

1) Create a Database called music.

```
node1.js | nodemongo1.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/music";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   console.log("Database created!");
7   db.close();
8 });
9
```

```
ukistu01@ukipc01:~/Desktop/node$ node nodemongo1.js
(node:6031) DeprecationWarning: current Server Discovery and Monitoring engine
is deprecated, and will be removed in a future version. To use the new Server D
iscover and Monitoring engine, pass option { useUnifiedTopology: true } to the
MongoClient constructor.
Database created!
```

2) Create a collection called songdetails.

```
node1.js | nodemongo1.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.createCollection("songdetails", function(err, res) {
8     if (err) throw err;
9     console.log("Collection created!");
10    db.close();
11  });
12 });
13
```

```
ukistu01@ukipc01:~/Desktop/node$ node nodemongo1.js
(node:6520) DeprecationWarning: current Server Discovery and Monitoring engine
is deprecated, and will be removed in a future version. To use the new Server D
iscover and Monitoring engine, pass option { useUnifiedTopology: true } to the
MongoClient constructor.
Collection created!
```

3) Create the above 5 song documents.

```
node1.js | nodemongo1.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 myobj = [
8     { Song_Name: 'Thaniye Thananthaniye', Film: 'Rhythm', Music_Director: 'A.R.Rahman', Singer: 'Shankar mahadevan'},
9     { Song_Name: 'Evano Oruvan', Film: 'Alai Payuthey', Music_Director: 'A.R.Rahman', Singer: 'Swarnalatha'},
10    { Song_Name: 'Roja Poonthottam', Film: 'Kannukkul Nilavu', Music_Director: 'Ilaiyaraaja', Singer: 'Unnikrishnan Anuradha Sr'},
11    { Song_Name: 'Vennilavae vennilavae Vinnaithaandi', Film: 'Minsara Kanavu', Music_Director: 'A.R.Rahman', Singer: 'Hariharan'},
12    { Song_Name: 'Sollamal Thottu Chellum Thendral', Film: 'Dheena', Music_Director: 'Yuvan Shankar Raja', Singer: 'Hariharan'}
13  ];
14  dbo.collection("songdetails").insertMany(myobj, function(err, res) {
15    if (err) throw err;
16    console.log("Number of documents inserted: " + res.insertedCount);
17    db.close();
18  });
19 });
20
```

ukistu01@ukipc01:~/Desktop/node\$ node nodemongo1.js
(node:6994) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Number of documents inserted: 5

4) List all documents created.

```
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({}).toArray(function(err, result) {
    if (err) throw err;
    console.log(result);
    db.close();
  });
});
```

```
ukistu01@ukitpc01:~/Desktop/node$ node nodemongo1.js
(node:8089) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
```

```
[
  {
    _id: 5e31593f096f101f4effff89,
    Song_Name: 'Thaniye Thananthanlye',
    Film: 'Rhythm',
    Music_Director: 'A.R.Rahman',
    Singer: 'Shankar mahadevan'
  },
  {
    _id: 5e31593f096f101f4effff8a,
    Song_Name: 'Evano Oruvan',
    Film: 'Alai Payuthey',
    Music_Director: 'A.R.Rahman',
    Singer: 'Swarnalatha'
  },
  {
    _id: 5e31593f096f101f4effff8b,
    Song_Name: 'Raja Poonthottam',
    Film: 'Kannukkul Nilavu',
    Music_Director: 'Ilaiyaraaja',
    Singer: 'Unnikrishnan Anuradha Srilam'
  },
  {
    _id: 5e31593f096f101f4effff8c,
    Song_Name: 'Vennilavae vennilavae Vinnaitthaandi',
    Film: 'Minsara Kanavu',
    Music_Director: 'A.R.Rahman',
    Singer: 'Hariharan Sadhana Sargam'
  },
  {
    _id: 5e31593f096f101f4effff8d,
    Song_Name: 'Sollamal Thottu Chellum Thendral',
    Film: 'Dheena',
    Music_Director: 'Yuvan Shankar Raja',
    Singer: 'Hariharan'
  }
]
```

