Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

Лабораторная работа №5
Работа с БД в СУБД MongoDB

Выполнил:

студент 2 курса ИКТ группа К32392,
Малышенко А. Р.

Проверила:

Говорова Марина Михайловна

Санкт-Петербург
2023

Цель работы:

Овладеть практическими навыками работы с CRUD-операциями, с вложенными объектами в коллекции базы данных MongoDB, агрегации и изменения данных, со ссылками и индексами в базе данных MongoDB.

Практическое задание:

Задание 8.1.1:

1. *Создайте базу данных learn.*

```
> use learn
switched to db learn
```

2. *Заполните коллекцию единорогов unicorns:*

```
> db.unicorns.insert({name: 'Horny', loves: ['carrot','papaya'], weight: 600, gender: 'm', vampires: 63});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Aurora', loves: ['carrot', 'grape'], weight: 450, gender: 'f', vampires: 43});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Unicrom', loves: ['energon', 'redbull'], weight: 984, gender: 'm', vampires: 182});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Roooooodles', loves: ['apple'], weight: 575, gender: 'm', vampires: 99});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Solnara', loves:['apple', 'carrot', 'chocolate'], weight:550, gender:'f', vampires:80});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name:'Ayna', loves: ['strawberry', 'lemon'], weight: 733, gender: 'f', vampires: 40});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name:'Kenny', loves: ['grape', 'lemon'], weight: 690,  gender: 'm', vampires: 39});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Raleigh', loves: ['apple', 'sugar'], weight: 421, gender: 'm', vampires: 2});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Leia', loves: ['apple', 'watermelon'], weight: 601, gender: 'f', vampires: 33});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Pilot', loves: ['apple', 'watermelon'], weight: 650, gender: 'm', vampires: 54});
WriteResult({ "nInserted" : 1 })
> db.unicorns.insert({name: 'Nimue', loves: ['grape', 'carrot'], weight: 540, gender: 'f'});
WriteResult({ "nInserted" : 1 })
```

3. *Используя второй способ, вставьте в коллекцию единорогов документ:*

```
> document = ({name: 'Dunx', loves: ['grape', 'watermelon'], weight: 704, gender: 'm', vampires: 165})
{
        "name" : "Dunx",
        "loves" : [
                "grape",
                "watermelon"
        ],
        "weight" : 704,
        "gender" : "m",
        "vampires" : 165
}
> db.learn.insert(document)
WriteResult({ "nInserted" : 1 })
```

4. *Проверьте содержимое коллекции с помощью метода find.*

```
> db.unicorns.find();
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "_id" : ObjectId("6477411a7b762700a6e16654"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("6477411a7b762700a6e16655"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 182 }
{ "_id" : ObjectId("6477411a7b762700a6e16656"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "_id" : ObjectId("6477411a7b762700a6e16657"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("6477411a7b762700a6e16658"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "_id" : ObjectId("6477411a7b762700a6e1665a"), "name" : "Raleigh", "loves" : [ "apple", "sugar" ], "weight" : 421, "gender" : "m", "vampires" : 2 }
{ "_id" : ObjectId("6477411a7b762700a6e1665b"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("6477411a7b762700a6e1665c"), "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "_id" : ObjectId("6477411a7b762700a6e1665d"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
```

Задание 8.1.2:

1. *Сформируйте запросы для вывода списков самцов и самок единорогов. Ограничьте список самок первыми тремя особями. Отсортируйте списки по имени.*

   Для самцов:

```
> db.unicorns.find({"gender": "m"}).limit(3).sort({"name": 1});
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
```

   Для самок:

```
> db.unicorns.find({"gender": "f"}).limit(3).sort({"name": 1});
{ "_id" : ObjectId("6477411a7b762700a6e16654"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("6477411a7b762700a6e16658"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "_id" : ObjectId("6477411a7b762700a6e1665b"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
```

2. *Найдите всех самок, которые любят carrot. Ограничьте этот список первой особью с помощью функций findOne и limit.*

```
> db.unicorns.findOne({"gender": "f", "loves": "carrot"})
{
        "_id" : ObjectId("6477411a7b762700a6e16654"),
        "name" : "Aurora",
        "loves" : [
                "carrot",
                "grape"
        ],
        "weight" : 450,
        "gender" : "f",
        "vampires" : 43
}
```

