EXERCISE 18

MONGODB

Name : Vidhyashree.J

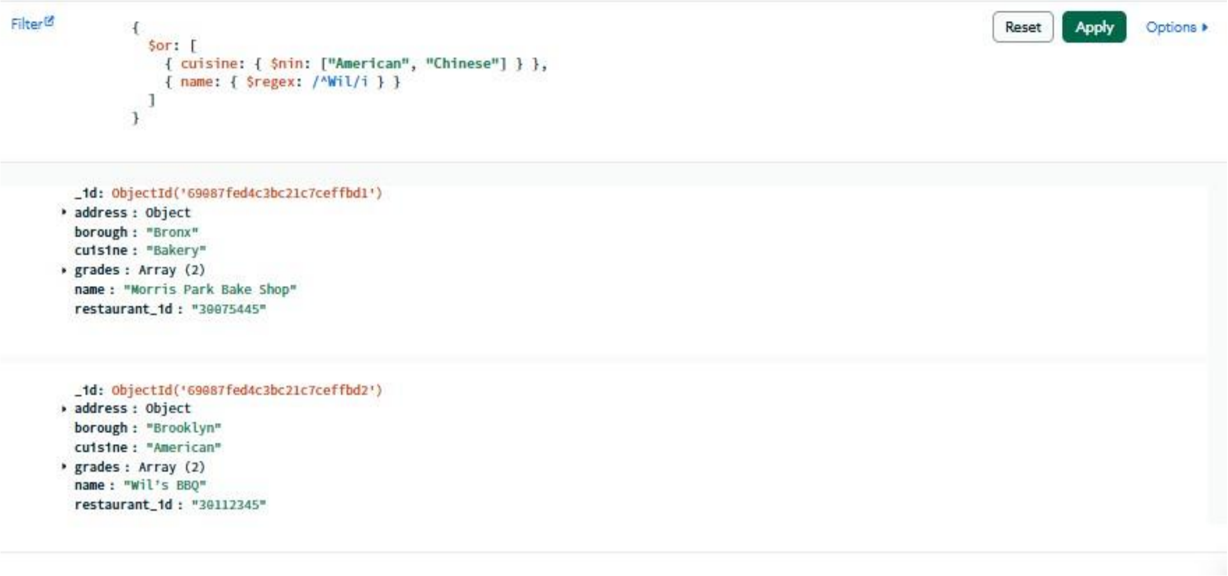
DEPARTMENT: AI&DS

ROLL NO: 241801310

Structure of 'restaurants' collection:

```
{
"address": {
"building": "1007",
"coord": [ -73.856077, 40.848447 ],
"street": "Morris Park Ave",
"zipcode": "10462"
},
"borough": "Bronx",
"cuisine": "Bakery",
"grades": [
{ "date": { "$date": 1393804800000 }, "grade": "A", "score": 2 },
{ "date": { "$date": 1378857600000 }, "grade": "A", "score": 6 },
{ "date": { "$date": 1358985600000 }, "grade": "A", "score": 10 },
{ "date": { "$date": 1322006400000 }, "grade": "A", "score": 9 },
{ "date": { "$date": 1299715200000 }, "grade": "B", "score": 14 }
],
"name": "Morris Park Bake Shop",
"restaurant_id": "30075445"
}
```

1. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'.



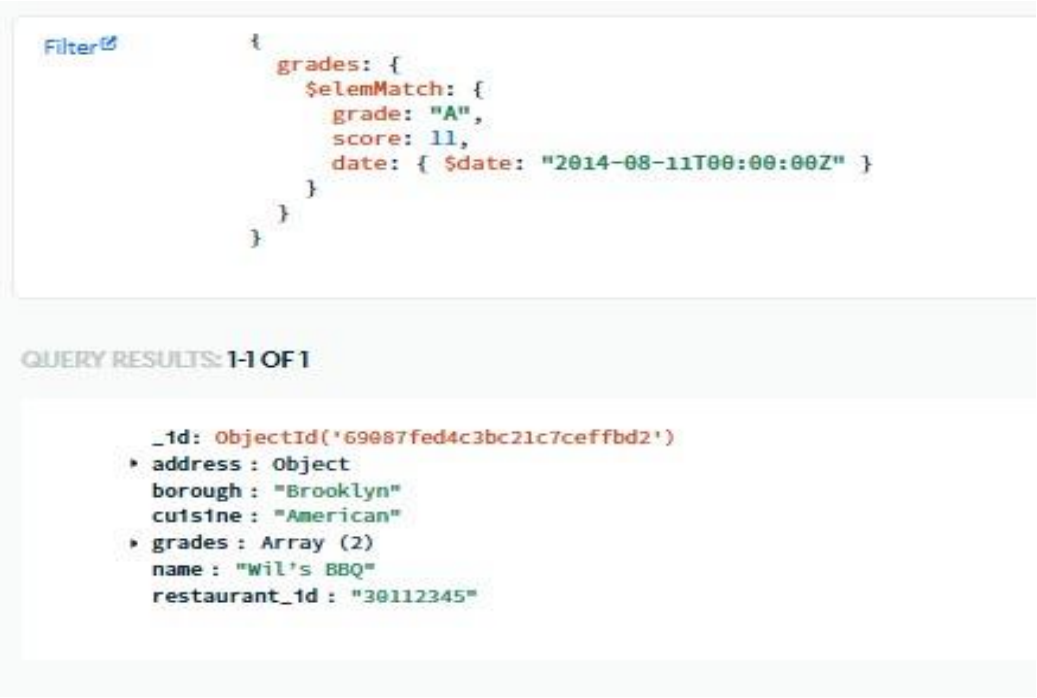
```
{
  $or: [
    { cuisine: { $nin: ["American", "Chinese"] } },
    { name: { $regex: /^Wil/i } }
  ]
}
```

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"
```

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd2')
▸ address : Object
  borough : "Brooklyn"
  cuisine : "American"
▸ grades : Array (2)
  name : "Wil's BBQ"
  restaurant_id : "30112345"
```

2. Write a MongoDB query to find the restaurant Id, name, and grades for those restaurants which achieved a grade of "A" and scored 11 on an ISODate "2014-08-

11T00:00:00Z" among many of survey dates..

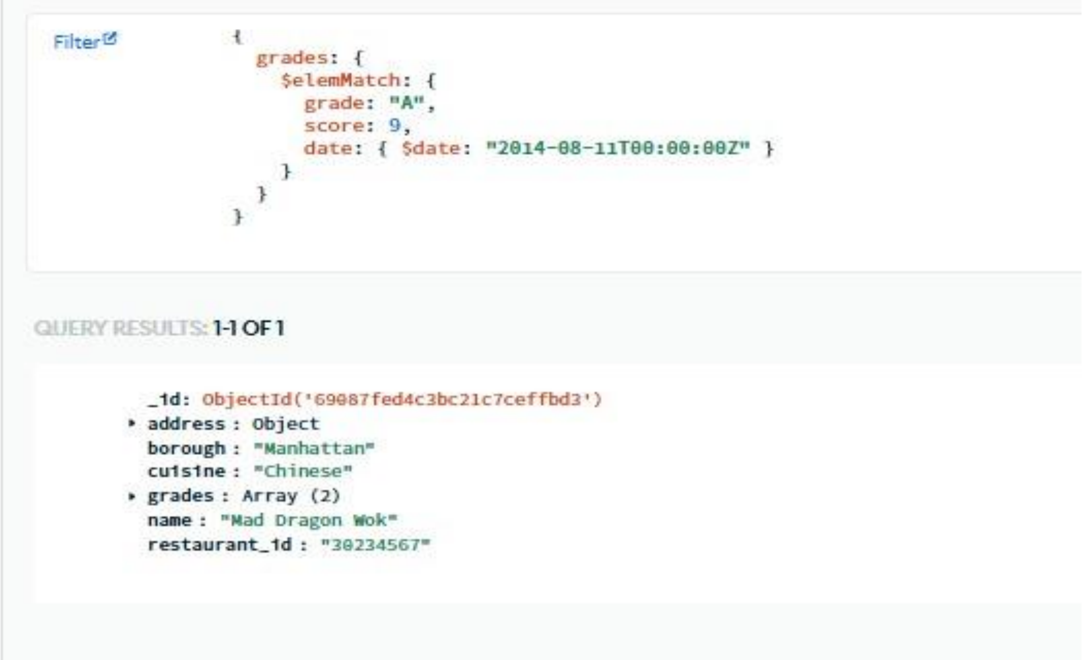


```
{
  grades: {
    $elemMatch: {
      grade: "A",
      score: 11,
      date: { $date: "2014-08-11T00:00:00Z" }
    }
  }
}
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd2')
▸ address : Object
  borough : "Brooklyn"
  cuisine : "American"
▸ grades : Array (2)
  name : "Wil's BBQ"
  restaurant_id : "30112345"
```

3. Write a MongoDB query to find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISODate "2014-08-11T00:00:00Z".

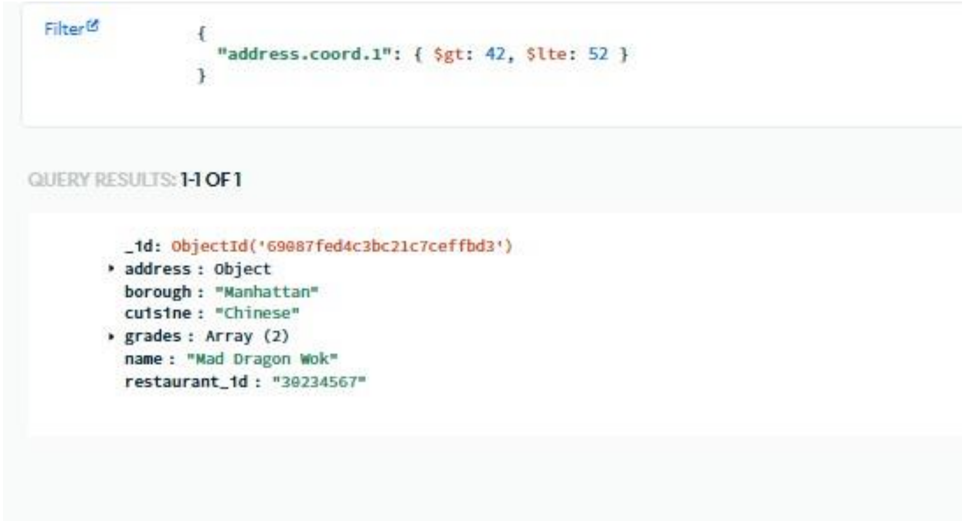


```
{
  grades: {
    $elemMatch: {
      grade: "A",
      score: 9,
      date: { $date: "2014-08-11T00:00:00Z" }
    }
  }
}
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd3')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Chinese"
▸ grades : Array (2)
  name : "Mad Dragon Wok"
  restaurant_id : "30234567"
```

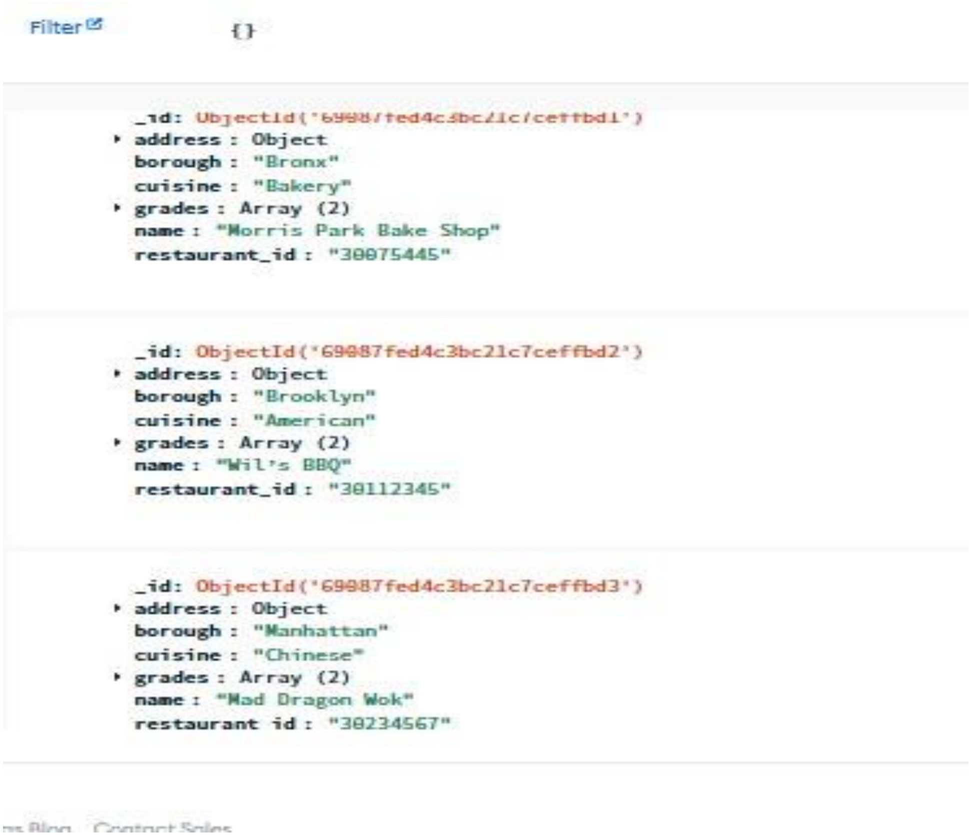
4. Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value which is more than 42 and upto 52..



```
{
  "address.coord.1": { $gt: 42, $lte: 52 }
}
```

