

Creating Collection

```
db.createCollection("Member")
db.Member.insert
(
[
{
    "member_id" : "1",
    "member_fname" : "Akshul",
    "member_lname" : "Agarwal",
    "member_insurance_status" : "1",
    "relation" : "Self",
    "sub_id" : "1",
    "group_id" : "1",
    "primary_doctor" : "1",
    count : NumberInt(0)
},
{
    "member_id" : "2",
    "member_fname" : "Gaurang",
    "member_lname" : "Panchal",
    "member_insurance_status" : "1",
    "relation" : "Group Member",
    "sub_id" : "1",
    "group_id" : "1",
    "primary_doctor" : "1",
    count : NumberInt(0)
},
{
    "member_id" : "3",
    "member_fname" : "Ujjval",
    "member_lname" : "Thakkar",
    "member_insurance_status" : "1",
    "relation" : "Group member",
    "sub_id" : "1",
    "group_id" : "1",
    "primary_doctor" : "2",
    count : NumberInt(0)
},
{
    "member_id" : "4",
    "member_fname" : "John",
    "member_lname" : "Doe",
    "member_insurance_status" : "1",
    "relation" : "Group member",
    "sub_id" : "1",
    "group_id" : "1",
```

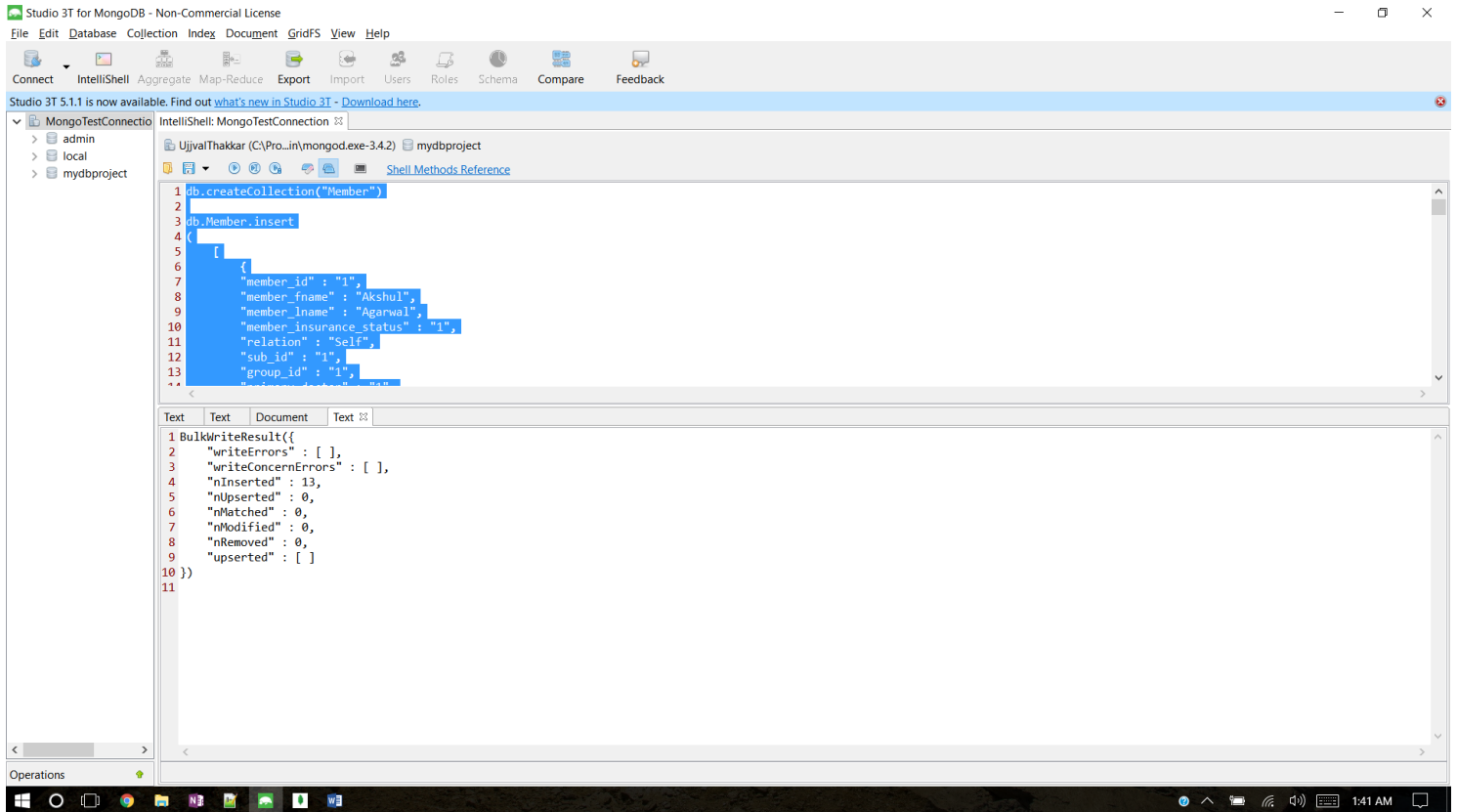
```
"primary_doctor" : "2",
count : NumberInt(0)
},
{
"member_id" : "5",
"member_fname" : "Peter",
    "member_lname" : "Samson",
    "member_insurance_status" : "1",
    "relation" : "Self",
    "sub_id" : "2",
    "group_id" : "2",
    "primary_doctor" : "3",
    count : NumberInt(0)
},
{
    "member_id" : "6",
    "member_fname" : "Harry",
    "member_lname" : "Potter",
    "member_insurance_status" : "1",
    "relation" : "Group member",
    "sub_id" : "2",
    "group_id" : "2",
    "primary_doctor" : "3",
    count : NumberInt(0)
},
{
    "member_id" : "7",
    "member_fname" : "Charles",
    "member_lname" : "Rhodes",
    "member_insurance_status" : "1",
    "relation" : "Group member",
    "sub_id" : "2",
    "group_id" : "2",
    "primary_doctor" : "4",
    count : NumberInt(0)
},
{
