

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИТМО»**

Отчет

по лабораторной работе «Работа с БД в СУБД MongoDB»
по дисциплине **«Проектирование и реализация баз данных»**

Автор: Пронина Мария Владимировна

Факультет: ИКТ

Группа: K32392

Преподаватель: Говорова М. М.

Дата: 31.05.2023



Санкт-Петербург 2023

Цель: 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.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД MongoDB 4+, 6.0.6 (текущая).

Выполнение:

8.1 CRUD-ОПЕРАЦИИ В СУБД MONGODB. ВСТАВКА ДАННЫХ. ВЫБОРКА ДАННЫХ

Практическое задание 8.1.1:

- 1) *Создайте базу данных learn.*
- 2) *Заполните коллекцию единорогов unicorns*

```
learn> db.unicorns.insert({name: 'Horny', loves: ['carrot', 'papaya'], weight: 600, gender: 'm', vampires: 63});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530131") }
}
learn> db.unicorns.insert({name: 'Aurora', loves: ['carrot', 'grape'], weight: 450, gender: 'f', vampires: 43});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530132") }
}
learn> db.unicorns.insert({name: 'Unicrom', loves: ['energon', 'redbull'], weight: 984, gender: 'm', vampires: 182});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530133") }
}
learn> db.unicorns.insert({name: 'Rooooooodles', loves: ['apple'], weight: 575, gender: 'm', vampires: 99});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530134") }
}
learn> _
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530135") }
}
learn> db.unicorns.insert({name: 'Ayna', loves: ['strawberry', 'lemon'], weight: 733, gender: 'f', vampires: 40});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530136") }
}
learn> db.unicorns.insert({name: 'Kenny', loves: ['grape', 'lemon'], weight: 690, gender: 'm', vampires: 39});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530137") }
}
learn> db.unicorns.insert({name: 'Raleigh', loves: ['apple', 'sugar'], weight: 421, gender: 'm', vampires: 2});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530138") }
}
learn> db.unicorns.insert({name: 'Leia', loves: ['apple', 'watermelon'], weight: 601, gender: 'f', vampires: 33});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af95530139") }
}
learn> db.unicorns.insert({name: 'Pilot', loves: ['apple', 'watermelon'], weight: 650, gender: 'm', vampires: 54});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af9553013a") }
}
learn> db.unicorns.insert({name: 'Nimue', loves: ['grape', 'carrot'], weight: 540, gender: 'f'});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6478798953c960af9553013b") }
}
```

- 3) *Используя второй способ, вставьте в коллекцию единорогов документ*

```
learn> document=({name: 'Dunx', loves: ['grape', 'watermelon'], weight: 704, gender: 'm', vampires: 165});
{
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 165
}
learn> db.unicorns.insert(document)
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("64787a6953c960af9553013c") }
}
learn>
```

4) Проверьте содержимое коллекции с помощью метода `find`.

```
learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 63
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 182
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 99
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 733,
    gender: 'f',
    vampires: 40
  },
]
```

```

{
  _id: ObjectId("6478798953c960af95530137"),
  name: 'Kenny',
  loves: [ 'grape', 'lemon' ],
  weight: 690,
  gender: 'm',
  vampires: 39
},
{
  _id: ObjectId("6478798953c960af95530138"),
  name: 'Raleigh',
  loves: [ 'apple', 'sugar' ],
  weight: 421,
  gender: 'm',
  vampires: 2
},
{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon' ],
  weight: 650,
  gender: 'm',
  vampires: 54
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 165
}
]

```

8.2.2 ВЫБОРКА ДАННЫХ ИЗ БД

Практическое задание 8.1.2:

1) Сформируйте запросы для вывода списков самок единорогов. Ограничьте список самок первыми тремя особями. Отсортируйте список по имени.

```
learn> db.unicorns.find({gender: 'f'}).sort({name: 1}).limit(3);
[
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 733,
    gender: 'f',
    vampires: 40
  },
  {
    _id: ObjectId("6478798953c960af95530139"),
    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.