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

```
node1.js | nodemongo1.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 query = { Music_Director: 'A.R.Rahman' };
8    dbo.collection("songdetails").find(query).toArray(function(err, result) {
9      if (err) throw err;
10     console.log(result);
11     db.close();
12   });
13 });
14
```

```
ukistu01@ukipc01:~/Desktop/node$ node nodemongo1.js
(node:8360) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[
  {
    _id: 5e31593f096f101f4effff89,
    Song_Name: 'ThanIye ThananthanIye',
    Film: 'Rhythm',
    Music_Director: 'A.R.Rahman',
    Singer: 'Shankar mahadevan'
  },
  {
    _id: 5e31593f096f101f4effff8a,
    Song_Name: 'Evano Oruvan',
    Film: 'Alai Payuthey',
    Music_Director: 'A.R.Rahman',
    Singer: 'Swarnalatha'
  },
  {
    _id: 5e31593f096f101f4effff8c,
    Song_Name: 'Vennilavae vennilavae Vinnalthaandi',
    Film: 'Minsara Kanavu',
    Music_Director: 'A.R.Rahman',
    Singer: 'Hariharan Sadhana Sargam'
  }
]
```


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

```

node1.js | nodemongo1.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 query = { Music_Director: 'A.R.Rahman', Singer: 'Unnikrishnan' };
8     dbo.collection("songdetails").find(query).toArray(function(err, result) {
9         if (err) throw err;
10        console.log(result);
11        db.close();
12    });
13 });
14

```

```

jksstu01@ukitpc01:~/Desktop/node$ node nodemongo1.js
(node:8541) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.

```

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

```

node1.js                                nodemongo1.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 = { Song_Name: 'Vennilavae vennilavae Vinnaiithaandi' };
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
uk1tsu01@uk1pc01:~/Desktop/node$ node nodemongo1.js
(node:8762) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
1 document deleted

```

8) Add new song which is your favourite.

```
node1.js | nodemongo1.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 myobj = { Song_Name: 'Roya Roy', Film: 'Kathalar thinam' ,Music_Director: 'A.R.Rahman', Singer: 'Lee'};
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
```

```
ukistu01@ukipc01:~/Desktop/node$ node nodemongo1.js
(node:9049) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
1 document inserted
ukistu01@ukipc01:~/Desktop/node$
```

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

```
node1.js | nodemongo1.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 query = { Film: 'Minsara Kanavu', Singer: 'Hariharan Sadhana Sargam' };
8   dbo.collection("songdetails").find(query).toArray(function(err, result) {
9     if (err) throw err;
10    console.log(result);
11    db.close();
12  });
13 });
14
```

```
ukistu01@ukipc01:~/Desktop/node$ node nodemongo1.js
(node:5522) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[
  {
    _id: 5e31593f096f101f4effff8c,
    Song_Name: 'Vennilavae vennilavae Vinnaithaandi',
    Film: 'Minsara Kanavu',
    Music_Director: 'A.R.Rahman',
    Singer: 'Hariharan Sadhana Sargam'
  }
]
```

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

```
node1.js | nodemongo1.js
1
2 var MongoClient = require('mongodb').MongoClient;
3 var url = "mongodb://localhost:27017/";
4
5 MongoClient.connect(url, function(err, db) {
6   if (err) throw err;
7   var dbo = db.db("music");
8   dbo.collection("songdetails").find({}, { projection: { _id: 0, Singer: 1 } }).toArray(function(err, result) {
9     if (err) throw err;
10    console.log(result);
11    db.close();
12  });
13 });
14
```

```
uk1stu01@uk1tpc01:~/Desktop/node$ node nodemongo1.js
(node:6209) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discovery and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[ { Singer: 'Shankar mahadevan' },
  { Singer: 'Swarnalatha' },
  { Singer: 'Unnikrishnan Anuradha Sriram' },
  { Singer: 'Haritharan Sadhana Sargam' },
  { Singer: 'Haritharan' },
  { Singer: 'Lee' } ]
```