Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра проектирования информационно-компьютерных систем

**ОТЧЕТ**

по лабораторной работе №5

«Основы NoSQL»

Выполнил: Проверил:

студент гр.914301 Коваленко Д.И.

Кондрашова В.А

Минск 2022

# Цель работы

Познакомиться с основами работы NoSQL базы данных MongoDB. Научится производить установку и запуск MongoDB.

Ознакомиться с консолью mongo, с утилитами для взаимодействия с базой данных mongo.

Научиться взаимодействовать с базой данных mongo посредством консоли и пользовательских интерфейсов сторонних фирм.

# Ответы на контрольные вопросы

**Что означает термин NoSQL.** NoSQL – Not only SQL – это ряд технологий, подходов, проектов направленных на реализацию моделей баз данных, имеющих существенные отличия от традиционных СУБД, работающих с языком SQL. Концепция NoSQL не отрицает SQL, а стремится решить проблемы и вопросы, с которыми недостаточно хорошо справляется РСУБД. Чаще всего данные в NoSQL решении представляются в виде хеш-таблиц, деревьев, документов и других структур.

**Какие преимущества предоставляют NoSQL базы данных по сравнению с РДБ.** Обозначим некоторые характеристики, присущие NoSQL базам данных:

* в NoSQL СУБД не используется SQL (имеется в виду DML4);
* отсутствие необходимости хранить данные с определенной структурой;
* высокая производительность решений, высокая доступность;
* распределенное хранение данных;
* способность к горизонтальному масштабированию по требованию для некоторого набора операций на многих серверах;
* эффективное использование распределенных индексов и памяти для запросов;
* отсутствие транзакций;
* простые протоколы доступа к хранимым данным;
* отсутствие поддержки транзакционной целостности ACID (atomicity, consistency, isolation, durability – атомарность, согласованность, изолированность, долговечность).

**Какими особенностями обладает MongoDB.** Основными особенностями MongoDB являются:

* документно-ориентированное хранилище;
* полная поддержка индексов;
* репликация данных;
* высокая доступность данных;
* способность к горизонтальному масштабированию;
* авто-шардинг;
* поддержка запросов;
* поддержка Map/Reduce;
* поддержка GridFS.

**Сколькими способами можно произвести установку MongoDB (краткое описание).** Для установки и запуска СУБД необходимо зайти на официальную страницу скачивания и скачать бесплатную версию MongoDB Community Server.

Далее необходимо распаковать архив в папку, в которой будет проходить работа с mongo.

После извлечения файлов mongo, необходимо определить место хранения базы данных. По умолчанию MongoDB ищет файлы БД в папке «C:\data\db».

Mongo может работать в двух режимах:

* в качестве обычного приложения Windows;
* в качестве Windows-сервиса.

В случае использования mongo как сервиса, он автоматически запускается при старте системы.

**На какие группы делятся приложения, входящие в состав MongoDB.** Все приложения в составе mongo можно условно разделить на следующие группы:

* приложения ядра базы данных;
* инструменты для дампа бинарных файлов MongoDB;
* инструменты импорта и экспорта данных;
* инструменты для диагностики;
* приложения распределенной файловой системы.

Основными приложениями являются приложения «mongod.exe», «mongos.exe» и «mongo.exe».

Приложение «mongod.exe» является главным процессом MongoDB. Оно работает с данными: обрабатывает запросы, управляет форматами данных и т.д. По сути, «mongod.exe» является сервером базы данных.

Приложение «mongos.exe» предназначено для шардинга данных. Оно представляет собой сервис маршрутизации для конфигурирования шардов, обрабатывает запросы от уровня приложений и определяет местоположение данных в кластере.

Приложение «mongo.exe» является консольным клиентом для «mongod.exe», и представляет собой интерфейс тестирования запросов для работы с базой данных и её администрирования.

**Создает ли MongoDB по умолчанию какую-либо БД (с каким именем).** По умолчанию MongoDB создает базу данных с именем «test».

**Какие существуют способы взаимодействия с БД mongo.** Помимо использования драйверов с сервером баз данных mongo можно взаимодействовать через графические утилиты, разрабатываемые сторонними фирмами.

Различные графические утилиты, предназначенные для взаимодействия с базами данных mongo, позиционируются как средства администрирования.

С MongoDB также можно взаимодействовать посредством http. Взаимодействие происходит с использованием либо HTTP, либо REST интерфейса.

**Существует ли возможность взаимодействовать с БД mongo посредством WEB.** С MongoDB также можно взаимодействовать посредством http. Взаимодействие происходит с использованием либо HTTP, либо REST интерфейса.

**Какими особенностями обладает Mongo Explorer.** MongoExplorer – это инструмент управления MongoDB. Основные особенности приложения:

* MongoExplorer легок в использовании;
* отображает все коллекции и документы базы данных;
* использует удобное дерево для отображения документов;
* полностью поддерживает drag’n’drop;
* in-place редактирование документов.

MongoExplorer представляет собой приложение, написанное на SilverLight. MongoExplorer предоставляет основные возможности по взаимодействию с БД mongo: добавление, редактирование и удаление коллекций/документов/полей. Взаимодействие с БД происходит с использованием графического интерфейса.

**Опишите интерфейс приложения Mongo Explorer.** В интерфейсе MongoExplorer можно выделить пять основных панелей:

* панель работы с коллекциями – содержит перечень коллекций, присутствующих в базе данных, а также предоставляет средства для создания, переименования и удаления коллекций;
* панель работы с документами – содержит перечень всех документов базы данных, и кнопки управления документами: создание документов и их удаление;
* панель ввода запросов – представляет собой поле, предназначенное для ввода запроса;
* панель команд – содержит следующие команды: подключение к базе данных (и отключение от нее), экспорт документа в JSON формате, импорт документа в JSON формате, создание резервной копии БД, сохранение файла на диск (при использовании GridFS9), поиск по документу и использованием фильтра, отображение всех документов, использовать значение поля в качестве имени документа, обновить документы, работа со списком пользователей;
* область работы с документом – представляет документ в графическом виде.

**Каков формат команд командной строки.** В консоли mongo есть несколько глобальных команд, например «help», «use». Не глобальные команды применяются к коллекциям или базе данных.

Команды, которые используются применительно к текущей базе данных, начинаются с указания объекта базы данных: «db», например «db.help()» или «db.stats()». Команды, которые используются применительно к конкретной коллекции, помимо объекта базы данных используют также имя коллекции: «db.<имя коллекции>».

Синтаксис команд, применимых к коллекции, следующий: «<объект бд>.<коллекция>.<функция> (<список параметров>)». Список параметров задается в формате JSON.

# Ход выполнения работы

**Часть 1**

Запуск сервера: mongod.exe (скриншот на рисунке 1).

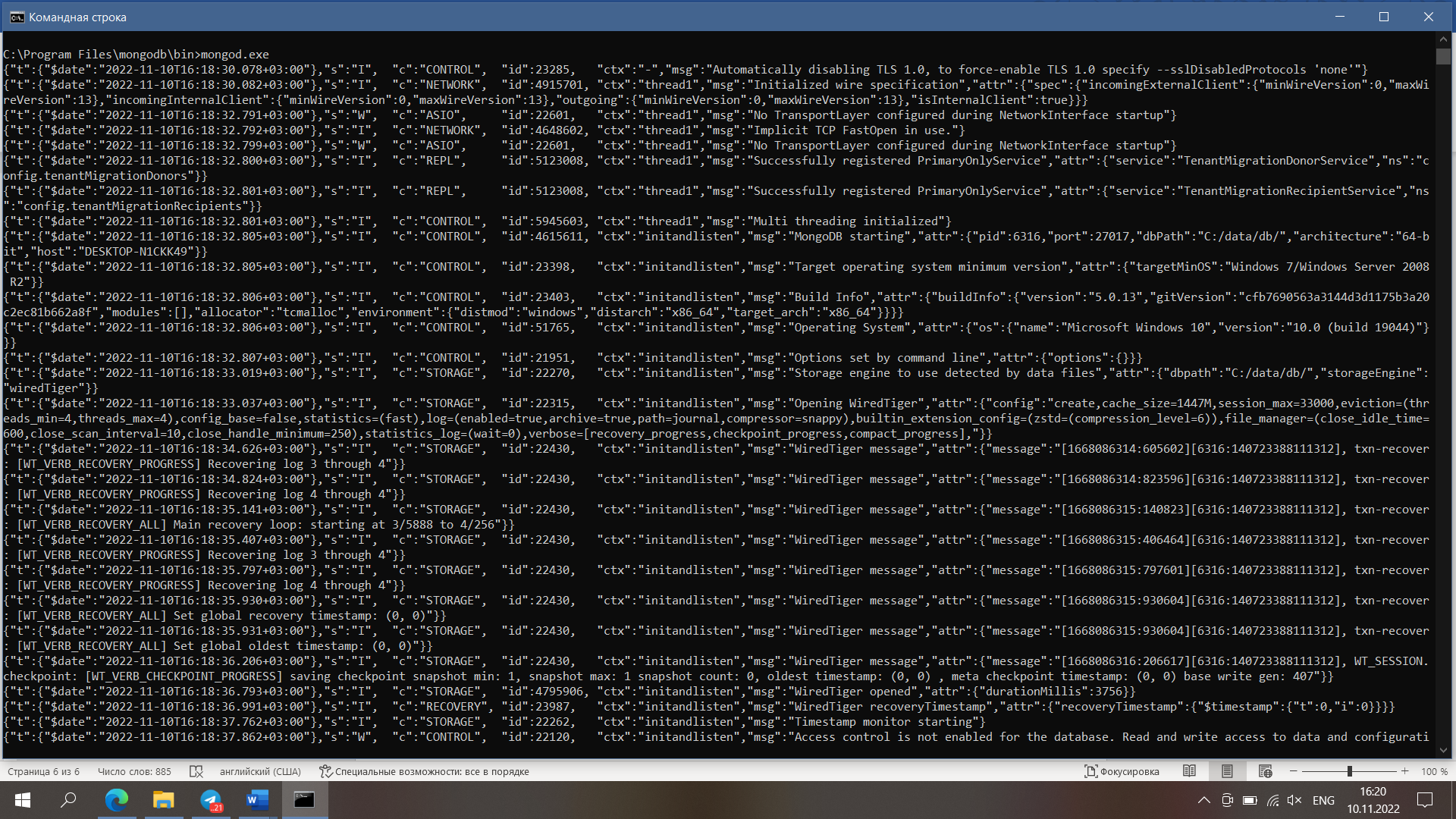


Рисунок 1 – Запуск сервера

Запуск клиента: mongo.exe (скриншот на рисунке 2).