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd3')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Chinese"
▸ grades : Array (2)
  name : "Mad Dragon Wok"
  restaurant_id : "30234567"
```

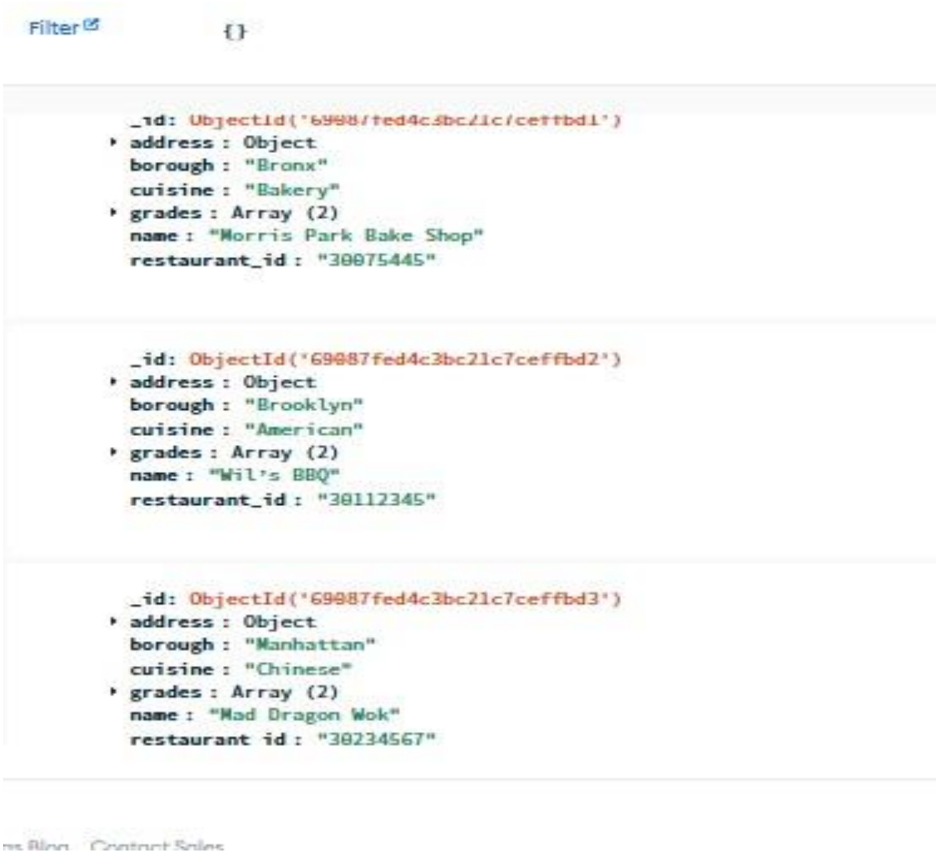
5. Write a MongoDB query to arrange the name of the restaurants in ascending order along with all the columns.



```
{}
```

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"

  _id: ObjectId('69087fed4c3bc21c7ceffbd2')
▸ address : Object
  borough : "Brooklyn"
  cuisine : "American"
▸ grades : Array (2)
  name : "Wil's BBQ"
  restaurant_id : "30112345"

  _id: ObjectId('69087fed4c3bc21c7ceffbd3')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Chinese"
▸ grades : Array (2)
  name : "Mad Dragon Wok"
  restaurant_id : "30234567"
```

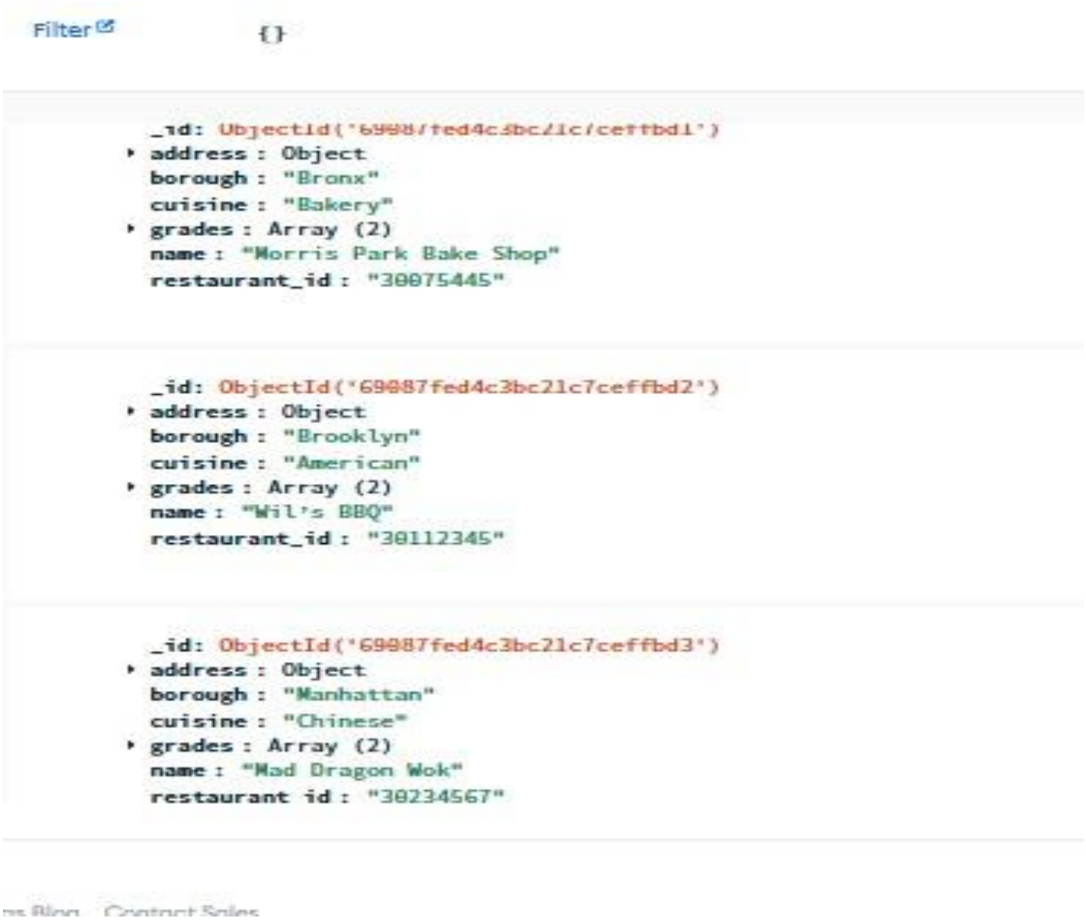
6. Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.



```
{}
```

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"

  _id: ObjectId('69087fed4c3bc21c7ceffbd2')
▸ address : Object
  borough : "Brooklyn"
  cuisine : "American"
▸ grades : Array (2)
  name : "Wil's BBQ"
  restaurant_id : "30112345"

  _id: ObjectId('69087fed4c3bc21c7ceffbd3')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Chinese"
▸ grades : Array (2)
  name : "Mad Dragon Wok"
  restaurant id : "30234567"
```

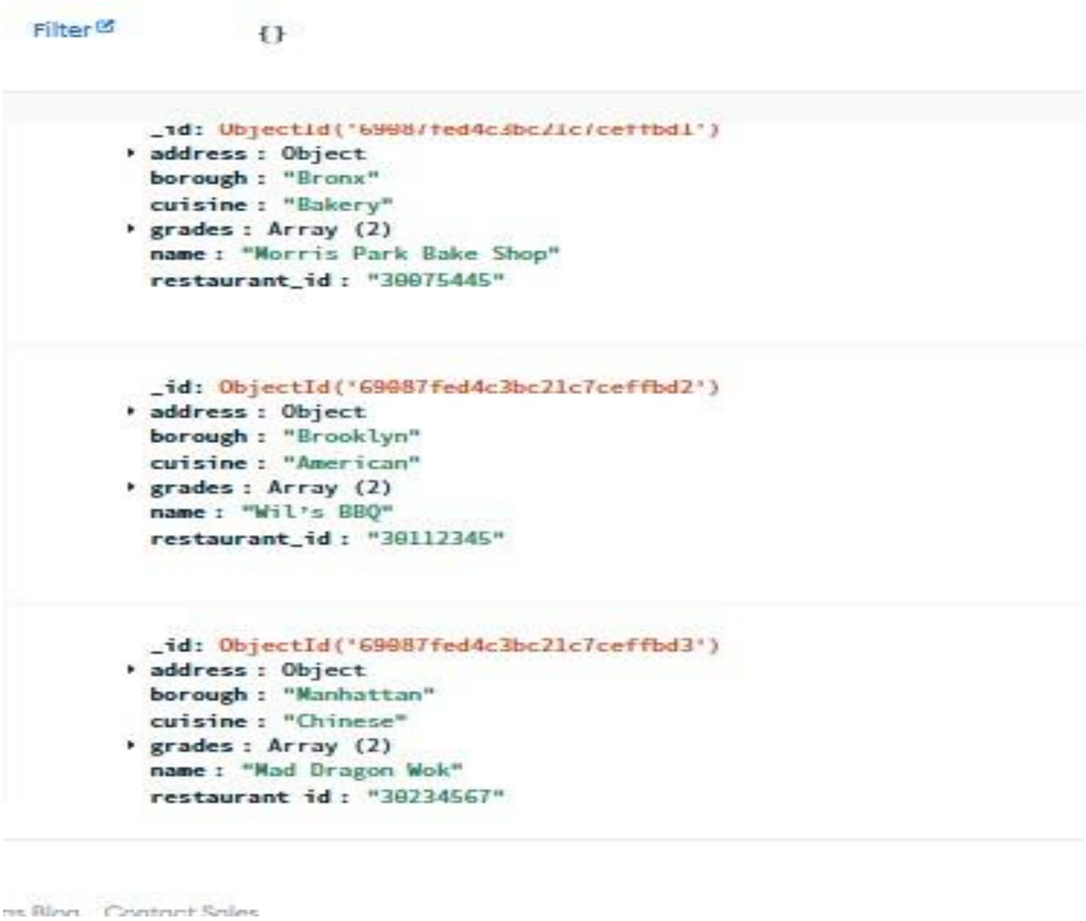
7. Write a MongoDB query to arranged the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.



