

MongoDB Data Analysis Using **sample_mflix** Dataset

By: Saumya Raghuvanshi(055040)

Shagun Seth(055042)

1. Find movies released after 2000:

The screenshot shows the MongoDB Compass interface for the **sample_mflix.movies** collection. At the top, it displays collection statistics: STORAGE SIZE: 19.1MB, LOGICAL DATA SIZE: 32.54MB, TOTAL DOCUMENTS: 21350, and INDEXES TOTAL SIZE: 17.02MB. Below this are tabs for Find, Indexes, Schema Anti-Patterns, Aggregation, and Search Indexes. A link to 'Generate queries from natural language in Compass' and an 'INSERT DOCUMENT' button are also present. The 'Filter' section shows a query: `{ year: { $gt: 2000 } }`. Below the query, a document is displayed with fields: `_id`, `plot`, `genres` (array), `runtime`, `metacritic`, `rated`, `cast` (array), `poster`, `title`, and `fullplot`. At the bottom, there are navigation buttons for 'PREVIOUS' and 'NEXT', and a status indicator '1-20 of many results'.

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

Generate queries from natural language in Compass [↗](#) [INSERT DOCUMENT](#)

Filter [↗](#) { year: { \$gt: 2000 } } [Reset](#) [Apply](#) [Options ▶](#)

```
_id: ObjectId('573a1393f29313caabdcdb42')
plot: "Kate and her actor brother live in N.Y. in the 21st Century. Her ex-bo..."
genres: Array (3)
runtime: 118
metacritic: 44
rated: "PG-13"
cast: Array (4)
poster: "https://m.media-amazon.com/images/M/MV5BNmNlN2VlOTctYTdhMS00NzUxLTg0ZG..."
title: "Kate & Leopold"
fullplot: "Kate and her actor brother live in N.Y. in the 21st Century. Her ex-bo..."
```

[◀](#) PREVIOUS 1-20 of many results NEXT [▶](#)

2. Find movies with a rating greater than 8:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find

Indexes

Schema Anti-Patterns 0

Aggregation

Search Indexes

[Generate queries from natural language in Compass](#)

INSERT DOCUMENT

Filter

{ "imdb.rating": { \$gt: 8 } }

Reset

Apply

Options

QUERY RESULTS: 1-20 OF MANY

```
{
  "_id": ObjectId('573a1391f29313caabcd6e2a'),
  "plot": "A newly wedded couple attempt to build a house with a prefabricated ki...",
  "genres": Array (2)
  "runtime": 25
  "cast": Array (2)
  "num_mflix_comments": 0
  "title": "One Week"
  "fullplot": "Buster and Sybil exit a chapel as newlyweds. Among the gifts is a port..."
}
```



PREVIOUS

1-20 of many results

NEXT



3. Find movies from the "Action" genre:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find

Indexes

Schema Anti-Patterns 0

Aggregation

Search Indexes

[Generate queries from natural language in Compass](#)

INSERT DOCUMENT

Filter

{ genres: "Action" }

Reset

Apply

Options

QUERY RESULTS: 1-20 OF MANY

```
{
  "_id": ObjectId('573a1390f29313caabcd5293'),
  "plot": "Young Pauline is left a lot of money when her wealthy uncle dies. Howe...",
  "genres": Array (1)
  "runtime": 199
  "cast": Array (4)
  "num_mflix_comments": 0
  "poster": "https://m.media-amazon.com/images/M/MV5BMzgxODk1Mzk2Ml5BMl5BanBnXkFtZT..."
}
```



PREVIOUS

1-20 of many results

NEXT



4. Find movies with a runtime between 90 and 150 minutes:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find

Indexes

Schema Anti-Patterns 0

Aggregation

Search Indexes

[Generate queries from natural language in Compass](#)