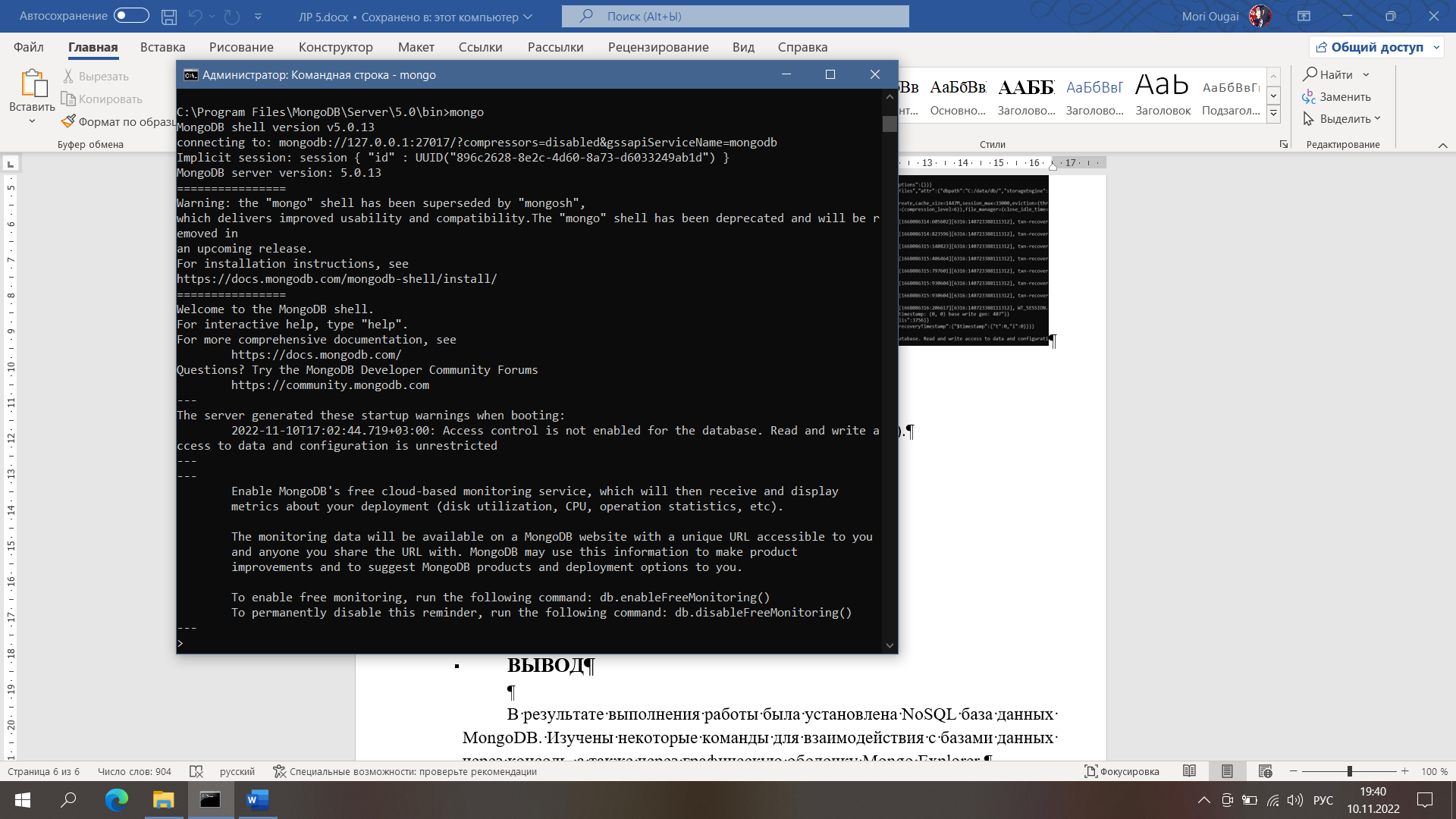


Рисунок 2 – Запуск клиента

Результат выполнения db.help() представлен на рисунке 3.

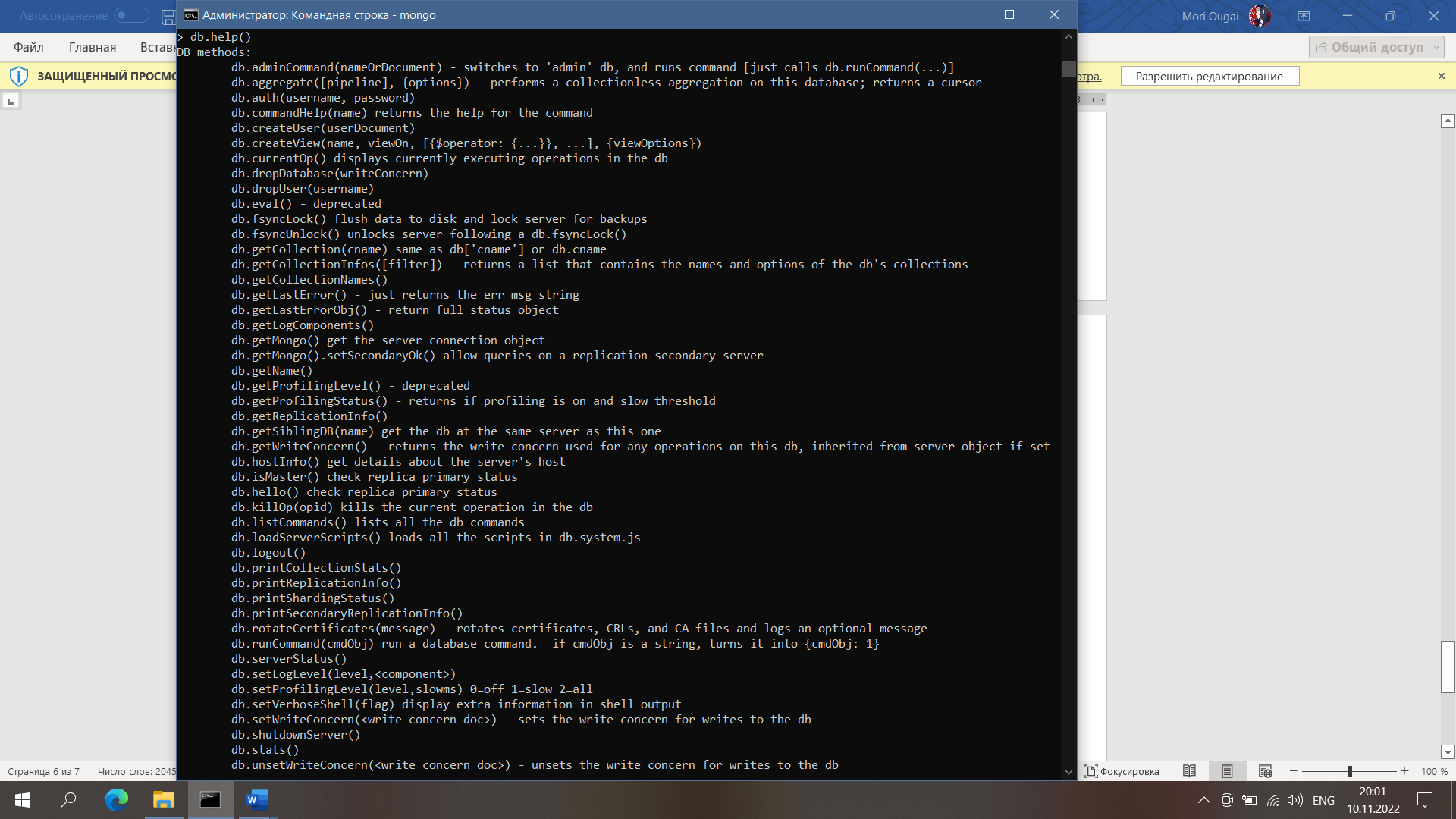


Рисунок 3 – Результат выполнения db.help()

Результат выполнения db.help представлен на рисунке 4.