```
learn> db.unicorns.findOne({gender: 'f', loves: 'carrot'});
{
  _id: ObjectId("6478798953c960af95530132"),
  name: 'Aurora',
  loves: [ 'carrot', 'grape' ],
  weight: 450,
  gender: 'f',
  vampires: 43
}
```

Практическое задание 8.1.3:

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

```
learn> db.unicorns.find({gender: 'm'}, {loves: 0, gender: 0});
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    weight: 600,
    vampires: 63
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    weight: 984,
    vampires: 182
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    weight: 575,
    vampires: 99
  },
  {
    _id: ObjectId("6478798953c960af95530137"),
    name: 'Kenny',
    weight: 690,
    vampires: 39
  },
  {
    _id: ObjectId("6478798953c960af95530138"),
    name: 'Raleigh',
    weight: 421,
    vampires: 2
  },
  {
    _id: ObjectId("6478798953c960af9553013a"),
    name: 'Pilot',
    weight: 650,
    vampires: 54
  },
  {
    _id: ObjectId("64787a6953c960af9553013c"),
    name: 'Dunx',
    weight: 704,
    vampires: 165
  }
]
```

Практическое задание 8.1.4:

Вывести список единорогов в обратном порядке добавления.

```
learn> db.unicorns.find().sort({$natural: -1});
[
  {
    _id: ObjectId("64787a6953c960af9553013c"),
    name: 'Dunx',
    loves: [ 'grape', 'watermelon' ],
    weight: 704,
    gender: 'm',
    vampires: 165
  },
  {
    _id: ObjectId("6478798953c960af9553013b"),
    name: 'Nimue',
    loves: [ 'grape', 'carrot' ],
    weight: 540,
    gender: 'f'
  },
  {
    _id: ObjectId("6478798953c960af9553013a"),
    name: 'Pilot',
    loves: [ 'apple', 'watermelon' ],
    weight: 650,
    gender: 'm',
    vampires: 54
  },
  {
    _id: ObjectId("6478798953c960af95530139"),
    name: 'Leia',
    loves: [ 'apple', 'watermelon' ],
    weight: 601,
    gender: 'f',
    vampires: 33
  },
  {
    _id: ObjectId("6478798953c960af95530138"),
    name: 'Raleigh',
    loves: [ 'apple', 'sugar' ],
    weight: 421,
    gender: 'm',
    vampires: 2
  },
  {
    _id: ObjectId("6478798953c960af95530137"),
    name: 'Kenny',
    loves: [ 'grape', 'lemon' ],
    weight: 690,
    gender: 'm',
    vampires: 39
  },
]
```



```

{
  _id: ObjectId("6478798953c960af95530136"),
  name: 'Ayna',
  loves: [ 'strawberry', 'lemon' ],
  weight: 733,
  gender: 'f',
  vampires: 40
},
{
  _id: ObjectId("6478798953c960af95530135"),
  name: 'Solnara',
  loves: [ 'apple', 'carrot', 'chocolate' ],
  weight: 550,
  gender: 'f',
  vampires: 80
},
{
  _id: ObjectId("6478798953c960af95530134"),
  name: 'Roooooodles',
  loves: [ 'apple' ],
  weight: 575,
  gender: 'm',
  vampires: 99
},
{
  _id: ObjectId("6478798953c960af95530133"),
  name: 'Unicrom',
  loves: [ 'energon', 'redbull' ],
  weight: 984,
  gender: 'm',
  vampires: 182
},
{
  _id: ObjectId("6478798953c960af95530132"),
  name: 'Aurora',
  loves: [ 'carrot', 'grape' ],
  weight: 450,
  gender: 'f',
  vampires: 43
},
{
  _id: ObjectId("6478798953c960af95530131"),
  name: 'Horny',
  loves: [ 'carrot', 'papaya' ],
  weight: 600,
  gender: 'm',
  vampires: 63
}
]

```

Практическое задание 8.1.5:

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

