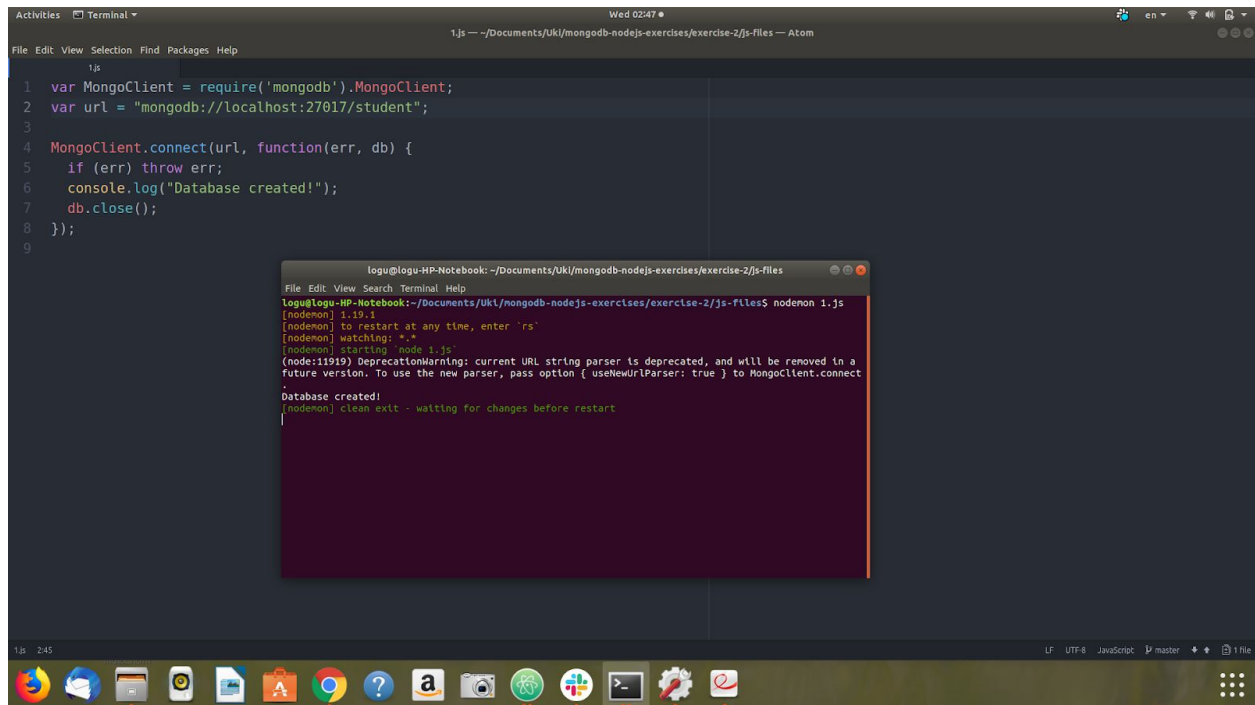


## 1) Create a Database called student

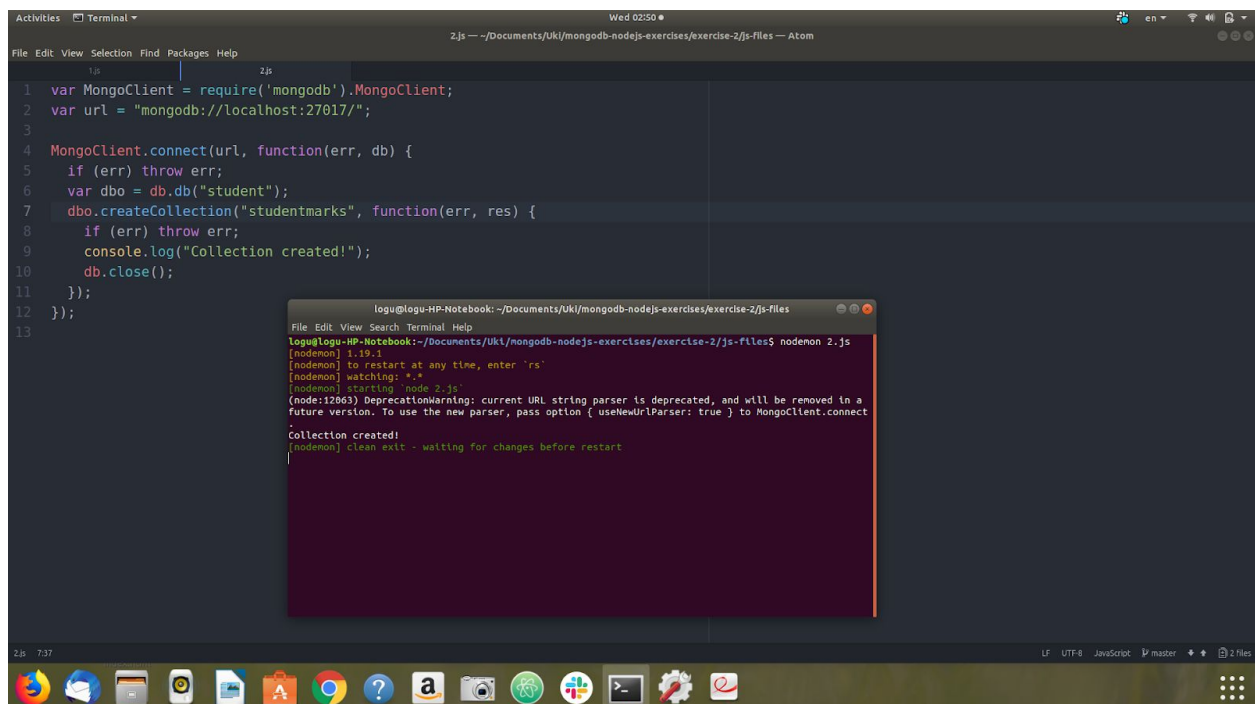


The screenshot shows an Atom editor window with a JavaScript file named `1.js`. The code connects to a MongoDB instance at `localhost:27017` and creates a database named `student`. A terminal window in the foreground shows the output of running `node 1.js`, which includes a deprecation warning and the message "Database created!".

```
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   console.log("Database created!");
7   db.close();
8 });
9
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 1.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting 'node 1.js'
(node:11919) 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!
[nodemon] clean exit - waiting for changes before restart
```