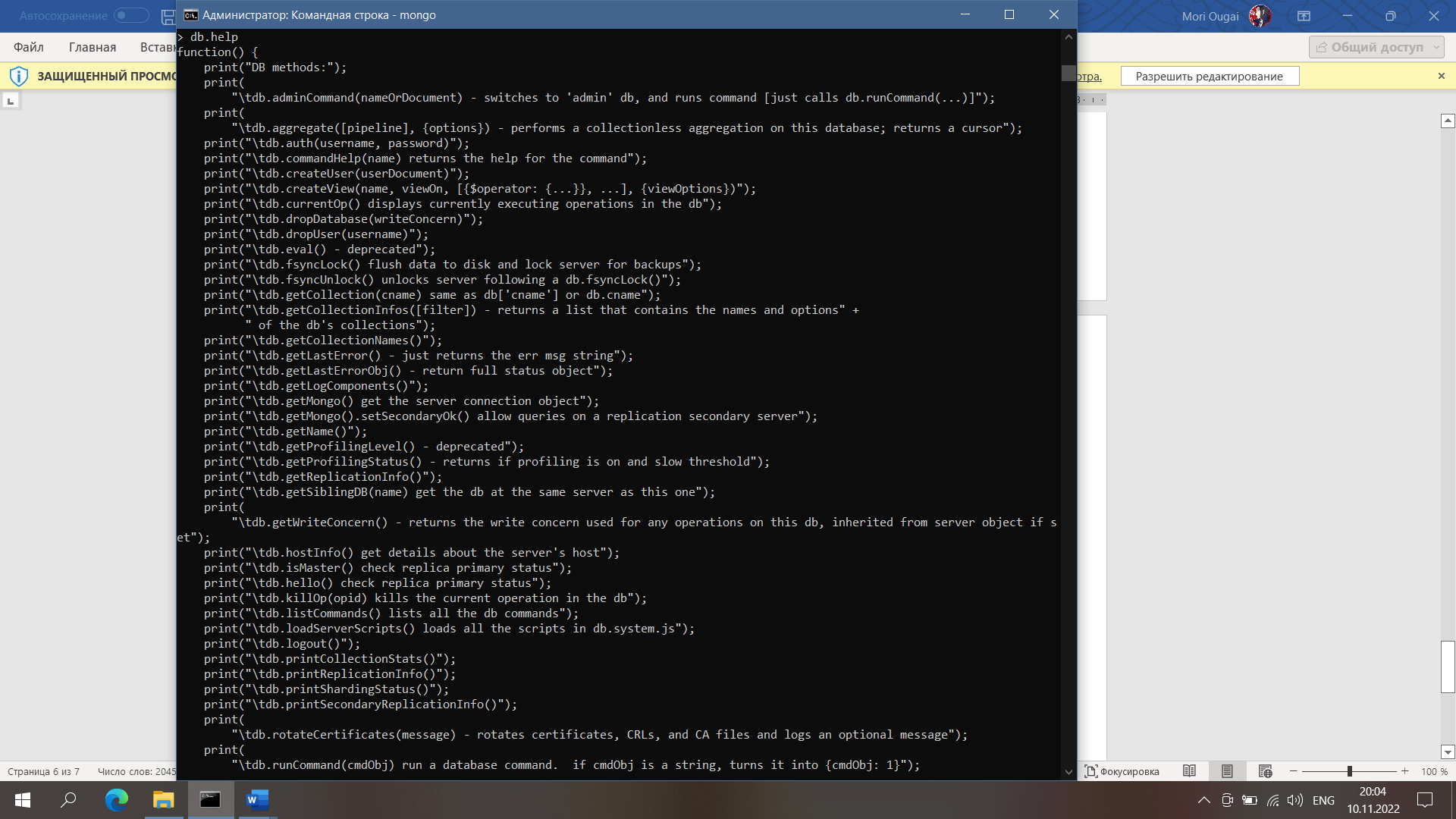


Рисунок 4 – Результат выполнения db.help

Результат выполнения db.stats() представлен на рисунке 5.

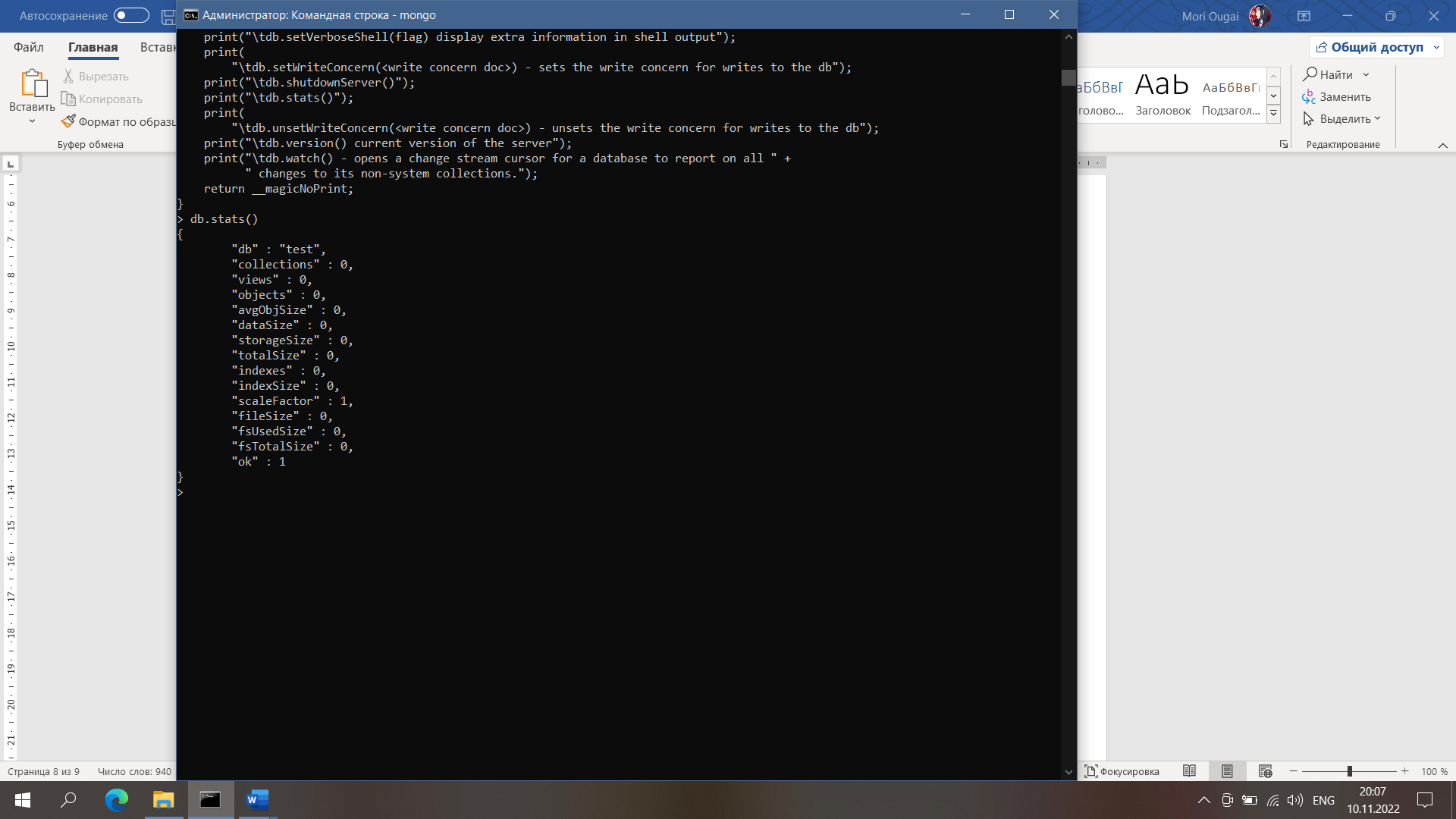


Рисунок 5 – Результат выполнения db.stats()

Создание базы данных learn, результат выполнения представлен на рисунке 6.

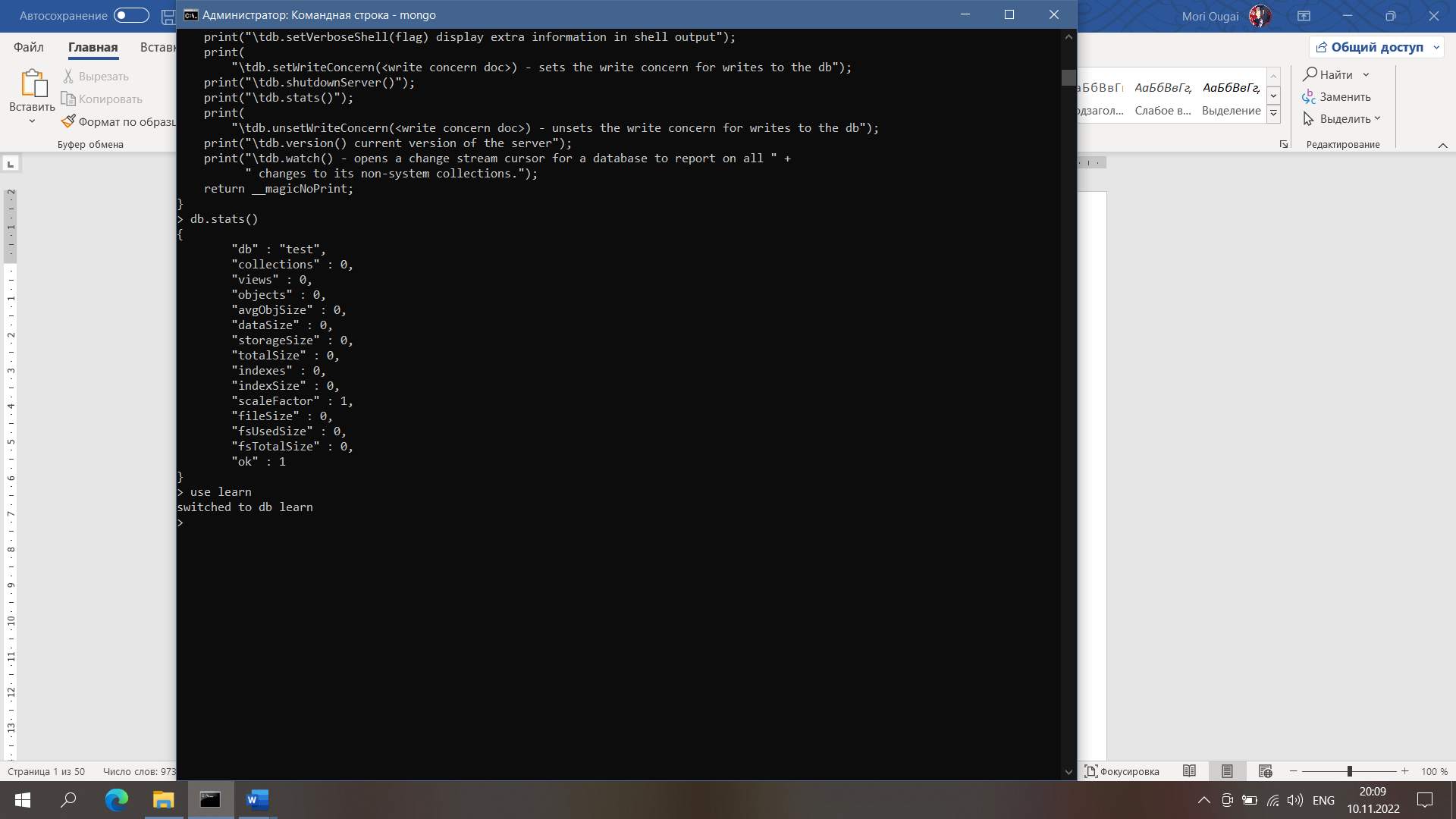


Рисунок 6 – Создание базы данных

Вставка документа в коллекцию unicorns, результат выполнения представлен на рисунке 7.