```
learn> db.unicorns.find({}, {_id: 0, loves: {$slice: 1}});
[
  {
    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: 'Rooooooodles',
    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
}
]
```

Практическое задание 8.1.6:

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

```
learn> db.unicorns.find({gender: 'f', weight: {$gte: 500, $lte: 700}}, {_id: 0});
[
  {
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    name: 'Leia',
    loves: [ 'apple', 'watermelon' ],
    weight: 601,
    gender: 'f',
    vampires: 33
  },
  {
    name: 'Nimue',
    loves: [ 'grape', 'carrot' ],
    weight: 540,
    gender: 'f'
  }
]
```

Практическое задание 8.1.7:

```
learn> db.unicorns.find({vampires: {$exists: false}});
[
  {
    _id: ObjectId("6478798953c960af9553013b"),
    name: 'Nimue',
    loves: [ 'grape', 'carrot' ],
    weight: 540,
    gender: 'f'
  }
]
```

Практическое задание 8.1.9:

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

```
learn> db.unicorns.find({gender: 'm'}, {_id: 0, name: 1, loves: {$slice: 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: 'Rooooooodles', loves: [ 'apple' ] },
  { name: 'Unicrom', loves: [ 'energon' ] }
]
```

8.2 ЗАПРОСЫ К БАЗЕ ДАННЫХ MONGODB. ВЫБОРКА ДАННЫХ. ВЛОЖЕННЫЕ ОБЪЕКТЫ. ИСПОЛЬЗОВАНИЕ КУРСОРОВ. АГРЕГИРОВАННЫЕ ЗАПРОСЫ. ИЗМЕНЕНИЕ ДАННЫХ

8.2.1 ЗАПРОС К ВЛОЖЕННЫМ ОБЪЕКТАМ

Практическое задание 8.2.1:

1) *Создайте коллекцию towns, включающую следующие документы*

```
learn> db.createCollection('towns');
{ ok: 1 }
learn> db.towns.insert({name: "Punxsutawney ", populatiuon: 6200, last_sensus: ISODate("2008-01-31"), famous_for: [], mayor: {name: "Jim Mehrle"}});
{ acknowledged: true,
  insertedIds: { '0': ObjectId("647881a353c960af95530141") } }
learn> db.towns.insert({name: "New York", populatiuon: 22200000, last_sensus: ISODate("2009-07-31"), famous_for: ["status of liberty", "food"], mayor: {name: "Michael Bloomberg", party: "I"}});
{ acknowledged: true,
  insertedIds: { '0': ObjectId("6478820d53c960af95530142") } }
learn> db.towns.insert({name: "Portland", populatiuon: 520000, last_sensus: ISODate("2009-07-20"), famous_for: ["beer", "food"], mayor: {name: "Sam Adams", party: "D"}});
{ acknowledged: true,
  insertedIds: { '0': ObjectId("6478824753c960af95530143") } }
```

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

```
learn> db.towns.find({"mayor.party": "I"}, {name: 1, mayor: 1, _id: 0});
[
  {
    name: 'New York',
    mayor: { name: 'Michael Bloomberg', party: 'I' }
  }
]
```

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

```
learn> db.towns.find({"mayor.party": "I"}, {name: 1, mayor: 1, _id: 0});
[
  {
    name: 'New York',
    mayor: { name: 'Michael Bloomberg', party: 'I' }
  }
]
```

Практическое 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.*
- 3) *Вывести результат, используя forEach.*

```
learn> function getMaleUnicorns() {return db.unicorns.find({gender: 'm'})};
learn> var cursor = getMaleUnicorns().sort({name: 1}).limit(2);
learn> cursor.forEach(function(unicorn) {print(unicorn.name)});
Dunx
Horny
```

8.2.2 АГРЕГИРОВАННЫЕ ЗАПРОСЫ

Практическое задание 8.2.3:

Вывести количество самок единорогов весом от полутонны до 600 кг.

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

Практическое задание 8.2.4:

Вывести список предпочтений.

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

Практическое задание 8.2.5:

Посчитать количество особей единорогов обоих полов.

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

8.2.5 РЕДАКТИРОВАНИЕ ДАННЫХ

Практическое задание 8.2.6:

1. Выполнить команду:

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

2. Проверить содержимое коллекции unicorns.

