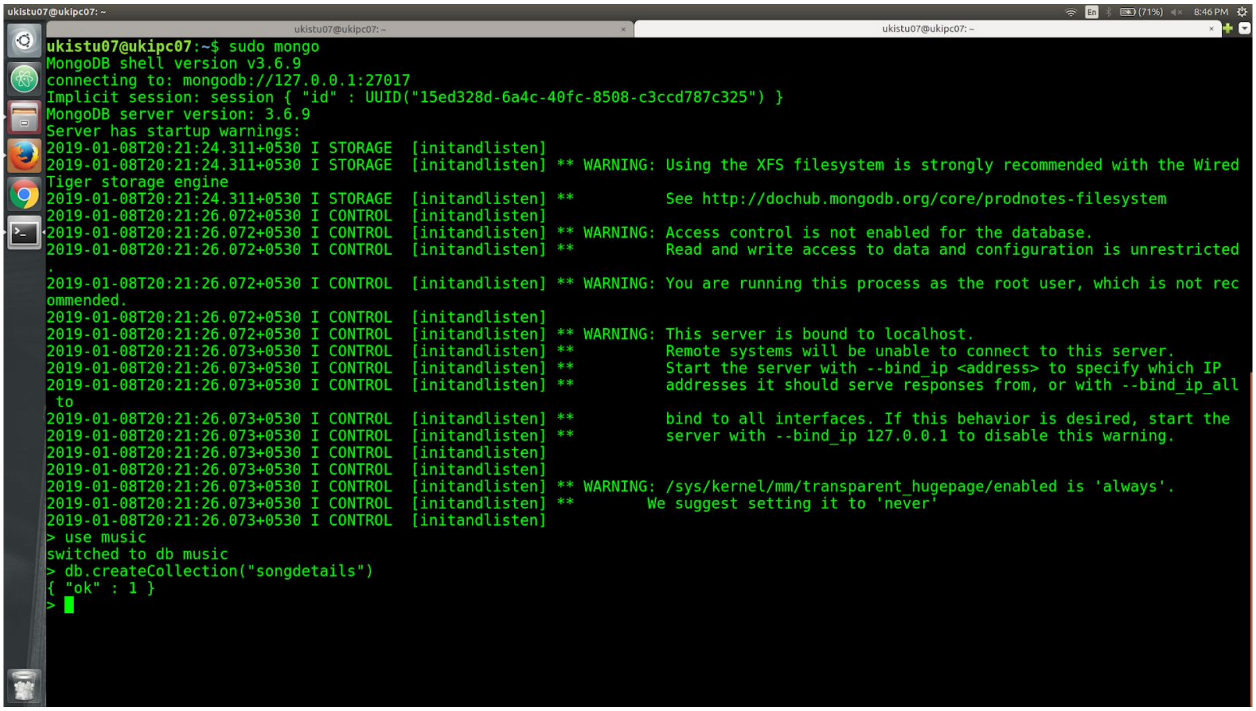
1.Create a Database called music.

> use music

2.Create a collection called songdetails.

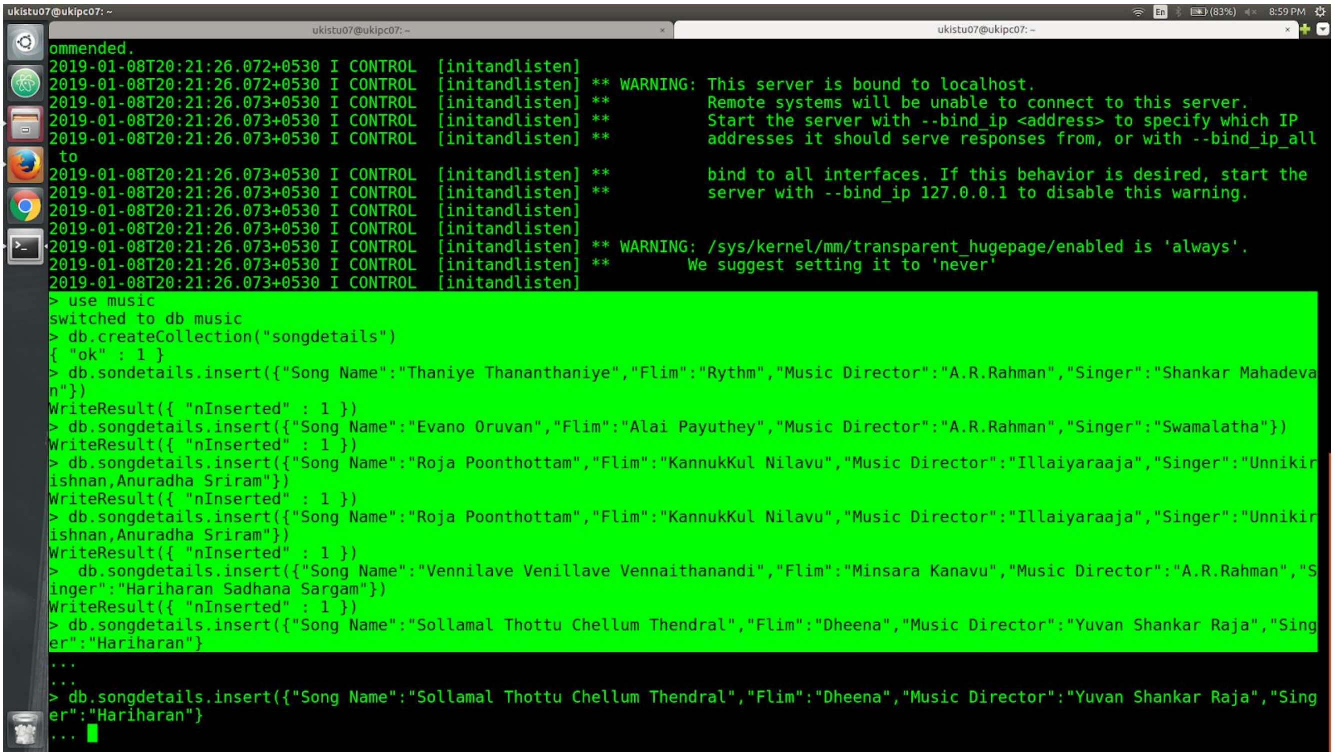
> db.createCollection("songdetails")



```
ukistu07@ukipc07:~$ sudo mongo
MongoDB shell version v3.6.9
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID("15ed328d-6a4c-40fc-8508-c3ccd787c325") }
MongoDB server version: 3.6.9
Server has startup warnings:
2019-01-08T20:21:24.311+0530 I STORAGE  [initandlisten]
2019-01-08T20:21:24.311+0530 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the Wired
Tiger storage engine
2019-01-08T20:21:24.311+0530 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted
.
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten] ** WARNING: You are running this process as the root user, which is not rec
ommended.
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten] ** WARNING: This server is bound to localhost.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          Remote systems will be unable to connect to this server.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          Start the server with --bind_ip <address> to specify which IP
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          addresses it should serve responses from, or with --bind_ip_all
 to
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          bind to all interfaces. If this behavior is desired, start the
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          server with --bind_ip 127.0.0.1 to disable this warning.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **        We suggest setting it to 'never'
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten]
> use music
switched to db music
> db.createCollection("songdetails")
{ "ok" : 1 }
> 
```

