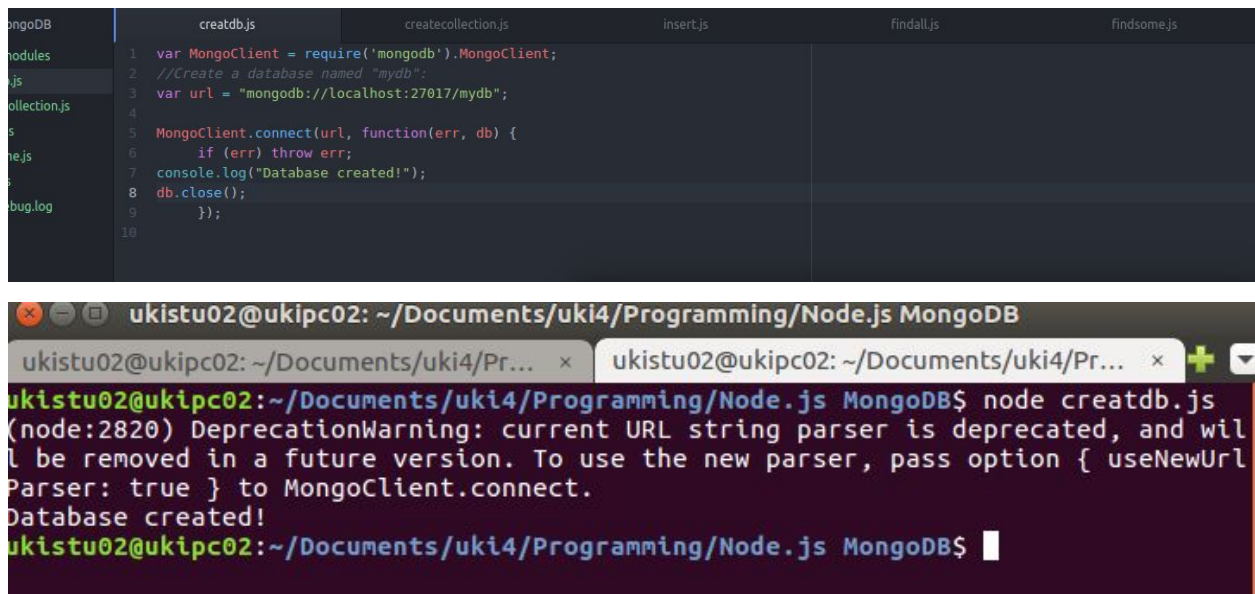


Node.js Exercise 2

Node.js MongoDB

1. Create a MongoClient object, then specify a connection URL with the correct ip address and the name of the database you want to create.

1) Create a Database called music.



The image shows a code editor with a file explorer on the left and a terminal window below it. The code editor has tabs for 'creatdb.js', 'createcollection.js', 'insert.js', 'findall.js', and 'findsome.js'. The 'creatdb.js' tab is active, showing the following code:

```
1 var MongoClient = require('mongodb').MongoClient;
2 //Create a database named "mydb":
3 var url = "mongodb://localhost:27017/mydb";
4
5 MongoClient.connect(url, function(err, db) {
6   if (err) throw err;
7   console.log("Database created!");
8   db.close();
9 });
10
```

The terminal window shows the command prompt for 'ukistu02@ukipc02: ~/Documents/uki4/Programming/Node.js MongoDB'. The user has run 'node creatdb.js', which outputs a deprecation warning and 'Database created!'. The prompt is now ready for the next command.

2) Create a collection called song details.

```
ukistu02@ukipc02: ~/Documents/uki4/Programming/Node.js MongoDB
ukistu02@ukipc02: ~/Documents/uki4/Pr... x ukistu02@ukipc02: ~/Documents/uki4/Pr... x
ukistu02@ukipc02:~/Documents/uki4/Programming/Node.js MongoDB$ node creatdb.js
(node:2820) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
Database created!
ukistu02@ukipc02:~/Documents/uki4/Programming/Node.js MongoDB$ node createcollection.js
(node:2974) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
Collection created!
ukistu02@ukipc02:~/Documents/uki4/Programming/Node.js MongoDB$
```

```
MongoDB      creatdb.js      createcollection.js      insert.js      fi
modules
db.js
ecollection.js
l.js
ome.js
.js
debug.log

1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("music");
7    dbo.createCollection("songdetails", function(err, res) {
8      if (err) throw err;
9      console.log("Collection created!");
10     db.close();
11   });
12 });
13
```

