

1) Create a Database called music.

> use music

switched to db music

```
ukistu01@ukipc01: ~  
ukistu01@ukipc01: ~  
http://docs.mongodb.org/  
Questions? Try the support group  
http://groups.google.com/group/mongodb-user  
Server has startup warnings:  
2019-01-08T11:47:51.763+0530 I STORAGE [initandlisten]  
2019-01-08T11:47:51.763+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS  
S filesystem is strongly recommended with the WiredTiger storage engine  
2019-01-08T11:47:51.763+0530 I STORAGE [initandlisten] ** See http://d  
ochub.mongodb.org/core/prodnotes-filesystem  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten]  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten] ** WARNING: Access contr  
ol is not enabled for the database.  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten] ** Read and wri  
te access to data and configuration is unrestricted.  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten]  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten]  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/  
mm/transparent_hugepage/enabled is 'always'.  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten] ** We suggest set  
ting it to 'never'  
2019-01-08T11:47:53.818+0530 I CONTROL [initandlisten]  
> use music  
switched to db music  
>
```

2) Create a collection called songdetails.

> db.createCollection("songdetails")

{ "ok":1 }

```
ukistu01@ukipc01: ~  
ukistu01@ukipc01: ~  
ukistu01@ukipc01:~$ sudo mongo  
[sudo] password for ukistu01:  
MongoDB shell version v3.6.9  
connecting to: mongodb://127.0.0.1:27017  
Implicit session: session { "id" : UUID("7bca0e89-007f-40e4-be0c-2cba4ce429d2") }  
MongoDB server version: 3.6.9  
Server has startup warnings:  
2019-01-09T21:19:54.058+0530 I STORAGE [initandlisten]  
2019-01-09T21:19:54.058+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine  
2019-01-09T21:19:54.058+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem  
2019-01-09T21:19:56.283+0530 I CONTROL [initandlisten]  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** WARNING: You are running this process as the root user, which is not recommended.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten]  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** WARNING: This server is bound to localhost.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** Remote systems will be unable to connect to this server.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** Start the server with --bind_ip <address> to specify which IP  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** addresses it should serve responses from, or with --bind_ip_all to  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** bind to all interfaces. If this behavior is desired, start the  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** server with --bind_ip 127.0.0.1 to disable this warning.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten]  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten]  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten]  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten] ** We suggest setting it to 'never'  
2019-01-09T21:19:56.284+0530 I CONTROL [initandlisten]  
> use music  
switched to db music  
> db.createCollection("songdetails")  
{ "ok" : 1 }  
>
```

3) Create the above 5 song documents.

```
>db.songdetails.insert({"Song Name":"ThaniyeThananthaniye","Film":"Rhythm","Music  
Director":"A.R.Rahman","Singer":"Shankar mahadevan"})  
WriteResult({ "nInserted" : 1 })
```

```
>db.songdetails.insert({"Song Name":"Evano Oruvan","Film":"Alai Payuthey","Music  
Director":"A.R.Rahman","Singer":"Swarnalatha"})  
WriteResult({ "nInserted" : 1 })
```

```
>db.songdetails.insert({"Song Name":"Roja poonthotam","Film":"Kannukul nilavuy","Music  
Director":"Ilaiyaraaja","Singer":"Unnikrishnan,Anuradha Sriram"})  
WriteResult({ "nInserted" : 1 })
```

```
>db.songdetails.insert({"Song Name":"Vennilavae Vennilavae Vinnaithaandi","Film":"Minsara Kanavu","Music  
Director":"A.R.Rahman","Singer":"Hariharan ,Sadhana Sargam"})  
WriteResult({ "nInserted" : 1 })
```