3.Create the above 5 song documents

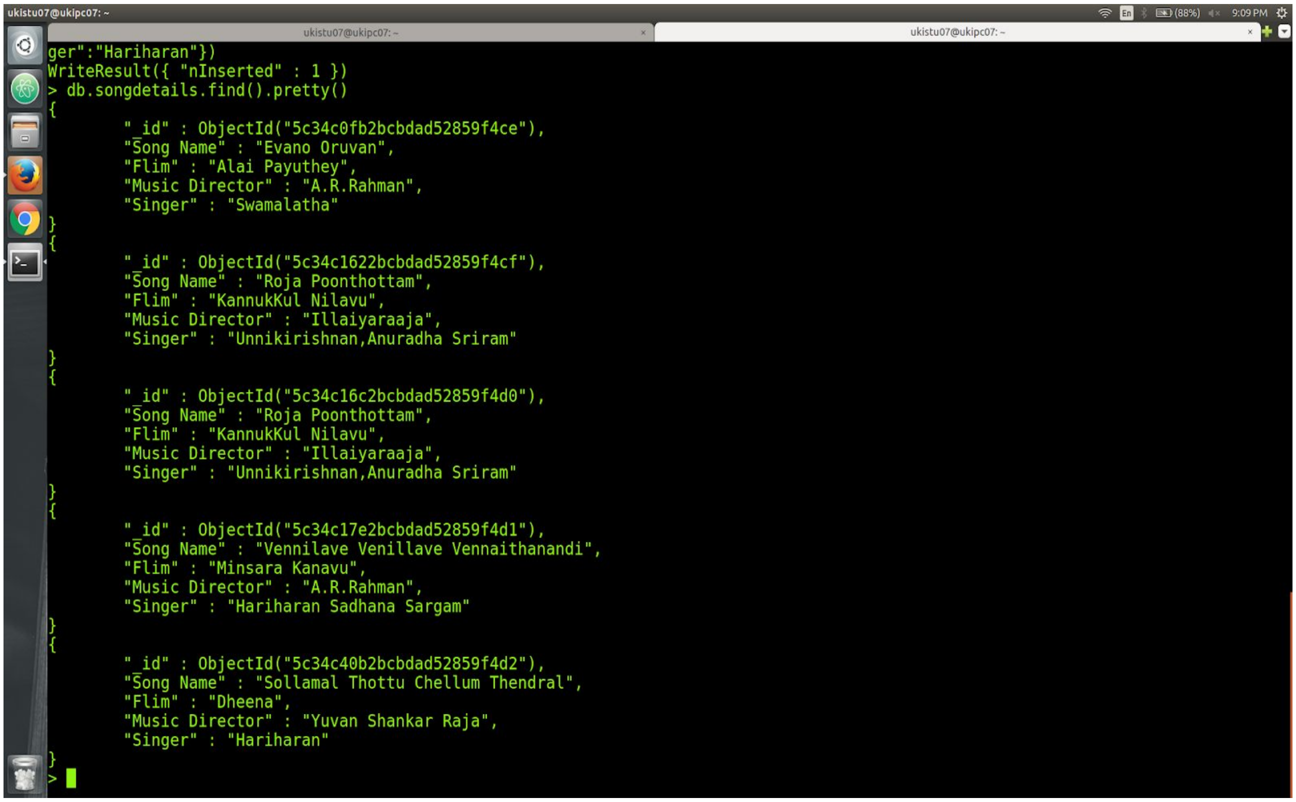
```
> db.songdetails.insert({"Song Name":"thaniye Thananthaniye","Flim":"Rythm","Music Director":"A.R.Rahman","Singer":"Shankar Mahadevan"})

> db.songdetails.insert({"Song Name":"Evano Oruvan","Flim":"Alai Payuthey","Music Director":"A.R.Rahman","Singer":"Swamalatha"})

> db.songdetails.insert({"Song Name":"Roja Poonthottam","Flim":"KannukKul Nilavu","Music
Director":"Illaiyaraaja","Singer":
"Unnikirishnan,Anuradha Sriram"})

    > db.songdetails.insert({"Song
Name":"Vennilave Venillave
Vennaithanandi","Flim":"Minsara
Kanavu","Music
Director":"A.R.Rahman","Singer":
"Hariharan Sadhana Sargam"})

    > db.songdetails.insert({"Song
Name":"Roja
Poonthottam","Film":"KannukKul
Nilavu","Music
Director":"Illaiyaraaja","Singer":
["Unnikirishnan","Anuradha Sriram"]})
```



```
ommended.
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.072+0530 I CONTROL  [initandlisten] ** WARNING: This server is bound to localhost.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          Remote systems will be unable to connect to this server.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          Start the server with --bind_ip <address> to specify which IP
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          addresses it should serve responses from, or with --bind_ip_all
 to
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          bind to all interfaces. If this behavior is desired, start the
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **          server with --bind_ip 127.0.0.1 to disable this warning.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten]
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten] **        We suggest setting it to 'never'
2019-01-08T20:21:26.073+0530 I CONTROL  [initandlisten]
> use music
switched to db music
> db.createCollection("songdetails")
{ "ok" : 1 }
> db.sondetails.insert({"Song Name":"Thaniye Thananthaniye","Flim":"Rythm","Music Director":"A.R.Rahman","Singer":"Shankar Mahadeva
n"})
WriteResult({ "nInserted" : 1 })
> db.songdetails.insert({"Song Name":"Evano Oruvan","Flim":"Alai Payuthey","Music Director":"A.R.Rahman","Singer":"Swamalatha"})
WriteResult({ "nInserted" : 1 })
> db.songdetails.insert({"Song Name":"Roja Poonthottam","Flim":"KannukKul Nilavu","Music Director":"Illaiyaraaja","Singer":"Unnikir
ishnan,Anuradha Sriram"})
WriteResult({ "nInserted" : 1 })
> db.songdetails.insert({"Song Name":"Roja Poonthottam","Flim":"KannukKul Nilavu","Music Director":"Illaiyaraaja","Singer":"Unnikir
ishnan,Anuradha Sriram"})
WriteResult({ "nInserted" : 1 })
>  db.songdetails.insert({"Song Name":"Vennilave Venillave Vennaithanandi","Flim":"Minsara Kanavu","Music Director":"A.R.Rahman","S
inger":"Hariharan Sadhana Sargam"})
WriteResult({ "nInserted" : 1 })
> db.songdetails.insert({"Song Name":"Sollamal Thottu Chellum Thendral","Flim":"Dheena","Music Director":"Yuvan Shankar Raja","Sing
er":"Hariharan"}
...
...
> db.songdetails.insert({"Song Name":"Sollamal Thottu Chellum Thendral","Flim":"Dheena","Music Director":"Yuvan Shankar Raja","Sing
er":"Hariharan"}
... 
```

