

1. And 2.

```

ukistu06@UkiPc06:~
08 03:23:25 UkiPc06 systemd[1]: Started MongoDB Database Server.
ukistu06@UkiPc06:~$ mongo
MongoDB shell version v4.2.2
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("77492b31-68c5-4d7f-b038-62681da07ae8") }
MongoDB server version: 4.2.2
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
http://docs.mongodb.org/
Questions? Try the support group
http://groups.google.com/group/mongodb-user
Server has startup warnings:
2020-01-08T03:23:25.324+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2020-01-08T03:23:25.324+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem
2020-01-08T03:23:25.324+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2020-01-08T03:23:25.324+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.
2020-01-08T03:23:25.324+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2020-01-08T03:23:25.324+0530 I CONTROL [initandlisten] ** We suggest setting it to 'never'
MongoDB Enterprise > dbs
2020-01-08T03:23:55.809+0530 E QUERY [js] uncaught exception: ReferenceError: dbs is not defined :
@shell:1:1
MongoDB Enterprise > showdbs
2020-01-08T03:24:00.013+0530 E QUERY [js] uncaught exception: ReferenceError: showdbs is not defined :
@shell:1:1
MongoDB Enterprise > show dbs
admin 0.000GB
config 0.000GB
local 0.000GB
MongoDB Enterprise > use music
switched to db music
MongoDB Enterprise > db.createCollection('songdetails')
{ "ok" : 1 }
MongoDB Enterprise > show collections
songdetails
MongoDB Enterprise >

```

3.

```
kistuko06@UkiPc:~  
ge engine  
2020-01-08T03:23:25.324+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem  
2020-01-08T03:23:32.110+0530 I CONTROL [initandlisten]  
2020-01-08T03:23:32.111+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.  
2020-01-08T03:23:32.111+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.  
2020-01-08T03:23:32.111+0530 I CONTROL [initandlisten]  
2020-01-08T03:23:32.111+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.  
2020-01-08T03:23:32.111+0530 I CONTROL [initandlisten] ** We suggest setting it to 'never'  
2020-01-08T03:23:32.111+0530 I CONTROL [initandlisten]  
MongoDB Enterprise > dbs  
2020-01-08T03:23:55.809+0530 E QUERY [js] uncaught exception: ReferenceError: dbs is not defined :  
@(shell):1:1  
MongoDB Enterprise > showdbs  
2020-01-08T03:24:00.013+0530 E QUERY [js] uncaught exception: ReferenceError: showdbs is not defined :  
@(shell):1:1  
MongoDB Enterprise > show dbs  
admin 0.000GB  
config 0.000GB  
local 0.000GB  
MongoDB Enterprise > use music  
switched to db music  
MongoDB Enterprise > db.createCollection('songdetails')  
{ "ok" : 1 }  
MongoDB Enterprise > show collections  
songdetails  
MongoDB Enterprise > db.songdetails.insert({"Song Name":"Thaniyea Thananathaniyea", "Film":"Rhythm","Music Director":"A.R.Rahman","Singer":"ShankarMahadevan"})  
2020-01-08T05:47:37.018+0530 E QUERY [js] uncaught exception: SyntaxError: "" literal not terminated before end of script :  
@(shell):1:139  
MongoDB Enterprise > db.songdetails.insert({"Song Name":"Thaniyea Thananathaniyea", "Film":"Rhythm","Music Director":"A.R.Rahman","Singer":"ShankarMahadevan"})  
WriteResult({ "nInserted" : 1 })  
MongoDB Enterprise > db.songdetails.find()  
{ "_id" : ObjectId("5e15faa6d0be719e1651"), "Song Name" : "Thaniyea Thananathaniyea", "Film" : "Rhythm", "Music Director" : "A.R.Rahman",  
"Singer" : "ShankarMahadevan" }  
MongoDB Enterprise > db.songdetails.insert({"Song Name":"Evano Oruvan", "Film":"Alai Payuthey","Music Director":"A.R.Rahman","Singer":"Swamalat  
ha"})  
WriteResult({ "nInserted" : 1 })  
MongoDB Enterprise > db.songdetails.insert({"Song Name":"Roja Poonthoodam", "Film":"Kannukkul Nilavu","Music Director":"Illayaraja","Singer":  
"Unnikrishnan","Anuradha Siram"})  
WriteResult({ "nInserted" : 1 })  
MongoDB Enterprise > 
```

Song Name	Film	Music Director	Singer
Thaniyea	Rhythm	A.R.Rahman	Shankar mahadevan
Alai Payuthey		A.R.Rahman	Swamalatha
Thodu	Dhruva	Yuvan Shankar Raja	Haritharan

