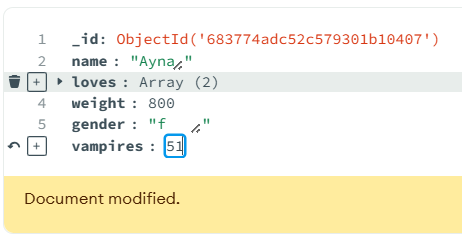
1. Проверить содержимое коллекции unicorns



Добавилась одна запись.

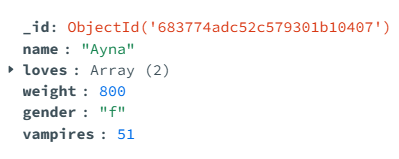
Практическое задание 3.3.2:

1. Для самки единорога Ayna внести изменения в БД: теперь ее вес 800, она убила 51 вампира



В compass есть возможность редактировать записи в ручном режиме.

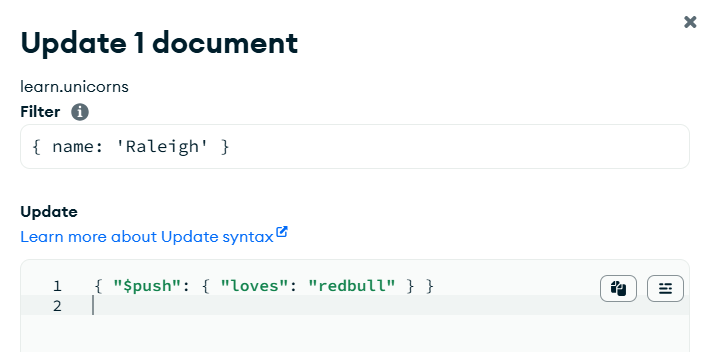
1. Проверить содержимое коллекции



Данные изменены.

Практическое задание 3.3.3:

1. Для самца единорога Raleigh внести изменения в БД: теперь он любит рэдбул.



Изменяем предпочтения через Update

Практическое задание 3.3.4:

1. Всем самцам единорогов увеличить количество убитых вампиров на 5.



Практическое задание 3.3.5:

1. Изменить информацию о городе Портланд: мэр этого города теперь беспартийный.



Практическое задание 3.3.6:

1. Изменить информацию о самце единорога Pilot: теперь он любит и шоколад.



Практическое задание 3.3.7:

1. Изменить информацию о самке единорога Aurora: теперь она любит еще и сахар, и лимоны.



### 3.4 Удаление данных из коллекции

Практическое задание 3.4.1:

1. Создайте коллекцию towns, включающую следующие документы

[

{

"name": "Punxsutawney",

"popujatiuon": 6200,

"last\_sensus": { "$date": "2008-01-31T00:00:00Z" },

"famous\_for": ["phil the groundhog"],

"mayor": {

"name": "Jim Wehrle"

}

},

{

"name": "New York",

"popujatiuon": 22200000,

"last\_sensus": { "$date": "2009-07-31T00:00:00Z" },

"famous\_for": ["status of liberty", "food"],

"mayor": {

"name": "Michael Bloomberg",

"party": "I"

}

},

{

"name": "Portland",

"popujatiuon": 528000,

"last\_sensus": { "$date": "2009-07-20T00:00:00Z" },

"famous\_for": ["beer", "food"],

"mayor": {

"name": "Sam Adams",

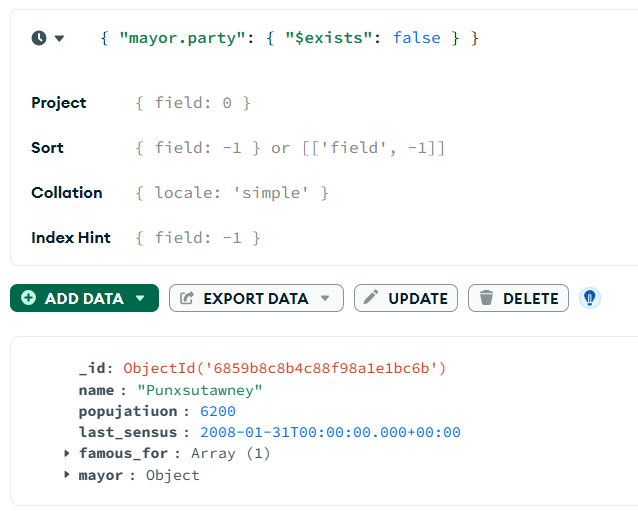
"party": "D"

}

}

]

1. Удалите документы с беспартийными мэрами.