4.List all documents created.

> db.songdetails.find().pretty()

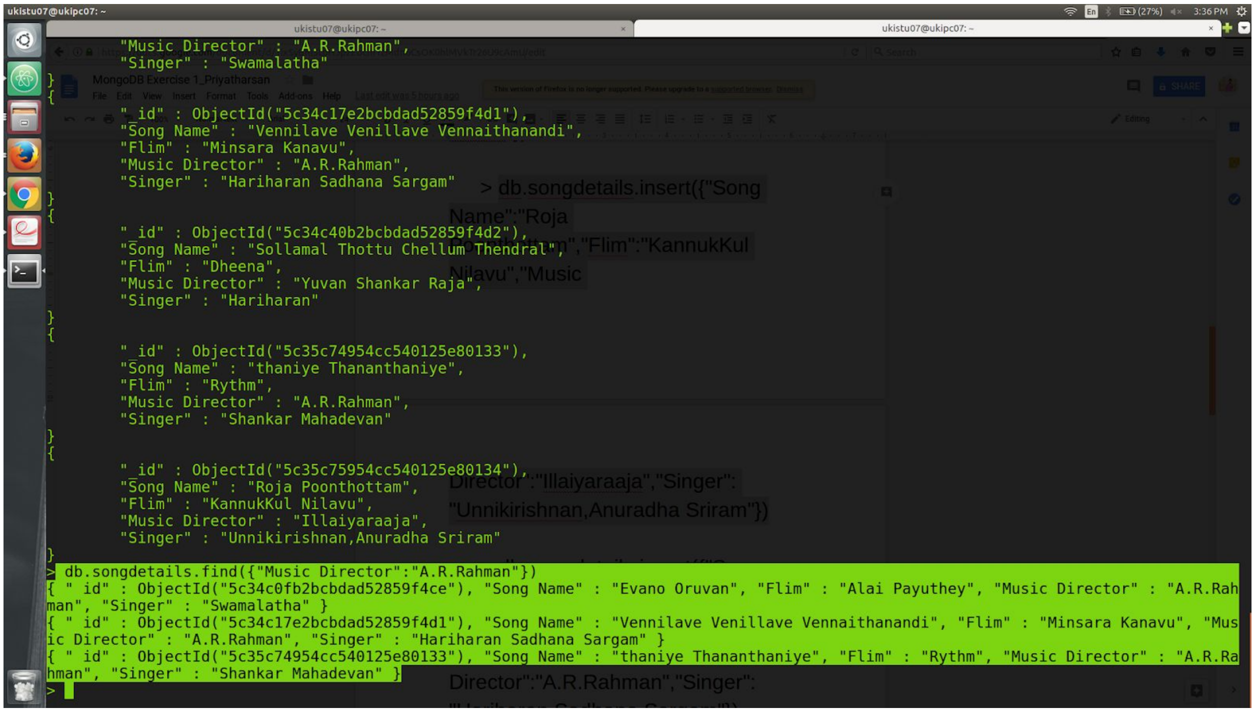


```
ger":"Hariharan"})
WriteResult({ "nInserted" : 1 })
> db.songdetails.find().pretty()
{
        "_id" : ObjectId("5c34c0fb2bcbdad52859f4ce"),
        "Song Name" : "Evano Oruvan",
        "Flim" : "Alai Payuthey",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Swamalatha"
}
{
        "_id" : ObjectId("5c34c1622bcbdad52859f4cf"),
        "Song Name" : "Roja Poonthottam",
        "Flim" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : "Unnikirishnan,Anuradha Sriram"
}
{
        "_id" : ObjectId("5c34c16c2bcbdad52859f4d0"),
        "Song Name" : "Roja Poonthottam",
        "Flim" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : "Unnikirishnan,Anuradha Sriram"
}
{
        "_id" : ObjectId("5c34c17e2bcbdad52859f4d1"),
        "Song Name" : "Vennilave Venillave Vennaithanandi",
        "Flim" : "Minsara Kanavu",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Hariharan Sadhana Sargam"
}
{
        "_id" : ObjectId("5c34c40b2bcbdad52859f4d2"),
        "Song Name" : "Sollamal Thottu Chellum Thendral",
        "Flim" : "Dheena",
        "Music Director" : "Yuvan Shankar Raja",
        "Singer" : "Hariharan"
}
> 
```

5.List A.R.Rahman’s songs.

Method 1.