    "member_id" : "8",
    "member_fname" : "Nick",
    "member_lname" : "Halden",
    "member_insurance_status" : "1",
    "relation" : "Group member",
    "sub_id" : "2",
    "group_id" : "2",
    "primary_doctor" : "4",
    count : NumberInt(0)
},
}
```

```
{
  "member_id" : "9",
  "member_fname" : "Dan",
  "member_lname" : "Summers",
  "member_insurance_status" : "1",
  "relation" : "Self",
  "sub_id" : "3",
  "group_id" : "3",
  "primary_doctor" : "5",
  count : NumberInt(0)
},
{
  "member_id" : "10",
  "member_fname" : "Steve",
  "member_lname" : "Harvey",
  "member_insurance_status" : "1",
  "relation" : "Group member",
  "sub_id" : "3",
  "group_id" : "3",
  "primary_doctor" : "5",
  count : NumberInt(0)
},
{
  "member_id" : "11",
  "member_fname" : "Jack",
  "member_lname" : "Reacher",
  "member_insurance_status" : "1",
  "relation" : "Self",
  "sub_id" : "4",
  "group_id" : "4",
  "primary_doctor" : "6",
  count : NumberInt(0)
},
{
  "member_id" : "12",
  "member_fname" : "Bill",
  "member_lname" : "Gates",
  "member_insurance_status" : "1",
  "relation" : "Group member",
  "sub_id" : "4",
  "group_id" : "4",
  "primary_doctor" : "7",
  count : NumberInt(0)
},
{
  "member_id" : "13",
  "member_fname" : "Mathew",
```

```

"member_lname" : "Hayden",
"member_insurance_status" : "1",
"relation" : "Group member",
"sub_id" : "4",
"group_id" : "4",
"primary_doctor" : "9",
count : NumberInt(0)
}
]
)

```



Search through the Members in your mongo collection and return a document based on searching for a name

```
db.Member.find(  
{"Member_fname":"Ujjval"}  
)
```

Studio 3T for MongoDB - Non-Commercial License

File Edit Database Collection Index Document GridFS View Help

Connect IntelliShell Aggregate Map-Reduce Export Import Users Roles Schema Compare Feedback

Studio 3T 5.1.1 is now available. Find out [what's new in Studio 3T](#) - [Download here](#).