4.

ukistu06@UKIPc06: ~

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

```
{
  "_id" : ObjectId("5e151faa6ae6d0be719e1651"),
  "Song Name" : "Thaniyea Thananthathaniyea",
  "Film" : "Rhythm",
  "Music Director" : "A.R.Rahman",
  "Singer" : "ShankarMahadevan"
}
```

MongoDB Exercise 3

Name	Film	Music Director	Singer
Thaniyea Thananthathaniyea	Rhythm	A.R.Rahman	Shankar mahadevan
Evano Oruvan	Alai Payuthey	A.R.Rahman	Swamalatha
Illayaraaja	Kannukkul Nilavu	Unnikrishnan, Anuradha Srinam	
Vennilavae Vennilvae Vinnaitaandi	Minsara Kanavu	A.R.Rahman	Haritharan, Sadhana Sargam
Sollamal Thottu Chellum Thendral	Dheena	Yuvan Shankar Raja	Haritharan

MongoDB Enterprise >

5.

ukistu06@UKIPc06: ~

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

```
{
  "_id" : ObjectId("5e151faa6ae6d0be719e1651"),
  "Song Name" : "Thaniyea Thananthathaniyea",
  "Film" : "Rhythm",
  "Music Director" : "A.R.Rahman",
  "Singer" : "ShankarMahadevan"
}
```

Analyze Query Performance

The cursor .explain("executionStats") and the db.collection.explain("executionStats") methods provide statistics about the performance of a query. This data output can be useful in measuring if and how a query uses an index.

db.collection.explain() provides information on the execution of other operations, such as db.collection.update(). See db.collection.explain() for details.

Evaluate the Performance of a Query

Consider a collection **inventory** with the following documents:

```
{ "item": 1, "name": "M1", "type": "Food", "quantity": 100 }
{ "item": 2, "name": "M2", "type": "Food", "quantity": 100 }
```

MongoDB Enterprise >

6.

MongoDB Enterprise > db.songdetails.find({\$and:[{"Music Director":"A.R.Rahman"}, {"Singer":"Unnikrishnan"}]}).pretty()

MongoDB Enterprise >

MongoDB (Documentation) > MONGODB MANUAL > Version 4.2 (current) > Query Documents > Query on Embedded/Nested Documents

Specify Conditions Using Query Operators

A query filter document can use the query operators to specify conditions in the following form:

```
{ <field> : { <operator> : <value> 1, ... 1 }
```

The following example retrieves all documents from the `inventory` collection where `status` equals either "A" or "B":

```
db.inventory.find( { status: { $in: [ "A", "B" ] } } )
```

NOTE:

Although you can express this query using the `$or` operator, use the `$in` operator rather than the `$or` operator when performing equality checks on the same field.

The operation corresponds to the following SQL statement:

YES NO

7.

```
" id" : ObjectId("5e151faa6ae6d0be719e1651"),
"Song Name" : "Thanlyea Thananthathaniyea",
"Film" : "Rhythm",
"Music Director" : "A.R.Rahman",
"Singer" : "ShankarMahadevan"

" id" : ObjectId("5e1520d6ae6d0be719e1652"),
"Song Name" : "Evano Oruvan",
"Film" : "Alai Payuthey",
"Music Director" : "A.R.Rahman",
"Singer" : "Swamalatha"

" id" : ObjectId("5e1520a06ae6d0be719e1653"),
"Song Name" : "Raja Poonthoodan",
"Film" : "Kannukkul Nilavu",
"Music Director" : "Illayaraja",
"Singer" : [
  "Unnikrishnan",
  "Anuradha Sritam"
]

" id" : ObjectId("5e152116ae6d0be719e1654"),
"Song Name" : "Vennilavae Vennilvae Vinnaitaandi",
"Film" : "Minsara Kanavu",
"Music Director" : "A.R.Rahman",
"Singer" : [
  "Hariharan",
  "Sadhana Sargam"
]

" id" : ObjectId("5e152168ae6d0be719e1655"),
"Song Name" : "Sollamal Thottu Chellum Thendral",
"Film" : "Dheena",
"Music Director" : "Yuvan Shankar Raja",
"Singer" : "Hariharan"

MongoDB Enterprise > db.songdetails.remove({"Film":"Kannukkul Nilavu"})
WriteResult({ "nRemoved" : 1 })
MongoDB Enterprise >
```