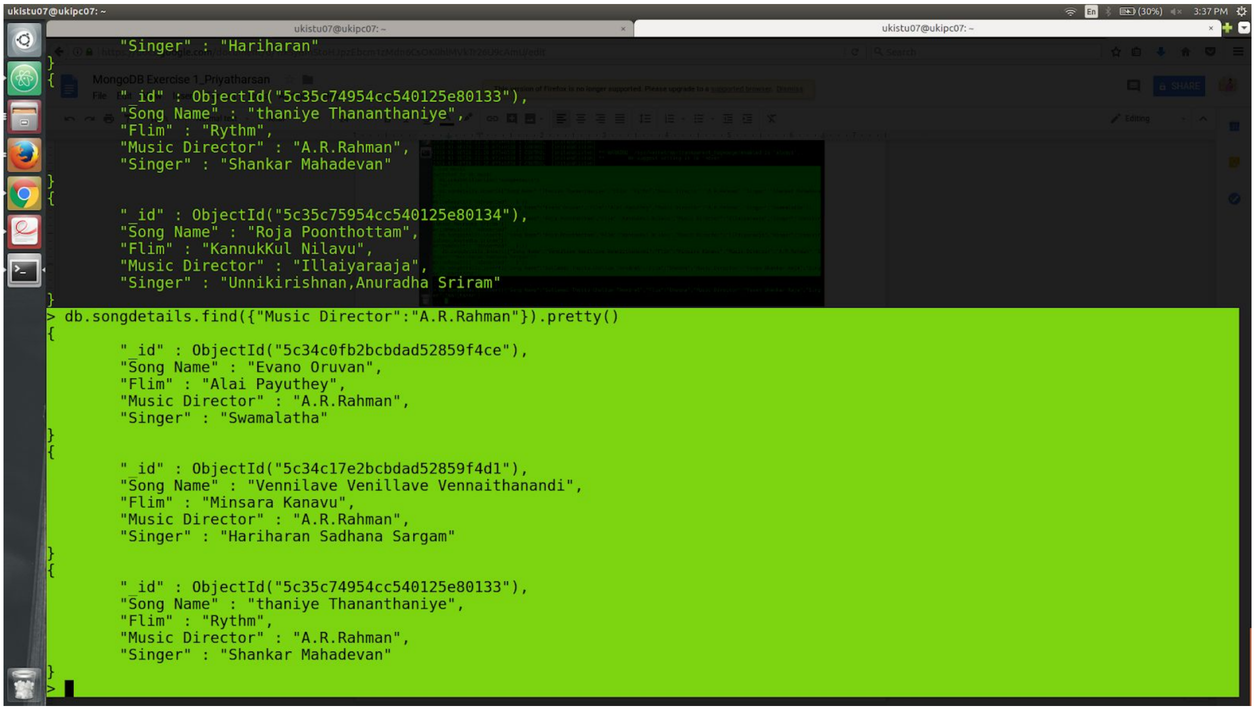
>db.songdetails.find({"Music Director":"A.R.Rahman"})



```
        "Music Director" : "A.R.Rahman",
        "Singer" : "Swamalatha"
}
{
        "_id" : ObjectId("5c34c17e2bcbdad52859f4d1"),
        "Song Name" : "Vennilave Venillave Vennaithanandi",
        "Flim" : "Minsara Kanavu",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Hariharan Sadhana Sargam"
}
{
        "_id" : ObjectId("5c34c40b2bcbdad52859f4d2"),
        "Song Name" : "Sollamal Thottu Chellum Thendral",
        "Flim" : "Dheena",
        "Music Director" : "Yuvan Shankar Raja",
        "Singer" : "Hariharan"
}
{
        "_id" : ObjectId("5c35c74954cc540125e80133"),
        "Song Name" : "thaniye Thananthaniye",
        "Flim" : "Rythm",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Shankar Mahadevan"
}
{
        "_id" : ObjectId("5c35c75954cc540125e80134"),
        "Song Name" : "Roja Poonthottam",
        "Flim" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : "Unnikirishnan,Anuradha Sriram"
}
> db.songdetails.find({"Music Director":"A.R.Rahman"})
{ "_id" : ObjectId("5c34c0fb2bcbdad52859f4ce"), "Song Name" : "Evano Oruvan", "Flim" : "Alai Payuthey", "Music Director" : "A.R.Rah
man", "Singer" : "Swamalatha" }
{ "_id" : ObjectId("5c34c17e2bcbdad52859f4d1"), "Song Name" : "Vennilave Venillave Vennaithanandi", "Flim" : "Minsara Kanavu", "Mus
ic Director" : "A.R.Rahman", "Singer" : "Hariharan Sadhana Sargam" }
{ "_id" : ObjectId("5c35c74954cc540125e80133"), "Song Name" : "thaniye Thananthaniye", "Flim" : "Rythm", "Music Director" : "A.R.Ra
hman", "Singer" : "Shankar Mahadevan" }
>
```

Method 2.

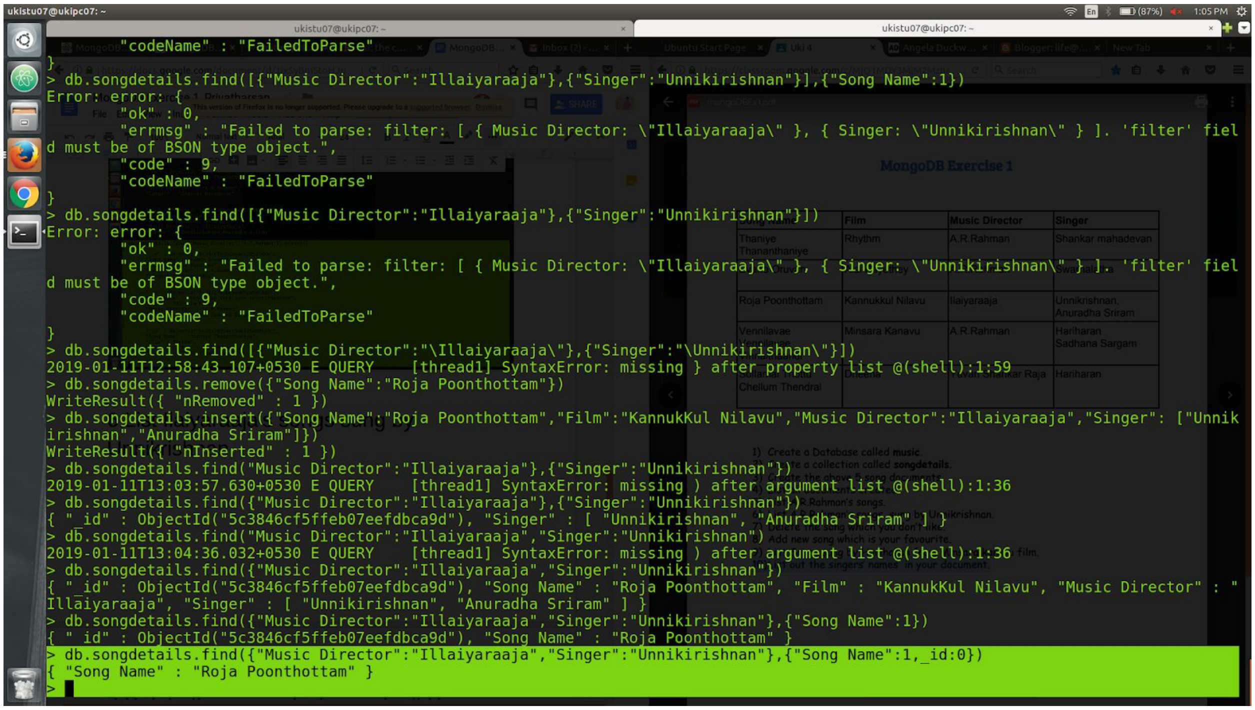
> db.songdetails.find({"Music Director":"A.R.Rahman"}).pretty()



```
        "Singer" : "Hariharan"
}
{
        "_id" : ObjectId("5c35c74954cc540125e80133"),
        "Song Name" : "thaniye Thananthaniye",
        "Flim" : "Rythm",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Shankar Mahadevan"
}
{

        "_id" : ObjectId("5c35c75954cc540125e80134"),
        "Song Name" : "Roja Poonthottam",
        "Flim" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : "Unnikirishnan,Anuradha Sriram"
}
> db.songdetails.find({"Music Director":"A.R.Rahman"}).pretty()
{
        "_id" : ObjectId("5c34c0fb2bcbdad52859f4ce"),
        "Song Name" : "Evano Oruvan",
        "Flim" : "Alai Payuthey",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Swamalatha"
}
{

        "_id" : ObjectId("5c34c17e2bcbdad52859f4d1"),
        "Song Name" : "Vennilave Venillave Vennaithanandi",
        "Flim" : "Minsara Kanavu",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Hariharan Sadhana Sargam"
}
{

        "_id" : ObjectId("5c35c74954cc540125e80133"),
        "Song Name" : "thaniye Thananthaniye",
        "Flim" : "Rythm",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Shankar Mahadevan"
}
> 
```