```
{}
```

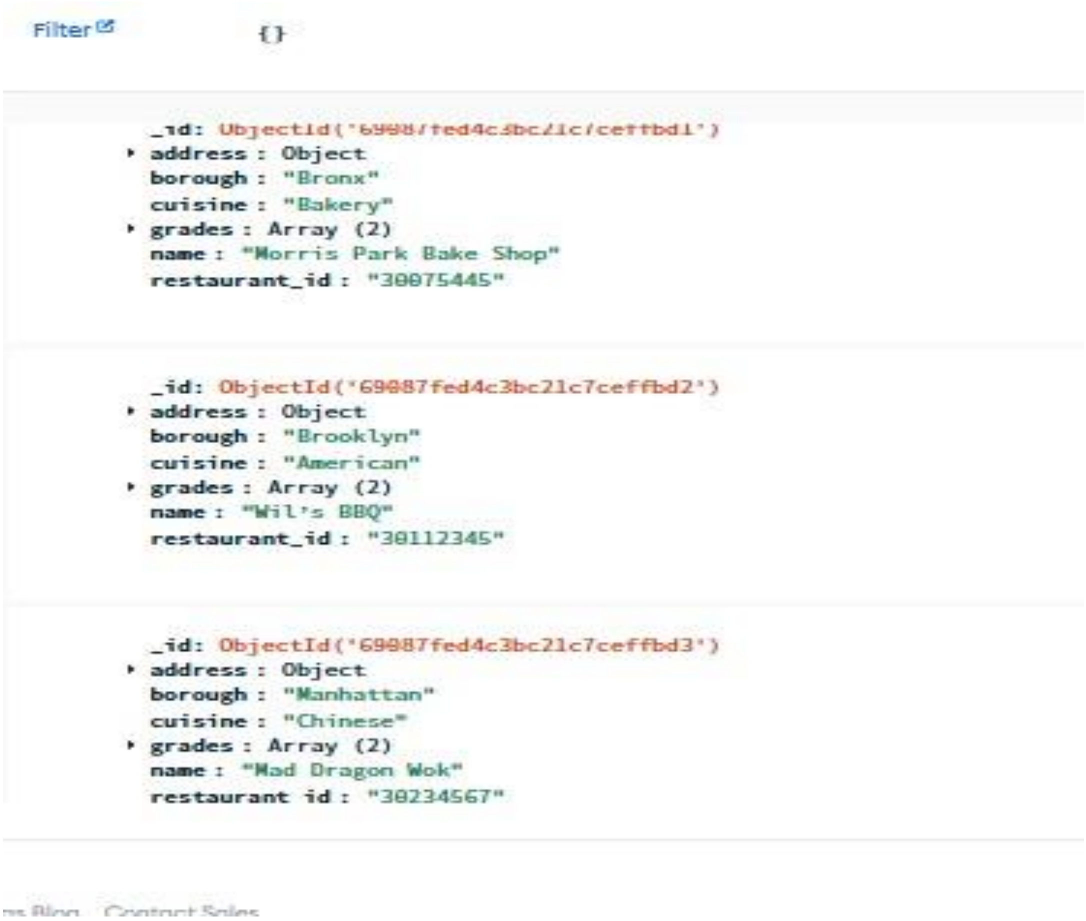
```
  _id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"

  _id: ObjectId('69087fed4c3bc21c7ceffbd2')
▸ address : Object
  borough : "Brooklyn"
  cuisine : "American"
▸ grades : Array (2)
  name : "Wil's BBQ"
  restaurant_id : "30112345"

  _id: ObjectId('69087fed4c3bc21c7ceffbd3')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Chinese"
▸ grades : Array (2)
  name : "Mad Dragon Wok"
  restaurant id : "30234567"
```

8. Write a MongoDB query to know whether all the addresses contains the street or not.



```
{}

    _id: ObjectId('69087fed4c3bc21c7ceffbd1')
  ▸ address : Object
    borough : "Bronx"
    cuisine : "Bakery"
  ▸ grades : Array (2)
    name : "Morris Park Bake Shop"
    restaurant_id : "30075445"

    _id: ObjectId('69087fed4c3bc21c7ceffbd2')
  ▸ address : Object
    borough : "Brooklyn"
    cuisine : "American"
  ▸ grades : Array (2)
    name : "Wil's BBQ"
    restaurant_id : "30112345"

    _id: ObjectId('69087fed4c3bc21c7ceffbd3')
  ▸ address : Object
    borough : "Manhattan"
    cuisine : "Chinese"
  ▸ grades : Array (2)
    name : "Mad Dragon Wok"
    restaurant id : "30234567"
```

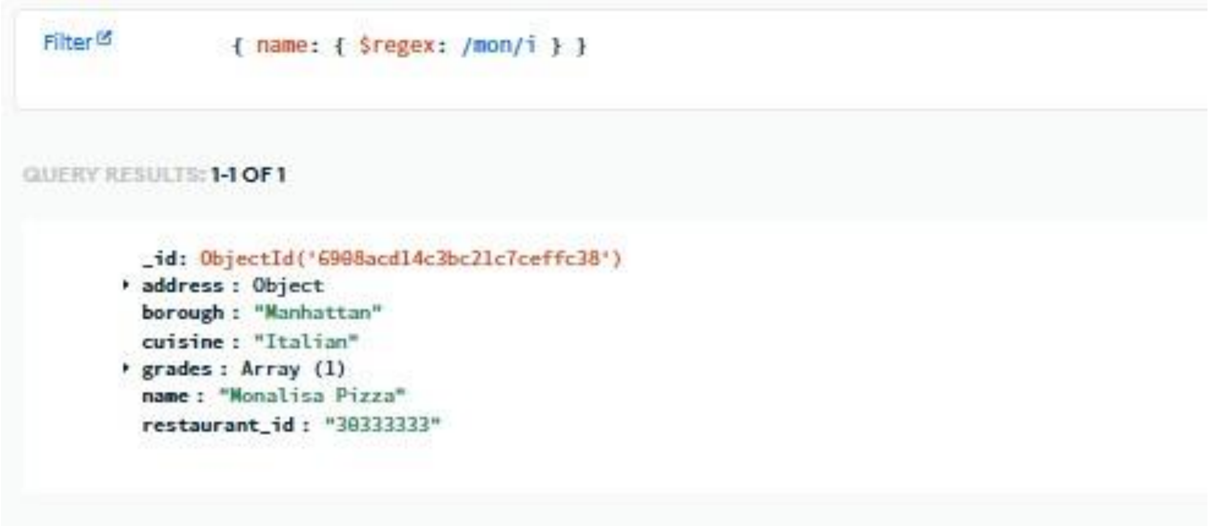
9. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.



```
{}
```

```
_id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"

_id: ObjectId('69087fed4c3bc21c7ceffbd2')
▸ address : Object
  borough : "Brooklyn"
  cuisine : "American"
▸ grades : Array (2)
  name : "Wil's BBQ"
  restaurant_id : "30112345"

_id: ObjectId('69087fed4c3bc21c7ceffbd3')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Chinese"
▸ grades : Array (2)
  name : "Mad Dragon Wok"
  restaurant id : "30234567"
```

10. Write a MongoDB query which will select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.

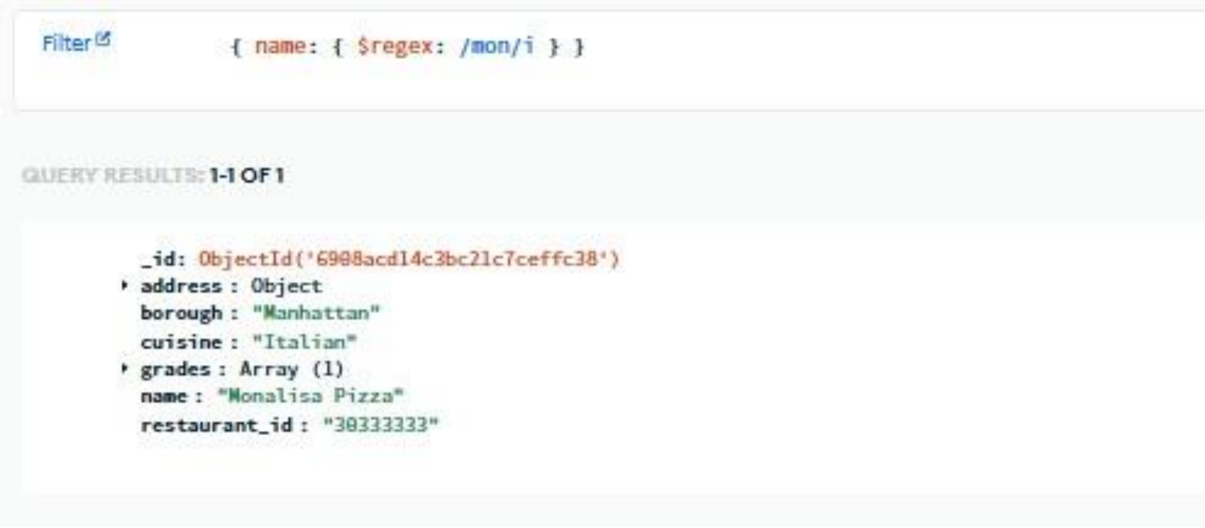


```
{ name: { $regex: /mon/i } }
```

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('6908acd14c3bc21c7ceffc38')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Italian"
▸ grades : Array (1)
  name : "Monalisa Pizza"
  restaurant_id : "30333333"
```

11. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name.

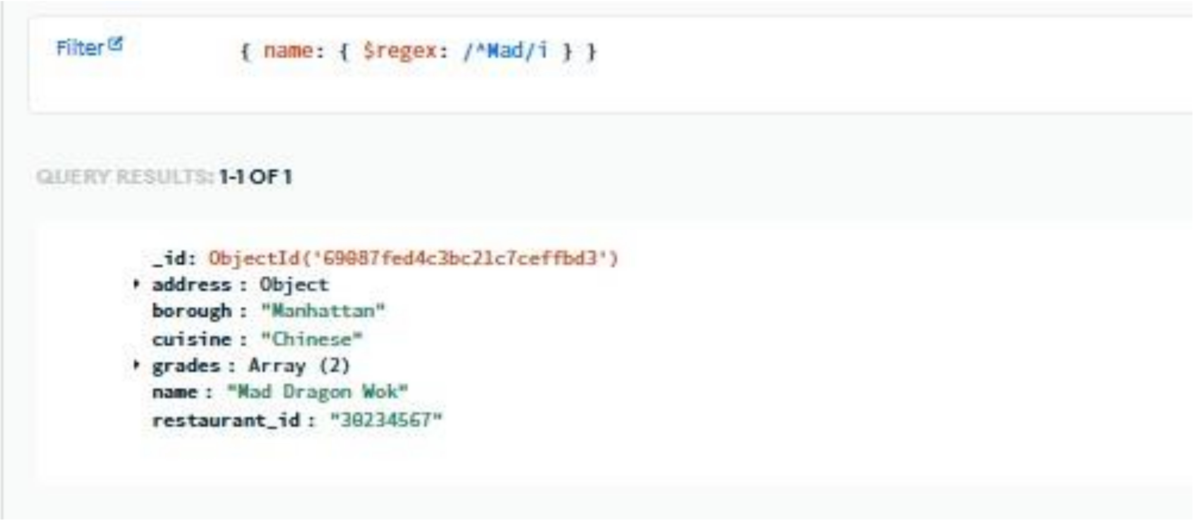


```
{ name: { $regex: /mon/i } }
```

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('6908acd14c3bc21c7ceffc38')
address : Object
borough : "Manhattan"
cuisine : "Italian"
grades : Array (1)
name : "Monalisa Pizza"
restaurant_id : "30333333"
```

12. Write a MongoDB query to find the restaurant name, borough, longitude and latitude and cuisine for those restaurants which contain 'Mad' as first three letters of its name.

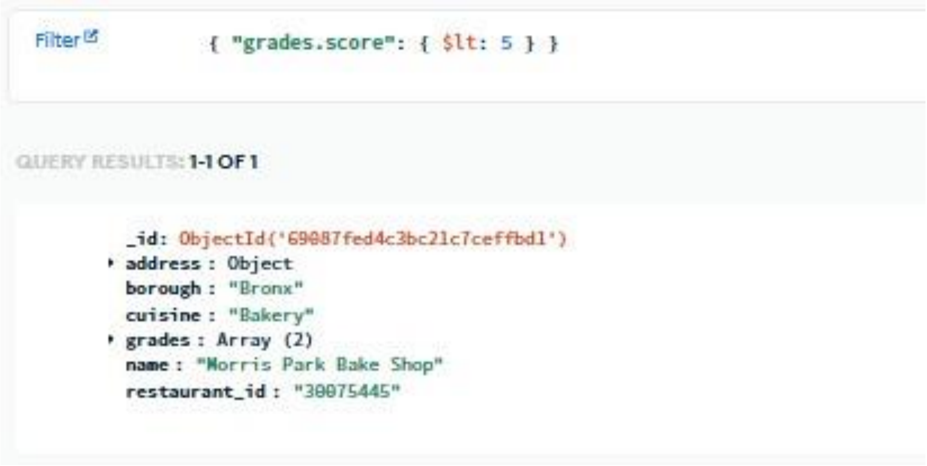


```
{ name: { $regex: /^Mad/i } }
```

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('69087fed4c3bc21c7ceffbd3')
address : Object
borough : "Manhattan"
cuisine : "Chinese"
grades : Array (2)
name : "Mad Dragon Wok"
restaurant_id : "30234567"
```

13. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5.