```

learn> db.unicorns.insert({name: 'Barney', loves: ['grape'], weight: 340, gender: 'm'});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("647884fb53c960af95530144") }
}
learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 63
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 182
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 99
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
]

```

```
{
  _id: ObjectId("6478798953c960af95530136"),
  name: 'Ayna',
  loves: [ 'strawberry', 'lemon' ],
  weight: 733,
  gender: 'f',
  vampires: 40
},
{
  _id: ObjectId("6478798953c960af95530137"),
  name: 'Kenny',
  loves: [ 'grape', 'lemon' ],
  weight: 690,
  gender: 'm',
  vampires: 39
},
{
  _id: ObjectId("6478798953c960af95530138"),
  name: 'Raleigh',
  loves: [ 'apple', 'sugar' ],
  weight: 421,
  gender: 'm',
  vampires: 2
},
{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon' ],
  weight: 650,
  gender: 'm',
  vampires: 54
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 165
},
{
  _id: ObjectId("647884fb53c960af95530144"),
  name: 'Barney',
  loves: [ 'grape' ],
  weight: 340,
  gender: 'm'
}
]
```


Практическое задание 8.2.7:

1. Для самки единорога Ауна внести изменения в БД: теперь ее вес 800, она убила 51 вампира.
2. Проверить содержимое коллекции `unicorns`.

```
learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 63
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 182
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 99
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 800,
    gender: 'f',
    vampires: 51
  },
  {
    _id: ObjectId("6478798953c960af95530137"),
    name: 'Kenny',
    loves: [ 'grape', 'lemon' ],
    weight: 690,
    gender: 'm',
    vampires: 39
  },
  {
    _id: ObjectId("6478798953c960af95530138"),
    name: 'Raleigh',
    loves: [ 'apple', 'sugar' ],
    weight: 421,
    gender: 'm',
    vampires: 2
  },
]
```

```

{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon' ],
  weight: 650,
  gender: 'm',
  vampires: 54
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 165
},
{
  _id: ObjectId("647884fb53c960af95530144"),
  name: 'Barney',
  loves: [ 'grape' ],
  weight: 340,
  gender: 'm'
}
]

```

Практическое задание 8.2.8:

1. Для самца единорога `Raleigh` внести изменения в БД: теперь он любит рэдбул.
2. Проверить содержимое коллекции `unicorns`.

```

learn> db.unicorns.updateOne({name: "Raleigh"}, {$push: {loves: "redbull"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
learn> dp.unicorns.find();
ReferenceError: dp is not defined
learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 63
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 182
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 99
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 800,
    gender: 'f',
    vampires: 51
  },
]

```

```

{
  _id: ObjectId("6478798953c960af95530137"),
  name: 'Kenny',
  loves: [ 'grape', 'lemon' ],
  weight: 690,
  gender: 'm',
  vampires: 39
},
{
  _id: ObjectId("6478798953c960af95530138"),
  name: 'Raleigh',
  loves: [ 'apple', 'sugar', 'redbull' ],
  weight: 421,
  gender: 'm',
  vampires: 2
},
{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon' ],
  weight: 650,
  gender: 'm',
  vampires: 54
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 165
},
{
  _id: ObjectId("647884fb53c960af95530144"),
  name: 'Barney',
  loves: [ 'grape' ],
  weight: 340,
  gender: 'm'
}
]

```

Практическое задание 8.2.9:

1. *Всем самцам единорогов увеличить количество убитых вампиров на 5.*
2. *Проверить содержимое коллекции unicorns.*

