

1) Create a Database called music.

```
01.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
```

```
ukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 01.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 01.js`
(node:6477) 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 songdetails.

```
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.createCollection("songdetails", function(err, res) {
    if (err) throw err;
    console.log("Collection created!");
    db.close();
  });
});
```

```
^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 02.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 02.js`
(node:6933) 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 above 5 song documents.

```
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");
  var myobj = [
    { SongName: 'Thaniye Thananthaniye', Film: 'Rhythm', MusicDirector:'A.R.Rahman', Singer:'Shankar Mahadevan' },
    { SongName: 'Evano Oruvan', Film: 'Alai Payuthey', MusicDirector:'A.R.Rahman', Singer:'Swarnalatha'},
    { SongName: 'Roja Poonthottam', Film: 'Kannukkul Nilavu', MusicDirector:'Ilaiyaraaja', Singer:'Unnikrishnan,Anuradha Sriram'},
    { SongName: 'Vennilavae Vennilavae Vinnaithaandi', Film: 'Minsara Kanavu', MusicDirector:'A.R.Rahman', Singer:'Hariharan S'},
    { SongName: 'Sollamal Thottu Chellum Thendral', Film: 'Dheena', MusicDirector:'Yuvan Shankar Raja', Singer:'Hariharan'}
  ];
  dbo.collection("songdetails").insertMany(myobj, function(err, res) {
    if (err) throw err;
    console.log("Number of documents inserted: " + res.insertedCount);
    db.close();
  });
});
```

```
^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 03.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 03.js`
(node:7907) 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: 5
[nodemon] clean exit - waiting for changes before restart
```

4) List all documents 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("music");
7    dbo.collection("songdetails").find({}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 });
13
```

```

^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 04.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 04.js`
(node:8421) 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 MongoCli
ent.connect.
[ { _id: 5d2430e3ffeae51ee3eebb5f,
  SongName: 'Thaniye Thananthaniye',
  Film: 'Rhythm',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Shankar Mahadevan' },
  { _id: 5d2430e3ffeae51ee3eebb60,
  SongName: 'Evano Oruvan',
  Film: 'Alai Payuthey',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Swarnalatha' },
  { _id: 5d2430e3ffeae51ee3eebb61,
  SongName: 'Raja Poonthottam',
  Film: 'Kannukkul Nilavu',
  MusicDirector: 'Ilaiyaraaja',
  Singer: 'Unnikrishnan,Anuradha Sriram' },
  { _id: 5d2430e3ffeae51ee3eebb62,
  SongName: 'Vennilavae Vennilavae Vinnaihaandi',
  Film: 'Minsara Kanavu',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Hariharan Sadhana Sargam' },
  { _id: 5d2430e3ffeae51ee3eebb63,
  SongName: 'Sollamal Thottu Chellum Thendral',
  Film: 'Dheena',
  MusicDirector: 'Yuwan Shankar Raja',
  Singer: 'Hariharan' },
  { _id: 5d2430e3ffeae51ee3eebb64,
  SongName: 'Sollamal Thottu Chellum Thendral',
  Film: 'Dheena',
  MusicDirector: 'Yuwan Shankar Raja',
  Singer: 'Hariharan' } ]
ent.connect.
[ { _id: 5d2430e3ffeae51ee3eebb5f,
  SongName: 'Thaniye Thananthaniye',
  Film: 'Rhythm',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Shankar Mahadevan' },
  { _id: 5d2430e3ffeae51ee3eebb60,
  SongName: 'Evano Oruvan',
  Film: 'Alai Payuthey',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Swarnalatha' },
  { _id: 5d2430e3ffeae51ee3eebb61,
  SongName: 'Raja Poonthottam',
  Film: 'Kannukkul Nilavu',
  MusicDirector: 'Ilaiyaraaja',
  Singer: 'Unnikrishnan,Anuradha Sriram' },
  { _id: 5d2430e3ffeae51ee3eebb62,
  SongName: 'Vennilavae Vennilavae Vinnaihaandi',
  Film: 'Minsara Kanavu',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Hariharan Sadhana Sargam' },
  { _id: 5d2430e3ffeae51ee3eebb63,
  SongName: 'Sollamal Thottu Chellum Thendral',
  Film: 'Dheena',
  MusicDirector: 'Yuwan Shankar Raja',
  Singer: 'Hariharan' },
  { _id: 5d2430e3ffeae51ee3eebb64,
  SongName: 'Sollamal Thottu Chellum Thendral',
  Film: 'Dheena',
  MusicDirector: 'Yuwan Shankar Raja',
  Singer: 'Hariharan' } ]

```

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



```

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");
  var query = { MusicDirector: "A.R.Rahman" };
  dbo.collection("songdetails").find(query).toArray(function(err, result) {
    if (err) throw err;
    console.log(result);
    db.close();
  });
});

```

```

^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 05.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 05.js`
(node:9324) 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 MongoCli
ent.connect.
[ { _id: 5d24374adb698922d6dee38d,
  SongName: 'Thaniye Thananthaniye',
  Film: 'Rhythm',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Shankar Mahadevan' },
  { _id: 5d24374adb698922d6dee38e,
  SongName: 'Evano Oruvan',
  Film: 'Alai Payuthey',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Swarnalatha' },
  { _id: 5d24374adb698922d6dee390,
  SongName: 'Vennilavae Vennilavae Vinnaiithaandi',
  Film: 'Minsara Kanavu',
  MusicDirector: 'A.R.Rahman',
  Singer: 'Hariharan Sadhana Sargam' } ]
[nodemon] clean exit - waiting for changes before restart

```

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

```
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 = { MusicDirector: "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
```

```
^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 06.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 06.js`
(node:9614) 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 MongoCli
ent.connect.
[]
[nodemon] clean exit - waiting for changes before restart
```

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

```

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 = { Film: 'Rhythm' };
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

```

```

^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 07.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 07.js`
(node:10200) 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 MongoCl
ient.connect.
1 document deleted
[nodemon] clean exit - waiting for changes before restart

```

8) Add new song which is your favourite.



```

^Cukistu02@ukipc02:~/Documents/UK15/NodeMongodb Exercises$ nodemon 08.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 08.js`
(node:10522) 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 MongoCl
ient.connect.
1 document inserted
[nodemon] clean exit - waiting for changes before restart

```

```

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");
  var myobj = { SongName: 'Nooru samykal', Film: 'Pichchaikaran', MusicDirector: 'Vijay Antony', Singer: 'Vijay Antony' };
  dbo.collection("songdetails").insertOne(myobj, function(err, res) {
    if (err) throw err;
    console.log("1 document inserted");
    db.close();
  });
});

```

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

```

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" };
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

```

```

^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 09.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 09.js`
(node:10775) 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 MongoCl
ient.connect.
[]
[nodemon] clean exit - waiting for changes before restart

```

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

```
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({}, { projection: { _id: 0, Singer: 1 } }).toArray(function(err, result) {
    if (err) throw err;
    console.log(result);
    db.close();
  });
});
```

```
^Cukistu02@ukipc02:~/Documents/uki5/NodeMongodb Exercises$ nodemon 10.js
[nodemon] 1.19.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node 10.js`
(node:11024) 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 MongoCl
ient.connect.
[ { Singer: 'Swarnalatha' },
  { Singer: 'Unnikrishnan,Anuradha Sriram' },
  { Singer: 'Hariharan Sadhana Sargam' },
  { Singer: 'Hariharan' },
  { Singer: 'Vijay Antony' },
  { Singer: 'Vijay Antony' },
  { Singer: 'Vijay Antony' } ]
[nodemon] clean exit - waiting for changes before restart
```