INSERT DOCUMENT

Filter

{ runtime: { \$gte: 90, \$lte: 150 } }

Reset

Apply

Options ▶

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('573a1391f29313caabcd70b4')
plot: "An extended family split up in France and Germany find themselves on o..."
genres: Array (3)
runtime: 150
rated: "PASSED"
cast: Array (4)
num_mflix_comments: 1
poster: "https://m.media-amazon.com/images/M/MV5B0TU1ODQyYTctODhkNy00YWVmLWE2Yz..."
```



PREVIOUS

1-20 of many results

NEXT



5. Find movies that has at least 3 cast members:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find

Indexes

Schema Anti-Patterns 0

Aggregation

Search Indexes

[Generate queries from natural language in Compass](#)

INSERT DOCUMENT

Filter

{ "cast.3": { \$exists: true } }

Reset

Apply

Options ▶

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('573a1390f29313caabcd42e8')
plot: "A group of bandits stage a brazen train hold-up, only to find a determ..."
genres: Array (2)
runtime: 11
cast: Array (4)
poster: "https://m.media-amazon.com/images/M/MV5BMTU3NjE5NzYtYTtyNS00MDVmLWIwYj..."
title: "The Great Train Robbery"
fullplot: "Among the earliest existing films in American cinema - notable as the ..."
languages: Array (1)
```



PREVIOUS

1-20 of many results

NEXT



6. Find movies released between 1990 and 2000:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Generate queries from natural language in Compass [↗](#) INSERT DOCUMENT

Filter [↗](#)