6.List Ilaiyaraaja’s songs sung by Unnikrishnan.

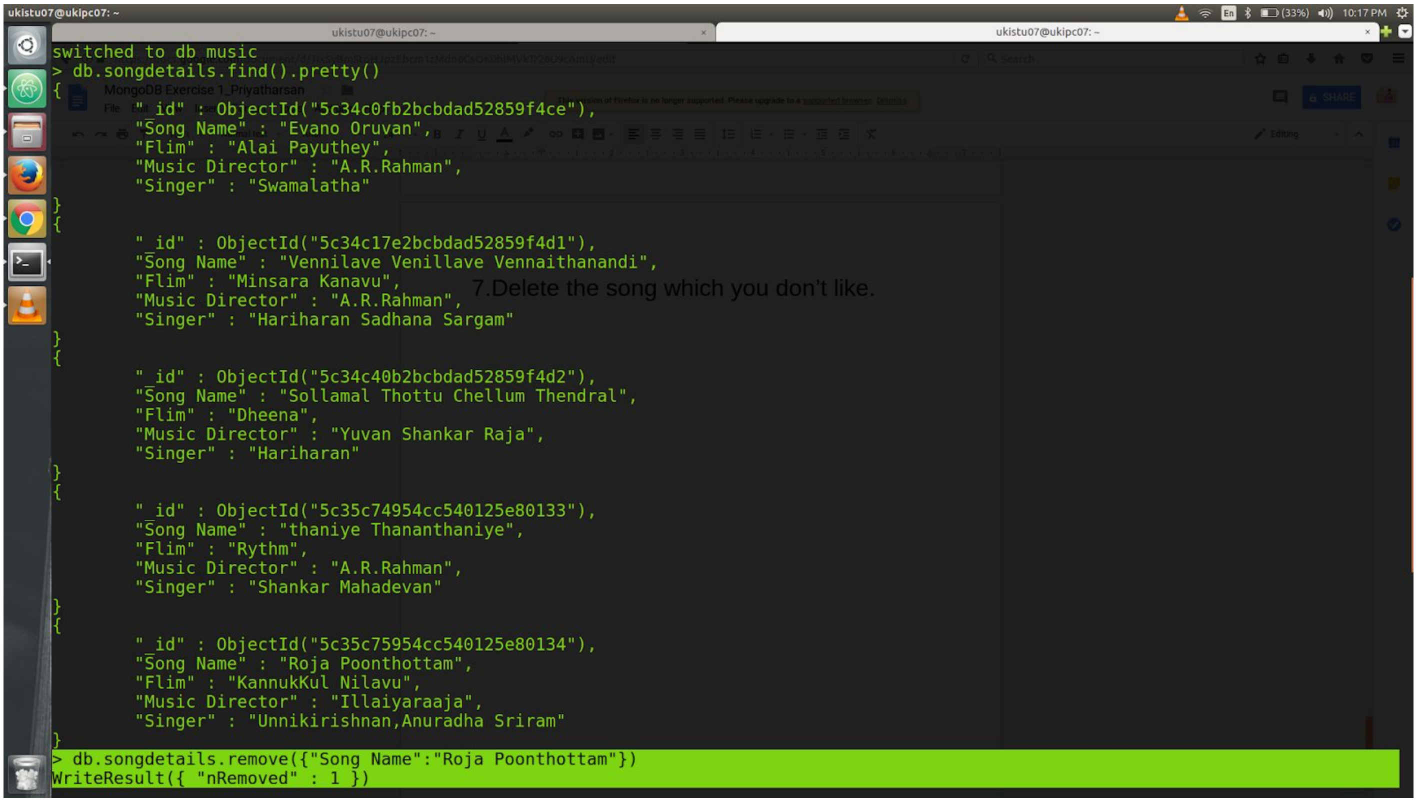
> db.songdetails.find({"Music Director":"Illaiyaraaja","Singer":"Unnikirishnan"},{"Song Name":1})



```
        "codeName" : "FailedToParse"
}
> db.songdetails.find([{"Music Director":"Illaiyaraaja"},{"Singer":"Unnikirishnan"}],{"Song Name":1})
Error: error: {
        "ok" : 0,
        "errmsg" : "Failed to parse: filter: [ { Music Director: \"Illaiyaraaja\" }, { Singer: \"Unnikirishnan\" } ]. 'filter' fiel
d must be of BSON type object.",
        "code" : 9,
        "codeName" : "FailedToParse"
}
> db.songdetails.find([{"Music Director":"Illaiyaraaja"},{"Singer":"Unnikirishnan"}])
Error: error: {
        "ok" : 0,
        "errmsg" : "Failed to parse: filter: [ { Music Director: \"Illaiyaraaja\" }, { Singer: \"Unnikirishnan\" } ]. 'filter' fiel
d must be of BSON type object.",
        "code" : 9,
        "codeName" : "FailedToParse"
}
> db.songdetails.find([{"Music Director":"\Illaiyaraaja\"},{"Singer":"\Unnikirishnan\"}])
2019-01-11T12:58:43.107+0530 E QUERY    [thread1] SyntaxError: missing } after property list @(shell):1:59
> db.songdetails.remove({"Song Name":"Roja Poonthottam"})
WriteResult({ "nRemoved" : 1 })
> db.songdetails.insert({"Song Name":"Roja Poonthottam","Film":"KannukKul Nilavu","Music Director":"Illaiyaraaja","Singer": ["Unnik
irishnan","Anuradha Sriram"]})
WriteResult({ "nInserted" : 1 })
> db.songdetails.find("Music Director":"Illaiyaraaja"},{"Singer":"Unnikirishnan"})
2019-01-11T13:03:57.630+0530 E QUERY    [thread1] SyntaxError: missing ) after argument list @(shell):1:36
> db.songdetails.find({"Music Director":"Illaiyaraaja"},{"Singer":"Unnikirishnan"})
{ "_id" : ObjectId("5c3846cf5ffeb07eefdbca9d"), "Singer" : [ "Unnikirishnan", "Anuradha Sriram" ] }
> db.songdetails.find("Music Director":"Illaiyaraaja","Singer":"Unnikirishnan")
2019-01-11T13:04:36.032+0530 E QUERY    [thread1] SyntaxError: missing ) after argument list @(shell):1:36
> db.songdetails.find({"Music Director":"Illaiyaraaja","Singer":"Unnikirishnan"})
{ "_id" : ObjectId("5c3846cf5ffeb07eefdbca9d"), "Song Name" : "Roja Poonthottam", "Film" : "KannukKul Nilavu", "Music Director" : "
Illaiyaraaja", "Singer" : [ "Unnikirishnan", "Anuradha Sriram" ] }
> db.songdetails.find({"Music Director":"Illaiyaraaja","Singer":"Unnikirishnan"},{"Song Name":1})
{ "_id" : ObjectId("5c3846cf5ffeb07eefdbca9d"), "Song Name" : "Roja Poonthottam" }
> db.songdetails.find({"Music Director":"Illaiyaraaja","Singer":"Unnikirishnan"},{"Song Name":1,_id:0})
{ "Song Name" : "Roja Poonthottam" }
> 
```