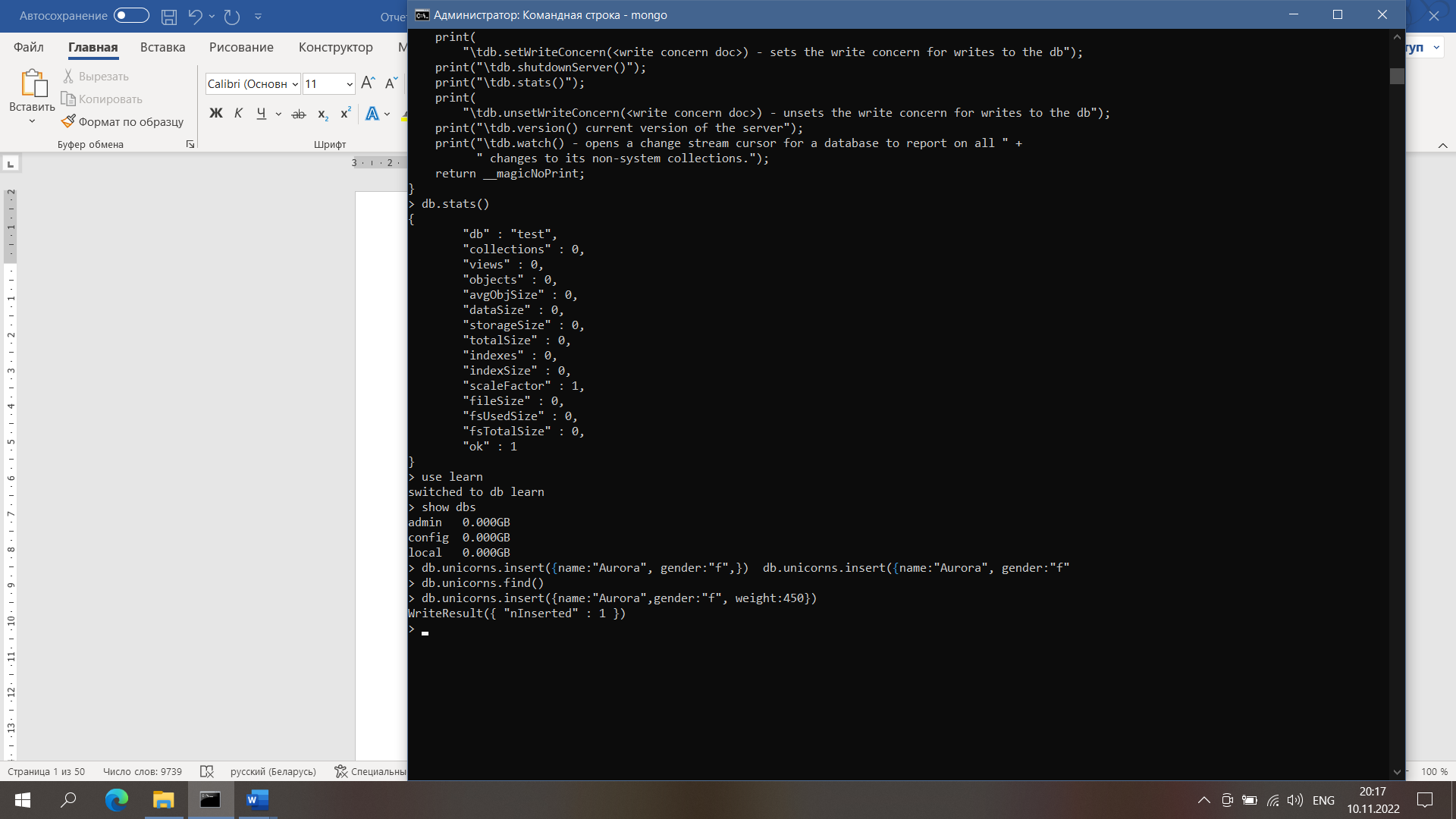


Рисунок 7 – Вставка документа

Список всех БД show dbs представлен на рисунке 8.

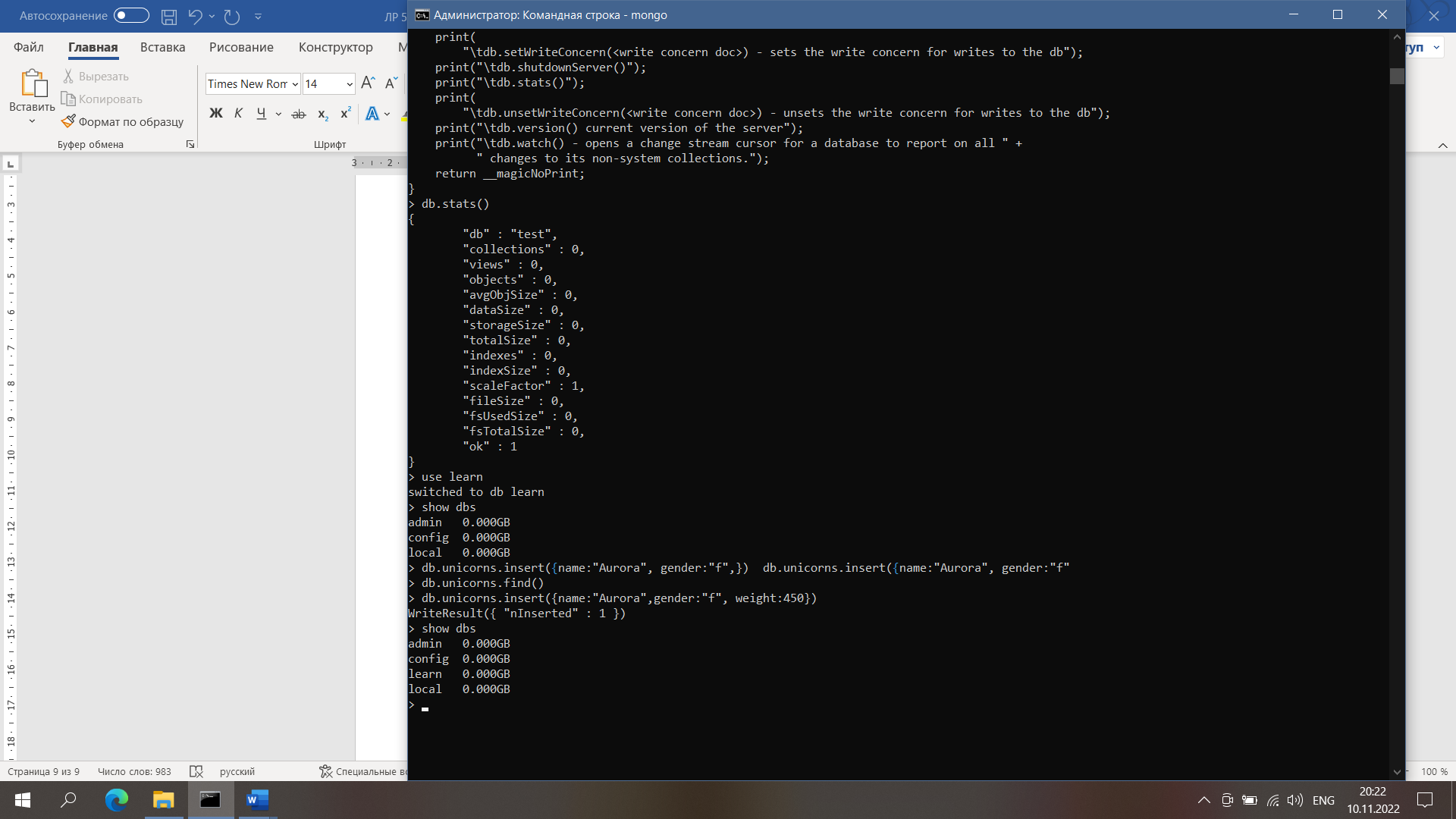


Рисунок 8 – Список БД

Список коллекций: show collections результат выполнения на рисунке 9.