{ year: { \$gte: 1990, \$lte: 2000 } }

Reset Apply Options ▶

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('573a1396f29313caabce3d17')
plot: "Set in the late 1940s, the film concerns the treatment of suspect "bou..."
▶ genres: Array (3)
  runtime: 94
▶ cast: Array (4)
  num_mflix_comments: 0
poster: "https://m.media-amazon.com/images/M/MV5BMjY2NzMyNmEtZmM0Ny00YzY4LWFlNW..."
title: "Larks on a String"
```

< PREVIOUS

1-20 of many results

NEXT >

7. Find movies with an IMDb rating less than 5:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Generate queries from natural language in Compass [↗](#) INSERT DOCUMENT

Filter [↗](#)

{ "imdb.rating": { \$lt: 5 } }

Reset Apply Options ▶

QUERY RESULTS: 1-20 OF MANY

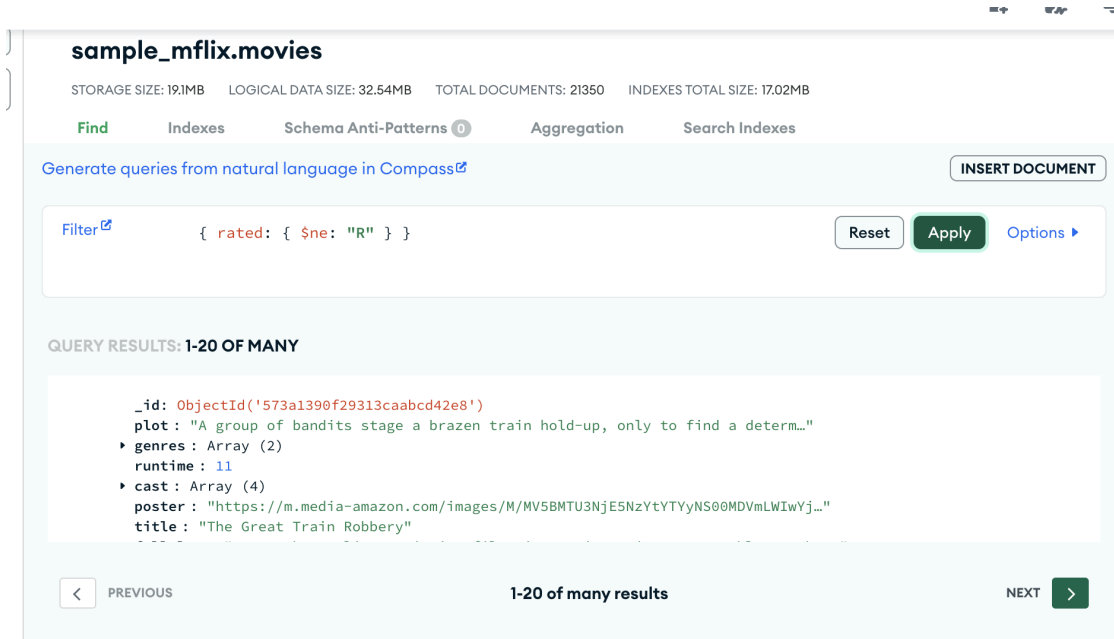
```
_id: ObjectId('573a1393f29313caabcedf75')
▶ countries: Array (1)
▶ genres: Array (1)
  runtime: 142
▶ cast: Array (4)
  title: "La mies es mucha"
lastupdated: "2014-12-08 00:26:46.587000000"
▶ languages: Array (1)
```

< PREVIOUS

1-20 of many results

NEXT >

8. Find movies that are not rated "R":



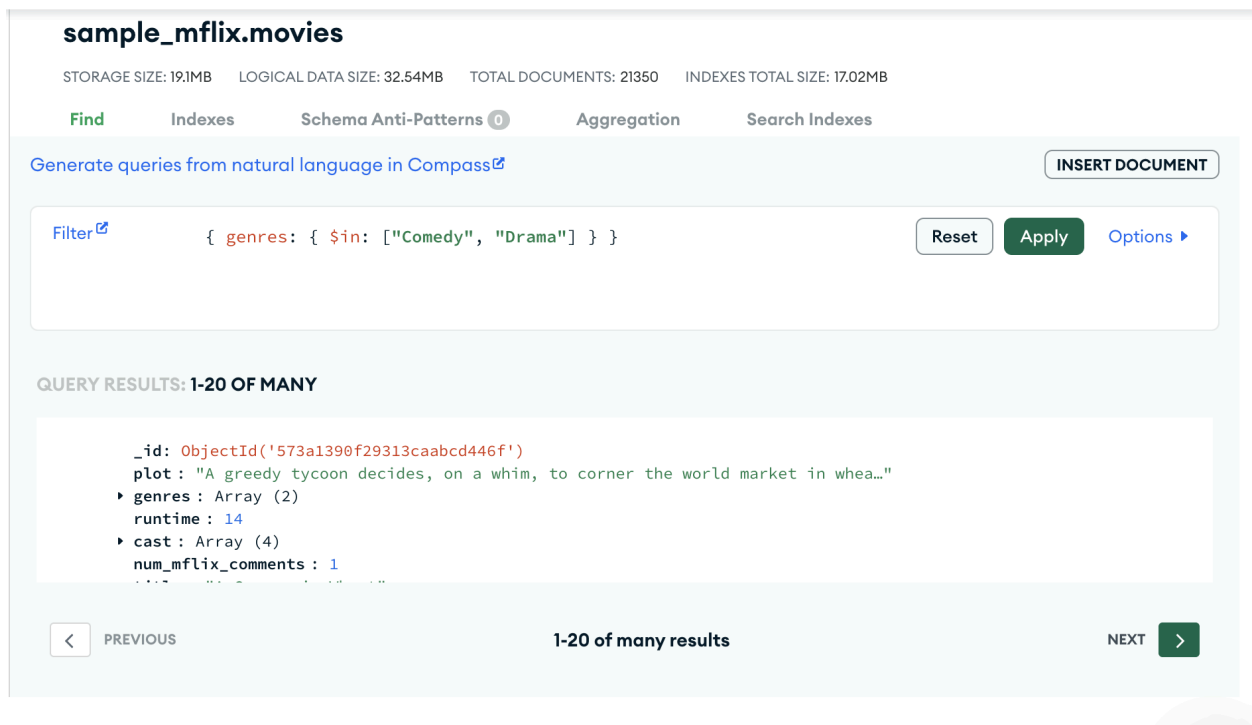
The screenshot shows the MongoDB Compass interface for the **sample_mflix.movies** collection. The top bar displays collection statistics: STORAGE SIZE: 19.1MB, LOGICAL DATA SIZE: 32.54MB, TOTAL DOCUMENTS: 21350, and INDEXES TOTAL SIZE: 17.02MB. The navigation tabs include Find, Indexes, Schema Anti-Patterns (0), Aggregation, and Search Indexes. A link to "Generate queries from natural language in Compass" is present, along with an "INSERT DOCUMENT" button.

The **Filter** tab is active, showing a query: `{ rated: { $ne: "R" } }`. Below the query, the **QUERY RESULTS: 1-20 OF MANY** section displays a document for "The Great Train Robbery":