## 2) Create a collection called studentmarks

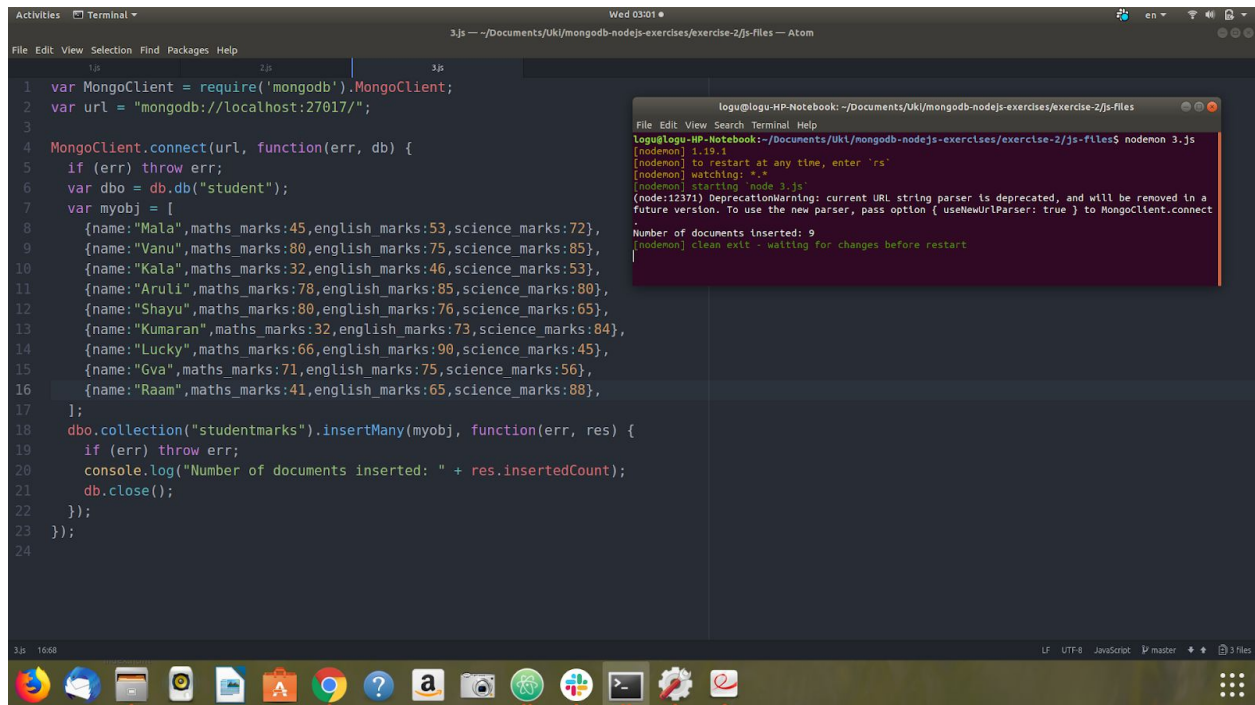


The screenshot shows an Atom editor window with a JavaScript file named `2.js`. The code connects to a MongoDB instance at `localhost:27017`, creates a database named `student`, and then creates a collection named `studentmarks`. A terminal window in the foreground shows the output of running `node 2.js`, which includes a deprecation warning and the message "Collection created!".

```
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.createCollection("studentmarks", function(err, res) {
8     if (err) throw err;
9     console.log("Collection created!");
10    db.close();
11  });
12 });
13
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 2.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting 'node 2.js'
(node:12063) 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!
[nodemon] clean exit - waiting for changes before restart
```

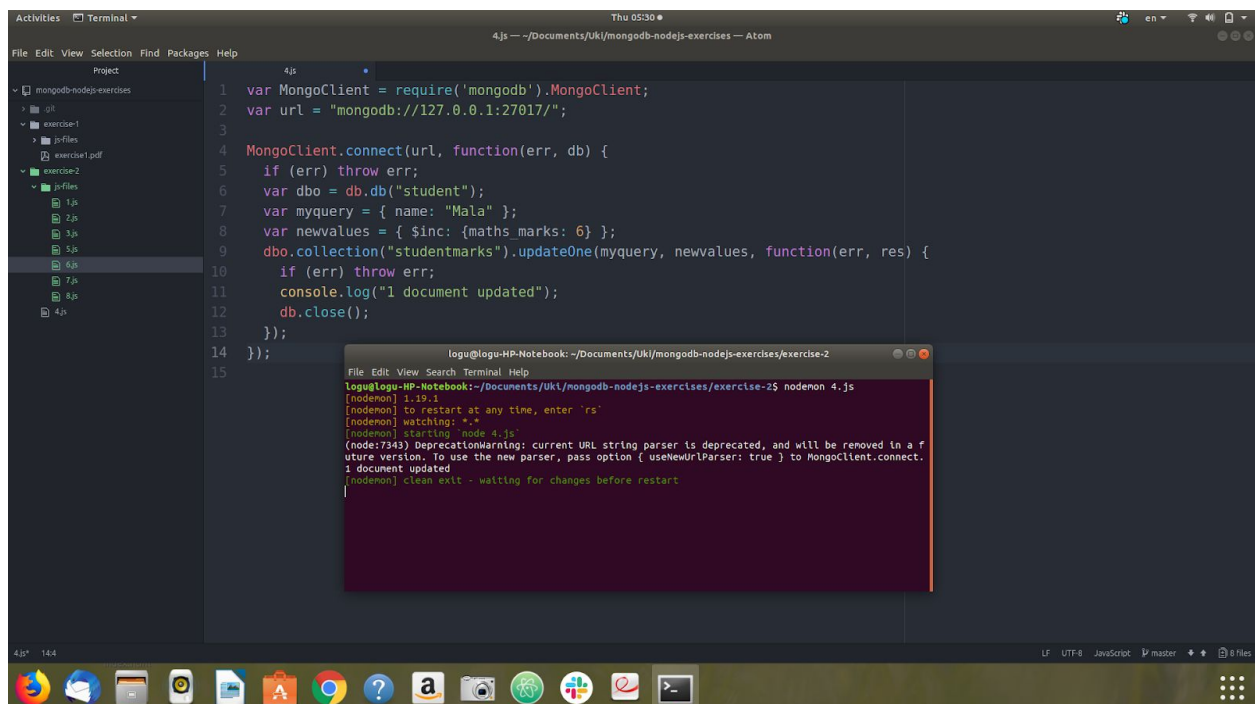
### 3) Create the documents listed in above table.



```
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:"Vanu",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:80},
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("studentmarks").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
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 3.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 3.js'
(node:12371) 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.
Number of documents inserted: 9
[nodemon] clean exit - waiting for changes before restart
```