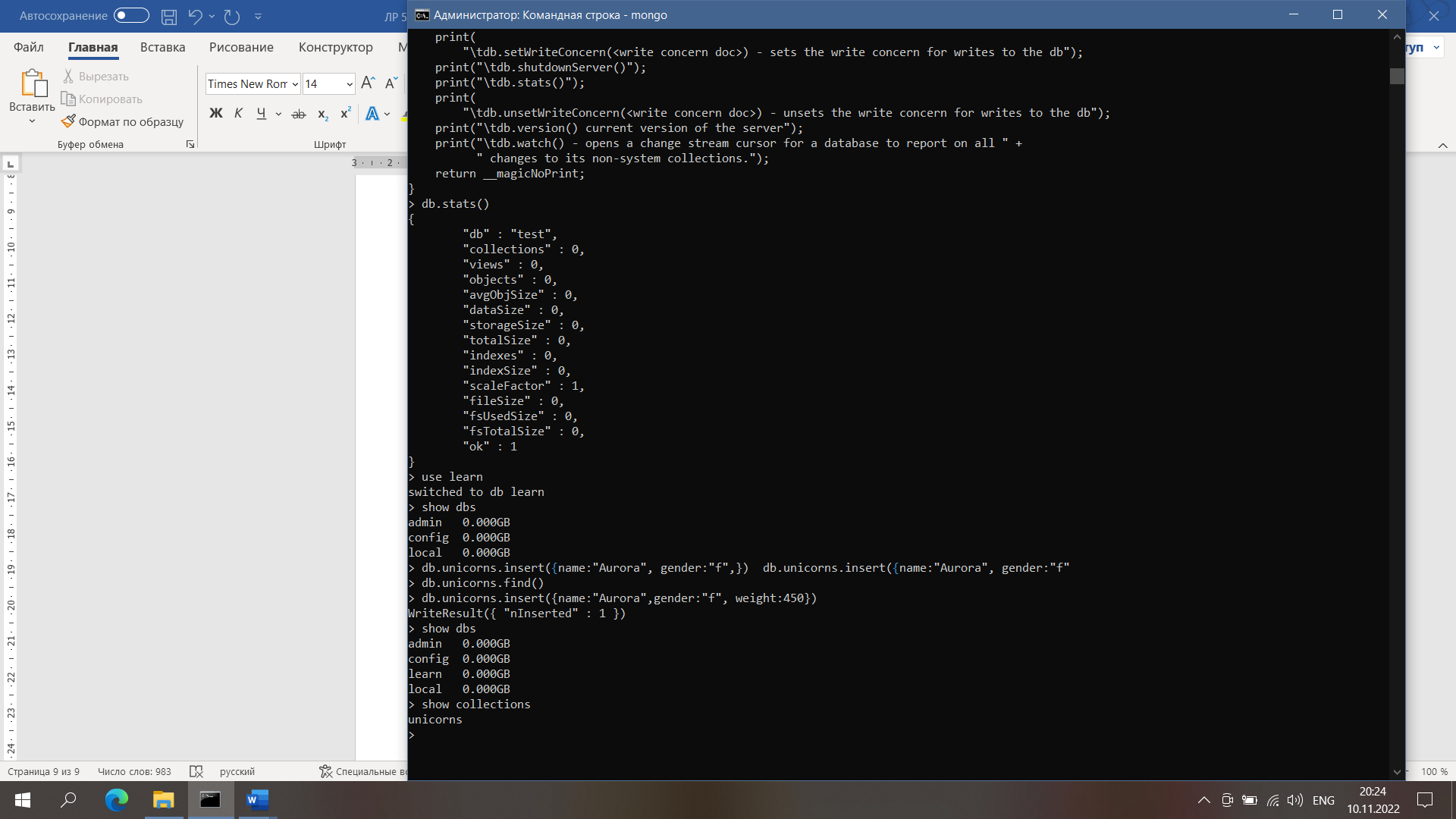


Рисунок 9 – Список коллекций БД

Переименование коллекции db.unicorn.renameCollection(“newName”) результат выполнения на рисунке 10.

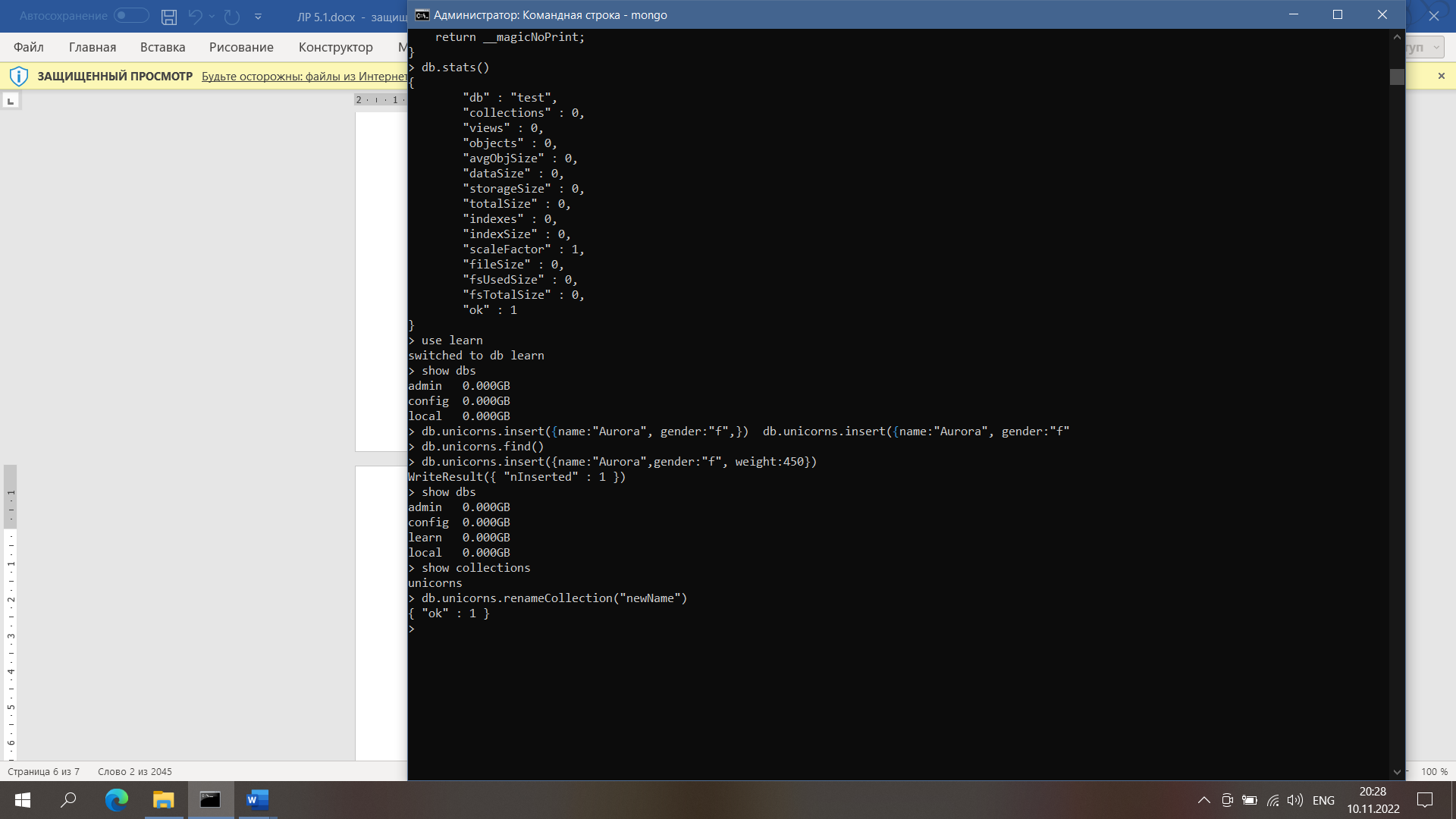


Рисунок 10 – Переименование коллекции

Статистика коллекции db.newName.stats() представлена на рисунке 11.

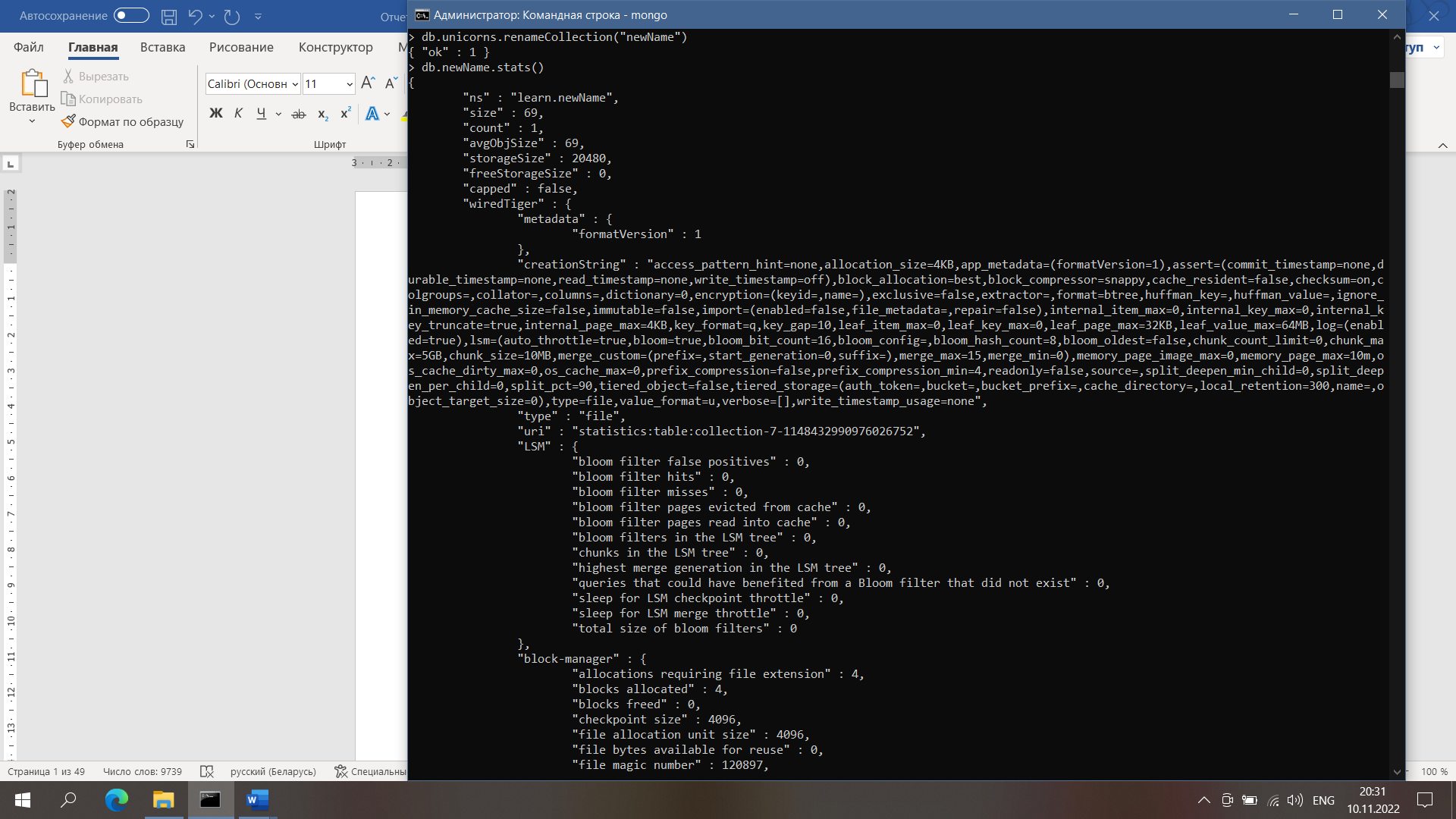


Рисунок 11 – Статистика коллекции

Удаление коллекции db.newPage.drop() результат представлен на рисунке 12.