```
{
  "_id": ObjectId('573a1390f29313caabcd42e8'),
  "plot": "A group of bandits stage a brazen train hold-up, only to find a determ...",
  "genres": Array (2)
  runtime: 11
  cast: Array (4)
  poster: "https://m.media-amazon.com/images/M/MV5BMTU3NjE5NzYtYTYyNS00MDVmLWIwYj...
  title: "The Great Train Robbery"
}
```

At the bottom, navigation controls show "< PREVIOUS", "1-20 of many results", and "NEXT >".

9. Find movies that are either Comedy or Drama:



The screenshot shows the MongoDB Compass interface for the **sample_mflix.movies** collection, similar to the previous one. The **Filter** tab is active, showing a query: `{ genres: { $in: ["Comedy", "Drama"] } }`.

The **QUERY RESULTS: 1-20 OF MANY** section displays a document for "A greedy tycoon decides, on a whim, to corner the world market in whea...":

```
{
  "_id": ObjectId('573a1390f29313caabcd446f'),
  "plot": "A greedy tycoon decides, on a whim, to corner the world market in whea...",
  "genres": Array (2)
  runtime: 14
  cast: Array (4)
  num_mflix_comments: 1
}
```

At the bottom, navigation controls show "< PREVIOUS", "1-20 of many results", and "NEXT >".

10. Find movies that are neither Horror nor Thriller:

sample_mflix.movies

STORAGE SIZE: 19.1MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21350 INDEXES TOTAL SIZE: 17.02MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Generate queries from natural language in Compass

INSERT DOCUMENT

Filter { genres: { \$nin: ["Horror", "Thriller"] } }

Reset Apply Options

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('573a1390f29313caabcd42e8')
plot: "A group of bandits stage a brazen train hold-up, only to find a determ..."
▶ genres: Array (2)
runtime: 11
▶ cast: Array (4)
```

PREVIOUS 1-20 of many results NEXT

11. CREATE : Insert a new movie into the database

r... Settings Access Manager Billing

All Clusters Get Help

Data Services Charts

ClusterO

Overview Real Time

DATABASES: 1 COLLECTIONS: 6

+ Create Database

Search Namespaces

sample_mflix

comments

embedded_movies

movies

sessions

theaters

users

Insert Document

To collection movies

VIEW

```
1 {
2   "title": "Inception",
3   "year": 2010,
4   "runtime": 148,
5   "cast": ["Leonardo DiCaprio", "Joseph Gordon-Levitt", "Elli
6   "genres": ["Action", "Sci-Fi", "Thriller"],
7   "metacritic": 86
8 }
9
```

Cancel Insert

VERSION 8.0.5 REGION AWS Mumbai

Performance Advisor Online Archive

VISUALIZE YOUR DATA

INDEXES TOTAL SIZE: 17.02MB

Search Indexes

INSERT

Reset Apply

the dangerous art of..."

[Find](#)[Indexes](#)[Schema Anti-Patterns 0](#)[Aggregation](#)[Search Indexes](#)

Generate queries from natural language in Compass[↗](#)

INSERT DOCUMENT

[Filter ↗](#)

{ "title": "Inception" }

ResetApplyOptions ▶

QUERY RESULTS: 1-2 OF 2

```
  _id: ObjectId('573a13c5f29313caabd6ee61')
  fullplot: "Dom Cobb is a skilled thief, the absolute best in the dangerous art of..."
  ▶ imdb: Object
    year: 2010
    plot: "A thief who steals corporate secrets through use of dream-sharing tech..."
  ▶ genres: Array (3)
    rated: "PG-13"
    metacritic: 74
    title: "Inception"
    lastupdated: "2015-09-12 00:29:01.430000000"
  ▶ languages: Array (3)
  ▶ writers: Array (1)
    type: "movie"
    ...
```