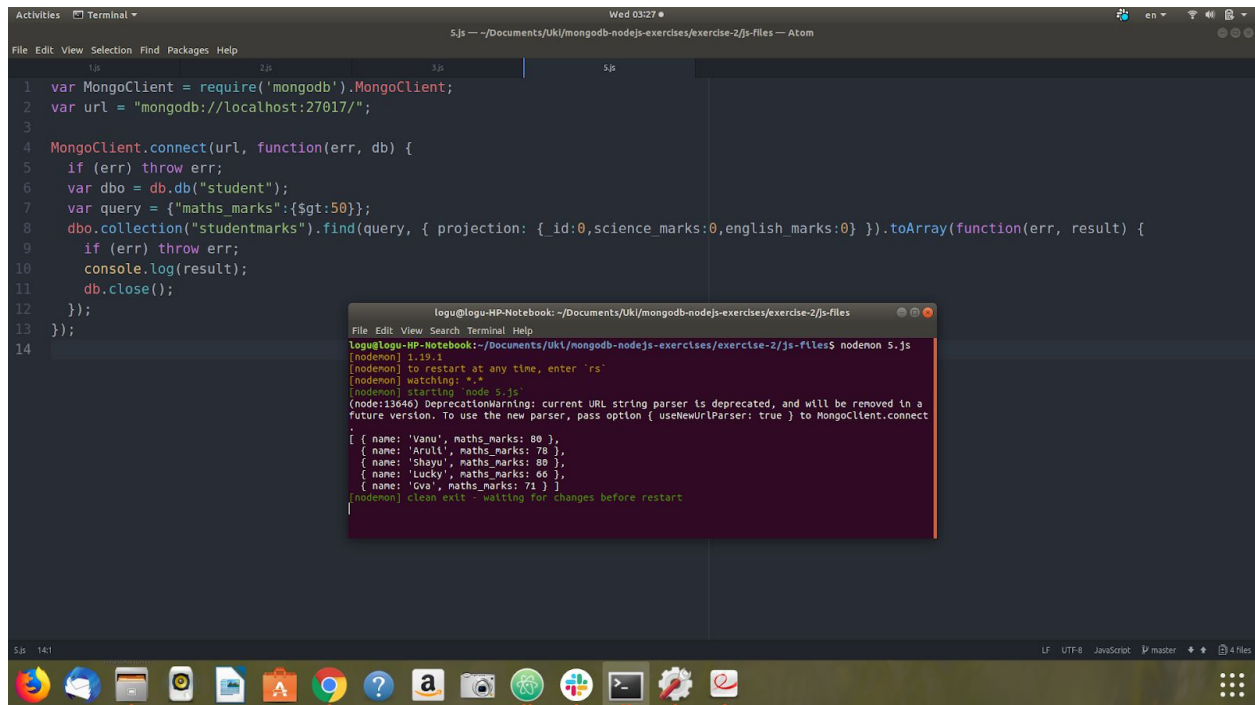
### 4) Increase the maths marks of Mala by 6 marks



```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://127.0.0.1:27017/";
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: {maths_marks: 6} };
9   dbo.collection("studentmarks").updateOne(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12     db.close();
13   });
14 });
15
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2$ node 4.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 4.js'
(node:7343) 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 updated
[nodemon] clean exit - waiting for changes before restart
```

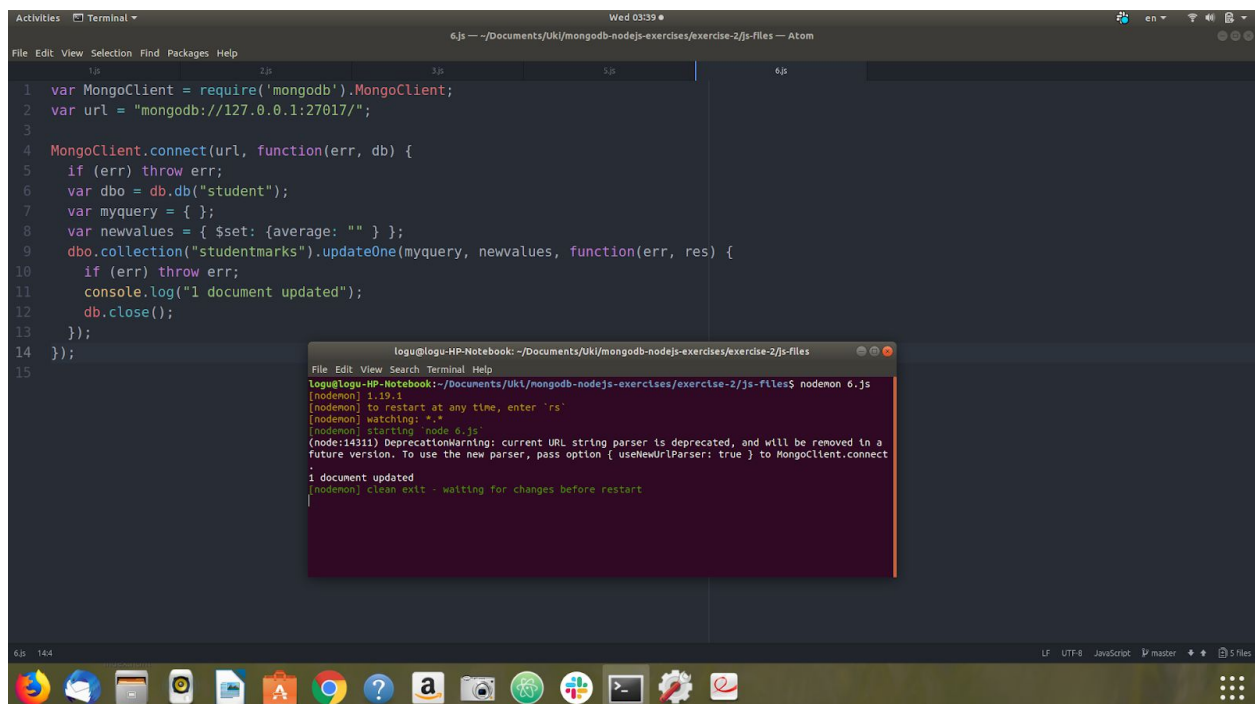
5) List the names of students who got more than 50 marks in Maths Subject.



```
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 query = {"maths_marks":{"$gt:50}};
8   dbo.collection("studentmarks").find(query, { projection: { _id:0,science_marks:0,english_marks:0 } }).toArray(function(err, result) {
9     if (err) throw err;
10    console.log(result);
11    db.close();
12  });
13 });
14
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 5.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 5.js'
(node:13646) 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
[ { name: 'Vanu', maths_marks: 80 },
  { name: 'Arul', maths_marks: 78 },
  { name: 'Shayu', maths_marks: 80 },
  { name: 'Lucky', maths_marks: 66 },
  { name: 'Gva', maths_marks: 71 } ]
[nodemon] clean exit - waiting for changes before restart
```

