**Лабораторная работа №3**

**Операции с документами**

**Удаление документов**

Cледует выполнять команды, выделенные **жирным шрифтом**

Для дальнейшего изучения возможностей Mongo создадим новую базу данных

**> use База1**,

а в ней коллекцию Авторы

**> db.createCollection(Авторы)**

В созданную коллекцию введем документы

**> db.Авторы.insert ({"Au\_lname":"Лесков", "Au\_fname": "Николай", "birthday": ISODate ("1831-02-16")})**

**> db.Авторы.insert ({\_id: "000–11–1111", "Au\_ lname": "Пелевин", " Au\_fname ": "Виˊктор", "year\_of\_birth": 1962, Contract: true})**

**> db.Авторы.insert({"Au\_lname":"Н. В. Гоголь", "year\_of\_ birth":1809})**

Для удаления отдельных документов из коллекции предназначен перегружаемый метод remove. При вызове метода с одним параметром db.<Коллекция >.remove ({<Селектор>}) происходит удаление всех документов, удовлетворяющих селектору. А в отсутствие селектора из коллекции удаляются все документы. При вызове с двумя параметрами db.<Коллекция>.remove ({<Селектор>}, true) удаляется только один (первый) документ из соответствующих селектору.

Например, команда

**> db.Авторы.remove ({"year\_of\_birth":{$gt:1907}})**

удалит документы с годом рождения авторов, большим, чем 1907.

Убедимся в этом

**> db.Авторы.find()**

{ "\_id" : ObjectId("586f5cc203ef629951278f6d"), "Au\_lname" : "Лесков", "Au\_fname" : "Николай", "birthday" : ISODate("1831-02-16T00:00:00Z") }

{ "\_id" : ObjectId("586f5f4303ef629951278f6e"), "Au\_lname" : "Н. В. Гоголь", "year\_of\_ birth" : 1809 }

**Изменение документов**

Изменения в документы коллекции вносятся методом update.

Метод может изменить определенные атрибуты одного или нескольких существующих документов или полностью заменить существующий документ новым. Вид выполняемого действия определяется используемыми параметрами и их модификаторами. Базовый вызов метода update имеет вид:

db. <Коллекция>.update ({<Селектор>}, {<Новый документ или модификатор с изменяемыми атрибутами>}, {<Опции обновления>});

Полная замена документа, удовлетворяющего селектору, выполняется, если новые значения атрибутов заданы без использования модификаторов

Пусть в коллекции Авторы имеется документ:

**> db.Авторы.insert({Au\_id: "0007", lname: "Гоголь", Years: [1809, 1852]})**

Применение метода

**db.authors.update ({Au\_id:"0007"}, {Au\_ lname:"Гоголь", Years: [1809, 1852], Books: ["Вий", "Тарас Бульба"]})**

приведет к замене выбранного селектором {Au\_id:"0007"} документа новым.

Заносим в переменную doc новый документ:

**> var doc = {Au\_id: "0007", Au\_lname: "Н. В. Гоголь", Years: [1809, 1852]}**

Выполняем замену документа:

**> db.Авторы.update ({Au\_id: "0007"}, doc)**

Убедимся в том, что операция выполнена

**> db.Авторы.find()**

{ "\_id" : ObjectId("586f69584652a7f8a3964bd7"), "Au\_lname" : "Лесков", "Au\_fname" : "Николай", "birthday" : ISODate("1831-02-16T00:00:00Z") }

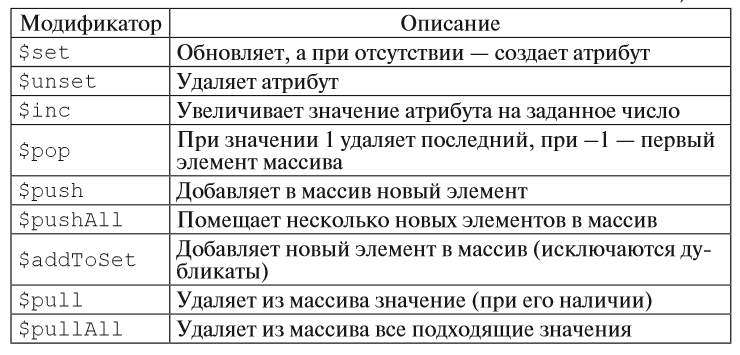
{ "\_id" : ObjectId("586f69a44652a7f8a3964bd8"), "Au\_lname" : "Н. В. Гоголь", "year\_of\_ birth" : 1809 }