12. UPDATE

Update Metacritic score of "Inception"

[Find](#)[Indexes](#)[Schema Anti-Patterns 0](#)[Aggregation](#)[Search Indexes](#)

Generate queries from natural language in Compass[↗](#)

INSERT DOCUMENT

[Filter ↗](#)

{ "title": "Inception" }

ResetApplyOptions ▶

QUERY RESULTS: 1-2 OF 2

```
  _id: ObjectId('573a13c5f29313caabd6ee61')
  fullplot: "Dom Cobb is a skilled thief, the absolute best in the dangerous art of..."
  ▶ imdb: Object
    year: 2010
    plot: "A thief who steals corporate secrets through use of dream-sharing tech..."
  ▶ genres: Array (3)
    rated: "PG-13"
    metacritic: 74
    title: "Inception"
    lastupdated: "2015-09-12 00:29:01.430000000"
  ▶ languages: Array (3)
  ▶ writers: Array (1)
    type: "movie"
    ...
```

4

year : 2010

int32

5

plot : "A thief who steals corporate secrets through use of dream-sharing tech"

String

6

▸ genres : Array (3)

Array

7

rated : "PG-13"

String

8

metacritic : 96

Int32

9

title : "Inception"

String

10

lastupdated : "2015-09-12 00:29:01.430000000"

String

11

▸ languages : Array (3)

Array

12

▸ writers : Array (1)

Array

```

plot : "A thief who steals corporate secrets through use of dream-sharing tech..."
▸ genres : Array (3)
rated : "PG-13"
metacritic : 96
title : "Inception"
lastupdated : "2015-09-12 00:29:01.430000000"
▸ languages : Array (3)
▸ writers : Array (1)
type : "movie"
tomatoes : Object

```

13. Aggregation Queries in MongoDB Atlas:

Find the average IMDb rating per genre:

```
[
  { "$unwind": "$genres" },
  { "$group": { "_id": "$genres", "avg_imdb_rating": { "$avg": "$imdb.rating" } } },
  { "$sort": { "avg_imdb_rating": -1 } }
]
```


STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline Pipeline Sunwind \$group \$sort

+ CREATE NEW

PREVIEW STAGES TEXT

Stage 1 Sunwind

```
1 /**
2  * path: Path to the array field.
3  * includeArrayIndex: Optional name for in
4  * preserveNullAndEmptyArrays: Optional
5  * toggle to unwind null and empty value
6  */
7
8  { "path": "$genres" }
```

Output after \$unwind stage (Sample of 10 documents)

```
title : "The Great Train Robbery"
languages : Array (1)
released : 1903-12-01T00:00:00.000+00:00
directors : Array (1)
year : 1903
tomatoes : Object
  rated : "TV-G"
imdb : Object
  type : "movie"
  num_mfli
  title : "The Great Train Robbery"
  released : 1903-12-01T00:00:00.000+00:00
  directors : Array (1)
  year : 1903
  tomatoes : Object
    rated : "TV-G"
  imdb : Object
    type : "movie"
```

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline Pipeline Sunwind \$group \$sort

+ CREATE NEW

PREVIEW STAGES TEXT

Stage 2 \$group

```
1
2 id: The id of the group.
3 fieldN: The first field name.
4
5
6
7 id: "$genres",
8 avg_imdb_rating: { "$avg": "$imdb.rating" }
9
```

Output after \$group stage (Sample of 10 documents)

```
_id: "Thriller"
avg_imdb_rating: 6.304498977505112
_id: "An
avg_imdb
```

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline ▾

\$unwind

\$group

\$sort

+ CREATE NEW </>

PREVIEW

STAGES

</> TEXT



Stage 3 (\$sort) 🔵

⌵ ⋮

```
1 ▾ /**
2   * Provide any number of field/order pairs
3   */
4 ▾ {
5   "avg_imdb_rating": -1
6 }
```