3) Create the above 5 song documents.

```

Number of documents inserted: 5
ukistu02@ukipc02:~/Documents/uki4/Programming/Node.js MongoDB$ nodemon insert.js

[nodemon] 1.11.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node insert.js`
(node:3261) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser:
Parser: true } to MongoClient.connect.
Number of documents inserted: 5
[nodemon] clean exit - waiting for changes before restart

```

Documents/uki4/Programming/Node.js MongoDB — Atom

creatdb.js	createcollection.js	insert.js	findall.js	findsome.js
<pre> 1 var MongoClient = require('mongodb').MongoClient; 2 var url = "mongodb://localhost:27017/"; 3 4 MongoClient.connect(url, function(err, db) { 5 if (err) throw err; 6 var dbo = db.db("music"); 7 var myobj = [8 { 9 Songname : "Thaniye Thananthaniye", flim: "Rhythm", Musicdirector: "A.R.Rahman", Singer: "Shankar mahadevan" 10 }, 11 { 12 Songname: "Evano oruvan", flim: "Alai payuthe", Musicdirector: "A.R.Rahman", Singer: "Swarnalatha" 13 }, 14 { 15 Songname: "Roja poonthooddam", flim: "Kannukkul nilavu", Musicdirector : "Ilaiyaraja", Singer: "unnikrishnan, Anuradha Sriram" 16 }, 17 { 18 Songname: "Vennilave venilave vinnaithaandi", flim: "Minsarak kanavu", Musicdirector: "A.R.Rahman", Singer: "Hariharan, Sadhana Sargam" 19 }, 20 { 21 Songname: "Sollamal thottu chellum thendral", flim: "Dheena", Musicdirector: "Yuvan Shankar Raja", Singer: "Hariharan" 22 } 23]; 24 25 26 dbo.collection("songdetails").insertMany(myobj, function(err, res) { 27 if (err) throw err; 28 console.log("Number of documents inserted: " + res.insertedCount); 29 db.close(); 30 }); 31 }); 32 </pre>				

4) List all documents created.

```
ukistu02@ukipc02: ~/Documents/uki4/Programming/Node.js MongoDB
ukistu02@ukipc02: ~/Documents/uki4/Pr... x ukistu02@ukipc02: ~/Documents/uki4/Pr... x
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node findall.js`
(node:3476) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
[ { _id: 5c383c8ea2164ac1a0a57478,
  'Song Name': 'Thaniye Thananthaniye',
  Film: 'Rhythm',
  'Music Director': 'A.R.Rahman',
  Singer: 'Shankar mahadevan' },
  { _id: 5c383db2a2164ac1a0a57479,
  'Song Name': 'Evano Oruvan',
  Film: 'Alai Payuthey',
  'Music Director': 'A.R.Rahman',
  Singer: 'Swarnalatha' },
  { _id: 5c38441fa2164ac1a0a5747e,
  'Song Name': 'Maruvarththai Peshathe',
  Film: 'Ennai Nokki Pajum Thotta',
  'Music Director': 'Dhanush',
  Singer: 'Sidsriram' },
  { _id: 5c3848bba2164ac1a0a57482,
  SongName: 'Roja Poonthottam',
  Film: 'Kannukul Nilavu',
```

```
DB      creatdb.js  createcollection.js  insert.js  findall.js  findsome.js
les
tion.js
log
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("music");
7    dbo.collection("songdetails").find({}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     dbo.close();
11   });
12 });
13
```

5) List A.R.Rahman's songs.


```

kistu02@ukipc02:~/Documents/uki4/Programming/Node.js MongoDB$ nodemon findsome.js
nodemon] 1.11.0
nodemon] to restart at any time, enter `rs`
nodemon] watching: *.*
nodemon] starting `node findsome.js`
node:3693) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
]
nodemon] clean exit - waiting for changes before restart
nodemon] restarting due to changes...
nodemon] starting `node findsome.js`
node:3750) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
]