{ "\_id" : ObjectId("586f6ce34652a7f8a3964bd9"), "Au\_id" : "0007", "Au\_lname" : "Н. В. Гоголь", "Years" : [ 1809, 1852 ] }

Третий параметр «Опции обновления» влияет на «поведение» метода update. Опция upsert (типа boolean) со значением true при отсутствии документа, удовлетворяющие условию селектора, добавляет документ с заданными атрибутами. При значении false новый документ не создается. По умолчанию действует false. Опция multi (типа boolean): по значению true обновляются все документы, удовлетворяющего условию селектора. В значении false обновляет один документ. Значение по умолчанию false. Добавление, удаление или изменение атрибутов в документе задается модификаторами, устанавливаемыми перед их значениями. Изменяемые атрибуты записываются в виде:

<Модификатор>:{<Атрибут>:<Значение>},..

Ниже приводится сводная таблица модификаторов, используемых в методе update ().



Рассмотрим использование модификаторов внесения изменений.

Пусть в коллекции Авторы находится следующий документ:

{"Au\_id": "000–11–0009", "LName": "Толстой", "Lang": "Русский", "Years": [1882, 1945], "Product": ["Аэлита", "Князь Серебряный"]}

**> db.Авторы.insert({"Au\_id": "000–11–0009", "LName": "Толстой", "Lang": "Русский", "Years": [1882, 1945], "Product": ["Аэлита", "Князь Серебряный"]})**

**> db.Авторы.insert({"Au\_id": "000–11–0009", "Au\_lname": "Толстой", "Years": [1882, 1945], "Books": ["Аэлита", "Князь Серебряный"]} )**

1. Замена всего документа

<db.authors.update ({Au\_id: "000–11–0009"}, {Au\_ id: "000–11–0009", Au\_lname: "Толстой", Years: [1882, 1945], Books: ["Аэлита", "Князь Серебряный"]})

2. Добавление нового атрибута. Модификатор $set добавляет атрибут, если его не было в документе:

**> db.authors.update ({Au\_id: "000–11–0009"}, {$set:{"Страна": "СССР"}})**

Запрос

**> db.Авторы.find({Au\_id: "000–11–0009"})**

{ "\_id" : ObjectId("586f7ddf4a7bd4ba7abfc2dc"), "Au\_id" : "000–11–0009", "Au\_lname" : "Толстой", "Years" : [ 1882, 1945 ], "Books" : [ "Аэлита", "Князь Серебряный" ], "Страна" : "СССР" }

подтверждает внесенные изменения

3. Изменение значения атрибута. Модификатор $set изменяет значение атрибута, если такой атрибут существует в документе.

Уточняем автора:

**> db.Авторы.update ({Au\_id: "000–11–0009"}, {$set:{Au\_lname: "А. Н. Толстой"}})**

Проверяем соответствующим запросом

**> db.Авторы.find({Au\_id: "000–11–0009"})**

{ "\_id" : ObjectId("586f7ddf4a7bd4ba7abfc2dc"), "Au\_id" : "000–11–0009", "Au\_lname" : "А. Н. Толстой", "Years" : [ 1882, 1945 ], "Books" : [ "Аэлита", "Князь Серебряный" ], "Страна" : "СССР" }

4. Удаление ошибочного элемента "Князь Серебряный" из атрибута-массива "Books":

**> db.Авторы.update ({Au\_id: "000–11–0009"}, {$pull:{"Books":"Князь Серебряный"}})**

Проверяем соответствующим запросом

**<db.Авторы.find({Au\_id: "000–11–0009"})**

{ "\_id" : ObjectId("586f7ddf4a7bd4ba7abfc2dc"), "Au\_id" : "000–11–0009", "Au\_lname" : "А. Н. Толстой", "Years" : [ 1882, 1945 ], "Books" : [ "Аэлита" ], "Страна" : "СССР" }