```

learn> db.unicorns.updateMany({gender: "m"}, {$inc: {"vampires": 5}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 8,
  modifiedCount: 8,
  upsertedCount: 0
}
learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 68
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 187
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 104
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 800,
    gender: 'f',
    vampires: 51
  },
]

```

```
{
  {
    _id: ObjectId("6478798953c960af95530137"),
    name: 'Kenny',
    loves: [ 'grape', 'lemon' ],
    weight: 690,
    gender: 'm',
    vampires: 44
  },
  {
    _id: ObjectId("6478798953c960af95530138"),
    name: 'Raleigh',
    loves: [ 'apple', 'sugar', 'redbull' ],
    weight: 421,
    gender: 'm',
    vampires: 7
  },
  {
    _id: ObjectId("6478798953c960af95530139"),
    name: 'Leia',
    loves: [ 'apple', 'watermelon' ],
    weight: 601,
    gender: 'f',
    vampires: 33
  },
  {
    _id: ObjectId("6478798953c960af9553013a"),
    name: 'Pilot',
    loves: [ 'apple', 'watermelon' ],
    weight: 650,
    gender: 'm',
    vampires: 59
  },
  {
    _id: ObjectId("6478798953c960af9553013b"),
    name: 'Nimue',
    loves: [ 'grape', 'carrot' ],
    weight: 540,
    gender: 'f'
  },
  {
    _id: ObjectId("64787a6953c960af9553013c"),
    name: 'Dunx',
    loves: [ 'grape', 'watermelon' ],
    weight: 704,
    gender: 'm',
    vampires: 170
  },
  {
    _id: ObjectId("647884fb53c960af95530144"),
    name: 'Barney',
    loves: [ 'grape' ],
    weight: 340,
    gender: 'm',
    vampires: 5
  }
}
```

Практическое задание 8.2.10:

1. Изменить информацию о городе Портланд: мэр этого города теперь беспартийный.
2. Проверить содержимое коллекции `towns`.

```
learn> db.towns.updateOne({name: "Portland"}, {$unset: {"mayor.party": ""}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
learn> db.towns.find();
[
  {
    _id: ObjectId("647881a353c960af95530141"),
    name: 'Punxsutawney ',
    populatiuon: 6200,
    last_sensus: ISODate("2008-01-31T00:00:00.000Z"),
    famous_for: [ '' ],
    mayor: { name: 'Jim Wehrle' }
  },
  {
    _id: ObjectId("6478820d53c960af95530142"),
    name: 'New York',
    populatiuon: 22200000,
    last_sensus: ISODate("2009-07-31T00:00:00.000Z"),
    famous_for: [ 'status of liberty', 'food' ],
    mayor: { name: 'Michael Bloomberg', party: 'I' }
  },
  {
    _id: ObjectId("6478824753c960af95530143"),
    name: 'Portland',
    populatiuon: 528000,
    last_sensus: ISODate("2009-07-20T00:00:00.000Z"),
    famous_for: [ 'beer', 'food' ],
    mayor: { name: 'Sam Adams' }
  }
]
```

Практическое задание 8.2.11:

1. Изменить информацию о самце единорога *Pilot*: теперь он любит и шоколад.
2. Проверить содержимое коллекции `unicorns`.