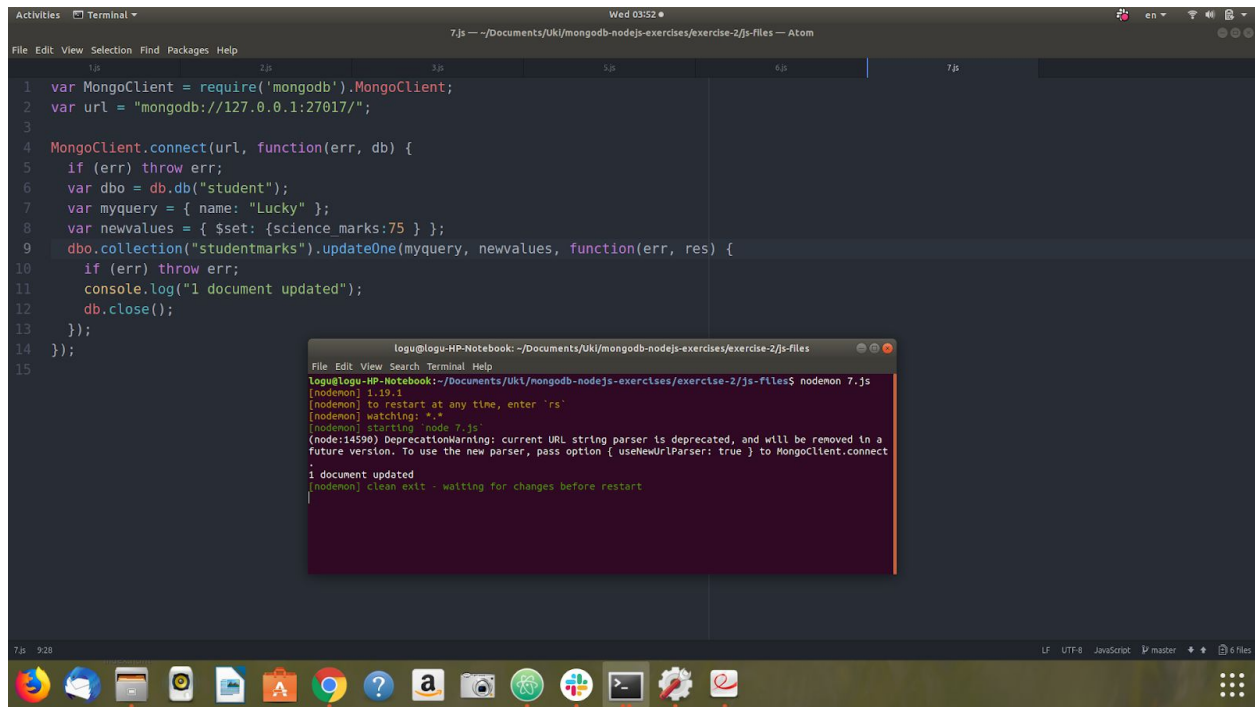
6) Add a new column(field) for Average for all students.



```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://127.0.0.1:27017/";
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("studentmarks").updateOne(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12     db.close();
13   });
14 });
15
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 6.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 6.js'
(node:14311) 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 updated
[nodemon] clean exit - waiting for changes before restart
```

7) Update Marks\_Science=75 to Lucky .

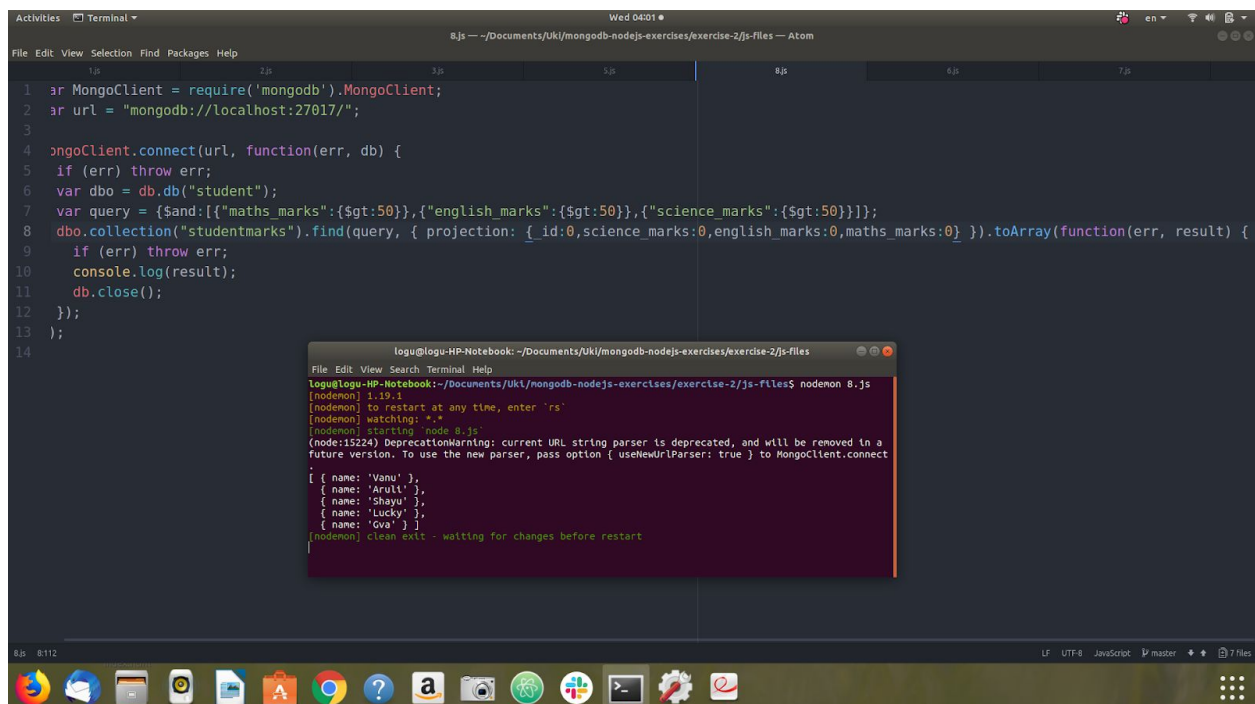


The screenshot shows an Atom editor with a JavaScript file named 7.js. The code connects to a MongoDB instance at 127.0.0.1:27017, selects the 'student' database, and updates the document where 'name' is 'Lucky' to set 'science\_marks' to 75. A terminal window in the foreground shows the command 'nodemon 7.js' being executed, with output indicating a successful update of 1 document.

```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://127.0.0.1:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("student");
7   var myquery = { name: "Lucky" };
8   var newvalues = { $set: {science_marks:75 } };
9   dbo.collection("studentmarks").updateOne(myquery, newvalues, function(err, res) {
10    if (err) throw err;
11    console.log("1 document updated");
12    db.close();
13  });
14 }
15
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ nodemon 7.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 7.js'
(node:14590) 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 updated
[nodemon] clean exit - waiting for changes before restart
```