5. Добавление нового элемента "Петр Первый" в массив "Books"

**db.Авторы.update ({Au\_id: "000–11–0009"}, {$push:{"Books": "Петр Первый"}})**

Проверяем соответствующим запросом

**> db.Авторы.find({Au\_id: "000–11–0009"})**

**{ "\_id" : ObjectId("586f7ddf4a7bd4ba7abfc2dc"), "Au\_id" : "000–11–0009", "Au\_lname" : "А. Н. Толстой", "Years" : [ 1882, 1945 ], "Books" : [ "Аэлита", "Петр Первый" ], "Страна" : "СССР" }**

**Добавление или замена документа в коллекции — метод save**

db.<Коллекция>.save (<Документ>)

Если указанный в параметре документ не содержит атрибут \_id, то выполняется добавление документа с созданием уникального идентификатора \_id.

Если атрибут \_id в документе задан и он совпадает с \_id документа в коллекции, то выполняется замена существующего документа, иначе добавляется новый документ с заданным ключом \_id.

По аналогии с методами insert и update параметром метода save может быть содержащая документ переменная.

Например,

**> var doc = {Au\_id: "0008", Au\_lname: "А. С. Пушкин", Years: [1799, 1837]};**

**> db.Авторы.save (doc)**

**> db.Авторы.find()**

{ "\_id" : ObjectId("586f69584652a7f8a3964bd7"), "Au\_lname" : "Лесков", "Au\_fname" : "Николай", "birthday" : ISODate("1831-02-16T00:00:00Z") }

{ "\_id" : ObjectId("586f69a44652a7f8a3964bd8"), "Au\_lname" : "Н. В. Гоголь", "year\_of\_ birth" : 1809 }

{ "\_id" : ObjectId("586f6ce34652a7f8a3964bd9"), "Au\_id" : "0007", "Au\_lname" : "Н. В. Гоголь", "Years" : [ 1809, 1852 ] }

{ "\_id" : ObjectId("586f7ddf4a7bd4ba7abfc2dc"), "Au\_id" : "000–11–0009", "Au\_lname" : "А. Н. Толстой", "Years" : [ 1882, 1945 ], "Books" : [ "Аэлита", "Петр Первый" ], "Страна" : "СССР" }

{ "\_id" : ObjectId("586f87fa4a7bd4ba7abfc2dd"), "Au\_id" : "0008", "Au\_lname" : "А. С. Пушкин", "Years" : [ 1799, 1837 ] }

**Задание для самостоятельной работы**

Создать базу данных с коллекцией, содержащей информацию о Ваших друзьях , родственниках и знакомых. Каждый документ должен содержать следующие атрибуты: Фамилия и инициалы, номер телефона, год рождения, город проживания, адрес (улица, номер дома, номер квартиры). Имя коллекции должно совпадать с Вашей фамилией. Атрибут-массив "номер телефона" в некоторых документов должен содержать более двух значений. Коллекция должна содержать не менее 15 документов. Для добавления документов в коллекцию следует использовать разные методы (insert, save,…). Выполните следующие действия

1.Добавьте в документы еще один атрибут – страна проживания.

2.Удалите один из телефонов из атрибут-массивов некоторых документах

3.Добавте номер телефона в атрибут-массивы некоторых документах

4. Изменение значения атрибута "город проживания" в некоторых документах

5. Удалите документы, у которых атрибут "год рождения" превосходит Ваш год рождения

**Вопросы для защиты работы**

1.Cинтаксис метода отдельных документов

2. Какой имеет вид базовый вызов метода update

3. Какие модификаторы, используются в методе update

4. В каком виде записываются изменяемые атрибуты в методе update

5. Cинтаксис метода save

**Содержание отчета**

1.Номер и название лабораторной работы

2. Экранные формы, показывающие порядок выполнения лабораторной

работы, и результаты, полученные в ходе её выполнения.

3.Постановка задачи для самостоятельной работы

4. Экранные формы, показывающие порядок выполнения задания для

Самостоятельной работы с соответствующими пояснениями, и результаты, полученные в ходе её выполнения.

5.Ответы на вопросы для защиты