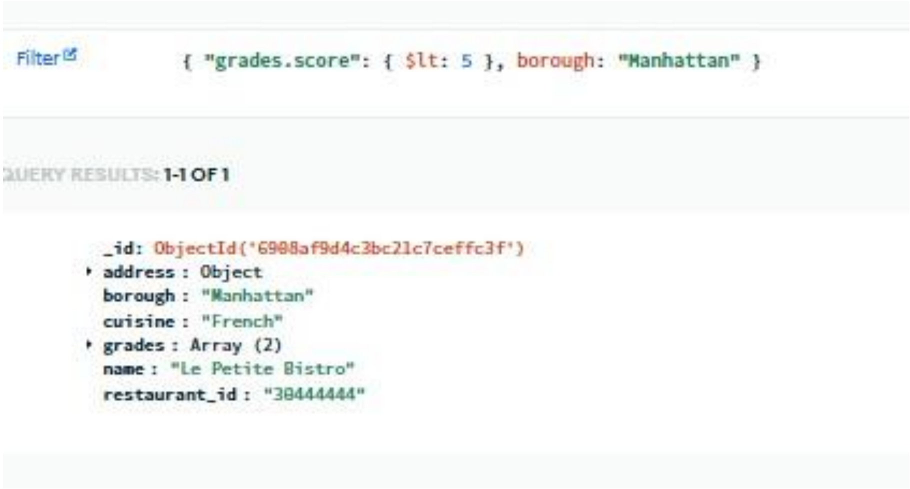


```
{ "grades.score": { $lt: 5 } }
```

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('69087fed4c3bc21c7ceffbd1')
address : Object
borough : "Bronx"
cuisine : "Bakery"
grades : Array (2)
name : "Morris Park Bake Shop"
restaurant_id : "30075445"
```

14. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan.

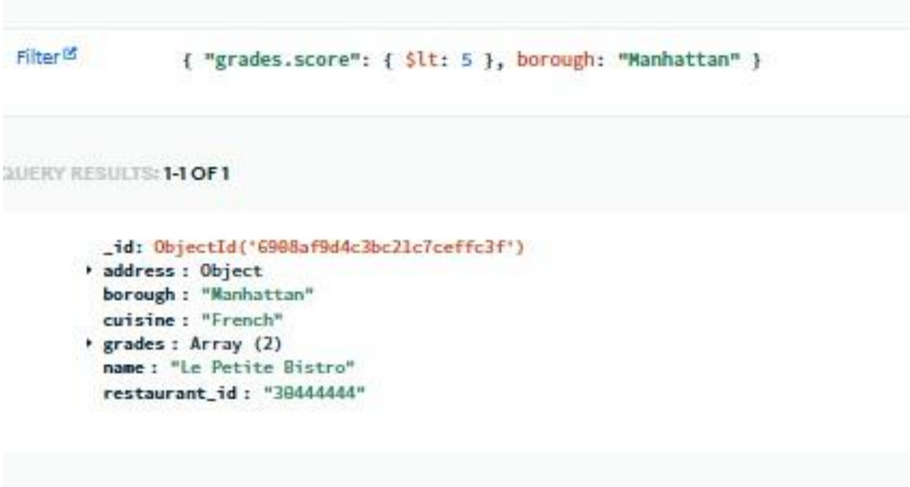


```
{ "grades.score": { $lt: 5 }, borough: "Manhattan" }
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('6908af9d4c3bc21c7ceffc3f')
▸ address : Object
  borough : "Manhattan"
  cuisine : "French"
▸ grades : Array (2)
  name : "Le Petite Bistro"
  restaurant_id : "30444444"
```

15. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn.

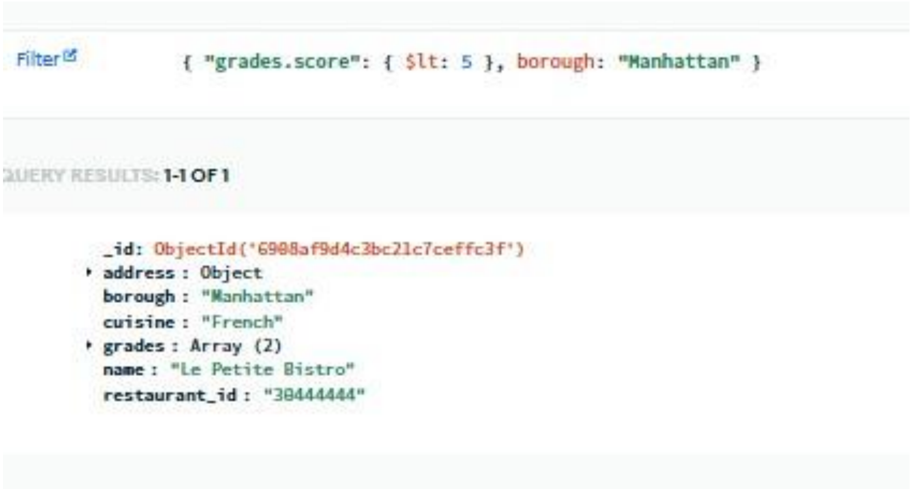


```
{ "grades.score": { $lt: 5 }, borough: "Manhattan" }
```

QUERY RESULTS: 1-1 OF 1

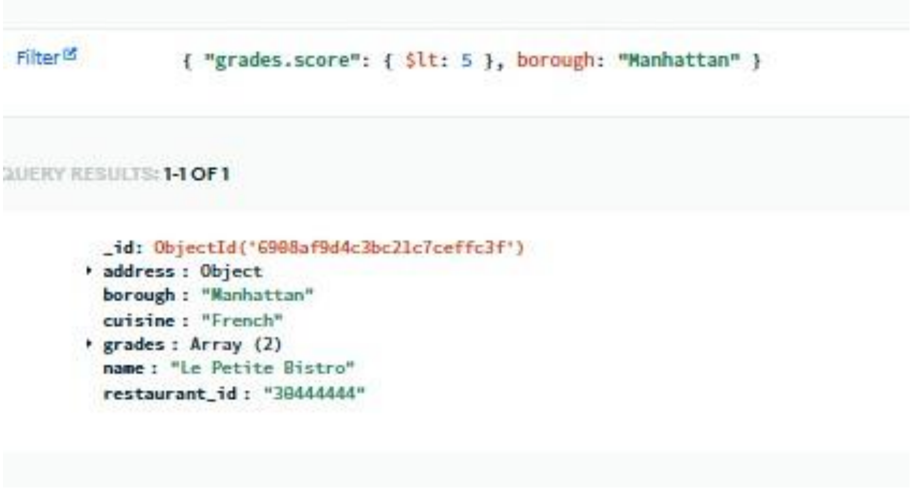
```
  _id: ObjectId('6908af9d4c3bc21c7ceffc3f')
▸ address : Object
  borough : "Manhattan"
  cuisine : "French"
▸ grades : Array (2)
  name : "Le Petite Bistro"
  restaurant_id : "30444444"
```

16. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.



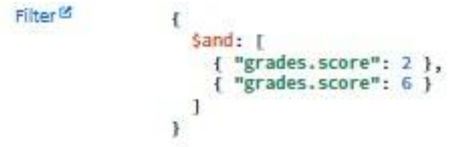
```
Filter    { "grades.score": { $lt: 5 }, borough: "Manhattan" }

QUERY RESULTS: 1-1 OF 1

  _id: ObjectId('6908af9d4c3bc21c7ceffc3f')
▸ address : Object
  borough : "Manhattan"
  cuisine : "French"
▸ grades : Array (2)
  name : "Le Petite Bistro"
  restaurant_id : "30444444"
```

17. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.

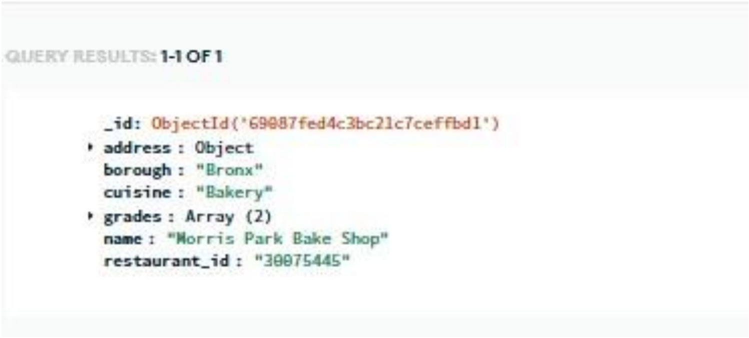


```
Filter    { "grades.score": { $lt: 5 }, borough: "Manhattan" }

QUERY RESULTS: 1-1 OF 1

  _id: ObjectId('6908af9d4c3bc21c7ceffc3f')
▸ address : Object
  borough : "Manhattan"
  cuisine : "French"
▸ grades : Array (2)
  name : "Le Petite Bistro"
  restaurant_id : "30444444"
```

18. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6.

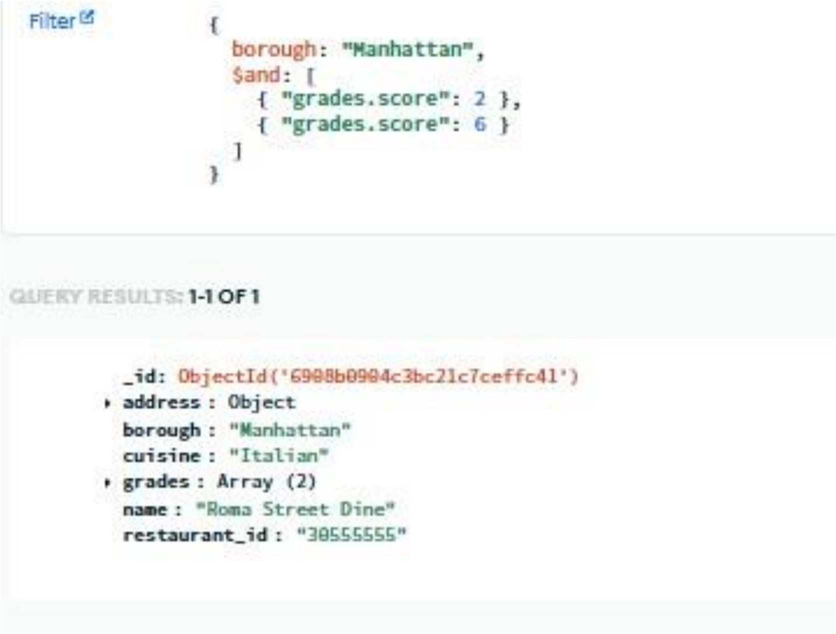


```
Filter    {
            $and: [
              { "grades.score": 2 },
              { "grades.score": 6 }
            ]
          }
```



```
QUERY RESULTS: 1-1 OF 1

  _id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"
```

19. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan.

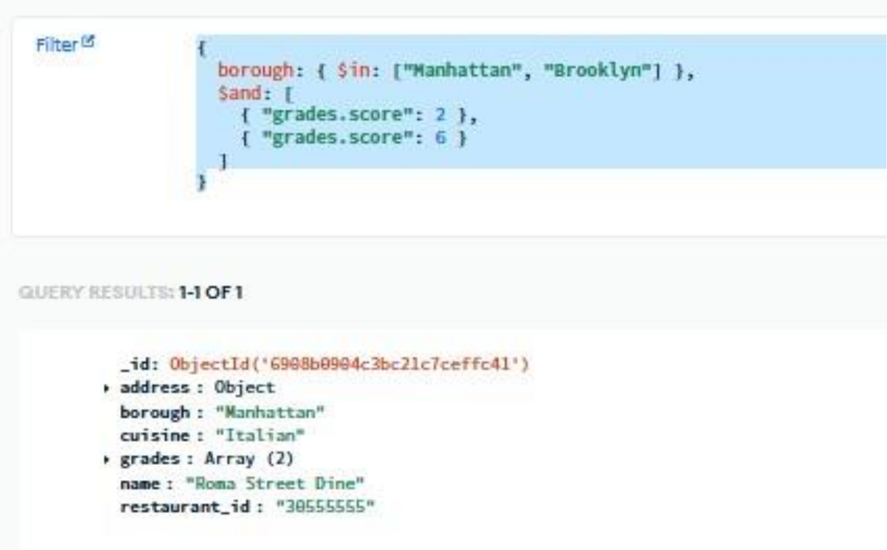