Задание 8.1.3:

Модифицируйте запрос для вывода списков самцов единорогов, исключив из результата информацию о предпочтениях и поле.

```
> db.unicorns.find({"gender": "m"}, {"loves": 0, "gender": 0}).sort({"name": 1})
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "weight" : 704, "vampires" : 165 }
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "weight" : 600, "vampires" : 63 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "weight" : 690, "vampires" : 39 }
{ "_id" : ObjectId("6477411a7b762700a6e1665c"), "name" : "Pilot", "weight" : 650, "vampires" : 54 }
{ "_id" : ObjectId("6477411a7b762700a6e1665a"), "name" : "Raleigh", "weight" : 421, "vampires" : 2 }
{ "_id" : ObjectId("6477411a7b762700a6e16656"), "name" : "Roooooodles", "weight" : 575, "vampires" : 99 }
{ "_id" : ObjectId("6477411a7b762700a6e16655"), "name" : "Unicrom", "weight" : 984, "vampires" : 182 }
```

Задание 8.1.4:

Вывести список единорогов в обратном порядке добавления.

Команда:

```
db.unicorns.find().sort({$natural: -1})
```

```
> db.unicorns.find().sort({$natural: -1})
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
{ "_id" : ObjectId("6477411a7b762700a6e1665d"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("6477411a7b762700a6e1665c"), "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "_id" : ObjectId("6477411a7b762700a6e1665b"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("6477411a7b762700a6e1665a"), "name" : "Raleigh", "loves" : [ "apple", "sugar" ], "weight" : 421, "gender" : "m", "vampires" : 2 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "_id" : ObjectId("6477411a7b762700a6e16658"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "_id" : ObjectId("6477411a7b762700a6e16657"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("6477411a7b762700a6e16656"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "_id" : ObjectId("6477411a7b762700a6e16655"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 182 }
{ "_id" : ObjectId("6477411a7b762700a6e16654"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
```

Задание 8.1.5:

Вывести список единорогов с названием первого любимого предпочтения, исключив идентификатор.

Команда

```
db. unicorns. find({}, {“loves”: {$slice: 1}, “_id”: 0})
```

```
> db.unicorns.find({}, {"loves": {$slice: 1}, "_id": 0})
{ "name" : "Horny", "loves" : [ "carrot" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "name" : "Aurora", "loves" : [ "carrot" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "name" : "Unicrom", "loves" : [ "energon" ], "weight" : 984, "gender" : "m", "vampires" : 182 }
{ "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "name" : "Solnara", "loves" : [ "apple" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "name" : "Ayna", "loves" : [ "strawberry" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "name" : "Kenny", "loves" : [ "grape" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "name" : "Raleigh", "loves" : [ "apple" ], "weight" : 421, "gender" : "m", "vampires" : 2 }
{ "name" : "Leia", "loves" : [ "apple" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "name" : "Pilot", "loves" : [ "apple" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "name" : "Nimue", "loves" : [ "grape" ], "weight" : 540, "gender" : "f" }
{ "name" : "Dunx", "loves" : [ "grape" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
```

Задание 8.1.6:

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

Команда

```
db.unicrons.find({“weight”: {$gte: 500, $lte: 700}},
{“_id”: 0})
```

```
> db.unicorns.find({"weight": {$gte: 500, $lte: 700}}, {"_id": 0})
{ "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
```

Задание 8.1.7:

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

Команда:

```
db.unicorns.find({“gender”: “m”, “loves”: {$in: [“grape”, “lemon”]}, “weight”: {$gte: 500}}, {“_id”: 0})
```

```
> db.unicorns.find({"gender": "m", "loves": {$in: ["grape", "lemon"]}, "weight": {$gte: 500}}, {"_id": 0})
{ "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
```

Задание 8.1.8:

Найти всех единорогов, не имеющих ключ `vampires`.