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



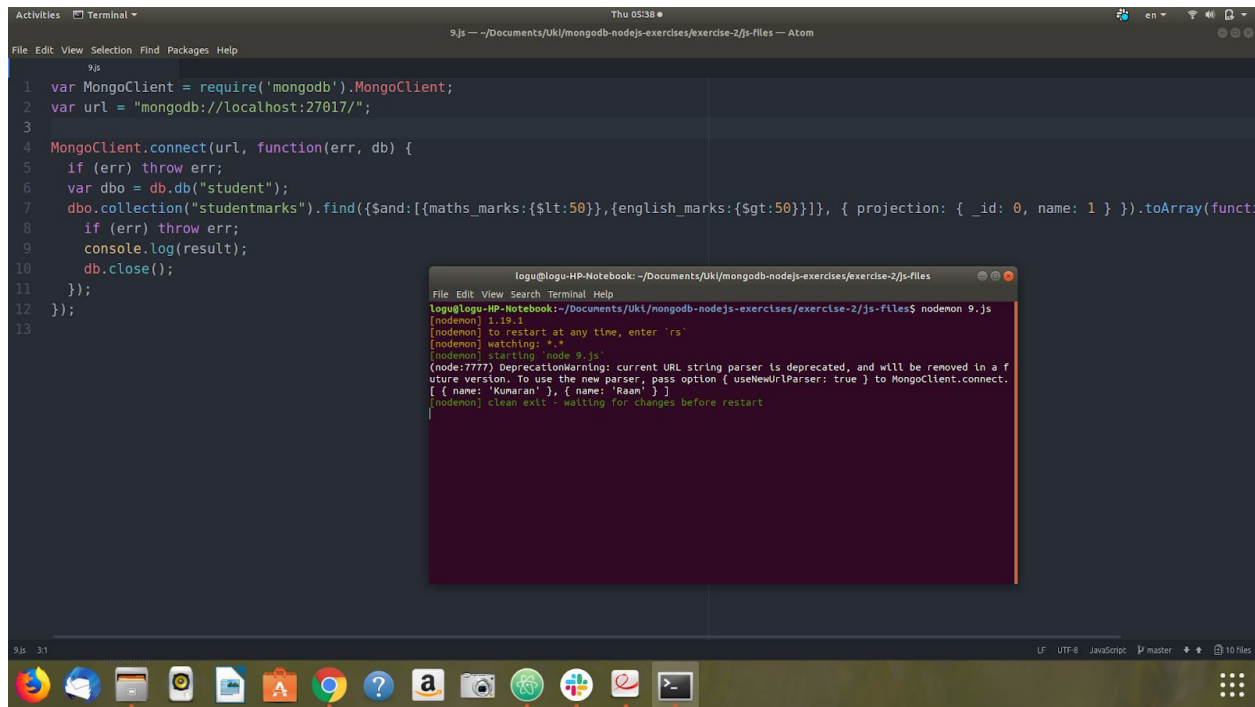
The screenshot shows an Atom editor with a JavaScript file named 8.js. The code connects to a MongoDB instance at localhost:27017, selects the 'student' database, and queries for documents where 'maths\_marks', 'english\_marks', and 'science\_marks' are all greater than 50. The results are printed to the console. A terminal window in the foreground shows the command 'nodemon 8.js' being executed, with output displaying the names of the students: Vanu, Arul, Shayu, Lucky, and Gva.

```
1 ar MongoClient = require('mongodb').MongoClient;
2 ar 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 query = {$and:[{"maths_marks":{$gt:50}}, {"english_marks":{$gt:50}}, {"science_marks":{$gt:50}}]};
8   dbo.collection("studentmarks").find(query, { projection: { _id:0, science_marks:0, english_marks:0, maths_marks:0 } }).toArray(function(err, result) {
9     if (err) throw err;
10    console.log(result);
11    db.close();
12  });
13 }
14
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ nodemon 8.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 8.js'
(node:15224) 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
[ { name: 'Vanu' },
  { name: 'Arul' },
  { name: 'Shayu' },
  { name: 'Lucky' },
  { name: 'Gva' } ]
[nodemon] clean exit - waiting for changes before restart
```



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

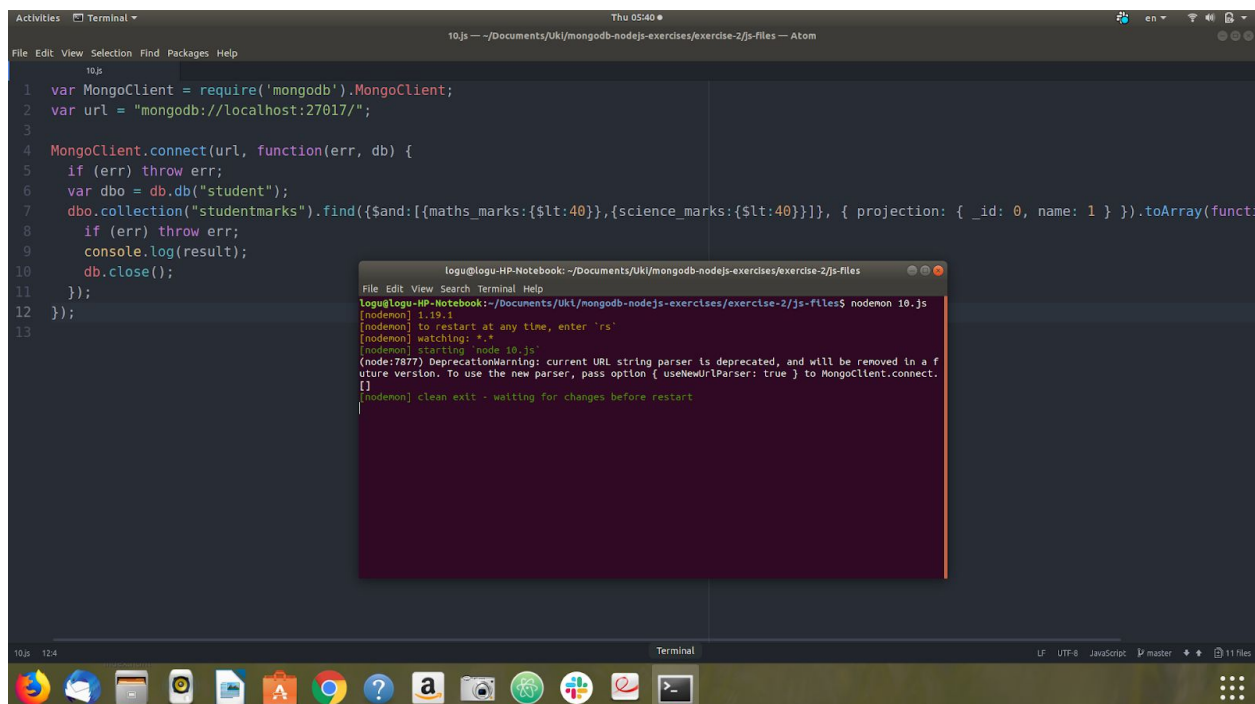


The screenshot shows the Atom editor with a JavaScript file named 9.js. The code connects to a MongoDB instance at localhost:27017 and queries the 'studentmarks' collection. The query filters for students with less than 50 marks in Maths and more than 50 marks in English, projecting the \_id and name. The results are logged to the console. A terminal window in the foreground shows the output of the query, displaying two names: 'Kumaran' and 'Raam'.

```
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("studentmarks").find({$and:[{maths_marks:{<50}}, {english_marks:{>50}}]}, { projection: { _id: 0, name: 1 } }).toArray(function(err, result) {
8     if (err) throw err;
9     console.log(result);
10    db.close();
11  });
12 });
13
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 9.js
[nodenon] 1.19.1
[nodenon] to restart at any time, enter 'rs'
[nodenon] watching: *.*
[nodenon] starting 'node 9.js'
(node:7777) 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.
[ { name: 'Kumaran' }, { name: 'Raam' } ]
[nodenon] clean exit - waiting for changes before restart
```