```
{
  borough: "Manhattan",
  $and: [
    { "grades.score": 2 },
    { "grades.score": 6 }
  ]
}
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('6908b0904c3bc21c7ceffc41')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Italian"
▸ grades : Array (2)
  name : "Roma Street Dine"
  restaurant_id : "30555555"
```

20. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn.

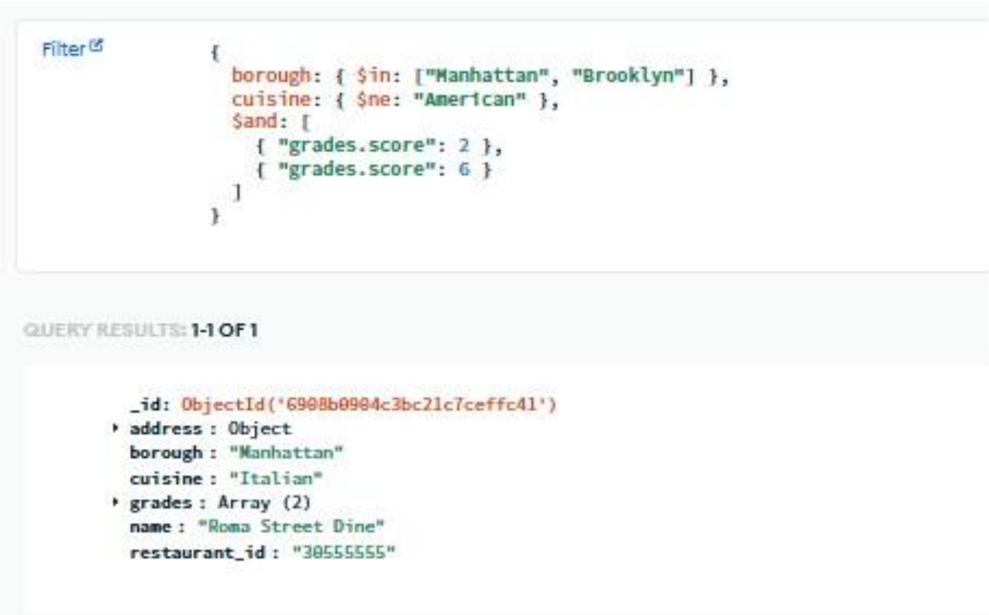


```
{
  borough: { $in: ["Manhattan", "Brooklyn"] },
  $and: [
    { "grades.score": 2 },
    { "grades.score": 6 }
  ]
}
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('6908b0904c3bc21c7ceffc41')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Italian"
▸ grades : Array (2)
  name : "Roma Street Dine"
  restaurant_id : "30555555"
```

21. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.

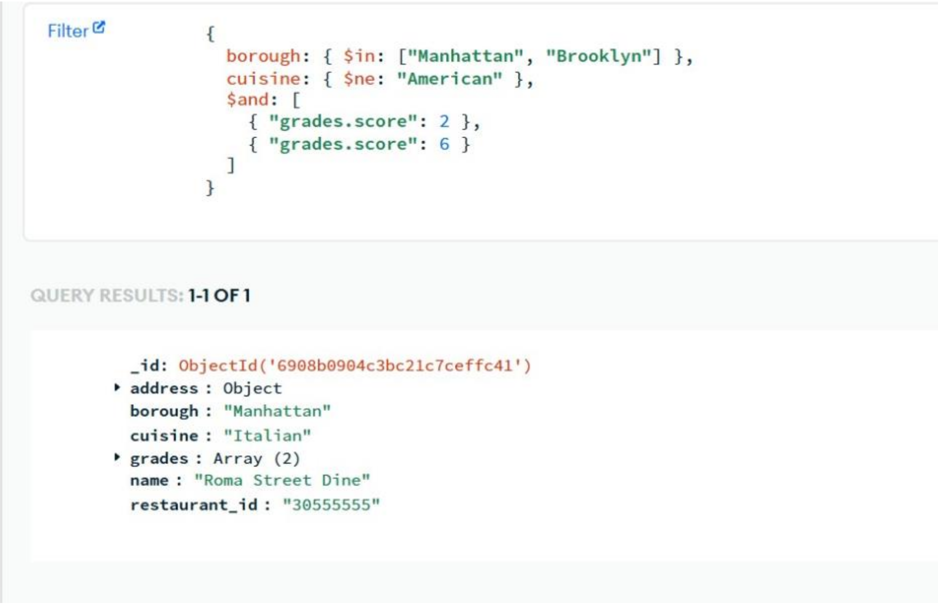


```
{
  borough: { $in: ["Manhattan", "Brooklyn"] },
  cuisine: { $ne: "American" },
  $and: [
    { "grades.score": 2 },
    { "grades.score": 6 }
  ]
}
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('6908b0904c3bc21c7ceffc41')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Italian"
▸ grades : Array (2)
  name : "Roma Street Dine"
  restaurant_id : "30555555"
```

22. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.

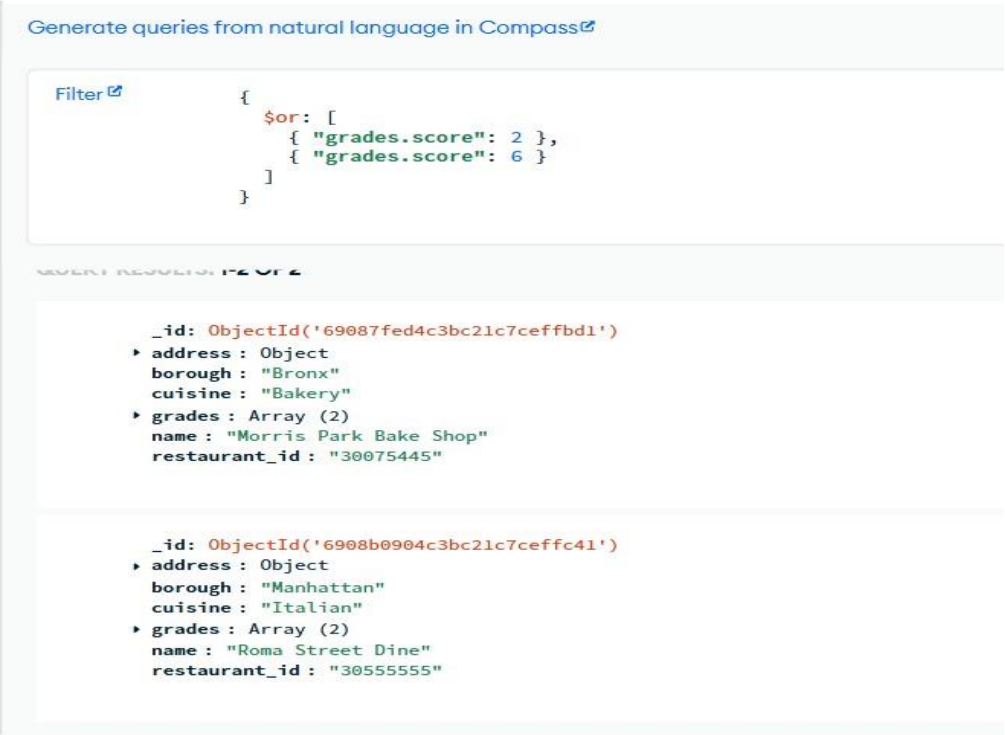


```
{
  borough: { $in: ["Manhattan", "Brooklyn"] },
  cuisine: { $ne: "American" },
  $and: [
    { "grades.score": 2 },
    { "grades.score": 6 }
  ]
}
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('6908b0904c3bc21c7ceffc41')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Italian"
▸ grades : Array (2)
  name : "Roma Street Dine"
  restaurant_id : "30555555"
```

23. Write a MongoDB query to find the restaurants that have a grade with a score of 2 or a grade with a score of 6.



```
{
  $or: [
    { "grades.score": 2 },
    { "grades.score": 6 }
  ]
}
```

QUERY RESULTS: 1-2 OF 2

```
  _id: ObjectId('69087fed4c3bc21c7ceffbd1')
▸ address : Object
  borough : "Bronx"
  cuisine : "Bakery"
▸ grades : Array (2)
  name : "Morris Park Bake Shop"
  restaurant_id : "30075445"
```

```
  _id: ObjectId('6908b0904c3bc21c7ceffc41')
▸ address : Object
  borough : "Manhattan"
  cuisine : "Italian"
▸ grades : Array (2)
  name : "Roma Street Dine"
  restaurant_id : "30555555"
```

Sample document of 'movies' collection

{

_id: ObjectId("573a1390f29313caabcd42e8"),

plot: 'A group of bandits stage a brazen train hold-up, only to find a determined posse hot on their heels.', genres: ['Short', 'Western'], runtime: 11,

cast: [

'A.C. Abadie',

"Gilbert M. 'Broncho Billy' Anderson",

'George Barnes',

'Justus D. Barnes'

],

poster: 'https://m.media-

amazon.com/images/M/MV5BMTU3NjE5NzYtYTYyNS00MDVmLWIwYjgtMmYwYWIxZDYyNzU2XkEyXkFqcG

deQXVyNzQzNzQxNzI@._V1_SY1000_SX677_AL_.jpg',

title: 'The Great Train Robbery',

fullplot: "Among the earliest existing films in American cinema - notable as the first film that presented a

narrative story to tell - it depicts a group of cowboy outlaws who hold up a train and rob the passengers.

They are then pursued by a Sheriff's posse. Several scenes have color included - all hand tinted.",

languages: ['English'], released: ISODate("1903-12-

01T00:00:00.000Z"), directors: ['Edwin S. Porter'],

rated: 'TV-G', awards: { wins: 1, nominations: 0, text:

'1 win.' }, lastupdated: '2015-08-13

00:27:59.177000000', year: 1903, imdb: { rating:

7.4, votes: 9847, id: 439 }, countries: ['USA'], type:

'movie', tomatoes: { viewer: { rating: 3.7,

numReviews: 2559, meter: 75 }, fresh: 6, critic: {

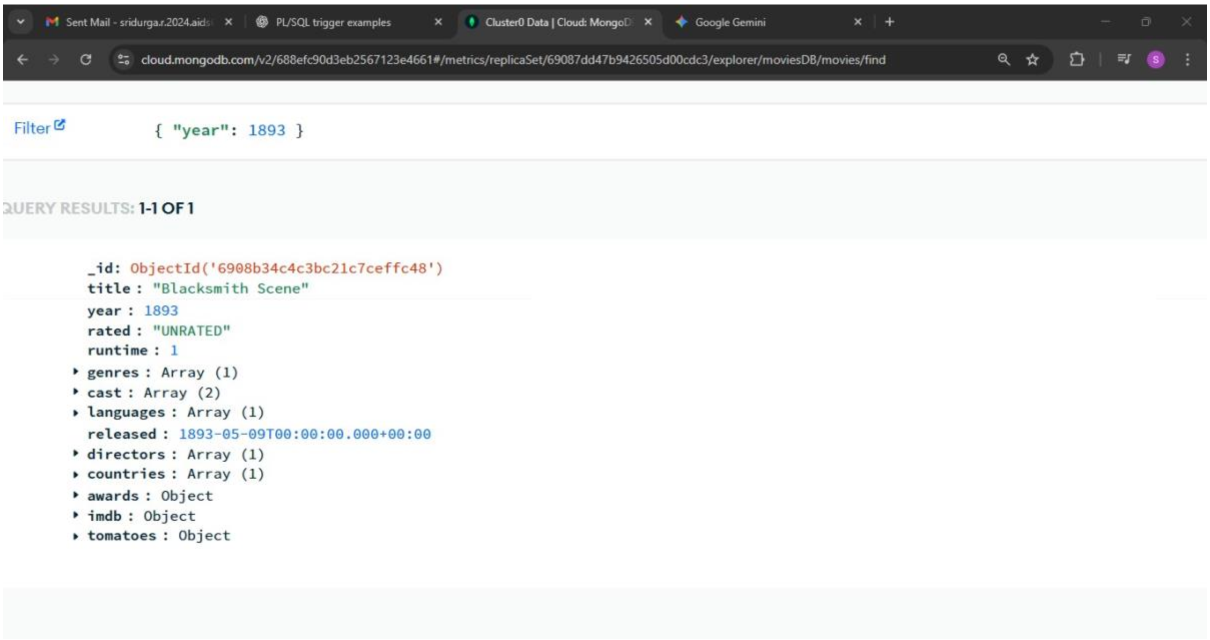
rating: 7.6, numReviews: 6, meter: 100 }, rotten: 0,

lastUpdated: ISODate("2015-08-08T19:16:10.000Z")

}

1.Find all movies with full information from the 'movies' collection that released in

the year 1893.

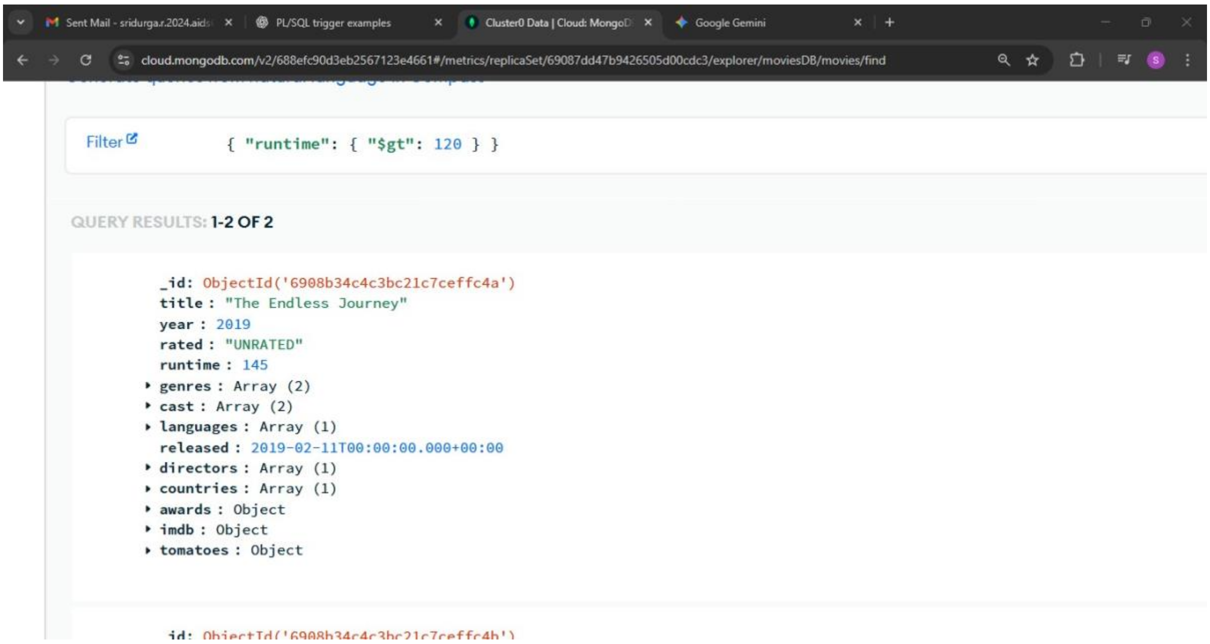

Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Filter