```

findsome.js — ~/Documents/uki4/Programming/Node.js MongoDB — Atom

```

modules
b.js
collection.js
js
me.js
is
debug.log
1  var MongoClient = require('mongodb').MongoClient;
2
3  var url = "mongodb://localhost:27017/music";
4
5  MongoClient.connect(url, function(err, db) {
6    if (err) throw err;
7    var dbo = db.db("music");
8    var query = { Musicdirector: "A.R.Rahman" };
9    dbo.collection("songdetails").find(query).toArray(function(err, result) {
10     if (err) throw err;
11     console.log(result);
12     db.close();
13   });
14 });
15

```

6) List A.R.Rahman's songs sung by Unnikrishnan.

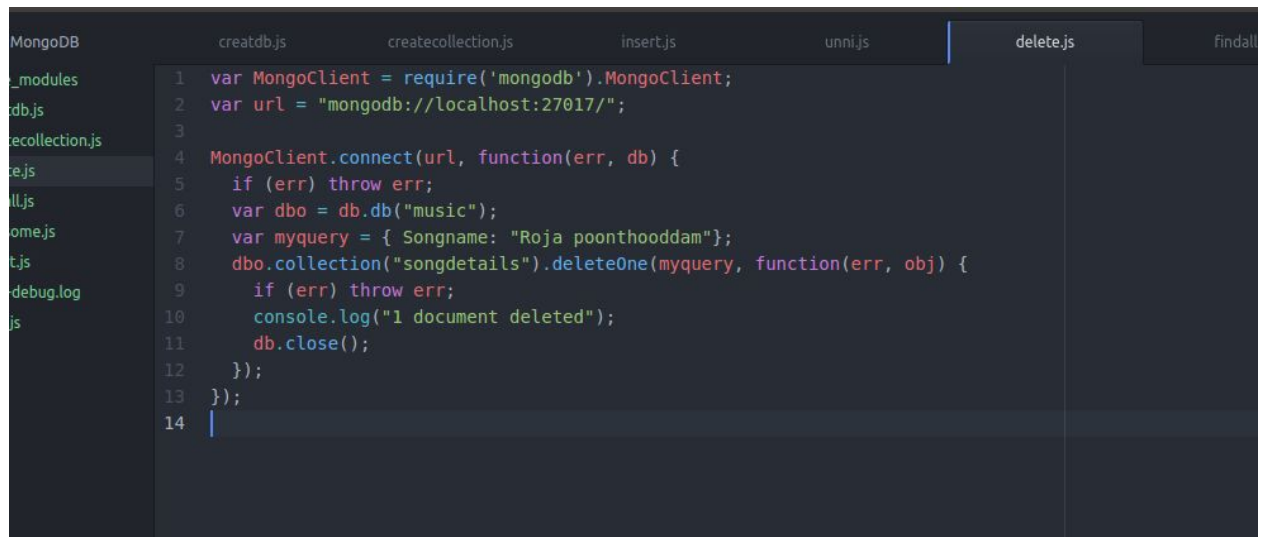
```
creatdb.js    createcollection.js    insert.js    unni.js    findall.js

1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("music");
7    /*Return only the documents with the address "Park Lane 38":*/
8    var query = { Musicdirector: "Ilaiyaraja", Singer: "unnikrishnan,Anuradha Sriram" };
9    dbo.collection("songdetails").find(query).toArray(function(err, result) {
10      if (err) throw err;
11      console.log(result);
12      db.close();
13    });
14  });
15
```

```
[ ]
ukistu02@ukipc02:~/Documents/uki4/Programming/Node.js MongoDB$ nodemon unni.js
[nodemon] 1.11.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node unni.js`
(node:4371) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser:
true } to MongoClient.connect.
[ ]
[nodemon] clean exit - waiting for changes before restart
```