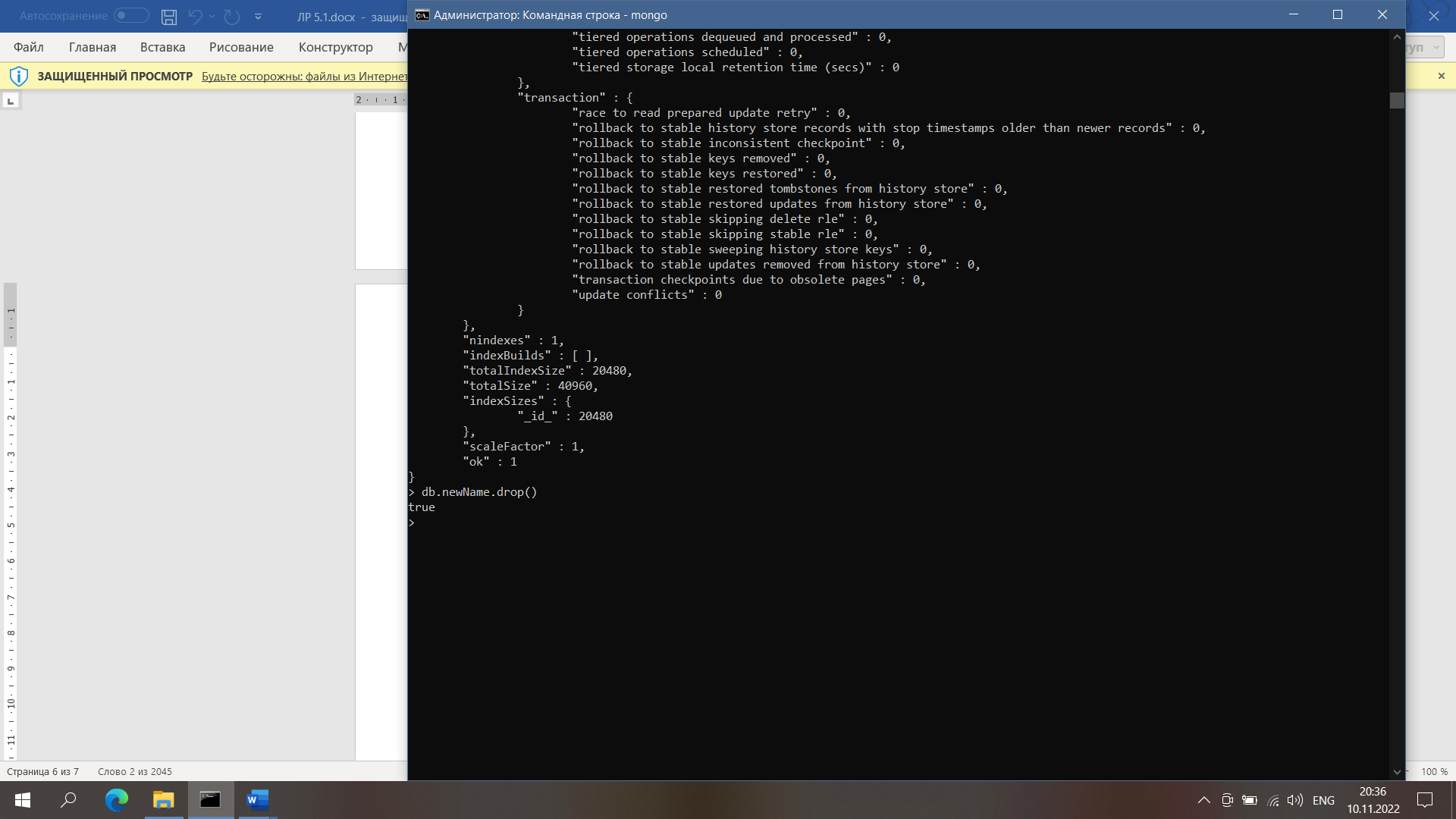


Рисунок 12 – Удаление коллекции

Удаление БД db.dropDatabase() на рисунке 13.

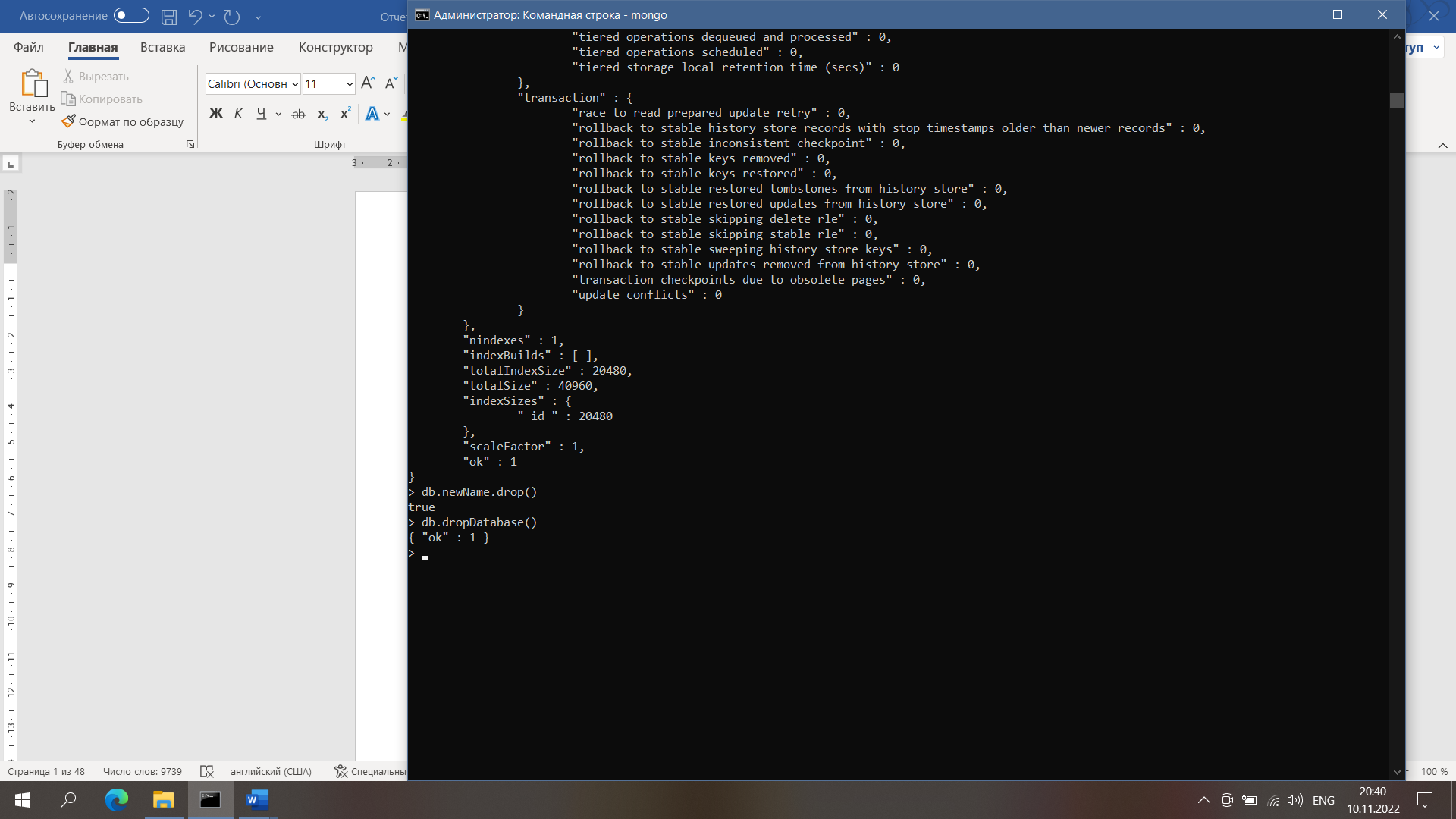


Рисунок 13 – Удаление базы данных

**Часть 2**

**Практическое задание 1.**

Вставка документов в коллекцию. Скрипт представлен ниже, а результат выполнения на рисунке 14.