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

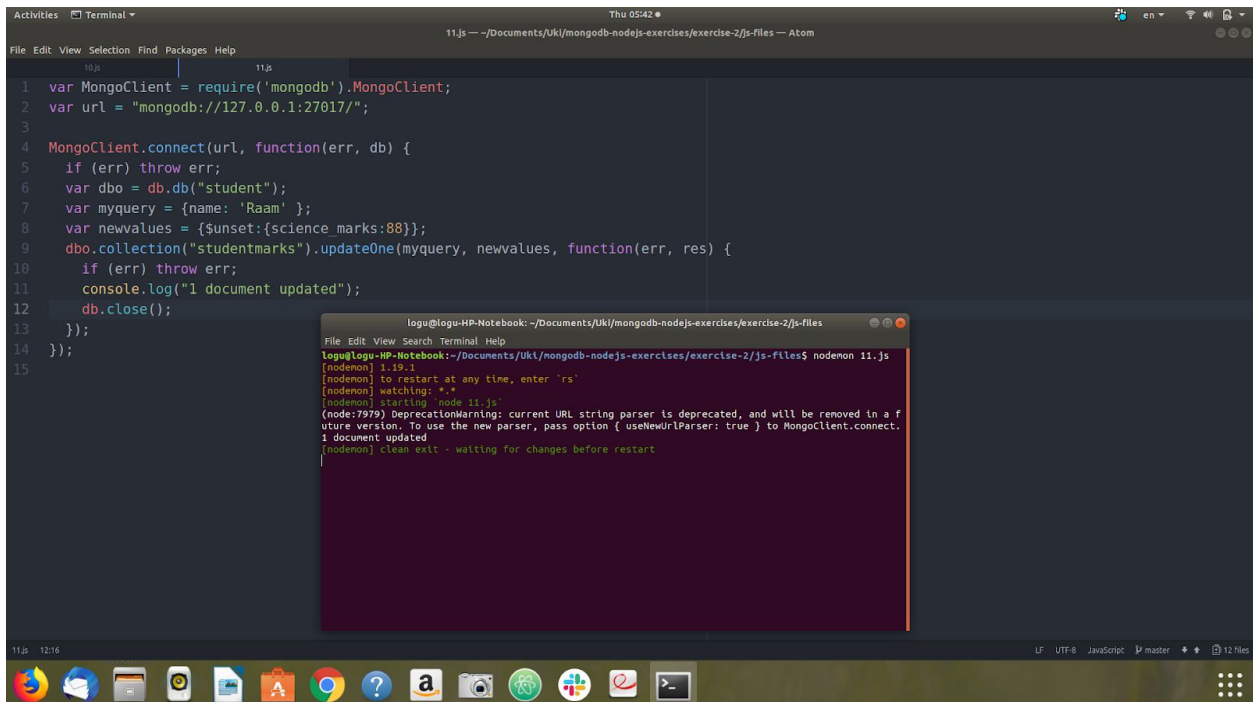


The screenshot shows the Atom editor with a JavaScript file named 10.js. The code connects to a MongoDB instance at localhost:27017 and queries the 'studentmarks' collection. The query filters for students with less than 40 marks in both Maths and Science, projecting the \_id and name. The results are logged to the console. A terminal window in the foreground shows the output of the query, displaying two names: 'Kumaran' and 'Raam'.

```
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("studentmarks").find({$and:[{maths_marks:{<40}}, {science_marks:{<40}}]}, { projection: { _id: 0, name: 1 } }).toArray(function(err, result) {
8     if (err) throw err;
9     console.log(result);
10    db.close();
11  });
12 });
13
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 10.js
[nodenon] 1.19.1
[nodenon] to restart at any time, enter 'rs'
[nodenon] watching: *.*
[nodenon] starting 'node 10.js'
(node:7877) 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.
[]
[nodenon] clean exit - waiting for changes before restart
```