Команда

```
db.unicorns.find({“vampires”: {$exists: false}})
```

```
> db.unicorns.find({"vampires": {$exists: false}})
{ "_id" : ObjectId("6477411a7b762700a6e1665d"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
```

Задание 8.1.9:

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

Команда:

```
db.unicorns.find({“gender”: “m”}, {“loves”: {$slice: 1}, “_id”: 0, “name”: 1}).sort({“name”: 1})
```

```
> db.unicorns.find({"gender": "m"}, {"loves": {$slice: 1}, "_id": 0, "name": 1}).sort({"name": 1})
{ "name" : "Dunx", "loves" : [ "grape" ] }
{ "name" : "Horny", "loves" : [ "carrot" ] }
{ "name" : "Kenny", "loves" : [ "grape" ] }
{ "name" : "Pilot", "loves" : [ "apple" ] }
{ "name" : "Raleigh", "loves" : [ "apple" ] }
{ "name" : "Roooooodles", "loves" : [ "apple" ] }
{ "name" : "Unicrom", "loves" : [ "energon" ] }
```

Задание 8.2.1:

1. *Создайте коллекцию towns, включающую следующие документы:*

```
> db.town.insert({name: "Punxsutawney ",
... populatiuon: 6200,
... last_sensus: ISODate("2008-01-31"),
... famous_for: [""],
... mayor: {
...     name: "Jim Wehrle"
... }}
... )
```

```
> db.town.insert({name: "New York",
... populatiuon: 22200000,
... last_sensus: ISODate("2009-07-31"),
... famous_for: ["status of liberty", "food"],
... mayor: {
...     name: "Michael Bloomberg",
... party: "I"}}
... )
WriteResult({ "nInserted" : 1 })
> db.town.insert({name: "Portland",
... populatiuon: 528000,
... last_sensus: ISODate("2009-07-20"),
... famous_for: ["beer", "food"],
... mayor: {
...     name: "Sam Adams",
... party: "D"}}
... )
WriteResult({ "nInserted" : 1 })
```

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

   Команда:

   `db.town.find({“mayor.party”: “I”}, {“_id”: 0, “name”: 1, “mayor”: 1})`

```
> db.town.find({"mayor.party": "I"}, {"_id": 0, "name": 1, "mayor": 1})
{ "name" : "New York", "mayor" : { "name" : "Michael Bloomberg", "party" : "I" } }
```

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

```
> db.town.find({"mayor.party": {$exists: 0}}, {"_id": 0, "name": 1, "mayor": 1})
{ "name" : "Punxsutawney ", "mayor" : { "name" : "Jim Wehrle" } }
```

Задание 8.2.2:

1. *Сформировать функцию для вывода списка самцов единорогов.*
2. *Создать курсор для этого списка из первых двух особей с сортировкой в лексикографическом порядке.*

```
> var cursor = db.unicorns.find({"gender": "m"}).sort({"name": 1}).limit(2); null;
null
```

3. *Вывести результат, используя forEach.*

```
> cursor.forEach(printjson)
{
        "_id" : ObjectId("6477427e7b762700a6e16660"),
        "name" : "Dunx",
        "loves" : [
                "grape",
                "watermelon"
        ],
        "weight" : 704,
        "gender" : "m",
        "vampires" : 165
}
{
        "_id" : ObjectId("6477411a7b762700a6e16653"),
        "name" : "Horny",
        "loves" : [
                "carrot",
                "papaya"
        ],
        "weight" : 600,
        "gender" : "m",
        "vampires" : 63
}
```

Задание 8.2.3:

Вывести количество самок единорогов весом от полутонны до 600 кг.

```
> db.unicorns.find({"gender": "f", "weight": {$gte: 500, $lte: 600}}).count()
2
```

Задание 8.2.4:

Вывести список предпочтений.

```
> db.unicorns.distinct("loves")
[
        "apple",
        "carrot",
        "chocolate",
        "energon",
        "grape",
        "lemon",
        "papaya",
        "redbull",
        "strawberry",
        "sugar",
        "watermelon"
]
```

Задание 8.2.5:

Посчитать количество особей единорогов обоих полов.

```
> db.unicorns.aggregate({$group: {_id: "$gender", count:{$sum: 1}}})
{ "_id" : "m", "count" : 7 }
{ "_id" : "f", "count" : 5 }
```

Задание 8.2.6:

1. *Выполнить команду:*

   ```
   > db.unicorns.save({name: 'Barny', loves: ['grape'],
   weight: 340, gender: 'm'})
   ```

```
> db.unicorns.save({name: 'Barny', loves: ['grape'],
... weight: 340, gender: 'm'})
WriteResult({ "nInserted" : 1 })
```