Output after \$sort stage (Sample of 10 documents)

```
_id: "Film-Noir"
avg_imdb_rating : 7.397402597402598
```

```
_id: "Sh
avg_imdb
```

14. Top 10 Most Popular Movies by IMDb Rating:

This query will:

- Filter out movies with at least 1000 votes (to avoid obscure ones)
- Sort them by IMDb rating in descending order
- Show only the top 10 movies
- Display only title and IMDb rating

Pipeline  \$match+ CREATE NEW  PREVIEW  STAGES  TEXT Stage 1 \$match 

```
1 /**
2  * query: The query in MQL.
3  */
4
5  { "imdb.votes": { "$gte": 1000 } }
6
```

Output after \$match stage (Sample of 10 documents)

```
▸ directors : Array (1)
  _id: ObjectId('573a1390f29313caabcd42e8')
  plot: "A group of bandits stage a brazen train
        hold-up, only to find a determ..."
▸ genres : Array (2)
▸ cast : Array (4)
  poster: "https://m.media-
          amazon.com/images/M/MV5BMTU3NjE5NzYtYTY...
  fullplot: "Among the earliest existing films in
            American cinema - notable as the "
```

```
title : '
▸ awards :
lastupda
type : "i
_id: Obj
▸ genres :
▸ imdb : Ol
▸ tomatoes
▸ language
```

Pipeline  \$match \$sort+ CREATE NEW  PREVIEW  STAGES  TEXT Stage 2 \$sort 

```
1 /**
2  * Provide any number of field/order pairs
3  */
4
5  { "imdb.rating": -1 }
6
```

Output after \$sort stage (Sample of 10 documents)

```
plot: "The story of Easy Company of the US Army
      101st Airborne division and t..."
runtime : 705
num_mflix_comments : 0
released : 2001-09-09T00:00:00.000+00:00
lastupdated : "2015-08-31 00:04:34.187000000"
▸ cast : Array (4)
  rated : "TV-MA"
  year : 2001
```

```
fullplot
released
lastupda
type : ":"
num_mfli
_id: Obj
poster :
```

Find Indexes Schema Anti-Patterns 0 **Aggregation** Search Indexes

Pipeline \$match \$sort \$limit

+ CREATE NEW

PREVIEW **STAGES** TEXT

▼ Stage 3 \$limit

1 ▼ /**
 2 * Provide the number of documents to limit
 3 */
 4 10

Output after \$limit stage (Sample of 10 documents)

```

  ▶ languages : Array (5)
  ▶ countries : Array (2)
  ▶ tomatoes : Object
  ▶ cast : Array (4)
  num_mflix_comments : 0
  poster : "https://m.media-
    amazon.com/images/M/MV5BMTI3ODc2ODc0M15...
  title : "Band of Brothers"
  _id : ObjectId('573a139ff29313caabd003c4')
```

poster :
 rated :
 awards :
 lastupdate :
 imdb : 0
 type :
 num_mflix_comments :
 runtime :

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 **Aggregation** Search Indexes

Pipeline \$match \$sort \$limit \$project

+ CREATE NEW

PREVIEW **STAGES** TEXT

▼ Stage 4 \$project

1 ▼ /**
 2 * specifications: The fields to
 3 * include or exclude.
 4 */
 5 {
 6 "title": 1, "imdb.rating": 1, "_id": 0
 7 }

Output after \$project stage (Sample of 10 documents)

```

  title : "Band of Brothers"
  ▶ imdb : Object
```

title :
 ▶ imdb : 0

```
[
  { "$match": { "imdb.votes": { "$gte": 1000 } } },
  { "$sort": { "imdb.rating": -1 } },
  { "$limit": 10 },
  { "$project": { "title": 1, "imdb.rating": 1, "_id": 0 } }
]
```