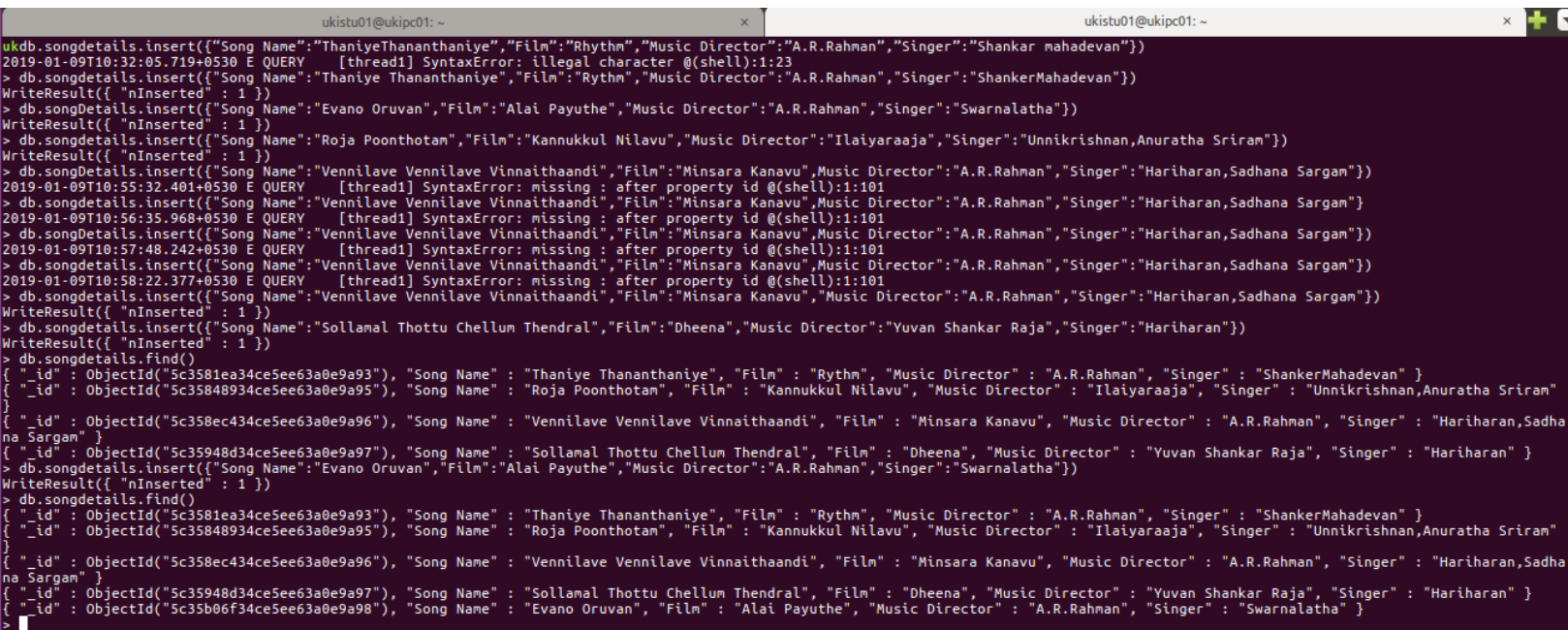
```
>db.songdetails.insert({"Song Name":"Sollamal Thottu Chellum Thendral","Film":"Dheena","Music  
Director":"Yuvan Shankar Raja","Singer":"Hariharan "})  
WriteResult({ "nInserted" : 1 })
```

```
ukdb.songdetails.insert({"Song Name":"ThaniyeThananthaniye","Film":"Rhythm","Music Director":"A.R.Rahman","Singer":"Shankar mahadevan"})  
2019-01-09T10:32:05.719+0530 E QUERY [thread1] SyntaxError: illegal character @(shell):1:23  
> db.songdetails.insert({"Song Name":"Thaniye Thananthaniye","Film":"Rythm","Music Director":"A.R.Rahman","Singer":"ShankerMahadevan"})  
WriteResult({ "nInserted" : 1 })  
> db.songDetails.insert({"Song Name":"Evano Oruvan","Film":"Alai Payuthe","Music Director":"A.R.Rahman","Singer":"Swarnalatha"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.insert({"Song Name":"Roja Poonthotam","Film":"Kannukkul Nilavu","Music Director":"Ilaiyaraaja","Singer":"Unnikrishnan,Anuradha Sriram"})  
WriteResult({ "nInserted" : 1 })  
> db.songDetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:55:32.401+0530 E QUERY [thread1] SyntaxError: missing : after property id @(shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:56:35.968+0530 E QUERY [thread1] SyntaxError: missing : after property id @(shell):1:101  
> db.songDetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:57:48.242+0530 E QUERY [thread1] SyntaxError: missing : after property id @(shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:58:22.377+0530 E QUERY [thread1] SyntaxError: missing : after property id @(shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.insert({"Song Name":"Sollamal Thottu Chellum Thendral","Film":"Dheena","Music Director":"Yuvan Shankar Raja","Singer":"Hariharan"})  
WriteResult({ "nInserted" : 1 })  
>
```