2. *Проверить содержимое коллекции unicorns.*

```
> db.unicorns.find()
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "_id" : ObjectId("6477411a7b762700a6e16654"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("6477411a7b762700a6e16655"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 182 }
{ "_id" : ObjectId("6477411a7b762700a6e16656"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "_id" : ObjectId("6477411a7b762700a6e16657"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("6477411a7b762700a6e16658"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "_id" : ObjectId("6477411a7b762700a6e1665a"), "name" : "Raleigh", "loves" : [ "apple", "sugar" ], "weight" : 421, "gender" : "m", "vampires" : 2 }
{ "_id" : ObjectId("6477411a7b762700a6e1665b"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("6477411a7b762700a6e1665c"), "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "_id" : ObjectId("6477411a7b762700a6e1665d"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
{ "_id" : ObjectId("64775d2b68eacd0b0a5a859e"), "name" : "Barny", "loves" : [ "grape" ], "weight" : 340, "gender" : "m" }
```

Задание 8.2.7:

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

```
> db.unicorns.update({"name": "Ayna", "gender": "f"}, {$set: {"weight": 800, "vampires": 51}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

2. *Проверить содержимое коллекции unicorns.*

```
> db.unicorns.find();
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "_id" : ObjectId("6477411a7b762700a6e16654"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("6477411a7b762700a6e16655"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 182 }
{ "_id" : ObjectId("6477411a7b762700a6e16656"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "_id" : ObjectId("6477411a7b762700a6e16657"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("6477411a7b762700a6e16658"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 800, "gender" : "f", "vampires" : 51 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "_id" : ObjectId("6477411a7b762700a6e1665a"), "name" : "Raleigh", "loves" : [ "apple", "sugar" ], "weight" : 421, "gender" : "m", "vampires" : 2 }
{ "_id" : ObjectId("6477411a7b762700a6e1665b"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("6477411a7b762700a6e1665c"), "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "_id" : ObjectId("6477411a7b762700a6e1665d"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
{ "_id" : ObjectId("64775d2b68eacd0b0a5a859e"), "name" : "Barny", "loves" : [ "grape" ], "weight" : 340, "gender" : "m" }
```

Задание 8.2.8:

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

```
> db.unicorns.update({"name": "Raleigh", "gender": "m"}, {$addToSet: {loves: "redbull"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

2. *Проверить содержимое коллекции unicorns.*

```
> db.unicorns.find()
{ "_id" : ObjectId("6477411a7b762700a6e16653"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 63 }
{ "_id" : ObjectId("6477411a7b762700a6e16654"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("6477411a7b762700a6e16655"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 182 }
{ "_id" : ObjectId("6477411a7b762700a6e16656"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 99 }
{ "_id" : ObjectId("6477411a7b762700a6e16657"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("6477411a7b762700a6e16658"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 800, "gender" : "f", "vampires" : 51 }
{ "_id" : ObjectId("6477411a7b762700a6e16659"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 39 }
{ "_id" : ObjectId("6477411a7b762700a6e1665b"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("6477411a7b762700a6e1665c"), "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 54 }
{ "_id" : ObjectId("6477411a7b762700a6e1665d"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("6477427e7b762700a6e16660"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 165 }
{ "_id" : ObjectId("64775d2b68eacd0b0a5a859e"), "name" : "Barny", "loves" : [ "grape" ], "weight" : 340, "gender" : "m" }
{ "_id" : ObjectId("647763aa68eacd0b0a5a859f"), "name" : "Raleigh", "loves" : [ "apple", "sugar", "redbull" ], "weight" : 421, "gender" : "m", "vampires" : 2 }
```

Задание 8.2.9:

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

```
> db.unicorns.updateMany({"gender": "m"}, {$inc: {"vampires": 5}})
{ "acknowledged" : true, "matchedCount" : 7, "modifiedCount" : 7 }
```