15. INSERT a New Movie (CREATE)

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('67d726639e4001f1c9df7e38')
title: "The AI Revolution"
year: 2025
▶ genres: Array (2)
▶ imdb: Object
```

16. UPDATE an Existing Movie

sample_mflix.movies

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

[Find](#) [Indexes](#) [Schema Anti-Patterns 0](#) [Aggregation](#) [Search Indexes](#)

[Generate queries from natural language in Compass](#)

INSERT DOCUMENT

Filter { "title": "The AI Revolution" }

Reset Apply Options ▶

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('67d726639e4001f1c9df7e38')
title: "The AI Revolution"
year: 2025
▶ genres: Array (2)
▼ imdb: Object
  rating: 9.8
  votes: 5000
```

sample_mflix.movies

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Generate queries from natural language in Compass

INSERT DOCUMENT

Filter { "title": "The AI Revolution" }

Reset

Apply

Options ▶

QUERY RESULTS: 1-1 OF 1

1	_id: ObjectId('67d726639e4001f1c9df7e38')	ObjectId
2	title: "The AI Revolution"	String
3	year: 2025	Int32
4	genres: Array (2)	Array
5	0: "Sci-Fi"	String
6	1: "Drama"	String
7	imdb: Object	Object
8	rating: 9.8	Double
9	votes: 5000	Int32

Document modified.

CANCEL

UPDATE

17. Delete

sample_mflix.movies

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Generate queries from natural language in Compass

INSERT DOCUMENT

Filter { "title": "The AI Revolution" }

Reset

Apply

Options ▶

QUERY RESULTS: 1-1 OF 1

▶ _id: ObjectId('67d726639e4001f1c9df7e38')
title: "The AI Revolution"
year: 2025
▶ genres: Array (2)
▶ imdb: Object



Remove document

sample_mflix.movies

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

FindIndexesSchema Anti-Patterns0AggregationSearch Indexes

Generate queries from natural language in Compass

INSERT DOCUMENT

Filter{ "title": "The AI Revolution" }ResetApplyOptions

QUERY RESULTS: 0

18. Count Movies per Year

sample_mflix.movies

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

FindIndexesSchema Anti-Patterns0AggregationSearch Indexes

Pipeline\$group

+ CREATE NEW</>PREVIEW{} STAGES</> TEXT

Stage1\$group

```
1 /**
2  * _id: The id of the group.
3  * fieldN: The first field name.
4  */
5 {
6
7   "_id": "$year",
8   "count": { "$sum": 1 }
9
10 }
```

Output after \$group stage (Sample of 10 documents)

_id: "1996"count : 1

_id: 198count : .

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline \$group \$sort

+ CREATE NEW

PREVIEW STAGES TEXT

Stage 2 \$sort

```
1 /**
2  * Provide any number of field/order pairs
3  */
4 {
5
6   "count": -1
7
8 }
```

Output after \$sort stage (Sample of 10 documents)

_id: 2013
count: 1105

_id: 201
count: 1

19. Find the Top 5 Highest Rated Movies

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline \$match

+ CREATE NEW

PREVIEW STAGES TEXT

Stage 1 \$match

```
1 /**
2  * query: The query in MQL.
3  */
4 {
5   "imdb.rating": { "$exists": true }
6 }
```

Output after \$match stage (Sample of 10 documents)

num_mflix_comments: 0
plot: "A group of bandits stage a brazen train hold-up, only to find a determ..."
runtime: 11
cast: Array (4)
released: 1903-12-01T00:00:00.000+00:00
year: 1903
countries: Array (1)
genres: Array (2)

imdb: 0
num_mfli
lastupda
year: 1
released
rated: '
awards: '
type: "i
plot: "i

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline

\$match \$sort

+ CREATE NEW

</>

 PREVIEW {} STAGES </> TEXT ⚙

Stage 2 \$sort

🔍

1 ▾ /**
2 * Provide any number of field/order pairs
3 */
4 ▾ {
5 | "imdb.rating": -1
6 | }
7

