MongoDB (формирование выборки, пагинация и сортировка)

Find

Чтобы извлечь все документы из коллекции, мы можем использовать следующую команду:

```
db.users.find()
```

Если нам надо получить не все документы, а только те, которые удовлетворяют определенному требованию, необходимо в find() указать условие. Например, выведем все документы, у которых name=”Anton”:

```
db.users.find({name: "Anton"})
```

Операторы

Сравнения:

- `$eq`: значения равны
- `$ne`: значения не равны
- `$gt`: значение больше другого значения
- `$gte`: значение больше или равно другому значению
- `$lt`: значение меньше другого значения
- `$lte`: значение меньше или равно другому значению
- `$in`: значение соответствует одному из значений в массиве

Примеры:

```
-- Вернет пользователей, чей возраст больше или равен 30
db.users.find({age: {$gte: 30}})

-- Вернет пользователей, чей возраст меньше 30 лет
db.users.find({age: {$lt: 30}})

-- Вернет пользователей, чей возраст равен 22 или 32
db.users.find ({age: {$in : [22, 32]}})
```

Логические:

- `$and`: возвращает документы, в которых выполняются оба условия
- `$or`: возвращает документы, в которых выполняется хотя бы одно условие
- `$nor`: возвращает документы, в которых оба условия не выполняются
- `$not`: возвращает документы, в которых условие не выполняется

```
-- Вернет пользователей, чей возраст находится в диапазоне от 22 до 30 лет
db.users.find(
  {$and: [
    {age: {$gte: 22}},
    {age: {$lte: 30}}
    ]
  }
)

-- Вернет то же, что и прошлый запрос. Короткий аналог указания двух условий
db.users.find({
  age: {$gte: 22, $lte: 30}
})

-- Вернет тех, кто отвечает хотя бы одному из условий
db.users.find(
  {$or: [
    {name: "Anton"},
    {age: {$lte: 30}}
    ]
```

```
  }
)
```

Пагинация и сортировка

limit()

Функция limit() задает максимально допустимое количество получаемых документов. Количество передается в виде числового параметра. Например, выведем три первых документа из коллекции:

```
db.users.find().limit(3)
```

skip()

Если мы хотим произвести выборку не сначала, а пропустив какое-то количество документов, то воспользуемся функцией skip(). Например, выведем все документы коллекции, кроме первых трех:

```
db.users.find().skip(3)
```

Функции limit() и skip() можно комбинировать:

```
db.users.find().skip(3).limit(3)
```

Оператор $slice

$slice является в некотором роде комбинацией функций `limit` и `skip`. Но в отличие от них $slice может работать с массивами.

```
-- В выборку попадет только один язык из массива languages, а не весь массив
db.users.find ({name: "Anton"}, {languages: {$slice : 1}})
```

sort()

Чтобы отсортировать полученную выборку, необходимо воспользоваться функцией sort(). Передавая в функцию значения 1 или -1, мы можем указать в каком порядке сортировать: по возрастанию (1) или по убыванию (-1).

```
db.users.find().sort({age: 1})
```