```

learn> db.unicorns.updateOne({name: "Pilot"}, {$push: {loves: "chocolate"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
learn> db.unicorns.find()
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 68
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 187
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 104
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 800,
    gender: 'f',
    vampires: 51
  },
]

```

```
{
  _id: ObjectId("6478798953c960af95530137"),
  name: 'Kenny',
  loves: [ 'grape', 'lemon' ],
  weight: 690,
  gender: 'm',
  vampires: 44
},
{
  _id: ObjectId("6478798953c960af95530138"),
  name: 'Raleigh',
  loves: [ 'apple', 'sugar', 'redbull' ],
  weight: 421,
  gender: 'm',
  vampires: 7
},
{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon', 'chocolate' ],
  weight: 650,
  gender: 'm',
  vampires: 59
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 170
},
{
  _id: ObjectId("647884fb53c960af95530144"),
  name: 'Barney',
  loves: [ 'grape' ],
  weight: 340,
  gender: 'm',
  vampires: 5
}
]
```

Практическое задание 8.2.12:

1. *Изменить информацию о самке единорога Aurora: теперь она любит еще и сахар, и лимоны.*
2. *Проверить содержимое коллекции unicorns.*

```

learn> db.unicorns.updateOne({name: 'Aurora'}, {$push: {loves: ["sugar", "lemons"]}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
learn> db.unicorn.find();

learn>

learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 68
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape', [ 'sugar', 'lemons' ] ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 187
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 104
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 800,
    gender: 'f',
    vampires: 51
  },
]

```

```
{
  _id: ObjectId("6478798953c960af95530137"),
  name: 'Kenny',
  loves: [ 'grape', 'lemon' ],
  weight: 690,
  gender: 'm',
  vampires: 44
},
{
  _id: ObjectId("6478798953c960af95530138"),
  name: 'Raleigh',
  loves: [ 'apple', 'sugar', 'redbull' ],
  weight: 421,
  gender: 'm',
  vampires: 7
},
{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon', 'chocolate' ],
  weight: 650,
  gender: 'm',
  vampires: 59
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 170
},
{
  _id: ObjectId("647884fb53c960af95530144"),
  name: 'Barney',
  loves: [ 'grape' ],
  weight: 340,
  gender: 'm',
  vampires: 5
}
]
```

8.2.6 УДАЛЕНИЕ ДАННЫХ ИЗ КОЛЛЕКЦИИ

Практическое задание 8.2.13:

1) *Создайте коллекцию towns, включающую следующие документы*

```
learn> db.towns.find();
[
  {
    _id: ObjectId("64788d540f1425dc9dd5ad9b"),
    name: 'Punxutawney ',
    popujatiuon: 6200,
    last_sensus: ISODate("2008-01-31T00:00:00.000Z"),
    famous_for: [ 'phil the groundhog' ],
    mayor: { name: 'Jim Wehrle' }
  },
  {
    _id: ObjectId("64788de70f1425dc9dd5ad9c"),
    name: 'New York',
    popujatiuon: 22200000,
    last_sensus: ISODate("2009-07-31T00:00:00.000Z"),
    famous_for: [ 'status of liberty', 'food' ],
    mayor: { name: 'Michael Bloomberg', party: 'I' }
  },
  {
    _id: ObjectId("64788e300f1425dc9dd5ad9d"),
    name: 'Portland',
    popujatiuon: 528000,
    last_sensus: ISODate("2009-07-20T00:00:00.000Z"),
    famous_for: [ 'beer', 'food' ],
    mayor: { name: 'Sam Adams', party: 'D' }
  }
]
```

- 2) *Удалите документы с беспартийными мэрами.*
- 3) *Проверьте содержание коллекции.*

```
learn> db.towns.deleteMany({"mayor.party": {$exists: false}});
{ acknowledged: true, deletedCount: 1 }
learn> db.towns.find();
[
  {
    _id: ObjectId("64788de70f1425dc9dd5ad9c"),
    name: 'New York',
    popujatiuon: 22200000,
    last_sensus: ISODate("2009-07-31T00:00:00.000Z"),
    famous_for: [ 'status of liberty', 'food' ],
    mayor: { name: 'Michael Bloomberg', party: 'I' }
  },
  {
    _id: ObjectId("64788e300f1425dc9dd5ad9d"),
    name: 'Portland',
    popujatiuon: 528000,
    last_sensus: ISODate("2009-07-20T00:00:00.000Z"),
    famous_for: [ 'beer', 'food' ],
    mayor: { name: 'Sam Adams', party: 'D' }
  }
]
```

- 4) Очистите коллекцию.
- 5) Просмотрите список доступных коллекций.