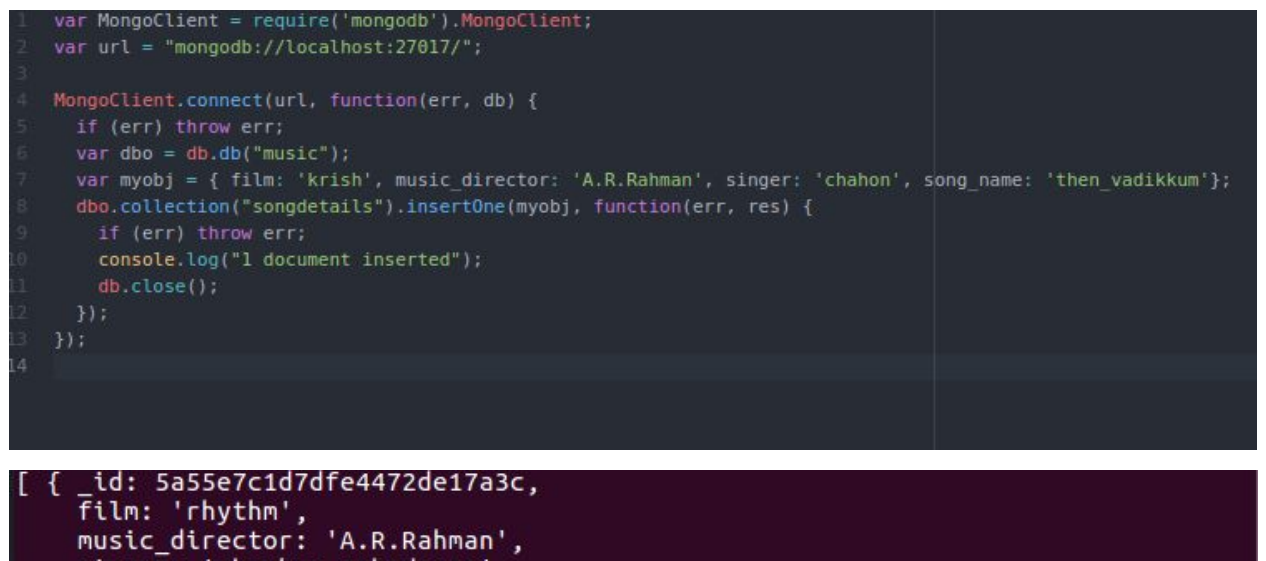
7) Delete the song which you don't like.

```
ukistu02@uklpc02:~/Documents/ukl4/Programming/Node.js MongoDB$ nodemon delete.js
[nodemon] 1.11.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node delete.js`
(node:4924) DeprecationWarning: current URL string parser is deprecated, and will
be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
1 document deleted
[nodemon] clean exit - waiting for changes before restart
```



```
MongoDB      creatdb.js      createcollection.js      insert.js      unni.js      delete.js      findall.js
_modules
db.js
ecollection.js
e.js
ll.js
ome.js
t.js
debug.log
js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("music");
7    var myquery = { Songname: "Roja poonthooddam"};
8    dbo.collection("songdetails").deleteOne(myquery, function(err, obj) {
9      if (err) throw err;
10     console.log("1 document deleted");
11     db.close();
12   });
13 });
14 |
```

8) Add new song which is your favourite.



```
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("music");
7    var myobj = { film: 'krish', music_director: 'A.R.Rahman', singer: 'chahon', song_name: 'then_vadikkum'};
8    dbo.collection("songdetails").insertOne(myobj, function(err, res) {
9      if (err) throw err;
10     console.log("1 document inserted");
11     db.close();
12   });
13 });
14 |
```

```
[ { _id: 5a55e7c1d7dfe4472de17a3c,
  film: 'rhythm',
  music_director: 'A.R.Rahman',
  singer: 'chahon',
  song_name: 'then_vadikkum' }
```

9) List Songs sung by Hariharan from Minsara kanavu film.

```
var MongoClient = require('mongodb').MongoClient;
var url = "mongodb://localhost:27017/";

MongoClient.connect(url, function(err, db) {
  if (err) throw err;
  var dbo = db.db("music");
  dbo.collection("songdetails").find({singer: "hariharan", film: "dheena"},{_id: false, name: true}).toArray(function(err, result) {
    if (err) throw err;
    console.log(result);
    db.close();
  });
});
```

```
S
[ { _id: 5a55e7c1d7dfe4472de17a40,
  film: 'dheena',
  music_director: 'yuvan_shankar_raja',
  singer: 'hariharan',
  song_name: 'sollamal thottu chellum thendral' } ]
```