7.Delete the song which you don’t like.

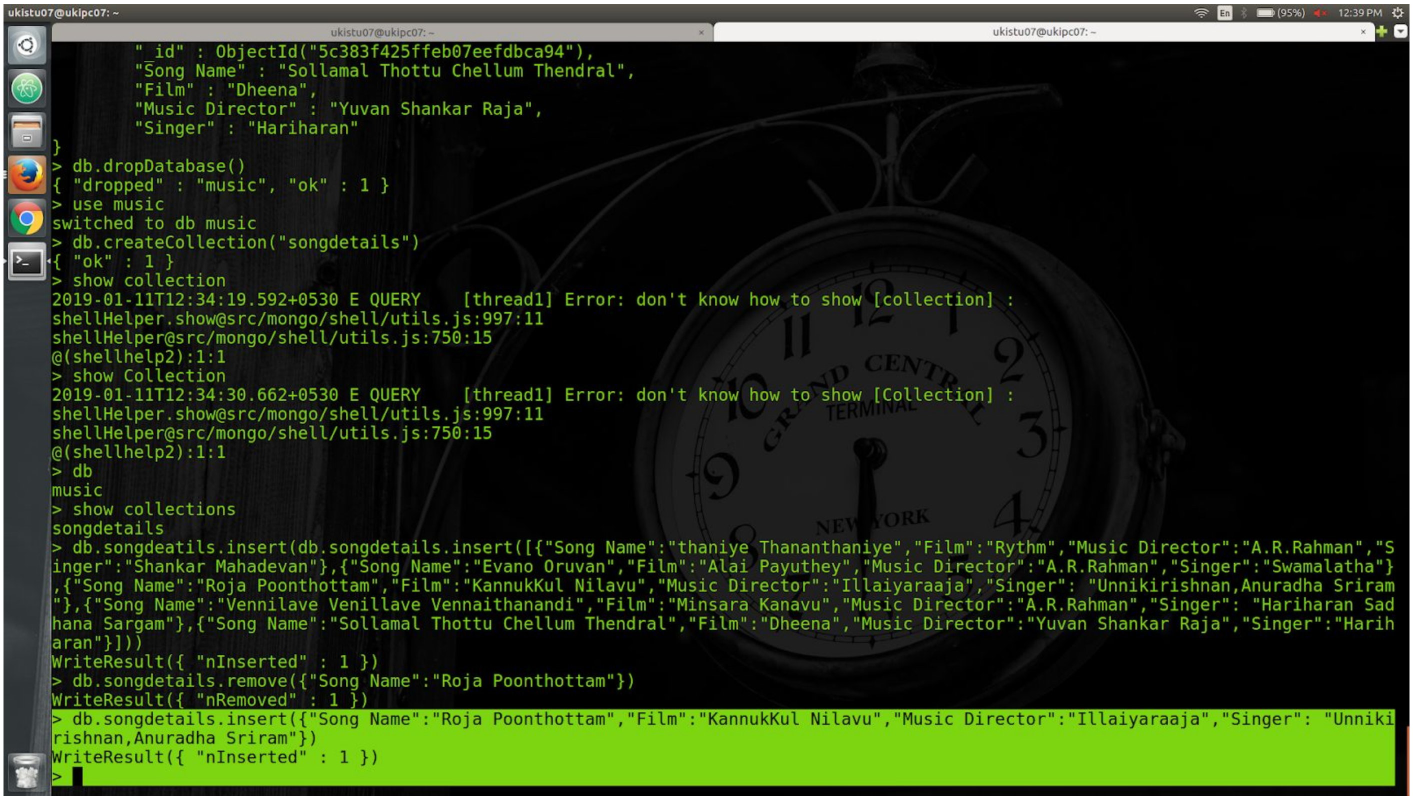
> db.songdetails.remove({"Song Name":"Roja Poonthottam"})



```
switched to db music
> db.songdetails.find().pretty()
{
        "_id" : ObjectId("5c34c0fb2bcbdad52859f4ce"),
        "Song Name" : "Evano Oruvan",
        "Flim" : "Alai Payuthey",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Swamalatha"
}
{
        "_id" : ObjectId("5c34c17e2bcbdad52859f4d1"),
        "Song Name" : "Vennilave Venillave Vennaithanandi",
        "Flim" : "Minsara Kanavu",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Hariharan Sadhana Sargam"
}
{
        "_id" : ObjectId("5c34c40b2bcbdad52859f4d2"),
        "Song Name" : "Sollamal Thottu Chellum Thendral",
        "Flim" : "Dheena",
        "Music Director" : "Yuvan Shankar Raja",
        "Singer" : "Hariharan"
}
{
        "_id" : ObjectId("5c35c74954cc540125e80133"),
        "Song Name" : "thaniye Thananthaniye",
        "Flim" : "Rythm",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Shankar Mahadevan"
}
{
        "_id" : ObjectId("5c35c75954cc540125e80134"),
        "Song Name" : "Roja Poonthottam",
        "Flim" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : "Unnikirishnan,Anuradha Sriram"
}
> db.songdetails.remove({"Song Name":"Roja Poonthottam"})
WriteResult({ "nRemoved" : 1 })
```

8.Add new song which is your favourite.