db.unicorns.insert({name: 'Horny', dob: new Date(1992,2,13,7,47), loves: ['carrot','papaya'], weight: 600, gender: 'm', vampires: 63});

db.unicorns.insert({name: 'Aurora', dob: new Date(1991, 0, 24, 13, 0), loves: ['carrot', 'grape'], weight: 450, gender: 'f', vampires: 43});

db.unicorns.insert({name: 'Unicrom', dob: new Date(1973, 1, 9, 22, 10), loves: ['energon', 'redbull'], weight: 984, gender: 'm', vampires: 182});

db.unicorns.insert({name: 'Roooooodles', dob: new Date(1979, 7, 18, 18, 44), loves: ['apple'], weight: 575, gender: 'm', vampires: 99});

db.unicorns.insert({name: 'Solnara', dob: new Date(1985, 6, 4, 2, 1), loves:['apple', 'carrot', 'chocolate'], weight:550, gender:'f', vampires:80});

db.unicorns.insert({name:'Ayna', dob: new Date(1998, 2, 7, 8, 30), loves: ['strawberry', 'lemon'], weight: 733, gender: 'f', vampires: 40});

db.unicorns.insert({name:'Kenny', dob: new Date(1997, 6, 1, 10, 42), loves: ['grape', 'lemon'], weight: 690, gender: 'm', vampires: 39});

db.unicorns.insert({name: 'Raleigh', dob: new Date(2005, 4, 3, 0, 57), loves: ['apple', 'sugar'], weight: 421, gender: 'm', vampires: 2});

db.unicorns.insert({name: 'Leia', dob: new Date(2001, 9, 8, 14, 53), loves: ['apple', 'watermelon'], weight: 601, gender: 'f', vampires: 33});

db.unicorns.insert({name: 'Pilot', dob: new Date(1997, 2, 1, 5, 3), loves: ['apple', 'watermelon'], weight: 650, gender: 'm', vampires: 54});

db.unicorns.insert({name: 'Nimue', dob: new Date(1999, 11, 20, 16, 15), loves: ['grape', 'carrot'], weight: 540, gender: 'f'});

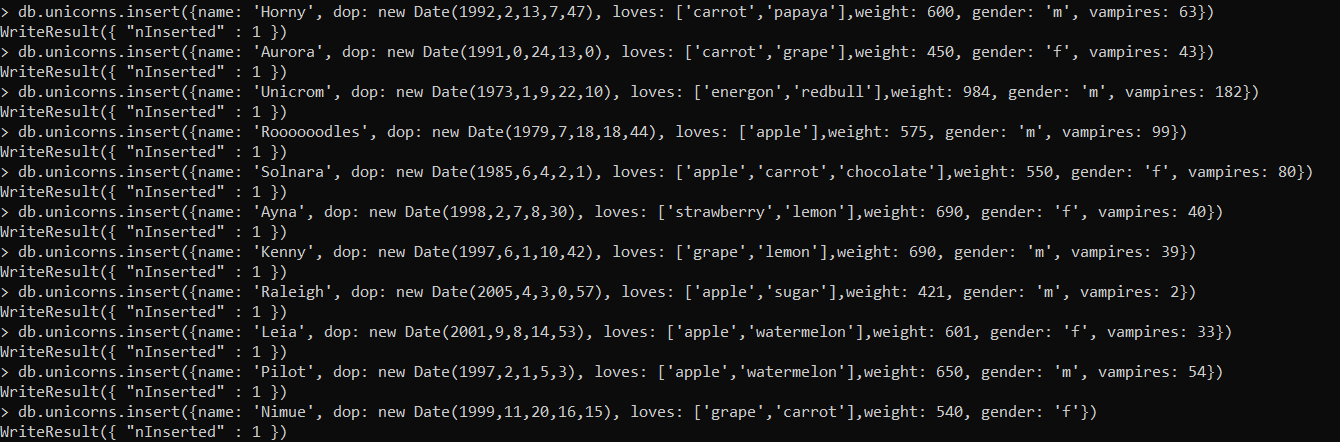


Рисунок 14 – Выполнение вставки

Вставка документов вторым способом представлена на рисунке 15.

document=({name: 'Dunx', dop: new Date(1976, 6, 18, 18, 18), loves: ['grape', 'watermelon'], weight: 704, gender: 'm', vampires: 165})

{

"name" : "Dunx",

"dop" : ISODate("1976-07-18T15:18:00Z"),

"loves" : [

"grape",

"watermelon"

],

"weight" : 704,

"gender" : "m",

"vampires" : 165

}

db.unicorns.insert(document)

WriteResult({ "nInserted" : 1 })

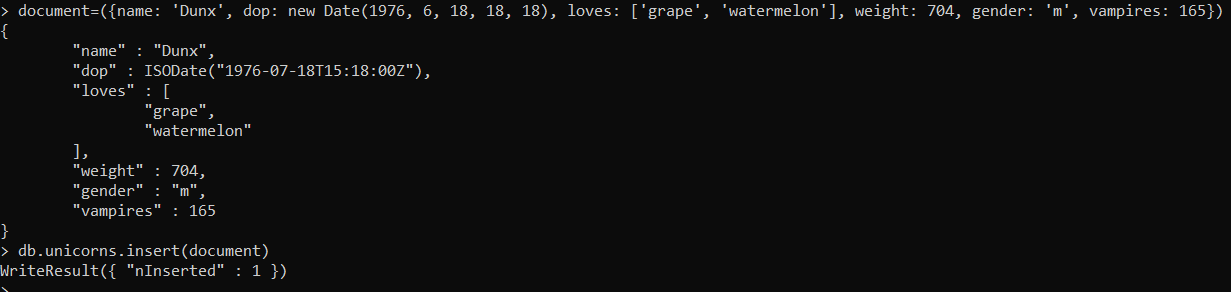


Рисунок 15 – Вставка документа вторым способом

Проверка содержимого коллекции db.unicorns.find(), результат выполнения на рисунке 16.



Рисунок 16 – Выборка

**Практическое задание 2**

Вывод самцов единорогов db.unicorns.find({gender: "m"}). Результат вывода представлен на рисунке 17.

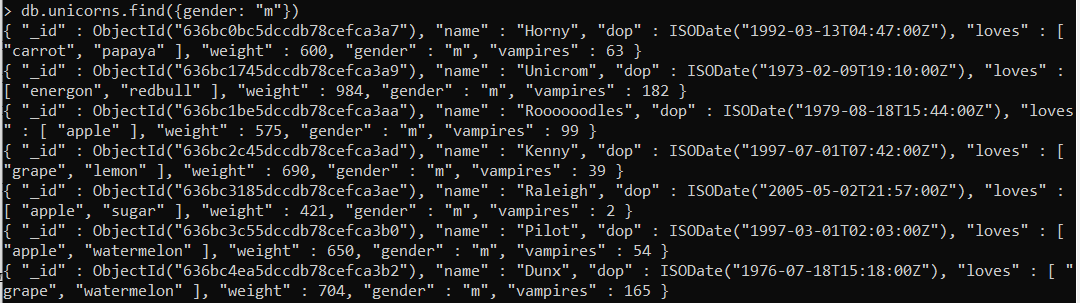


Рисунок 17 – Вывод самцов единорогов

Вывод самок единорогов db.unicorns.find({gender: "f"}). Результат вывода представлен на рисунке 18.

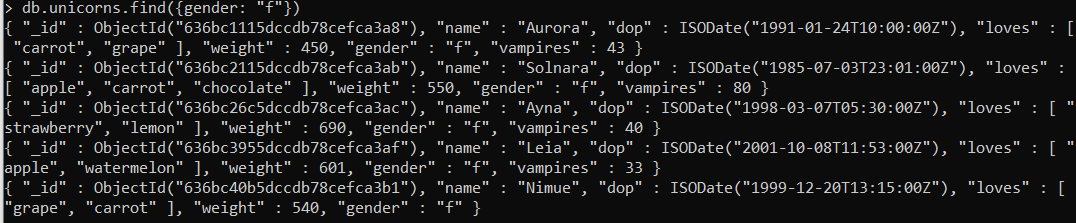


Рисунок 18 – Вывод самок единорогов

Вывод первых трех особей db.unicorns.find({gender: "f"}).limit(3). Результат выполнения представлен на рисунке 19.

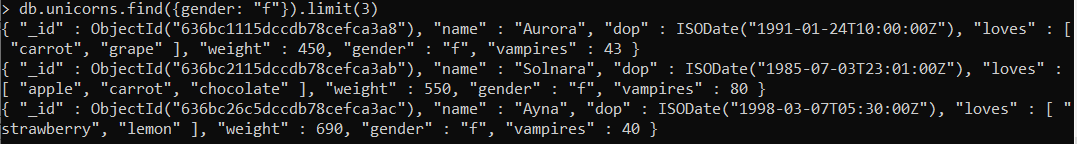


Рисунок 19 – Вывод трех особей

С учетом сортировки по имени db.unicorns.find({gender: "f"}).limit(3).sort({name: 1}). Результат выполнения на рисунке 20.

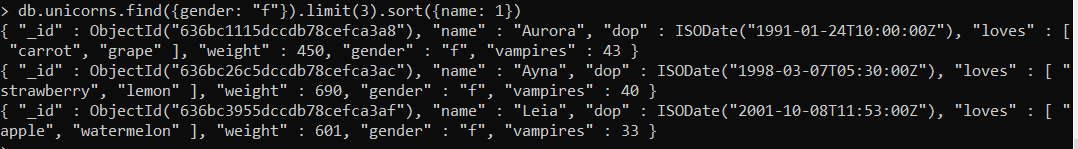


Рисунок 20 – Результат выполнения запроса

Поиск всех самок, которые любят carrot db.unicorns.find({gender: "f", loves: "carrot"}). Результат выполнения на рисунке 21.

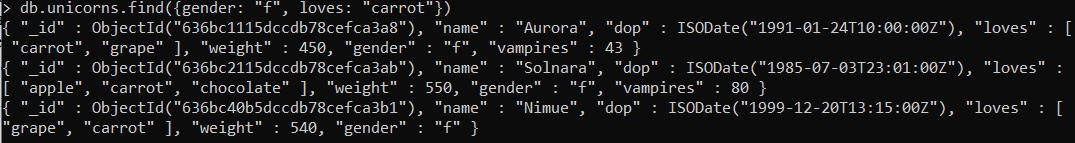


Рисунок 21 – Результат поиска

Поиск одной особи с помощью limit(1): db.unicorns.find({gender: "f", loves: "carrot"}).limit(1). Результат выполнения на рисунке 22.