10) List out the singers' names in your document.

```
creatdb.js  createcollection.js  insert.js  unni.js  delete.js  favourite.js  list.js

1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("music");
7    dbo.collection("songdetails").find({}, { _id:0,Singer:1,Songname:0, film:0,Musicedirector:0 });
8    if (err) throw err;
9    console.log(result);
10   db.close();
11  });
12  });
13
```

```
node:internal/modules/cjs/loader:114
(node:15329) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
[ { _id: 5c51b4c7899ca338f3edaa71,
  Songname: 'Thaniye Thananthaniye',
  film: 'Rhythm',
  Musicedirector: 'A.R.Rahman',
  Singer: 'Shankar mahadevan' },
  { _id: 5c51b4c7899ca338f3edaa72,
  Songname: 'Evano oruvan',
  film: 'Alal payuthe',
  Musicedirector: 'A.R.Rahman',
  Singer: 'Swarnalatha' },
  { _id: 5c51b4c7899ca338f3edaa74,
  Songname: 'Vennilave venilave vinnalthaandl',
  film: 'Minsarak kanavu',
  Musicedirector: 'A.R.Rahman',
  Singer: 'Hariharan,Sadhana Sargan' },
  { _id: 5c51b4c7899ca338f3edaa75,
  Songname: 'Sollamal thottu chellum thendral',
  film: 'dheena',
  Musicedirector: 'Yuvan Shankar Raja',
  Singer: 'Hariharan' },
  { _id: 5c51b8e22bb4df3ac2f79c00,
  Songname: 'kannala kane',
  film: 'visuvasan',
  Musicedirector: 'Ilaiyaraaja',
  Singer: 'Hariharan' } ]
```


2. Complete mongodb ex1 and mongodb ex2 with the MongoClient object you just created in Node.js.

1) Create a Database called student

```
File Edit View Selection Find Packages Help
qus1.js qus2.js qus3.js qus4.js qus5.js qus6.js qus7.js qus8.js qus9.js qus10.js qus11.js qus12.js qus13.js qus14.js qus15.js
1 var MongoClient = require('mongodb').MongoClient;
2 //Create a database named "mydb";
3 var url = "mongodb://localhost:27017/student";
4
5 MongoClient.connect(url, function(err, db) {
6   if (err) throw err;
7   console.log("Database created!");
8   db.close();
9 });
10
```

2) Create a collection called student marks

```
File Edit View Selection Find Packages Help
qus1.js qus2.js qus3.js qus4.js qus5.js qus6.js qus7.js qus8.js qus9.js qus10.js qus11.js qus12.js qus13.js qus14.js qus15.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/student";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var uki = db.db("student");
7   uki.createCollection("student_marks", function(err, res) {
8     if (err) throw err;
9     console.log("Collection created!");
10    db.close();
11  });
12 });
13
```

3) Create the documents listed in above table.

```
qus1.js qus2.js qus3.js qus4.js qus5.js qus6.js qus7.js qus8.js qus9.js qus10.js qus11.js qus12.js qus13.js qus14.js qus15.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("student");
7   var myobj = [
8     { name: 'Mala', maths_marks: 45, english_marks: 53, science_marks: 72},
9     { name: 'Venu', maths_marks: 80, english_marks: 75, science_marks: 85},
10    { name: 'Kala', maths_marks: 32, english_marks: 46, science_marks: 53},
11    { name: 'Aruli', maths_marks: 78, english_marks: 85, science_marks: 88},
12    { name: 'Shayu', maths_marks: 80, english_marks: 76, science_marks: 65},
13    { name: 'Kumaran', maths_marks: 32, english_marks: 73, science_marks: 84},
14    { name: 'Lucky', maths_marks: 66, english_marks: 90, science_marks: 45},
15    { name: 'Gva', maths_marks: 71, english_marks: 75, science_marks: 56},
16    { name: 'Raam', maths_marks: 41, english_marks: 65, science_marks: 88}
17  ];
18  dbo.collection("student_marks").insertMany(myobj, function(err, res) {
19    if (err) throw err;
20    console.log("Number of documents inserted: " + res.insertedCount);
21    db.close();
22  });
23 });
24
```