2. *Проверить содержимое коллекции unicorns.*

```
> db.unicorns.find()
{ "_id" : ObjectId("647765647bf0e7a923c7939a"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 73 }
{ "_id" : ObjectId("647765647bf0e7a923c7939b"), "name" : "Aurora", "loves" : [ "carrot", "grape" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("647765647bf0e7a923c7939c"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 192 }
{ "_id" : ObjectId("647765647bf0e7a923c7939d"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 109 }
{ "_id" : ObjectId("647765647bf0e7a923c7939e"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80 }
{ "_id" : ObjectId("647765647bf0e7a923c7939f"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "_id" : ObjectId("647765647bf0e7a923c793a0"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 49 }
{ "_id" : ObjectId("647765647bf0e7a923c793a1"), "name" : "Raleigh", "loves" : [ "apple", "sugar" ], "weight" : 421, "gender" : "m", "vampires" : 12 }
{ "_id" : ObjectId("647765647bf0e7a923c793a2"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("647765647bf0e7a923c793a3"), "name" : "Pilot", "loves" : [ "apple", "watermelon" ], "weight" : 650, "gender" : "m", "vampires" : 64 }
{ "_id" : ObjectId("647765647bf0e7a923c793a4"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("647765e07bf0e7a923c793a5"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 170 }
```

Задание 8.2.10:

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

```
> db.town.update({"name": "Portland"}, {$unset: {"mayor.party": 1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

2. *Проверить содержимое коллекции towns.*

```
> db.town.find()
{ "_id" : ObjectId("6477550568eacd0b0a5a859b"), "name" : "Punxsutawney ", "populatiuon" : 6200, "last_sensus" : ISODate("2008-01-31T00:00:00Z"), "famous_for" : [ "" ], "mayor" : { "name" : "Jim Wehrle" } }
{ "_id" : ObjectId("6477555a68eacd0b0a5a859c"), "name" : "New York", "populatiuon" : 22200000, "last_sensus" : ISODate("2009-07-31T00:00:00Z"), "famous_for" : [ "status of liberty", "food" ], "mayor" : { "name"
: "Michael Bloomberg", "party" : "I" } }
{ "_id" : ObjectId("6477556d68eacd0b0a5a859d"), "name" : "Portland", "populatiuon" : 528000, "last_sensus" : ISODate("2009-07-20T00:00:00Z"), "famous_for" : [ "beer", "food" ], "mayor" : { "name" : "Sam Adams" }
}
```

Задание 8.2.11:

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

```
> db.unicorns.update({"gender": "m", "name": "Pilot"}, {$addToSet: {loves: "chocolate"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

2. *Проверить содержимое коллекции unicorns.*

```
> db.unicorns.find({"name": "Pilot"})
{ "_id" : ObjectId("647765647bf0e7a923c793a3"), "name" : "Pilot", "loves" : [ "apple", "watermelon", "chocolate" ], "weight" : 650, "gender" : "m", "vampires" : 64 }
```

Задание 8.2.12:

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

   Команда:

   ```
   Db.unicorns.update({“gender”: “f”, “name”: “Aurora”}, {$addToSet: {“loves”: {$each: [“sugar”, “lemon”]}}})
   ```

```
> db.unicorns.update({"gender": "f", "name": "Aurora"}, {$addToSet: {"loves": {$each: ["sugar", "lemon"]}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

2. *Проверить содержимое коллекции unicorns.*

```
> db.unicorns.find({"name": "Aurora"})
{ "_id" : ObjectId("647765647bf0e7a923c7939b"), "name" : "Aurora", "loves" : [ "carrot", "grape", "sugar", "lemon" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
```

Задание 8.2.13:

1. *Создайте коллекцию towns, включающую следующие документы:*

```
> db.towns.insert({name: "Punxsutawney ",
... popujatiuon: 6200,
... last_sensus: ISODate("2008-01-31"),
... famous_for: ["phil the groundhog"],
... mayor: {
...     name: "Jim Wehrle"
... }}
... )
WriteResult({ "nInserted" : 1 })
> db.towns.insert({name: "New York",
... popujatiuon: 22200000,
... last_sensus: ISODate("2009-07-31"),
... famous_for: ["status of liberty", "food"],
... mayor: {
...     name: "Michael Bloomberg",
... party: "I"}}
... )
WriteResult({ "nInserted" : 1 })
> db.towns.insert({name: "Portland",
... popujatiuon: 528000,
... last_sensus: ISODate("2009-07-20"),
... famous_for: ["beer", "food"],
... mayor: {
...     name: "Sam Adams",
... party: "D"}}
... )
WriteResult({ "nInserted" : 1 })
```

2. *Удалите документы с беспартийными мэрами*