4) List all documents created.

```
> db.songdetails.find()
```

```
{ "_id" : ObjectId("5c3581ea34ce5ee63a0e9a93"), "Song Name" : "Thaniye Thananthaniye", "Film" :  
"Rythm", "Music Director" : "A.R.Rahman", "Singer" : "ShankerMahadevan" }  
{ "_id" : ObjectId("5c35848934ce5ee63a0e9a95"), "Song Name" : "Roja Poonthotam", "Film" :  
"Kannukkul Nilavu", "Music Director" : "Ilaiyaraaja", "Singer" : "Unnikrishnan,Anuratha Sriram" }  
{ "_id" : ObjectId("5c358ec434ce5ee63a0e9a96"), "Song Name" : "Vennilave Vennilave  
Vinnaithaandi", "Film" : "Minsara Kanavu", "Music Director" : "A.R.Rahman", "Singer" :  
"Hariharan,Sadhana Sargam" }  
{ "_id" : ObjectId("5c35948d34ce5ee63a0e9a97"), "Song Name" : "Sollamal Thottu Chellum Thendral",  
"Film" : "Dheena", "Music Director" : "Yuvan Shankar Raja", "Singer" : "Hariharan" }  
{ "_id" : ObjectId("5c35b06f34ce5ee63a0e9a98"), "Song Name" : "Evano Oruvan", "Film" : "Alai  
Payuthe", "Music Director" : "A.R.Rahman", "Singer" : "Swarnalatha" }  
>
```



```
ukistu01@ukipc01: ~  
ukdb.songdetails.insert({"Song Name":"ThaniyeThananthaniye","Film":"Rhythm","Music Director":"A.R.Rahman","Singer":"Shanker mahadevan"})  
2019-01-09T10:32:05.719+0530 E QUERY [thread1] SyntaxError: illegal character @ (shell):1:23  
> db.songdetails.insert({"Song Name":"Thaniye Thananthaniye","Film":"Rythm","Music Director":"A.R.Rahman","Singer":"ShankerMahadevan"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.insert({"Song Name":"Evano Oruvan","Film":"Alai Payuthe","Music Director":"A.R.Rahman","Singer":"Swarnalatha"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.insert({"Song Name":"Roja Poonthotam","Film":"Kannukkul Nilavu","Music Director":"Ilaiyaraaja","Singer":"Unnikrishnan,Anuratha Sriram"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:55:32.401+0530 E QUERY [thread1] SyntaxError: missing : after property id @ (shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:56:35.968+0530 E QUERY [thread1] SyntaxError: missing : after property id @ (shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:57:48.242+0530 E QUERY [thread1] SyntaxError: missing : after property id @ (shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
2019-01-09T10:58:22.377+0530 E QUERY [thread1] SyntaxError: missing : after property id @ (shell):1:101  
> db.songdetails.insert({"Song Name":"Vennilave Vennilave Vinnaithaandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer":"Hariharan,Sadhana Sargam"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.insert({"Song Name":"Sollamal Thottu Chellum Thendral","Film":"Dheena","Music Director":"Yuvan Shankar Raja","Singer":"Hariharan"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.find()  
{ "_id" : ObjectId("5c3581ea34ce5ee63a0e9a93"), "Song Name" : "Thaniye Thananthaniye", "Film" : "Rythm", "Music Director" : "A.R.Rahman", "Singer" : "ShankerMahadevan" }  
{ "_id" : ObjectId("5c35848934ce5ee63a0e9a95"), "Song Name" : "Roja Poonthotam", "Film" : "Kannukkul Nilavu", "Music Director" : "Ilaiyaraaja", "Singer" : "Unnikrishnan,Anuratha Sriram"  
{ "_id" : ObjectId("5c358ec434ce5ee63a0e9a96"), "Song Name" : "Vennilave Vennilave Vinnaithaandi", "Film" : "Minsara Kanavu", "Music Director" : "A.R.Rahman", "Singer" : "Hariharan,Sadha  
na Sargam" }  
{ "_id" : ObjectId("5c35948d34ce5ee63a0e9a97"), "Song Name" : "Sollamal Thottu Chellum Thendral", "Film" : "Dheena", "Music Director" : "Yuvan Shankar Raja", "Singer" : "Hariharan" }  
> db.songdetails.insert({"Song Name":"Evano Oruvan","Film":"Alai Payuthe","Music Director":"A.R.Rahman","Singer":"Swarnalatha"})  
WriteResult({ "nInserted" : 1 })  
> db.songdetails.find()  
{ "_id" : ObjectId("5c3581ea34ce5ee63a0e9a93"), "Song Name" : "Thaniye Thananthaniye", "Film" : "Rythm", "Music Director" : "A.R.Rahman", "Singer" : "ShankerMahadevan" }  
{ "_id" : ObjectId("5c35848934ce5ee63a0e9a95"), "Song Name" : "Roja Poonthotam", "Film" : "Kannukkul Nilavu", "Music Director" : "Ilaiyaraaja", "Singer" : "Unnikrishnan,Anuratha Sriram"  
{ "_id" : ObjectId("5c358ec434ce5ee63a0e9a96"), "Song Name" : "Vennilave Vennilave Vinnaithaandi", "Film" : "Minsara Kanavu", "Music Director" : "A.R.Rahman", "Singer" : "Hariharan,Sadha  
na Sargam" }  
{ "_id" : ObjectId("5c35948d34ce5ee63a0e9a97"), "Song Name" : "Sollamal Thottu Chellum Thendral", "Film" : "Dheena", "Music Director" : "Yuvan Shankar Raja", "Singer" : "Hariharan" }  
{ "_id" : ObjectId("5c35b06f34ce5ee63a0e9a98"), "Song Name" : "Evano Oruvan", "Film" : "Alai Payuthe", "Music Director" : "A.R.Rahman", "Singer" : "Swarnalatha" }
```

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

6) List ilaiyaraaja songs sung by Unnikrishnan.

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

8) Add new song which is your favourite.

- 9) List Songs sung by Hariharan from Minsara kanavu film.
- 10) List out the singers' names in your document.