```
{ "year": 1893 }
```

QUERY RESULTS: 1-1 OF 1

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc48')
  title : "Blacksmith Scene"
  year : 1893
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (1)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

2. Find all movies with full information from the 'movies' collection that have a runtime greater than 120 minutes. x

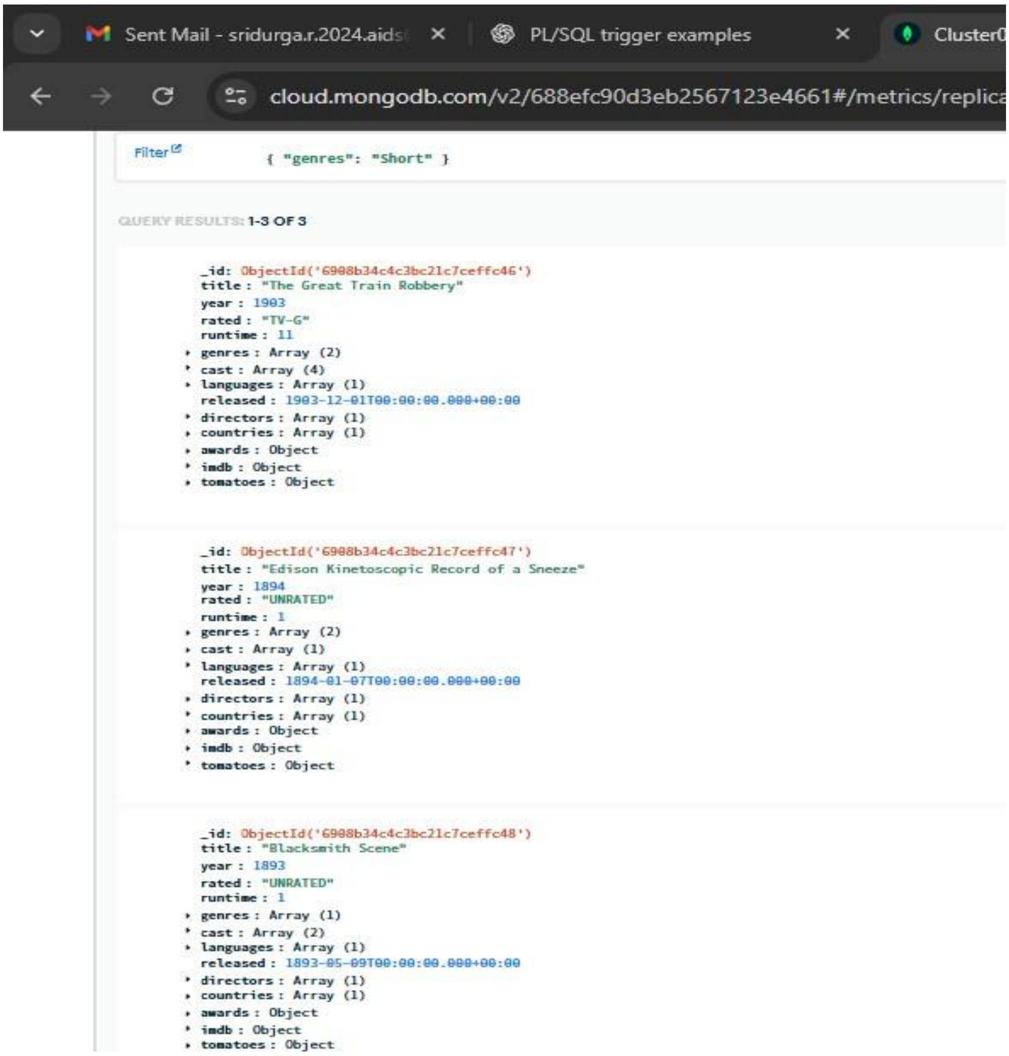

Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Filter

```
{ "runtime": { "$gt": 120 } }
```

QUERY RESULTS: 1-2 OF 2

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4b')
```

3. Find all movies with full information from the 'movies' collection that have "Short" genre.

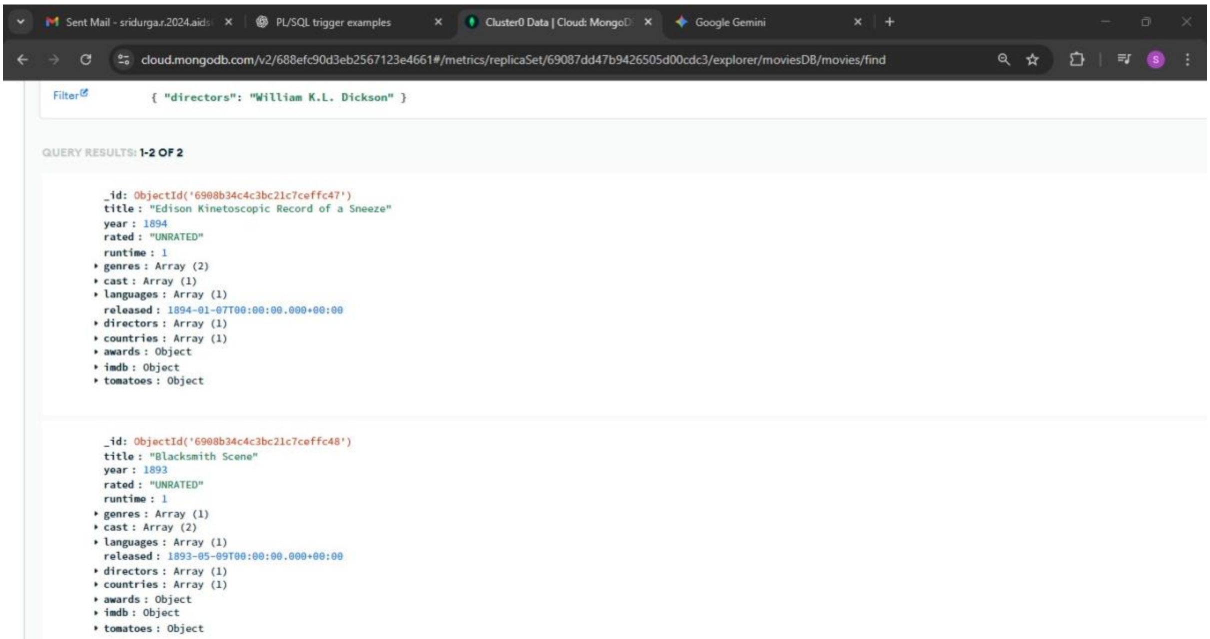

Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replica
Filter

```
{ "genres": "Short" }
```

QUERY RESULTS: 1-3 OF 3

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc46')
  title : "The Great Train Robbery"
  year : 1903
  rated : "TV-G"
  runtime : 11
▸ genres : Array (2)
▸ cast : Array (4)
▸ languages : Array (1)
  released : 1903-12-01T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc47')
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (2)
▸ cast : Array (1)
▸ languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc48')
  title : "Blacksmith Scene"
  year : 1893
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (1)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

4. Retrieve all movies from the 'movies' collection that were directed by "William K.L. Dickson" and include complete information for each movie.



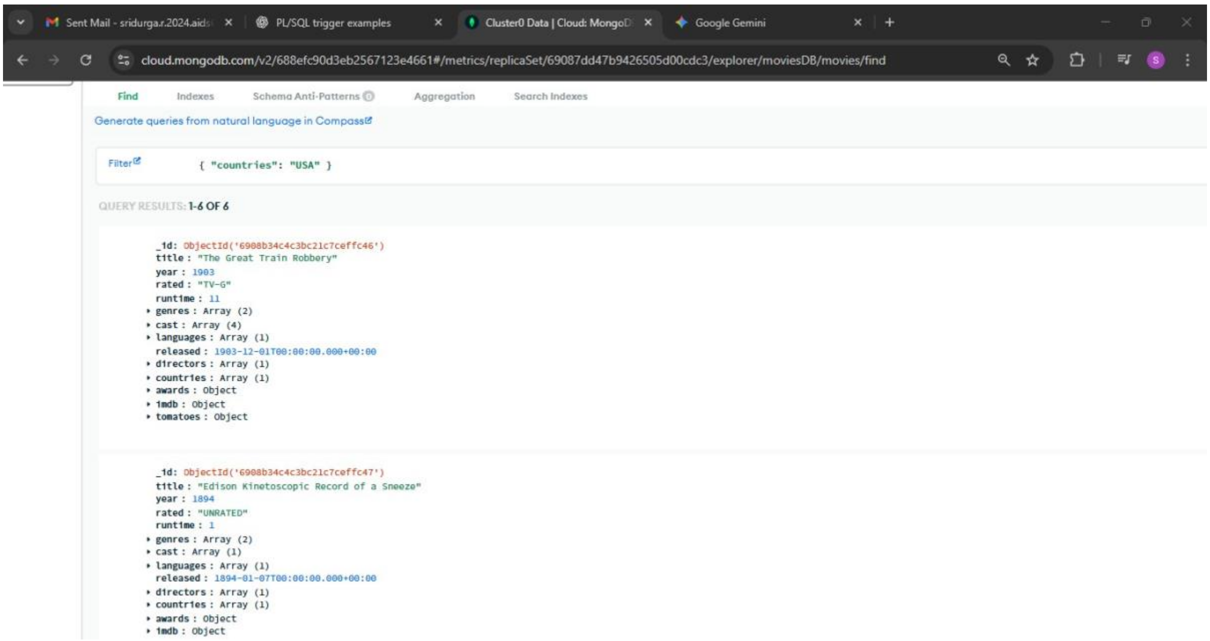
Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Filter

```
{ "directors": "William K.L. Dickson" }
```

