

ПРАКТИЧЕСКАЯ РАБОТА 6. ЗАПРОСЫ: МОДИФИКАТОРЫ МАССИВОВ. ПОЗИЦИОННЫЕ МОДИФИКАТОРЫ МАССИВОВ

6.1 Цель и содержание

Цель работы: знакомство с модификаторами массивов.

Задачи работы: научиться работать с массивами.

6.2 Модификаторы массивов в запросах

Для добавления элемента в массив используется модификатор «\$push», который используется как параметр метода «update». Синтаксис модификатора:

{ \$push: { <массив>: <значение> }, где:

<массив> – массив, в который происходит добавление элемента;

<значение> – значение добавляемого элемента. Пример использования модификатора:

```
db.phones.update({ name: "NL920" }, { $push: { cost: 19990 } })
```

Данный запрос добавит элемент «19990» в массив «cost».

Примечания

Массив будет создан, в случае если он отсутствует.

Если обновляемое поле не является массивом, то в результате выполнения метода возникнет ошибка.

Если значение является массивом, \$push добавляет весь массив как отдельный элемент. Для того чтобы добавить несколько элементов, необходимо использовать модификатор \$each:

Следующий пример добавляет числа из массива [90, 92, 85] как отдельные элементы в массив cost:

```
db.phones.update({name: "NL920"}, {$push: {cost: {$each: [19900, 20000, 20100]}}})13
```

Для удаления элемента из массива используется модификатор «\$pop».

Синтаксис модификатора:

{ \$pop: { <поле>: <опц. удаления> } },

¹³ \$push – MongoDBManual 2.4.4[Электронный ресурс]. – Режим доступа:<http://docs.mongodb.org/manual/reference/operator/push/>

где:

<поле> – имя массива;

<опц. удаления> – 1 – если требуется удалить последний элемент в массиве, -1 – если требуется удалить первый элемент в массиве.

Удаление первого элемента в массиве выглядит следующим образом:

```
db.phones.update({ name: "NL920" }, { $pop: { cost: 1 } })
```

Модификатор «\$push» добавляет элементы в массив, не проверяя их на уникальность. Для того чтобы добавить в массив только уникальные элементы используется модификатор «\$addToSet». Синтаксис модификатора аналогичен синтаксису «\$push».

Для удаления элемента массива по определенному критерию используется модификатор «\$pull». Синтаксис модификатора:

{ \$pull: { <массив>: <запрос> } }, где:

<массив> – имя массива;

<запрос> – запрос для отыскания требуемого элемента.

Например, для удаления элемента «20100» из массива «cost» используется запрос:

```
db.phones.update({ name: "NL920" }, { $pull: { cost: 20100 } })
```

6.3 Позиционные модификаторы массивов

Для доступа к конкретному элементу массива используется два способа: по конкретной позиции и с помощью использования позиционного оператора (символ '\$').

Для доступа к элементу массива по номеру, после запроса на выборку документа в качестве имени изменяемого поля используется: «<массив>.<№ элемента>»

Например, для увеличения нулевого элемента массива на 100 используется следующий запрос:

```
db.phones.update({ "name": "NL920" }, {$inc: {"cost.0": 100}})
```

Если требуется изменить элемент массива, не зная его позиции в массиве, то его необходимо отыскать с помощью запроса, и применить требуемый модификатор. В данном случае в качестве имени изменяемого поля будет выступать: «<массив>.<символ '\$'>».

Например, для того чтобы увеличить элемент массива равный «19900» на 200 требуется ввести в консоль:

```
db.phones.update({ "cost": 19900 }, {$inc: {"cost.$": 200}})
```

6.4 Запросы в массивах

Иногда требуется составить запрос, в котором происходит поиск по массиву значений. В таком случае применяются запросы в массивах. Выше было показано, как можно отыскать элемент в массиве. Для того чтобы найти элемент массива, равный «19990», используется следующий запрос:

```
db.phones.find({ "cost": 19900 })
```

Данный запрос успешно найдет документы, в которых массив с именем «cost» будет содержать элемент «19990».

Для того чтобы выбрать документы больше, чем по одному элементу массива, можно использовать оператор «\$all».

Синтаксис оператора:

{ <поле>: { \$all: [<значение1>, <значение 2>, ...] } }, где:

<поле> – поле, по которому проводится поиск;

<значение 1, 2, ...> – значения массива, которые нужно найти.

Для получения документов, в которых массив содержит элементы «19990», «20000» и «20100» используется запрос:

```
db.phones.find({cost: {$all: [19990, 20000, 20100]}})
```

Для получения документов по полному совпадению элементов в массиве необходимо просто перечислить искомые элементы так же, как и при вставке:

```
db.phones.find({cost: [19990, 20000, 20100]})
```

Для управления количеством возвращаемых элементов в массиве используется оператор «\$slice». Синтаксис оператора:

{ \$slice: <количество элементов> }.

Для возвращения элементов массива, начиная с его конца, оператору передается отрицательное значение, с начала – положительное.

Оператор \$slice может действовать подобно «skip» и «limit», только в отношении массивов. В этом случае оператор имеет следующий синтаксис:

{ \$slice: [<skip>, <limit>] }, где:

<skip> – пропускаемое количество элементов;

<limit> – возвращаемое количество элементов.

В качестве примера рассмотрим запрос для получения первых десяти элементов массива «cost», который выглядит следующим образом:

```
db.phones.find( {}, {cost : {$slice: 10}})
```

Если нужно получить последние десять записей, то число «10» в запросе будет отрицательным.

Для ограничения/пропуска элементов, или для получения элементов из середины массива необходимо использовать вторую форму оператора «\$slice»:

```
db.phones.find( {}, {cost : {$slice: [5, 10]}})
```

Данный запрос пропустит 5 начальных элементов массива и вернет элементы с 6 по 16.

6.5 Методика и порядок выполнения работы

Индивидуальное задание:

1. Создайте несколько запросов для вставки данных в массив.
2. Создайте запросы, производящие обновление данных в массиве: как по позиции элемента в массиве, так и по его значению.
3. Создайте запросы, удаляющие элементы из массива: по позиции элемента в массиве и по его значению.