## 11) Remove Science column/field for Raam

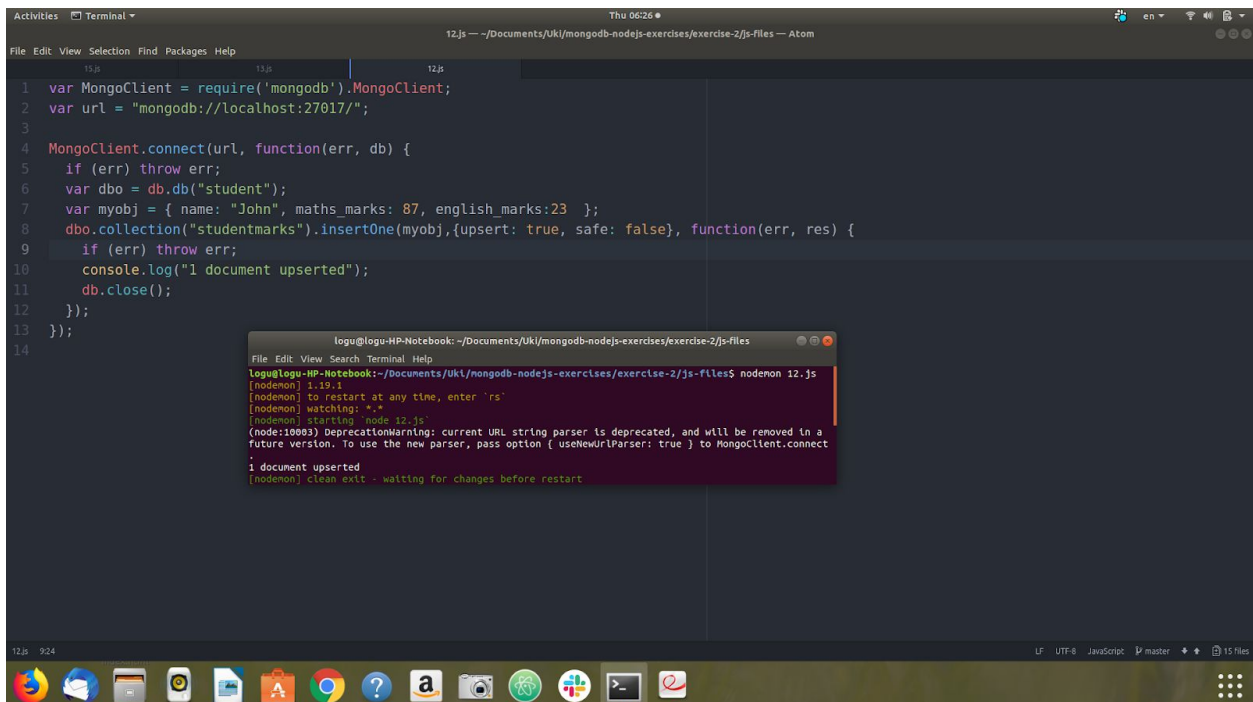


The screenshot shows a terminal window with a Node.js script being executed. The script connects to a MongoDB instance at 127.0.0.1:27017, updates a document in the 'studentmarks' collection, and logs the result. A second terminal window shows the output of the script, indicating that the document was successfully updated.

```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://127.0.0.1:27017/";
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 = {$unset:{science marks:88}};
9   dbo.collection("studentmarks").updateOne(myquery, newvalues, function(err, res) {
10     if (err) throw err;
11     console.log("1 document updated");
12   });
13   db.close();
14 });
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 11.js
[nodenon] 1.19.1
[nodenon] to restart at any time, enter 'rs'
[nodenon] watching: *.*
[nodenon] starting 'node 11.js'
(node:7979) 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 updated
[nodenon] clean exit - waiting for changes before restart
```

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

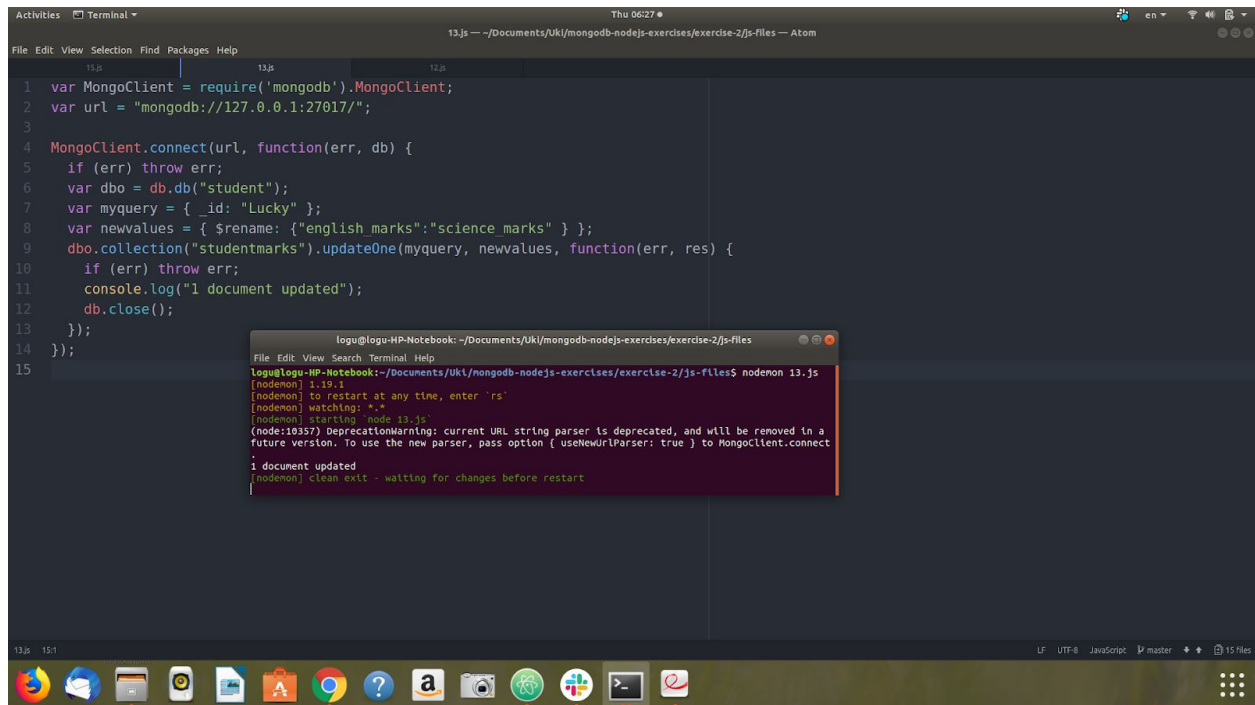


The screenshot shows a terminal window with a Node.js script being executed. The script connects to a MongoDB instance at localhost:27017, inserts or updates a document in the 'studentmarks' collection, and logs the result. A second terminal window shows the output of the script, indicating that the document was successfully upserted.

```
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:23 };
8   dbo.collection("studentmarks").insertOne(myobj,{upsert: true, safe: false}, function(err, res) {
9     if (err) throw err;
10    console.log("1 document upserted");
11    db.close();
12  });
13  });
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ node 12.js
[nodenon] 1.19.1
[nodenon] to restart at any time, enter 'rs'
[nodenon] watching: *.*
[nodenon] starting 'node 12.js'
(node:10003) 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 upserted
[nodenon] clean exit - waiting for changes before restart
```