```
> db.towns.deleteMany({"mayor.party": {$exists: 0}})
{ "acknowledged" : true, "deletedCount" : 1 }
```

3. *Проверьте содержание коллекции.*

```
> db.towns.find()
{ "_id" : ObjectId("64776bde7bf0e7a923c793a7"), "name" : "New York", "popujatiuon" : 22200000, "last_sensus" : ISODate("2009-07-31T00:00:00Z"), "famous_for" : [ "status of liberty", "food" ], "mayor" : { "name"
: "Michael Bloomberg", "party" : "I" } }
{ "_id" : ObjectId("64776beb7bf0e7a923c793a8"), "name" : "Portland", "popujatiuon" : 528000, "last_sensus" : ISODate("2009-07-20T00:00:00Z"), "famous_for" : [ "beer", "food" ], "mayor" : { "name" : "Sam Adams",
"party" : "D" } }
```

4. *Очистите коллекцию.*

```
> db.towns.deleteMany({})
{ "acknowledged" : true, "deletedCount" : 2 }
```

5. *Просмотрите список доступных коллекций.*

```
> show collections
towns
unicorns
```

p.s. если удалить коллекцию через `db.towns.drop()`*, тогда:*

```
> show collections
unicorns
```

Задание 8.3.1:

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

```
> db.habitatAreas.insert({"_id": "DT", "name": "Desert", "description": "Бескрайние пустыни с жарким климатом"})
WriteResult({ "nInserted" : 1 })
> db.habitatAreas.insert({"_id": "GA", "name": "Green area", "description": "Зеленые леса, зеленые луга, зеленая вода, короче все зеленое"})
WriteResult({ "nInserted" : 1 })
> db.habitatAreas.insert({"_id": "CE", "name": "Cave", "description": "Просто пещера с пауками"})
WriteResult({ "nInserted" : 1 })
> db.habitatAreas.insert({"_id": "LT", "name": "Last Town", "description": "Последний город с выжившими людьми"})
WriteResult({ "nInserted" : 1 })
```

2. *Включите для нескольких единорогов в документы ссылку на зону обитания, использую второй способ автоматического связывания.*

```
> db.unicorns.update({"name": "Pilot"}, {$set: {habitatAreas: {$ref: "habitatAreas", $id: "CE"}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.unicorns.update({"name": "Unicrom"}, {$set: {habitatAreas: {$ref: "habitatAreas", $id: "GA"}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.unicorns.update({"name": "Kenny"}, {$set: {habitatAreas: {$ref: "habitatAreas", $id: "LT"}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.unicorns.update({"name": "Solnara"}, {$set: {habitatAreas: {$ref: "habitatAreas", $id: "DT"}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
```

3. *Проверьте содержание коллекции единорогов.*

```
> db.unicorns.find()
{ "_id" : ObjectId("647765647bf0e7a923c7939a"), "name" : "Horny", "loves" : [ "carrot", "papaya" ], "weight" : 600, "gender" : "m", "vampires" : 73 }
{ "_id" : ObjectId("647765647bf0e7a923c7939b"), "name" : "Aurora", "loves" : [ "carrot", "grape", "sugar", "lemon" ], "weight" : 450, "gender" : "f", "vampires" : 43 }
{ "_id" : ObjectId("647765647bf0e7a923c7939c"), "name" : "Unicrom", "loves" : [ "energon", "redbull" ], "weight" : 984, "gender" : "m", "vampires" : 192, "habitatAreas" : DBRef("habitatAreas", "GA") }
{ "_id" : ObjectId("647765647bf0e7a923c7939d"), "name" : "Roooooodles", "loves" : [ "apple" ], "weight" : 575, "gender" : "m", "vampires" : 109 }
{ "_id" : ObjectId("647765647bf0e7a923c7939e"), "name" : "Solnara", "loves" : [ "apple", "carrot", "chocolate" ], "weight" : 550, "gender" : "f", "vampires" : 80, "habitatAreas" : DBRef("habitatAreas", "DT") }
{ "_id" : ObjectId("647765647bf0e7a923c7939f"), "name" : "Ayna", "loves" : [ "strawberry", "lemon" ], "weight" : 733, "gender" : "f", "vampires" : 40 }
{ "_id" : ObjectId("647765647bf0e7a923c793a0"), "name" : "Kenny", "loves" : [ "grape", "lemon" ], "weight" : 690, "gender" : "m", "vampires" : 49, "habitatAreas" : DBRef("habitatAreas", "LT") }
{ "_id" : ObjectId("647765647bf0e7a923c793a1"), "name" : "Raleigh", "loves" : [ "apple", "sugar" ], "weight" : 421, "gender" : "m", "vampires" : 12 }
{ "_id" : ObjectId("647765647bf0e7a923c793a2"), "name" : "Leia", "loves" : [ "apple", "watermelon" ], "weight" : 601, "gender" : "f", "vampires" : 33 }
{ "_id" : ObjectId("647765647bf0e7a923c793a3"), "name" : "Pilot", "loves" : [ "apple", "watermelon", "chocolate", "sugar", "lemon" ], "weight" : 650, "gender" : "m", "vampires" : 64, "habitatAreas" : DBRef("habi
tatAreas", "CE") }
{ "_id" : ObjectId("647765647bf0e7a923c793a4"), "name" : "Nimue", "loves" : [ "grape", "carrot" ], "weight" : 540, "gender" : "f" }
{ "_id" : ObjectId("647765e07bf0e7a923c793a5"), "name" : "Dunx", "loves" : [ "grape", "watermelon" ], "weight" : 704, "gender" : "m", "vampires" : 170 }
```

Задание 8.3.2:

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