```
learn> db.towns.deleteMany({});
{ acknowledged: true, deletedCount: 2 }
learn> db.getCollectionNames();
[ 'towns', 'unicorns' ]
```

Контрольные вопросы:

1. Как используется оператор точка?
Обращение к полям вложенных документов или элементов массива
2. Как можно использовать курсор?
Курсор используется для обработки итеративного доступа к результатам запроса.
3. Какие возможности агрегирования данных существуют в MongoDB?
Группировка, сортировка, фильтрация, объединение коллекций и т.д.
4. Какая из функций save или update более детально позволит настроить редактирование документов коллекции?
Update (больше операторов (\$push, \$addToSet, \$inc и тд)).
5. Как происходит удаление документов из коллекции по умолчанию?
deleteMany({}).

8.3.2 НАСТРОЙКА ИНДЕКСОВ

Практическое задание 8.3.1:

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

```
learn> db.zone.insertOne({_id: "zone1", name: "Zone 1", description: "Zone 1 description"});
{ acknowledged: true, insertedId: 'zone1' }
learn> db.zone.insertOne({_id: "zone2", name: "Zone 2", description: "Zone 2 description"});
{ acknowledged: true, insertedId: 'zone2' }
learn> db.zones.find();

learn> db.zones.find();

learn> db.zone.find();
[
  { _id: 'zone1', name: 'Zone 1', description: 'Zone 1 description' },
  { _id: 'zone2', name: 'Zone 2', description: 'Zone 2 description' }
]
```

- 2) *Включите для нескольких единорогов в документы ссылку на зону обитания, используя второй способ автоматического связывания.*
- 3) *Проверьте содержание коллекции единорогов.*


```

learn> db.unicorns.updateMany({gender: 'm'}, {$set: {zone: "zone1"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 8,
  modifiedCount: 8,
  upsertedCount: 0
}
learn> db.unicorns.find();
[
  {
    _id: ObjectId("6478798953c960af95530131"),
    name: 'Horny',
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 68,
    zone: 'zone1'
  },
  {
    _id: ObjectId("6478798953c960af95530132"),
    name: 'Aurora',
    loves: [ 'carrot', 'grape', [ 'sugar', 'lemons' ] ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId("6478798953c960af95530133"),
    name: 'Unicrom',
    loves: [ 'energon', 'redbull' ],
    weight: 984,
    gender: 'm',
    vampires: 187,
    zone: 'zone1'
  },
  {
    _id: ObjectId("6478798953c960af95530134"),
    name: 'Rooooooodles',
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 104,
    zone: 'zone1'
  },
  {
    _id: ObjectId("6478798953c960af95530135"),
    name: 'Solnara',
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId("6478798953c960af95530136"),
    name: 'Ayna',
    loves: [ 'strawberry', 'lemon' ],
    weight: 800,
    gender: 'f',
    vampires: 51
  },
]

```