### 13) Rename the english\_marks column/field for John to science\_marks

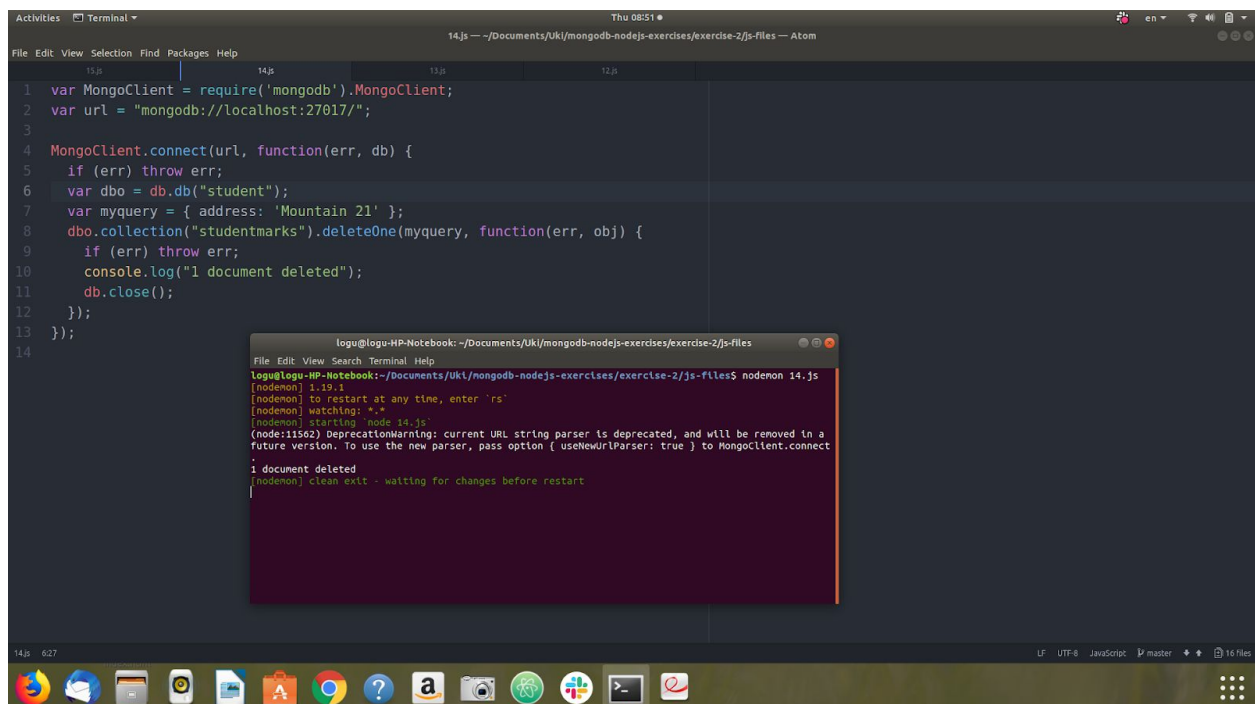


The screenshot shows an Atom editor window with a JavaScript file named 13.js. The code connects to a MongoDB instance at localhost:27017, updates the 'english\_marks' field to 'science\_marks' for the document with '\_id: "Lucky"', and logs the result. A terminal window in the foreground shows the command 'nodemon 13.js' being executed, with output indicating a successful update of 1 document.

```
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 = { _id: "Lucky" };
8   var newvalues = { $rename: { "english_marks": "science_marks" } };
9   dbo.collection("studentmarks").updateOne(myquery, newvalues, function(err, res) {
10    if (err) throw err;
11    console.log("1 document updated");
12    db.close();
13  });
14 });
15
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ nodemon 13.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 13.js'
(node:10357) 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 updated
[nodemon] clean exit - waiting for changes before restart
```

### 14) Remove Kumaran's document from collection

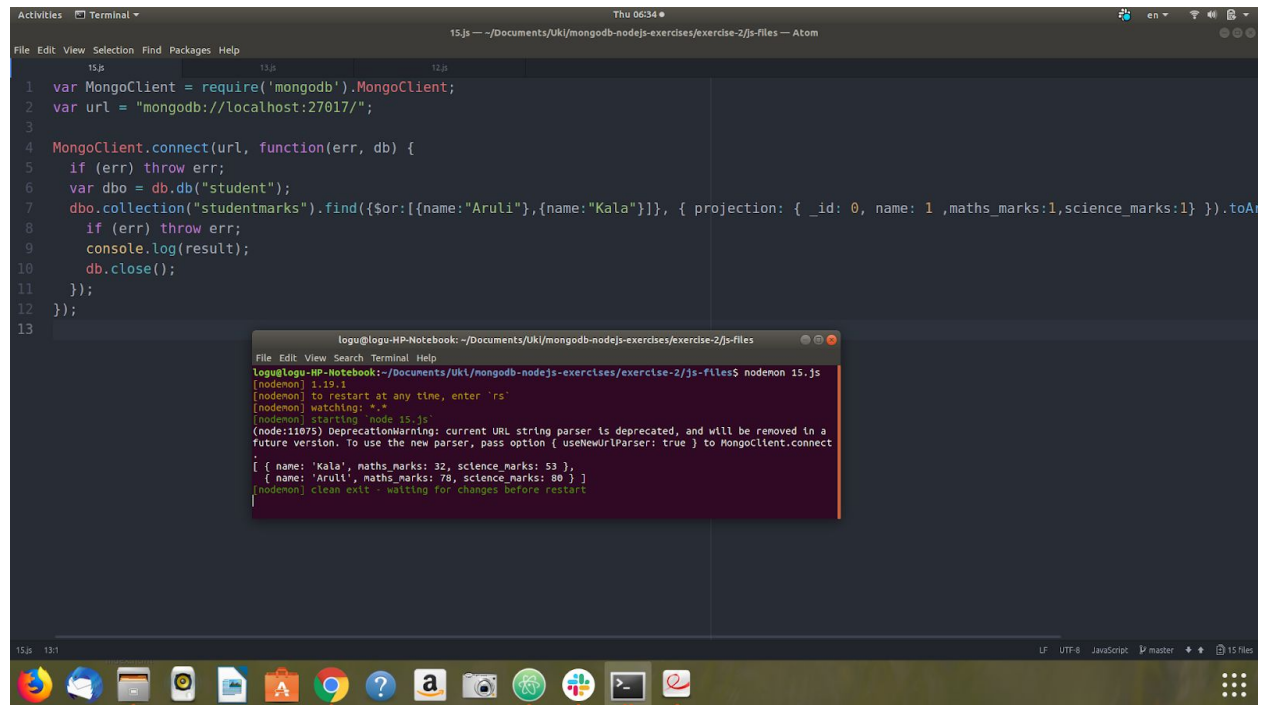


The screenshot shows an Atom editor window with a JavaScript file named 14.js. The code connects to a MongoDB instance at localhost:27017, deletes the document with 'address: "Mountain 21"' from the 'studentmarks' collection, and logs the result. A terminal window in the foreground shows the command 'nodemon 14.js' being executed, with output indicating a successful deletion of 1 document.

```
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 = { address: 'Mountain 21' };
8   dbo.collection("studentmarks").deleteOne(myquery, function(err, obj) {
9     if (err) throw err;
10    console.log("1 document deleted");
11    db.close();
12  });
13 });
14
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ nodemon 14.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 14.js'
(node:11562) 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
```

## 15) Find Kala's or Aruli's math\_marks and science\_marks



The screenshot shows an Atom editor window with a JavaScript file named `15.js` and a terminal window running the script. The JavaScript code connects to a MongoDB database, queries the `studentmarks` collection for students named 'Aruli' or 'Kala', and logs their marks. The terminal output shows the results of the query.

```
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("studentmarks").find({$or:[{name:"Aruli"},{name:"Kala"}]}).toArray(function(err, result) {
8     if (err) throw err;
9     console.log(result);
10    db.close();
11  });
12 });
13
```

```
logu@logu-HP-Notebook: ~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files
File Edit View Search Terminal Help
logu@logu-HP-Notebook:~/Documents/Uki/mongodb-nodejs-exercises/exercise-2/js-files$ nodemon 15.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching: *.*
[nodemon] starting 'node 15.js'
(node:11075) 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
[ { name: 'Kala', maths_marks: 32, science_marks: 53 },
  { name: 'Aruli', maths_marks: 78, science_marks: 80 } ]
[nodemon] clean exit - waiting for changes before restart
```