```
learn> db.unicorns.ensureIndex({"name": 1}, {"unique": true})
[ 'name_1' ]
```

p.s. индекс для ключа name успешно создался.

Задание 8.3.3:

1. *Получите информацию обо всех индексах коллекции unicorns.*

```
learn> db.unicorns.getIndexes()
[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { name: 1 }, name: 'name_1', unique: true }
]
```

2. *Удалите все индексы, кроме индекса для идентификатора.*

```
learn> db.unicorns.dropIndex("name_1")
{ nIndexesWas: 2, ok: 1 }
```

3. *Попытайтесь удалить индекс для идентификатора.*

```
learn> db.unicorns.dropIndex("_id_")
MongoServerError: cannot drop _id index
```

Задание 8.3.4:

1. *Создайте объемную коллекцию numbers, задействовав курсор:*

```
learn> for(i = 0; i < 100000; i++){db.numbers.insert({value: i})}
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.

{
  acknowledged: true,
  insertedIds: { '0': ObjectId("64777c590292ad9ca907bf3f") }
}
```

2. *Выберите последних четыре документа.*

```
learn> db.numbers.find({}).skip(99996)
[
  { _id: ObjectId("64777c590292ad9ca907bf3c"), value: 99996 },
  { _id: ObjectId("64777c590292ad9ca907bf3d"), value: 99997 },
  { _id: ObjectId("64777c590292ad9ca907bf3e"), value: 99998 },
  { _id: ObjectId("64777c590292ad9ca907bf3f"), value: 99999 }
]
```

3. *Проанализируйте план выполнения запроса 2. Сколько потребовалось времени на выполнение запроса? (по значению параметра executionTimeMillis)*

```
learn> db.numbers.explain("executionStats").find({}).skip(99996)
```

```
executionStats: {
  executionSuccess: true,
  nReturned: 4,
  executionTimeMillis: 27,
```

4. *Создайте индекс для ключа value.*

```
learn> db.numbers.ensureIndex({"value": 1}, {"unique": true})
[ 'value_1' ]
```

5. *Получите информацию обо всех индексах коллекции numbres.*

```
learn> db.numbers.getIndexes()
[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { value: 1 }, name: 'value_1', unique: true }
]
```

6. *Выполните запрос 2.*
7. *Проанализируйте план выполнения запроса с установленным индексом. Сколько потребовалось времени на выполнение запроса?*

```
learn> db.numbers.explain("executionStats").find({}).skip(99996)
```

```
executionStats: {
  executionSuccess: true,
  nReturned: 4,
  executionTimeMillis: 23,
```

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

   В целом скорость выполнения запроса с индексом и без в среднем отличается на 1–3 мс. Можно сделать вывод, что индексы не значительно ускоряют выполнение запроса для больших баз данных

Выводы:

В ходе лабораторной работы я овладел навыками работы с СУБД MongoDB. В том числе создание баз данных, коллекций для них, работа с запросами и индексами.