MongoTestConnection

UjjvalThakkar (C:\Pro...in\mongod.exe-3.4.2) mydbproject

```
142 "member_insurance_status" : "1",  
143 "relation" : "Group member",  
144 "sub_id" : "4",  
145 "group_id" : "4",  
146 "primary_doctor" : "9",  
147 count : NumberInt(0)  
148 }  
149 1  
150 }  
151  
152 db.Member.find(  
153 {"member_fname":"Ujjval"}  
154 )
```

Find

50 Documents 1 to 1

Table View

_id	count	group_id	member_fname	member_id	member_insurance...	member_lname	primary_doctor	relation	sub_id
58e7257945b2...	0	1	Ujjval	3	1	Thakkar	2	Group member	1

Operations

Displaying 1 document

Count Documents 0.001s

Update the document and change the name

```
db.Member.update({'member_fname':'Ujjval'},{$set: {'member_fname':'Ujjval Thakkar'}})
```

```
db.Member.find(
{"member_fname":{$gte:"Ujjval"}}
)
```

Explanation:

First query, Updates the document with fname as “Ujjval” to fname as “Ujjval Thakkar”

The second query, Finds a member with name >= “Ujjval”

\$gte => Greater than or equal to (>=)

The screenshot shows the Studio 3T for MongoDB interface. The left sidebar displays the database structure with 'admin', 'local', and 'mydbproject' collections. The main window shows the MongoDB Shell with the following commands:

```
155
156
157 db.Member.update({'member_fname':'Ujjval'},{$set: {'member_fname':'Ujjval Thakkar'}})
158
159 db.Member.find(
160 {'member_fname':{$gte:'Ujjval'}}
161 )
162
163 db.Member.update(
164   {'member_fname':'Ujjval Thakkar'},
165   {'$inc: { count: NumberInt(1)} }
166 )
167
168 db.Member.find(
```

Below the shell, a table view shows the document structure for the 'Member' collection. The table has columns: _id, count, group_id, member_fname, member_id, member_insurance_..., member_lname, primary_doctor, relation, and sub_id. The first document is displayed:

_id	count	group_id	member_fname	member_id	member_insurance_...	member_lname	primary_doctor	relation	sub_id
58e7257945b2...	0	1	Ujjval Thakkar	3	1	Thakkar	2	Group member	1

The status bar at the bottom indicates 'Displaying 1 document' and 'Count Documents 0.001s'.

Update the document again and increment a integer value if you collection doesn't have any integers add one to it and then increment it.