Рисунок 22 – Ограничение на количество записей limit

Поиск одной особи с помощью findOne(): db.unicorns.findOne({gender: "f", loves: "carrot"}). Результат выполнения на рисунке 23.

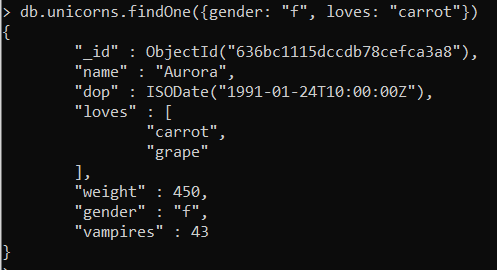


Рисунок 23 – Ограничение на количество записей findOne

**Практическое задание 3.**

Вывод самцов единорогов, исключая сведения о дате рождения и поле: db.unicorns.find({gender: "m"}, {dop: 0, gender: false}). Результат выполнения на рисунке 24.

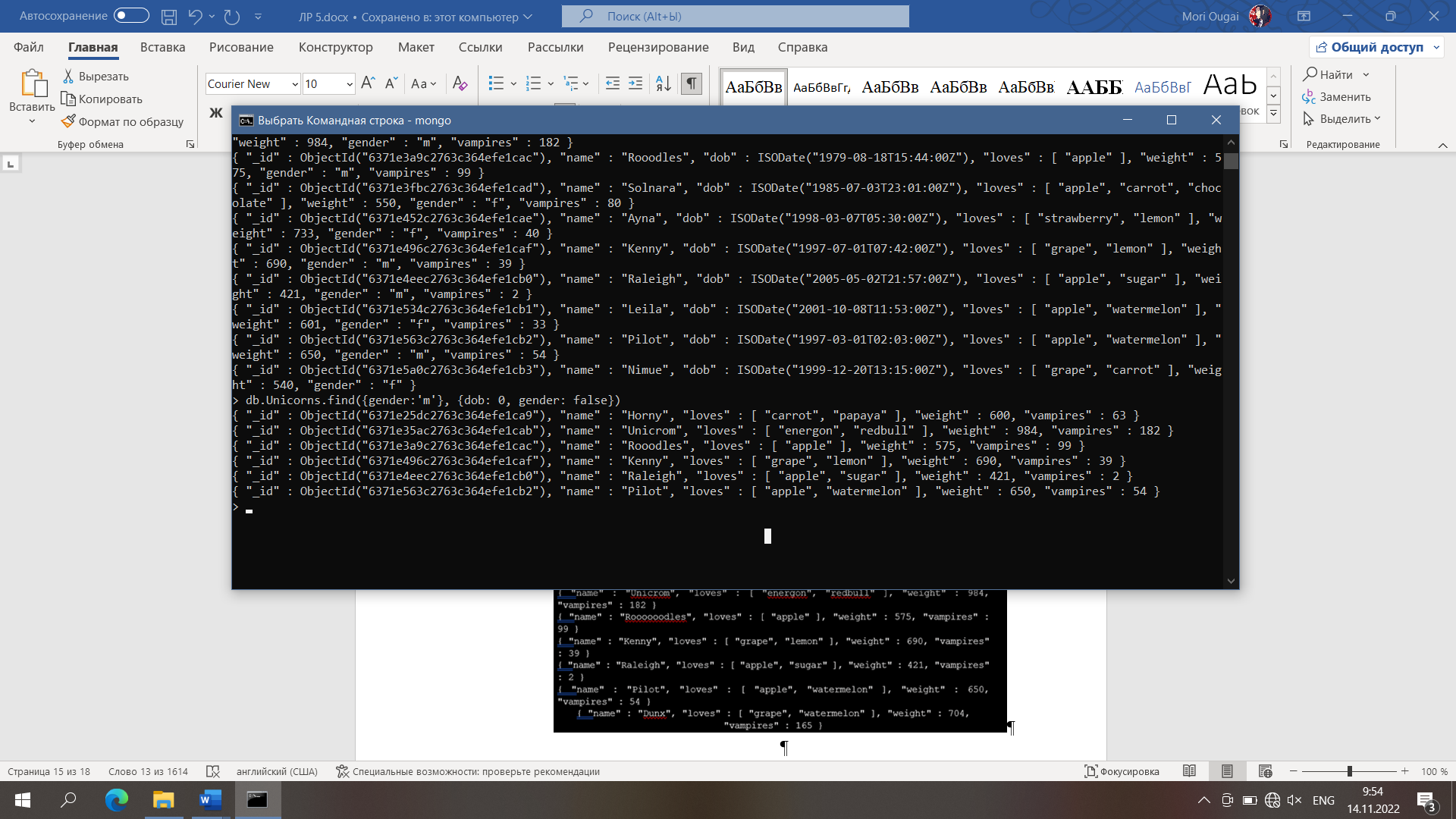


Рисунок 24 – Результат выборки

**Практическое задание 4.**

Вывести записи в обратном порядке добавления: db.unicorns.find().sort({ $natural: -1}). Результат выполнения представлен на рисунке 25.



Рисунок 25 – Результат выполнения

**Практическое задание 5.**

Вывести список единорогов с названия первого любимого фрукта, исключив идентификатор: db.unicorns.find({}, {"loves": {$slice: 1}}, {\_id:false}). Результат на рисунке 26.

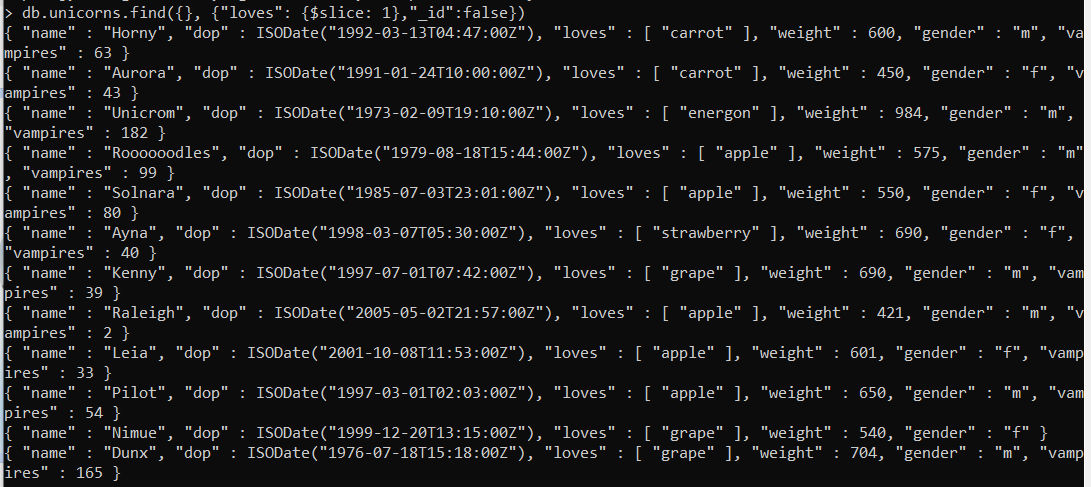


Рисунок 26 – Результат выбора

**Практическое задание 6.**

Вывести список самок единорогов весом от полутонны до 700 кг, исключив вывод идентификатора. db.unicorns.find({weight:{$gt :500, $lt:700}},{gender: "f"} ,{\_id:0}). Результат на рисунке 27.

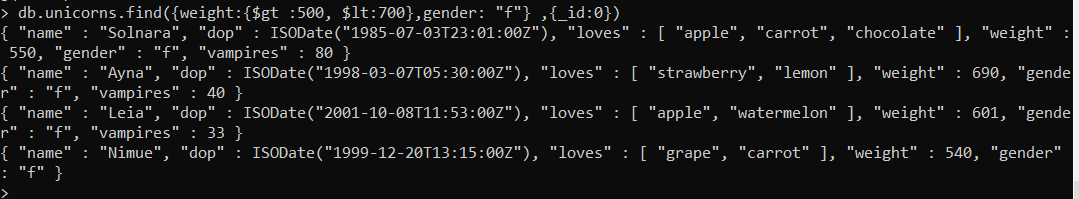


Рисунок 27 – Результат выполнения

**Практическое задание 7.**

Вывести список самцов единорогов весом от полтонны и предпочитающих grape и lemon, исключив вывод идентификатора. db.unicorns.find({weight:{$gt :500},gender: "m", loves:{$all: ["grape","lemon"]}} ,{\_id:0}). Результат выполнения на рисунке 28.

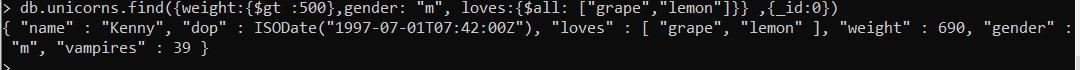


Рисунок 28 – Результат выполнения

**Практическое задание 8.**

Найти всех единорогов, не имеющих ключ vampires. db.unicorns.find({vampires:{$exists: true}}). Результат выполнения приведен на рисунке 29.

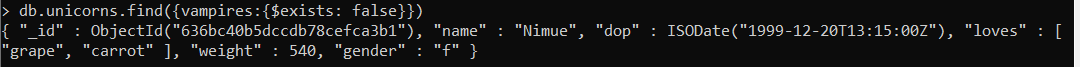


Рисунок 29 – Результат выбора

# Вывод

В результате выполнения работы была установлена NoSQL база данных MongoDB. Изучены некоторые команды для взаимодействия с базами данных через консоль, а также через графическую оболочку Mongo Explorer.