QUERY RESULTS: 1-2 OF 2

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc47')
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (2)
▸ cast : Array (1)
▸ languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc48')
  title : "Blacksmith Scene"
  year : 1893
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (1)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

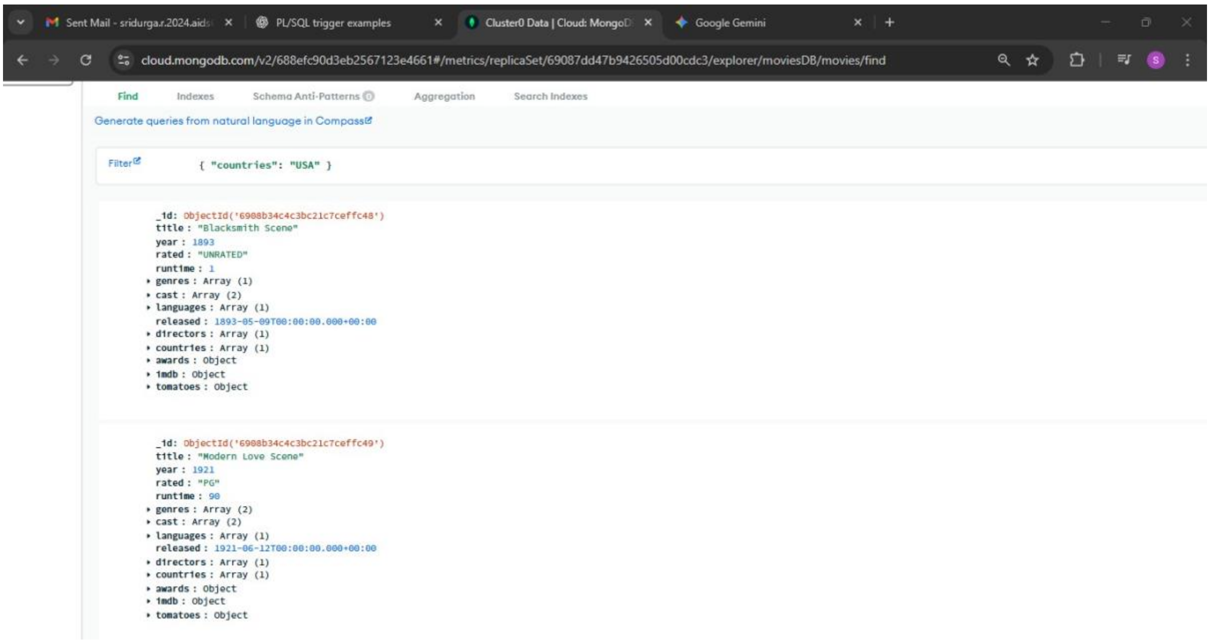
5. Retrieve all movies from the 'movies' collection that were released in the USA and include complete information for each movie.


Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass
Filter

```
{ "countries": "USA" }
```

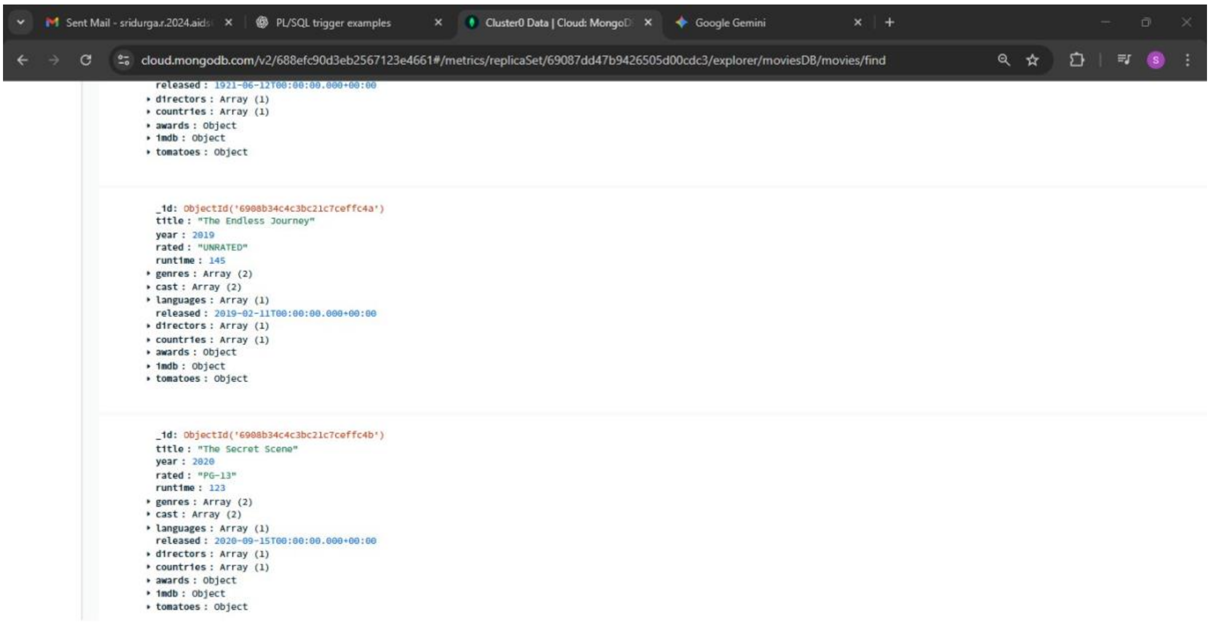
QUERY RESULTS: 1-6 OF 6

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc46')
  title : "The Great Train Robbery"
  year : 1903
  rated : "TV-G"
  runtime : 11
• genres : Array (2)
• cast : Array (4)
• languages : Array (1)
  released : 1903-12-01T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc47')
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  rated : "UNRATED"
  runtime : 1
• genres : Array (2)
• cast : Array (1)
• languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
```


Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass
Filter

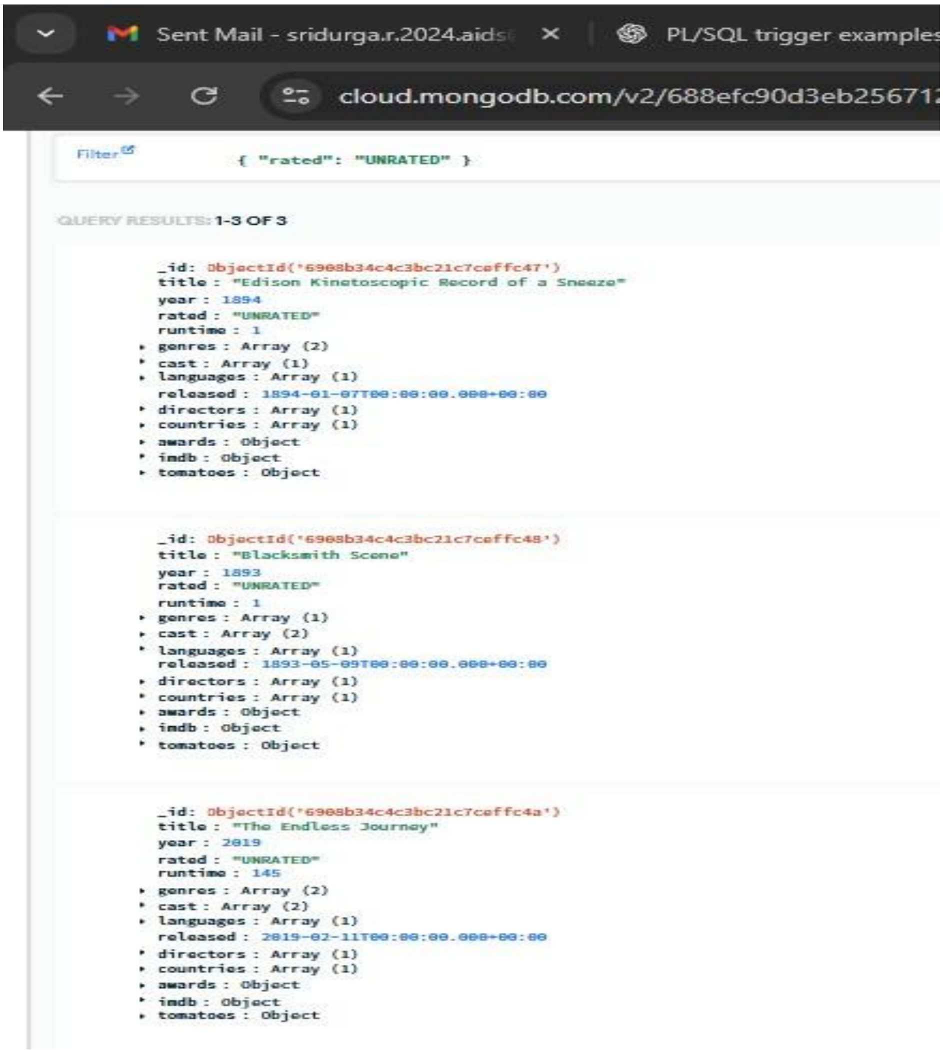
```
{ "countries": "USA" }
```

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc48')
  title : "Blacksmith Scene"
  year : 1893
  rated : "UNRATED"
  runtime : 1
• genres : Array (1)
• cast : Array (2)
• languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc49')
  title : "Modern Love Scene"
  year : 1921
  rated : "PG"
  runtime : 90
• genres : Array (2)
• cast : Array (2)
• languages : Array (1)
  released : 1921-06-12T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object
```


Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find

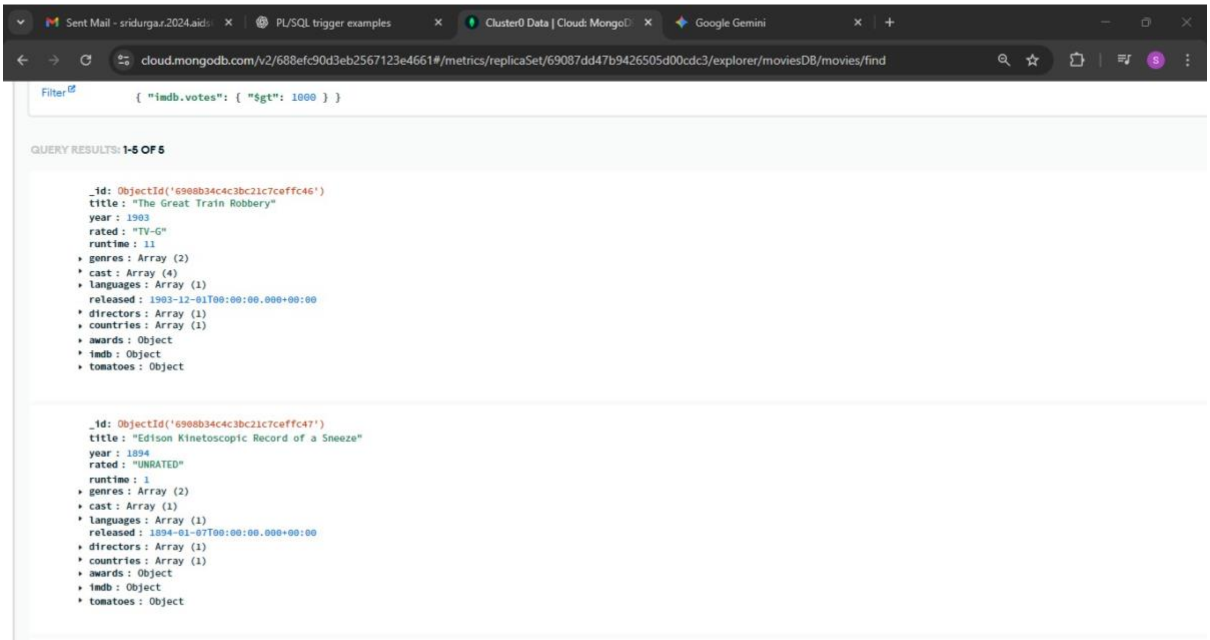
```
  released : 1921-06-12T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4b')
  title : "The Secret Scene"
  year : 2020
  rated : "PG-13"
  runtime : 123
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

6. Retrieve all movies from the 'movies' collection that have complete information and are rated as "UNRATED".


Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
cloud.mongodb.com/v2/688efc90d3eb256712