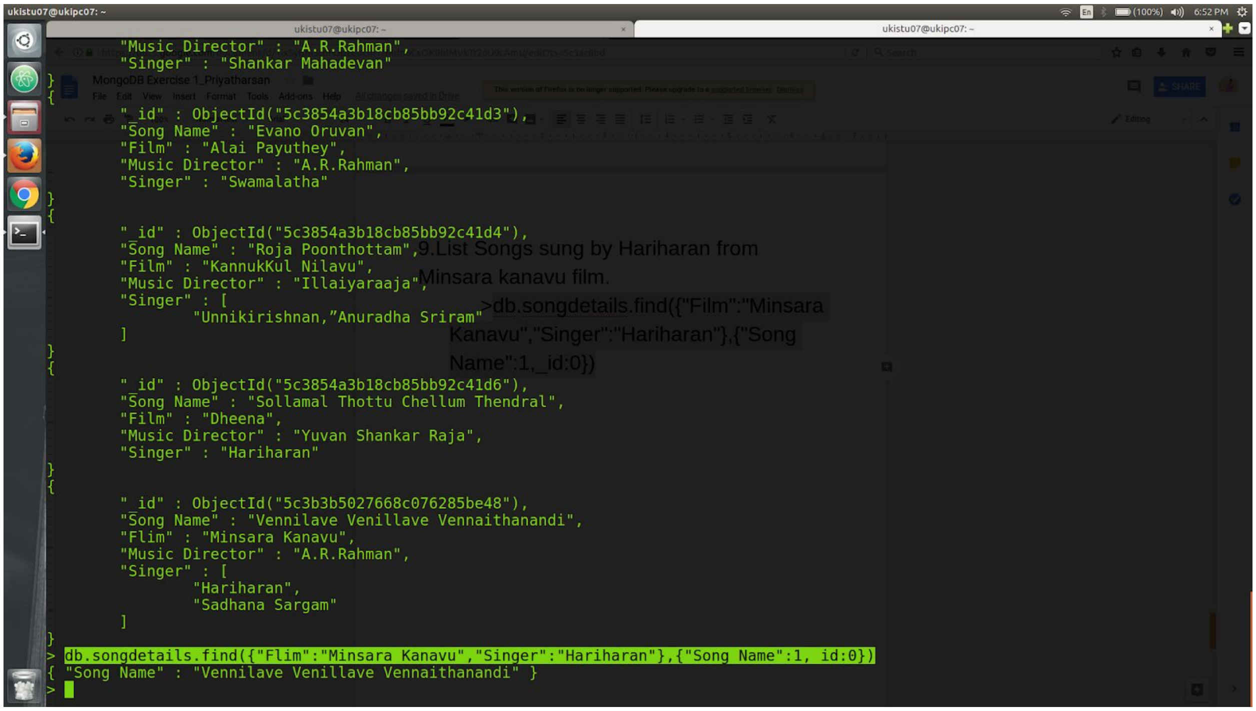
>db.songdetails.insert({"Song Name":"Roja Poonthottam","Film":"KannukKul Nilavu","Music Director":"Illaiyaraaja","Singer": "Unnikirishnan,Anuradha Sriram"})



```
        "_id" : ObjectId("5c383f425ffeb07eefdbca94"),
        "Song Name" : "Sollamal Thottu Chellum Thendral",
        "Film" : "Dheena",
        "Music Director" : "Yuvan Shankar Raja",
        "Singer" : "Hariharan"
}
> db.dropDatabase()
{ "dropped" : "music", "ok" : 1 }
> use music
switched to db music
> db.createCollection("songdetails")
{ "ok" : 1 }
> show collection
2019-01-11T12:34:19.592+0530 E QUERY    [thread1] Error: don't know how to show [collection] :
shellHelper.show@src/mongo/shell/utils.js:997:11
shellHelper@src/mongo/shell/utils.js:750:15
@(shellhelp2):1:1
> show Collection
2019-01-11T12:34:30.662+0530 E QUERY    [thread1] Error: don't know how to show [Collection] :
shellHelper.show@src/mongo/shell/utils.js:997:11
shellHelper@src/mongo/shell/utils.js:750:15
@(shellhelp2):1:1
> db
music
> show collections
songdetails
> db.songdeatils.insert(db.songdetails.insert([{"Song Name":"thaniye Thananthaniye","Film":"Rythm","Music Director":"A.R.Rahman","S
inger":"Shankar Mahadevan"},{"Song Name":"Evano Oruvan","Film":"Alai Payuthey","Music Director":"A.R.Rahman","Singer":"Swamalatha"}
,{"Song Name":"Roja Poonthottam","Film":"KannukKul Nilavu","Music Director":"Illaiyaraaja","Singer": "Unnikirishnan,Anuradha Sriram
"},{"Song Name":"Vennilave Venillave Vennaithanandi","Film":"Minsara Kanavu","Music Director":"A.R.Rahman","Singer": "Hariharan Sad
hana Sargam"},{"Song Name":"Sollamal Thottu Chellum Thendral","Film":"Dheena","Music Director":"Yuvan Shankar Raja","Singer":"Harih
aran"}]))
WriteResult({ "nInserted" : 1 })
> db.songdetails.remove({"Song Name":"Roja Poonthottam"})
WriteResult({ "nRemoved" : 1 })
> db.songdetails.insert({"Song Name":"Roja Poonthottam","Film":"KannukKul Nilavu","Music Director":"Illaiyaraaja","Singer": "Unniki
rishnan,Anuradha Sriram"})
WriteResult({ "nInserted" : 1 })
>
```

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

>db.songdetails.find({"Flim":"Minsara Kanavu","Singer":"Hariharan"},{"Song Name":1,_id:0})