4) Increase the maths marks of Mala by 6 marks

```

qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    var myquery = {name: "Mala"};
8    var newvalues = {$inc: {marks: 6}};
9    dbo.collection("student_marks").updateOne(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12     db.close();
13   });
14 });
15

```

5) List the names of students who got more than 50 marks in Maths Subject.

```

qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    dbo.collection("student_marks").find({maths_marks: {$gt :50}, {name: true, _id: false}}).toArray(function(err, result) {
8    if (err) throw err;
9    console.log(result);
10   db.close();
11 });
12 });
13

```

6)Add a new column(field) for Average for all students.

```

qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    var myquery = {};
8    var newvalues = {$set: {Average: ""}};
9    dbo.collection("student_marks").updateMany(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log(res.result.nModified + "document(s) updated");
12     db.close();
13   });
14 });
15

```

7) Update Marks_Science=75 to Lucky .

```

qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    var myquery = {};
8    var newvalues = {$set: {science_marks: 75}};
9    dbo.collection("student_marks").updateMany(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12     db.close();
13   });
14 });
15

```

8) List the names who got more than 50 marks in all subjects.

```
qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    dbo.collection("student_marks").find({maths_marks: {$gt :50}, english_marks: {$gt :50}, science_marks: {$gt :50}, {name: true, _id: false}}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 });
13
```

9) List the names who got less than 50 marks in Maths subject and more than 50 marks in English.

```
File Edit View Selection Find Packages Help
qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    dbo.collection("student_marks").find({maths_marks: {$lt :50}, english_marks: {$gt :50}, {name: true, _id: false}}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 });
13
```

10) List the names who got less than 40 in both Maths and Science.

```
qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    dbo.collection("student_marks").find({maths_marks: {$lt :40}, science_marks: {$lt :40}, {name: true, _id: false}}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 });
13
```

11) Remove Science column/field for Raam.

```
File Edit View Selection Find Packages Help
qus1.js qus2.js qus3.js qus4.js qus5.js qus6.js qus7.js qus8.js qus9.js qus10.js qus11.js qus12.js qus13.js qus14.js qus15.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("student");
7   var myquery = {name: 'Raam'}
8   var newvalues = {$set: {science_marks: 75}};
9   dbo.collection("student_marks").updateMany(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12     db.close();
13   });
14 });
15
```

12) Update John's Math mark as 87 and English mark as 23, if john not available upsert.

```
qus1.js qus2.js qus3.js qus4.js qus5.js qus6.js qus7.js qus8.js qus9.js qus10.js qus11.js qus12.js qus13.js qus14.js qus15.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("student");
7   var myobj = { name: 'John', maths_marks: 87, english_marks: 27}
8   dbo.collection("student_marks").insertOne(myobj, function(err, res) {
9     if (err) throw err;
10    console.log("1 documents inserted");
11    db.close();
12  });
13 });
14
```

13) Rename the english_marks column/field for John to science_marks.

```
qus1.js qus2.js qus3.js qus4.js qus5.js qus6.js qus7.js qus8.js qus9.js qus10.js qus11.js qus12.js qus13.js qus14.js qus15.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("student");
7   var myquery = {name: 'John'}
8   var newvalues = {$rename: {english_marks : "science_marks"}};
9   dbo.collection("student_marks").updateOne(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12     db.close();
13   });
14 });
15
```

14) Remove Kumaran's document from collection


```
qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    var myquery = {name: 'Kumaran'}
8    dbo.collection("student_marks").deleteOne(myquery, newvalues, function(err, res) {
9      if (err) throw err;
10     console.log("1 document deleted");
11     db.close();
12   });
13 });
14
```

15) Find Kala's or Aruli's math_marks and science_marks.

```
qus1.js  qus2.js  qus3.js  qus4.js  qus5.js  qus6.js  qus7.js  qus8.js  qus9.js  qus10.js  qus11.js  qus12.js  qus13.js  qus14.js  qus15.js
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("student");
7    dbo.collection("student_marks").find({$or: [{name: 'Aruli'}, {name: 'Kala'}]}, {maths_marks: true, science_marks: true}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 });
13
```