Output after \$sort stage (Sample of 10 documents)

plot: "The remarkable love story inspired by the lives of artists Lili Elbe a..."
genres: Array (2)
runtime: 120
num_mflix_comments: 1
writers: Array (2)
type: "movie"
_id: ObjectId('573a13b8f29313caabd4d540')
cast: Array (4)

released
writers
num_mfli
countrie
fullplot
title: '
language
director

Atlas Blog Contact Sales

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes

Pipeline

\$match \$sort \$limit

+ CREATE NEW

</>

 PREVIEW {} STAGES </> TEXT ⚙

Stage 3 \$limit

🔍

1 ▾ /**
2 * Provide the number of documents to limit
3 */
4 5
5

Output after \$limit stage (Sample of 5 documents)

num_mflix_comments: 1
directors: Array (1)
lastupdated: "2015-09-01 01:17:41.067000000"
type: "movie"
runtime: 120
awards: Object
year: 2015
tomatoes: Object
_id: ObjectId('573a13b8f29313caabd4d540')

_id: Obj
genres:
title:
writers
runtime
num_mfli
language
released
director

STORAGE SIZE: 19.07MBLOGICAL DATA SIZE: 32.54MBTOTAL DOCUMENTS: 21352INDEXES TOTAL SIZE: 17.23MB

FindIndexesSchema Anti-Patterns0AggregationSearch Indexes

PipelinePipeline\$match\$sort\$limit\$project

+ CREATE NEW</>

PREVIEWSTAGES</> TEXT

Stage 4\$project

```
1 /**
2  * specifications: The fields to
3  * include or exclude.
4  */
5 {
6   | "title": 1, "imdb.rating": 1, "_id": 0
7 }
```

Output after \$project stage (Sample of 5 documents)

title : "The Danish Girl"

imdb : Object

title : "

imdb : OI

20. Find the Most Common Movie Languages

PipelinePipeline\$unwind

+ CREATE NEW</>

PREVIEWSTAGES</> TEXT

Stage 1\$unwind

```
1 /**
2  * path: Path to the array field.
3  * includeArrayIndex: Optional name for in
4  * preserveNullAndEmptyArrays: Optional
5  * toggle to unwind null and empty value
6  */
7 {
8   | "path": "$languages"
9 }
10
```

Output after \$unwind stage (Sample of 10 documents)

type : "movie"

languages : "English"

directors : Array (1)

awards : Object

title : "The Great Train Robbery"

released : 1903-12-01T00:00:00.000+00:00

imdb : Object

countries : Array (1)

tomatoes : Object

_id: Obj

plot : "

num_mfli

language

type : "I

genres :

runtime

rated : "

Atlas Playground

Find Indexes Schema Anti-Patterns 0 **Aggregation** Search Indexes

Pipeline \$unwind \$group

+ CREATE NEW

☒ PREVIEW **{}** STAGES TEXT

▼ Stage 2 \$group

```
1  /**
2   * _id: The id of the group.
3   * fieldN: The first field name.
4   */
5   {
6
7     "_id": "$languages",
8     "count": { "$sum": 1 }
9   }
```

Output after \$group stage (Sample of 10 documents)

_id: "Fur"
count : 1

_id: "Ba
count : !

STORAGE SIZE: 19.07MB LOGICAL DATA SIZE: 32.54MB TOTAL DOCUMENTS: 21352 INDEXES TOTAL SIZE: 17.23MB

Find Indexes Schema Anti-Patterns 0 **Aggregation** Search Indexes

Pipeline \$unwind \$group \$sort

+ CREATE NEW

☒ PREVIEW **{}** STAGES TEXT

▼ Stage 3 \$sort

```
1  /**
2   * Provide any number of field/order pairs
3   */
4   {
5     "count": -1
6   }
```

Output after \$sort stage (Sample of 10 documents)

_id: "English"
count : 14639

_id: "Fr
count : :

```
[{"_id": "English", "count": 8500 },
 {"_id": "French", "count": 3200 },
 {"_id": "Spanish", "count": 2500 },
 {"_id": "Hindi", "count": 1800 }]
```