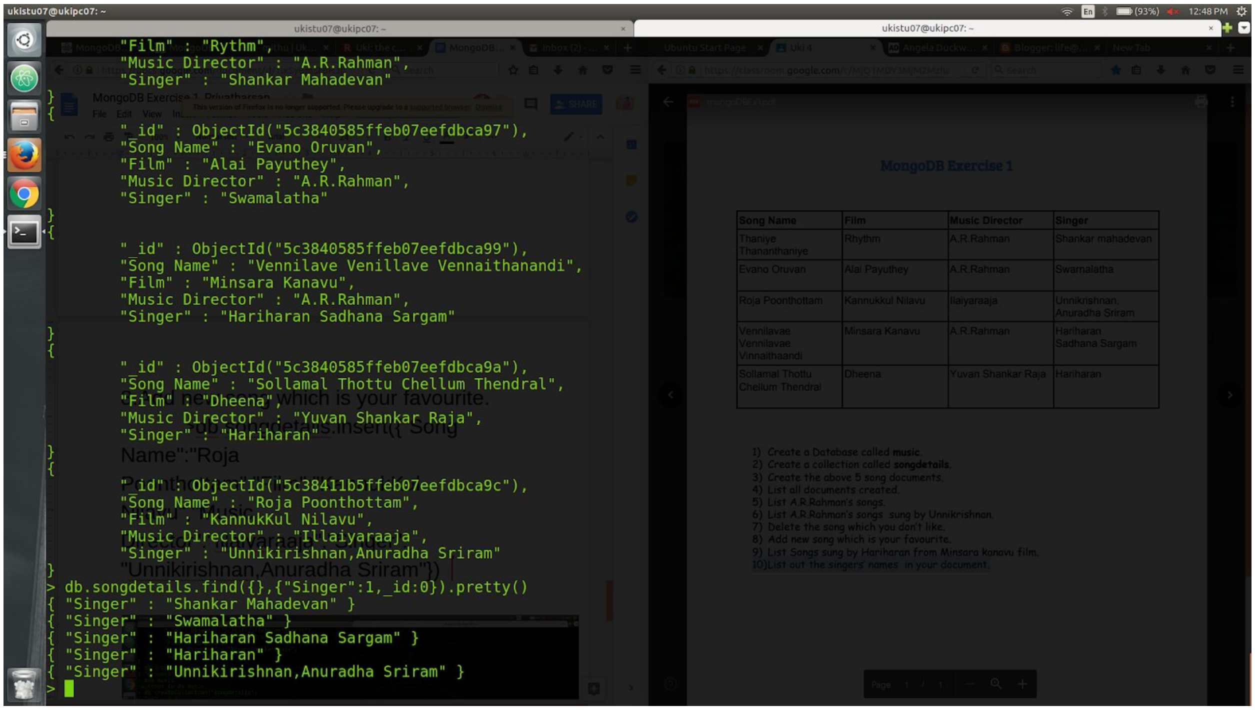


```
        "Music Director" : "A.R.Rahman",
        "Singer" : "Shankar Mahadevan"
}
{
        "_id" : ObjectId("5c3854a3b18cb85bb92c41d3"),
        "Song Name" : "Evano Oruvan",
        "Film" : "Alai Payuthey",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Swamalatha"
}
{
        "_id" : ObjectId("5c3854a3b18cb85bb92c41d4"),
        "Song Name" : "Roja Poonthottam",
        "Film" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : [
                "Unnikirishnan,”Anuradha Sriram"
        ]
}
{
        "_id" : ObjectId("5c3854a3b18cb85bb92c41d6"),
        "Song Name" : "Sollamal Thottu Chellum Thendral",
        "Film" : "Dheena",
        "Music Director" : "Yuvan Shankar Raja",
        "Singer" : "Hariharan"
}
{
        "_id" : ObjectId("5c3b3b5027668c076285be48"),
        "Song Name" : "Vennilave Venillave Vennaithanandi",
        "Flim" : "Minsara Kanavu",
        "Music Director" : "A.R.Rahman",
        "Singer" : [
                "Hariharan",
                "Sadhana Sargam"
        ]
}
> db.songdetails.find({"Flim":"Minsara Kanavu","Singer":"Hariharan"},{"Song Name":1, _id:0})
{ "Song Name" : "Vennilave Venillave Vennaithanandi" }
>
```

9.List Songs sung by Hariharan from Minsara kanavu film.
>db.songdetails.find({"Film":"Minsara Kanavu","Singer":"Hariharan"},{"Song Name":1,_id:0})

10.List out the singers’ names in your document.

>db.songdetails.find({},{"Singer":1, _id:0}).pretty()



```
        "Film" : "Rythm",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Shankar Mahadevan"
}
{
        "_id" : ObjectId("5c3840585ffeb07eefdbca97"),
        "Song Name" : "Evano Oruvan",
        "Film" : "Alai Payuthey",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Swamalatha"
}
{
        "_id" : ObjectId("5c3840585ffeb07eefdbca99"),
        "Song Name" : "Vennilave Venillave Vennaithanandi",
        "Film" : "Minsara Kanavu",
        "Music Director" : "A.R.Rahman",
        "Singer" : "Hariharan Sadhana Sargam"
}
{
        "_id" : ObjectId("5c3840585ffeb07eefdbca9a"),
        "Song Name" : "Sollamal Thottu Chellum Thendral",
        "Film" : "Dheena",
        "Music Director" : "Yuvan Shankar Raja",
        "Singer" : "Hariharan"
}
{
        "_id" : ObjectId("5c38411b5ffeb07eefdbca9c"),
        "Song Name" : "Roja Poonthottam",
        "Film" : "KannukKul Nilavu",
        "Music Director" : "Illaiyaraaja",
        "Singer" : "Unnikirishnan,Anuradha Sriram"
}
> db.songdetails.find({},{"Singer":1,_id:0}).pretty()
{ "Singer" : "Shankar Mahadevan" }
{ "Singer" : "Swamalatha" }
{ "Singer" : "Hariharan Sadhana Sargam" }
{ "Singer" : "Hariharan" }
{ "Singer" : "Unnikirishnan,Anuradha Sriram" }
>
```

MongoDB Exercise 1

Song Name	Film	Music Director	Singer
Thaniye Thananthaniye	Rhythm	A.R.Rahman	Shankar mahadevan
Evano Oruvan	Alai Payuthey	A.R.Rahman	Swarnalatha
Roja Poonthottam	Kannukkul Nilavu	Ilaiyaraaja	Unnikrishnan, Anuradha Sriram
Vennilavae Vennilavae Vinnaithaandi	Minsara Kanavu	A.R.Rahman	Hariharan Sadhana Sargam
Sollamal Thottu Chellum Thendral	Dheena	Yuvan Shankar Raja	Hariharan

1) Create a Database called **music**.
2) Create a collection called **songdetails**.
3) Create the above 5 song documents.
4) List all documents created.
5) List A.R.Rahman's songs.
6) List A.R.Rahman's 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.