```
{
  _id: ObjectId("6478798953c960af95530137"),
  name: 'Kenny',
  loves: [ 'grape', 'lemon' ],
  weight: 690,
  gender: 'm',
  vampires: 44,
  zone: 'zone1'
},
{
  _id: ObjectId("6478798953c960af95530138"),
  name: 'Raleigh',
  loves: [ 'apple', 'sugar', 'redbull' ],
  weight: 421,
  gender: 'm',
  vampires: 7,
  zone: 'zone1'
},
{
  _id: ObjectId("6478798953c960af95530139"),
  name: 'Leia',
  loves: [ 'apple', 'watermelon' ],
  weight: 601,
  gender: 'f',
  vampires: 33
},
{
  _id: ObjectId("6478798953c960af9553013a"),
  name: 'Pilot',
  loves: [ 'apple', 'watermelon', 'chocolate' ],
  weight: 650,
  gender: 'm',
  vampires: 59,
  zone: 'zone1'
},
{
  _id: ObjectId("6478798953c960af9553013b"),
  name: 'Nimue',
  loves: [ 'grape', 'carrot' ],
  weight: 540,
  gender: 'f'
},
{
  _id: ObjectId("64787a6953c960af9553013c"),
  name: 'Dunx',
  loves: [ 'grape', 'watermelon' ],
  weight: 704,
  gender: 'm',
  vampires: 170,
  zone: 'zone1'
},
{
  _id: ObjectId("647884fb53c960af95530144"),
  name: 'Barney',
  loves: [ 'grape' ],
  weight: 340,
  gender: 'm',
  vampires: 5,
  zone: 'zone1'
}
]
```

Практическое задание 8.3.2:

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

```
learn> db.unicorns.createIndex({name: 1}, {unique: true});
name_1
```

Практическое задание 8.3.3:

- 1) Получите информацию о 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.dropIndexes();
{
  nIndexesWas: 2,
  msg: 'non-_id indexes dropped for collection',
  ok: 1
}
```

- 3) Попробуйте удалить индекс для идентификатора.

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

Практическое задание 8.3.4:

- 1) Создайте объемную коллекцию `numbers`, задействовав курсор:

```
for(i = 0; i < 100000; i++){db.numbers.insert({value: i})}
```

- 2) Выберите последних четыре документа.

```
learn> for (i = 0; i < 10000; i++) {db.numbers.insertOne({value: i})};

learn>

learn>

learn> db.numbers.find().sort({_id: -1}).limit(4);
[
  { _id: ObjectId("647893196c1428b6c800cd25"), value: 9999 },
  { _id: ObjectId("647893196c1428b6c800cd24"), value: 9998 },
  { _id: ObjectId("647893196c1428b6c800cd23"), value: 9997 },
  { _id: ObjectId("647893196c1428b6c800cd22"), value: 9996 }
]
```

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

360ms

- 4) Создайте индекс для ключа `value`.
- 5) Получите информацию о всех индексах коллекции `numbers`.

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

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

0?? ms

```

learn> db.numbers.find().sort({_id: -1}).limit(4);
[
  { _id: ObjectId("647893196c1428b6c800cd25"), value: 9999 },
  { _id: ObjectId("647893196c1428b6c800cd24"), value: 9998 },
  { _id: ObjectId("647893196c1428b6c800cd23"), value: 9997 },
  { _id: ObjectId("647893196c1428b6c800cd22"), value: 9996 }
]
learn> db.users.explain("executionStats").find({})
{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'learn.users',
    indexFilterSet: false,
    parsedQuery: {},
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    winningPlan: { stage: 'EOF' },
    rejectedPlans: []
  },
  executionStats: {
    executionSuccess: true,
    nReturned: 0,
    executionTimeMillis: 0,
    totalKeysExamined: 0,
    totalDocsExamined: 0,
    executionStages: {
      stage: 'EOF',
      nReturned: 0,
      executionTimeMillisEstimate: 0,
      works: 1,
      advanced: 0,
      needTime: 0,
      needYield: 0,
      saveState: 0,
      restoreState: 0,
      isEOF: 1
    }
  }
},

```

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

С индексом эффективнее.??

Контрольные вопросы:

1. Назовите способы связывания коллекций в MongoDB.

Встраивание: по значению полей, ссылки на документы: связь осуществляется по идентификатору документа

2. Сколько индексов можно установить на одну коллекцию в БД MongoDB. До 64 индексов.

3. Как получить информацию о всех индексах базы данных MongoDB? `getIndexes()`.