Song Name	Film	Music Director	Singer
Rhythm		A.R. Rahman	Shankar Mahadevan
Evano Oruvan	Alai Payuthey	A.R. Rahman	Swamalatha
Raja Poonthoodan	Kannukkul Nilavu	Illayaraja	Unnikrishnan, Anuradha Sritam
Vennilavae Vennilvae Vinnaitaandi	Minsara Kanavu	A.R. Rahman	Hariharan, Sadhana Sargam
Sollamal Thottu Chellum Thendral	Dheena	Yuvan Shankar Raja	Hariharan

8.

ukistu06@UkiPC06: ~

```

"Music Director" : "A.R. Rahman",
"Singer" : "ShankarMahadevan"
}
{
  "id" : ObjectId("5e1520d06ae6d0be719e1652"),
  "Song Name" : "Evano Oruvan",
  "Film" : "Alai Payuthey",
  "Music Director" : "A.R. Rahman",
  "Singer" : "Swamalatha"
}
{
  "id" : ObjectId("5e1520a06ae6d0be719e1653"),
  "Song Name" : "Roja Poonthoodam",
  "Film" : "Kannukkul Nilavu",
  "Music Director" : "Illayaraja",
  "Singer" : [
    "Unnikrishnan",
    "Anuradha Sriram"
  ]
}
{
  "id" : ObjectId("5e152116ae6d0be719e1654"),
  "Song Name" : "Vennilavae Vennilavae Vinnaithaandi",
  "Film" : "Minsara Kanavu",
  "Music Director" : "A.R. Rahman",
  "Singer" : [
    "Hariharan",
    "Sadhana Sargam"
  ]
}
{
  "id" : ObjectId("5e1521686ae6d0be719e1655"),
  "Song Name" : "Sollanai Thottu Chellum Thendral",
  "Film" : "Dheena",
  "Music Director" : "Yuvan Shankar Raja",
  "Singer" : "Hariharan"
}
MongoDB Enterprise > db.songdetails.remove({"Film": "Kannukkul Nilavu"})
WriteResult({ "nRemoved" : 1 })
MongoDB Enterprise > db.songdetails.insert({"Song Name": "Chakarakady", "Film": "Meesaya Murukku", "Music Director": "Hiphop Tamizha", "Singer": "Hiphop Tamizha"})
WriteResult({ "nInserted" : 1 })
MongoDB Enterprise >

```

Meesaya Murukku
Soundtrack album by Hiphop Tamizha

Released: 14 April 2017
Recorded: 2014-2017

Genre: Feature film soundtrack
Language: Tamil
Label: Thirai Music, Thirai Music, Aasi Music
Producer: Hiphop Tamizha

Hiphop Tamizha chronology

Album	Year
Meesaya Murukku	2017
Chakarakady	2018

Song Details

Song Name	Film	Music Director	Singer(s)	Length
Evano Oruvan	Alai Payuthey	A.R. Rahman	ShankarMahadevan	4:21
Roja Poonthoodam	Kannukkul Nilavu	Illayaraja	Unnikrishnan, Anuradha Sriram	2:29
Vennilavae Vennilavae Vinnaithaandi	Minsara Kanavu	A.R. Rahman	Karunakrishnan, Hiphop Tamizha	4:05
Sollanai Thottu Chellum Thendral	Dheena	Yuvan Shankar Raja	Hariharan	3:17
Chakarakady	Meesaya Murukku	Hiphop Tamizha	Hiphop Tamizha	3:05
Sadhana Sargam	Minsara Kanavu	A.R. Rahman	Hariharan, Sadhana Sargam	3:42
Chellum Thendral	Dheena	Yuvan Shankar Raja	Hariharan	3:09
Evano Oruvan	Alai Payuthey	A.R. Rahman	ShankarMahadevan	4:21
Total length:				28:38

9.

10.

ukistu06@UkiPC06: ~

```

MongoDB Enterprise > db.songdetails.find({}, {"Singer": true})
{ "_id" : ObjectId("5e151fa0ae6d0be719e1651"), "Singer" : "ShankarMahadevan" }
{ "_id" : ObjectId("5e1520d06ae6d0be719e1652"), "Singer" : "Swamalatha" }
{ "_id" : ObjectId("5e152116ae6d0be719e1654"), "Singer" : [ "Hariharan", "Sadhana Sargam" ] }
{ "_id" : ObjectId("5e1521686ae6d0be719e1655"), "Singer" : "Hariharan" }
{ "_id" : ObjectId("5e1523c86ae6d0be719e1656"), "Singer" : "Hiphop Tamizha" }
MongoDB Enterprise >

```

Chellum Thendral

Song Name	Film	Music Director	Singer(s)	Length
Chellum Thendral	Dheena	Yuvan Shankar Raja	Hariharan	3:09

- 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.