Находим и удаляем

1. Удаляем коллекцию вручную

## 4 ссылки и работа с индексами в базе данных MongoDB

практическое задание 4.4.1:

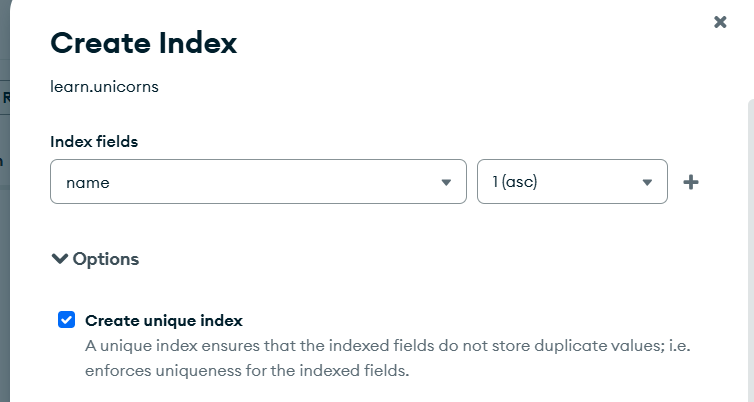
1. Создайте коллекцию зон обитания единорогов, указав в качестве идентификатора кратко название зоны, далее включив полное название и описание.

Создаем коллекцию habitats

Проверьте содержание коллекции единорогов.  

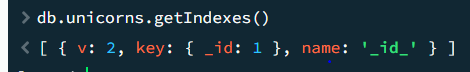
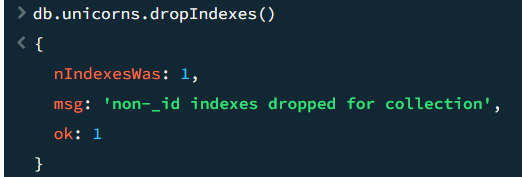

## 4.2 Настройка индексов

Практическое задание 4.2.1:

1. Проверьте, можно ли задать для коллекции unicorns индекс для ключа name с флагом unique.  
     
   Такой индекс нельзя создать.

## 4.3 Управление индексами

практическое задание 4.3.1:

1. Получите информацию о всех индексах коллекции unicorns .  
   
2. Удалите все индексы, кроме индекса для идентификатора.  
   
3. Попытайтесь удалить индекс для идентификатора.

* Этот индекс нельзя удалить, пример выше.

## 4.4 План запроса

Практическое задание 4.4.1:

1. Создайте объемную коллекцию numbers, задействовав курсор:

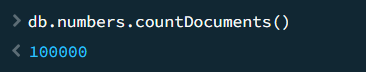
for(i = 0; i < 100000; i++){db.numbers.insert({value: i})}  
  
Вставляем документы в коллекцию:  
const ops = [];

for (let i = 0; i < 100000; i++) {

ops.push({ insertOne: { document: { value: i } } });

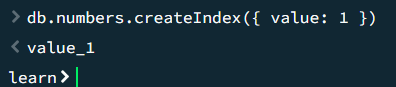
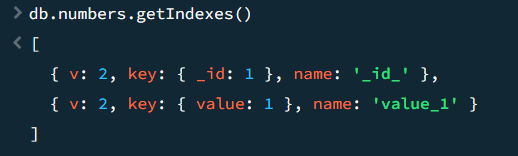
}

db.numbers.bulkWrite(ops);



1. Выберите последних четыре документа.  
   
2. Проанализируйте план выполнения запроса 2. Сколько потребовалось времени на выполнение запроса? (по значению параметра executionTimeMillis)

На выполнение запроса потребовалось менее 1 миллисекунды, MongoDB просмотрел 4 последних документа.

1. Создайте индекс для ключа value.  
   
2. Получите информацию о всех индексах коллекции numbres.  
   
3. Выполните запрос 2.
4. Проанализируйте план выполнения запроса с установленным индексом. Сколько потребовалось времени на выполнение запроса?

На выполнение запроса также потребовалось менее 1 миллисекунды.

1. Сравните время выполнения запросов с индексом и без. Дайте ответ на вопрос: какой запрос более эффективен?

Запрос с индексом по полю value гораздо эффективнее, так как он использует индекс для быстрой сортировки и поиска.  
Запрос без индекса требует полного сканирования коллекции (COLLSCAN), что занимает гораздо больше времени.