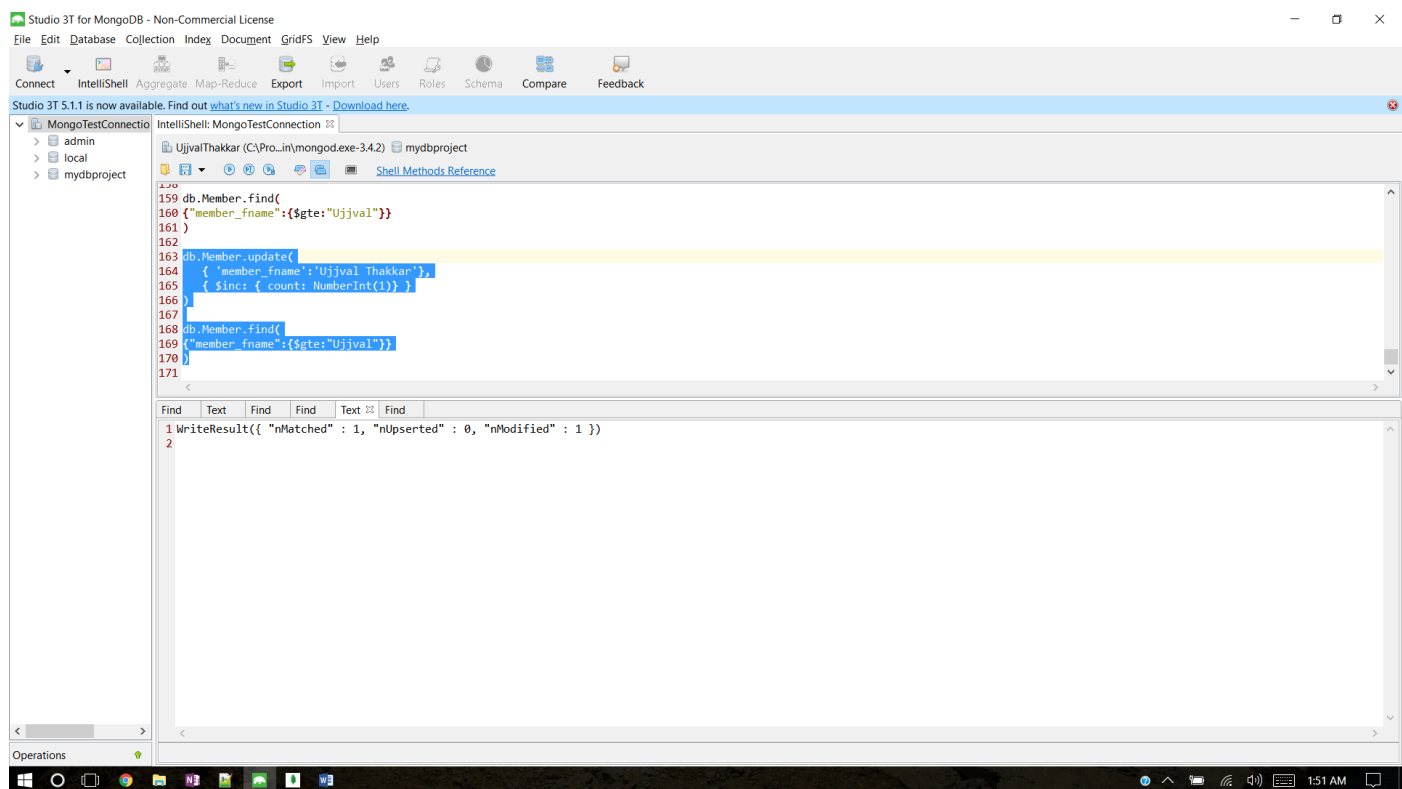
```
db.Member.update(  
  { 'member_fname':'Ujjval Thakkar'},  
  { $inc: { count: NumberInt(1)} }  
)
```

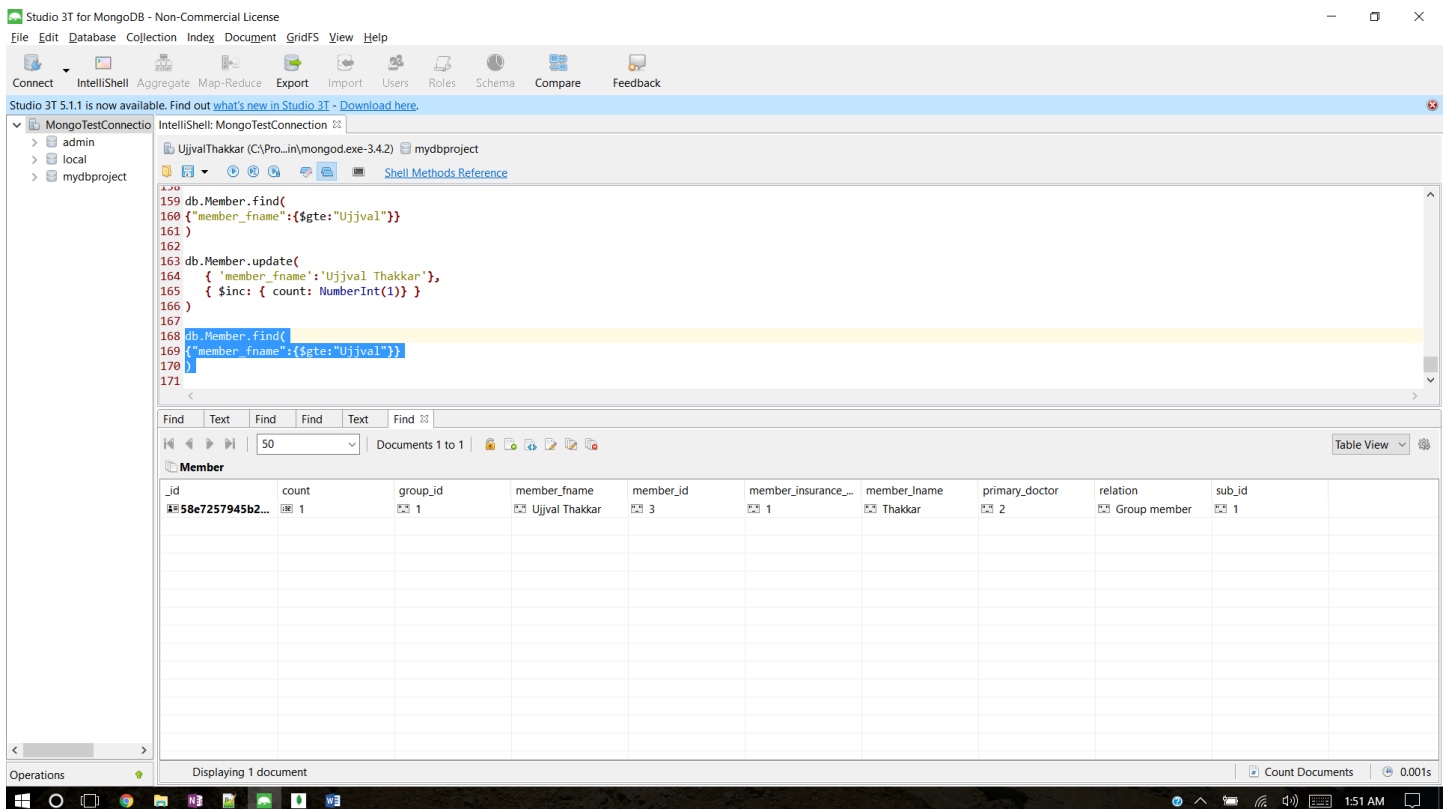
```
db.Member.find(  
{"member_fname":{$gte:"Ujjval"}}  
)
```

Explanation:

The first query, increments and adds an Integer value '1' to the existing integer value of 'count', which is updated from 0 to 1.

The second query, Finds a member with name >= "Ujjval" and confirms the increment update.
\$gte => Greater than or equal to (>=)





Explain the difference between save and insert in mongo

- The save() method replaces the existing document with the new document passed in the save() method.
- To insert data into MongoDB collection, you need to use MongoDB's insert() or save() method.
- To insert the document you can use db.post.save(document) also. If you don't specify _id in the document then save() method will work same as insert() method. If you specify _id then it will replace whole data of document containing _id as specified in save() method.
- If a document does not exist with the specified _id value, the save() method performs an insert with the specified fields in the document.
- If a document exists with the specified _id value, the save() method performs an update, replacing all field in the existing record with the fields from the document.
- For save, If the document contains _id, it will upsert querying the collection on the _id field, If not, it will insert.
 - **"UPSERT" definition.** "UPSERT" is a DBMS feature that allows a DML statement's author to atomically either insert a row, or on the basis of the row already existing, UPDATE that existing row instead, while safely giving little to no further thought to concurrency.