```
Filter      { "rated": "UNRATED" }

QUERY RESULTS: 1-3 OF 3

  _id: ObjectId('6908b34c4c3bc21c7ceffc47')
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (2)
▸ cast : Array (1)
▸ languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc48')
  title : "Blacksmith Scene"
  year : 1893
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (1)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

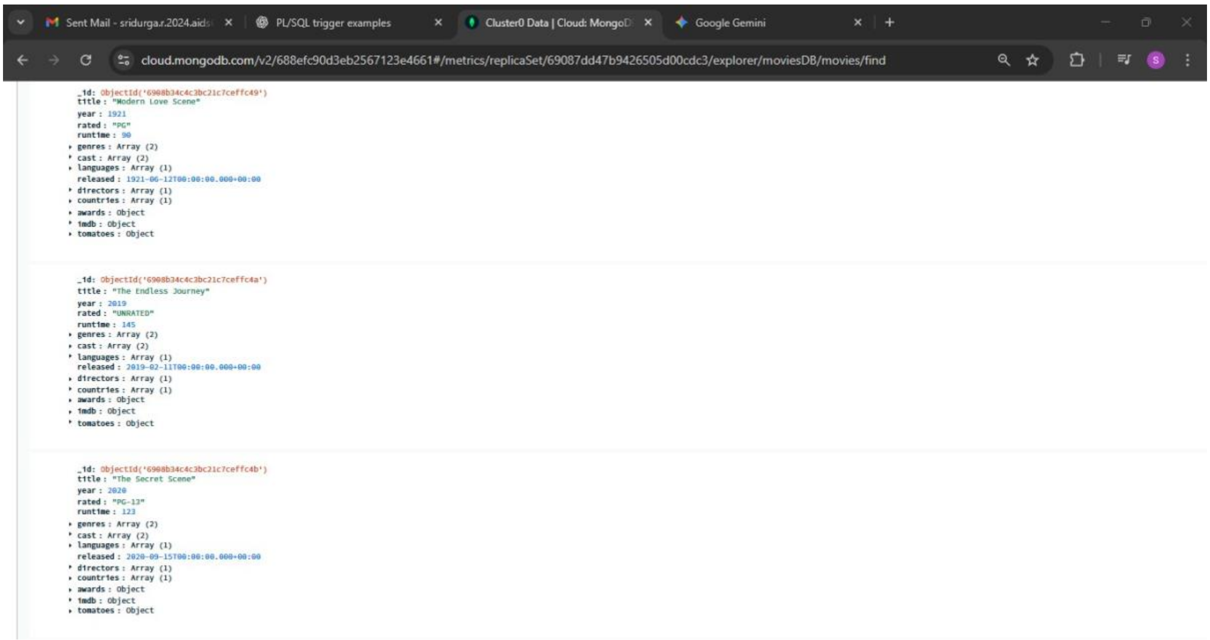
7. Retrieve all movies from the 'movies' collection that have complete information and have received more than 1000 votes on IMDb.


Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Filter

```
{ "imdb.votes": { "$gt": 1000 } }
```

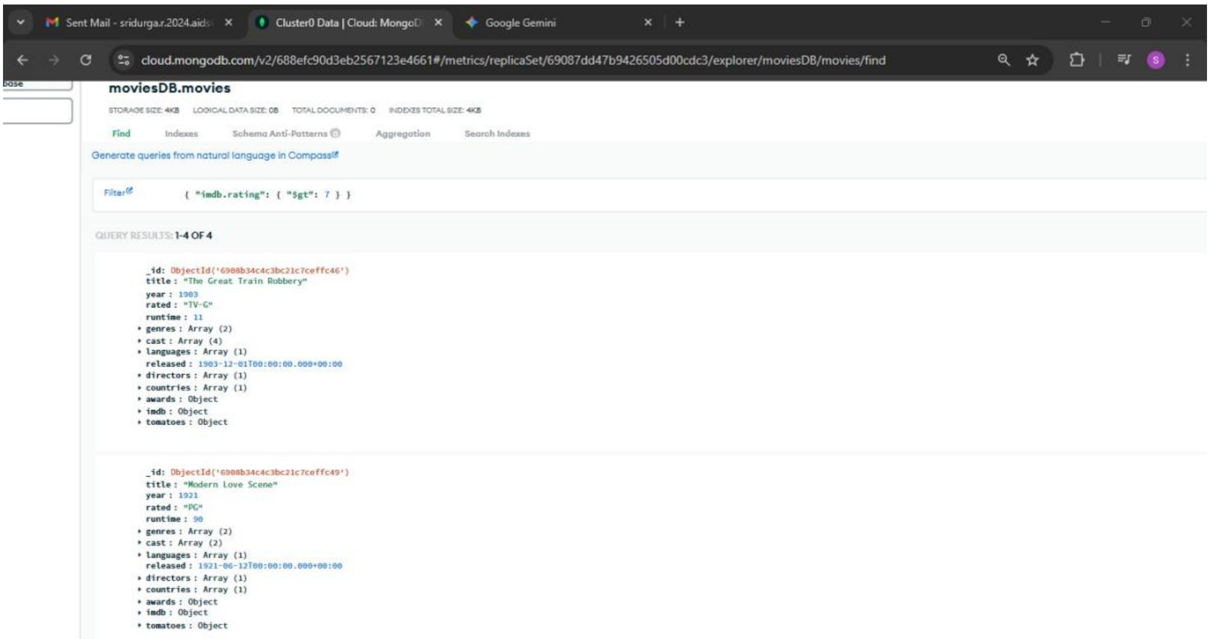
QUERY RESULTS: 1-5 OF 5

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc46')
  title : "The Great Train Robbery"
  year : 1903
  rated : "TV-G"
  runtime : 11
▸ genres : Array (2)
▸ cast : Array (4)
▸ languages : Array (1)
  released : 1903-12-01T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc47')
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (2)
▸ cast : Array (1)
▸ languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```


Sent Mail - sridurga.r.2024.aids
PL/SQL trigger examples
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc49')
  title : "Modern Love Scene"
  year : 1921
  rated : "PG"
  runtime : 90
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1921-06-12T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4b')
  title : "The Secret Scene"
  year : 2020
  rated : "PG-13"
  runtime : 123
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

8. Retrieve all movies from the 'movies' collection that have complete information and have an IMDb rating higher than 7.


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass

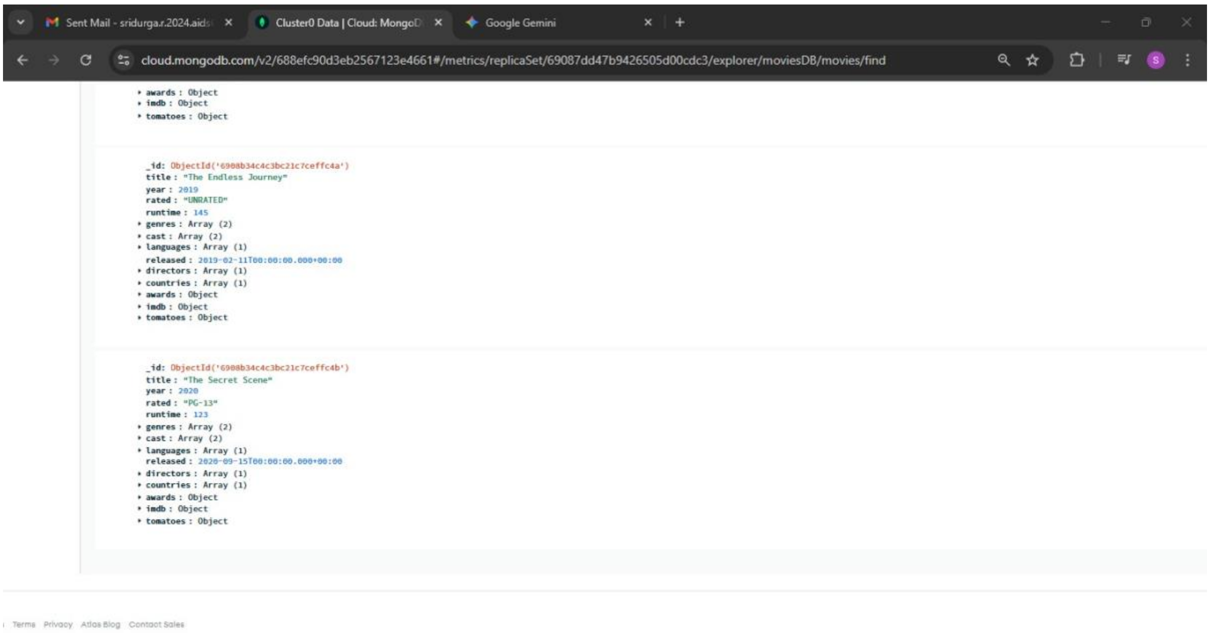
moviesDB.movies

STORAGE SIZE: 4KB LOGICAL DATA SIZE: 0B TOTAL DOCUMENTS: 0 INDEXES TOTAL SIZE: 4KB

Filter `{ "imdb.rating": { "$gt": 7 } }`

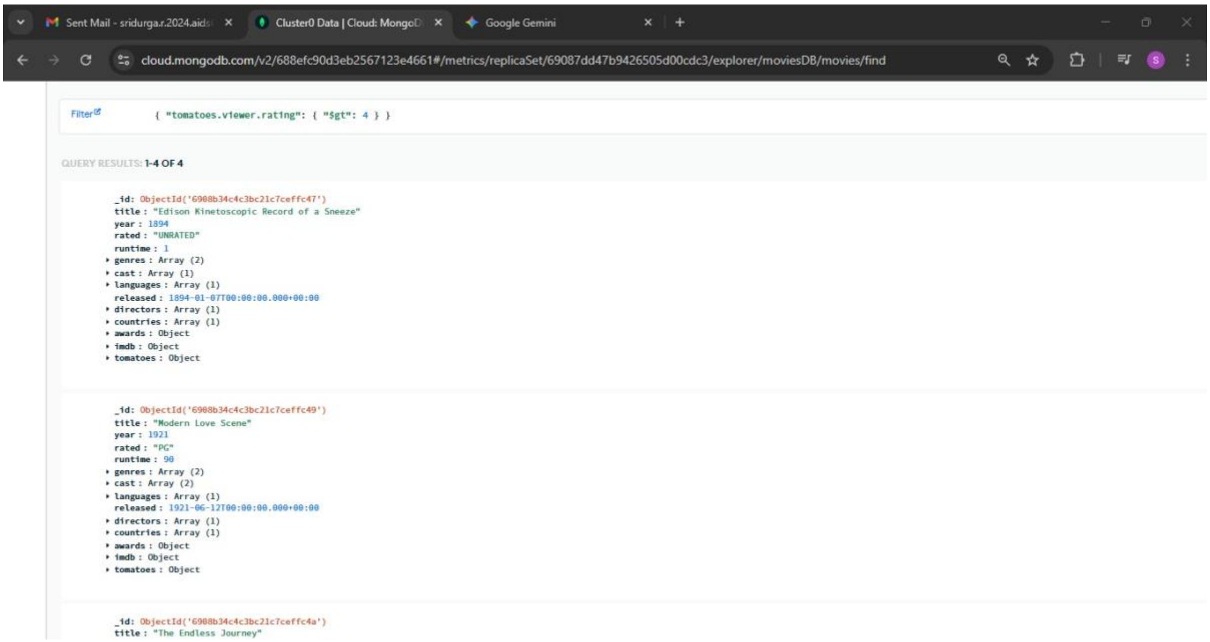
QUERY RESULTS: 1-4 OF 4

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc46')
  title : "The Great Train Robbery"
  year : 1903
  rated : "TV-G"
  runtime : 11
• genres : Array (2)
• cast : Array (4)
• languages : Array (1)
  released : 1903-12-01T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc49')
  title : "Modern Love Scene"
  year : 1921
  rated : "PG"
  runtime : 90
• genres : Array (2)
• cast : Array (2)
• languages : Array (1)
  released : 1921-06-12T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object
```


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Terms
Privacy
Atlas Blog
Contact Sales

```
• awards : Object
• imdb : Object
• tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
• genres : Array (2)
• cast : Array (2)
• languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4b')
  title : "The Secret Scene"
  year : 2020
  rated : "PG-13"
  runtime : 123
• genres : Array (2)
• cast : Array (2)
• languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
• directors : Array (1)
• countries : Array (1)
• awards : Object
• imdb : Object
• tomatoes : Object
```

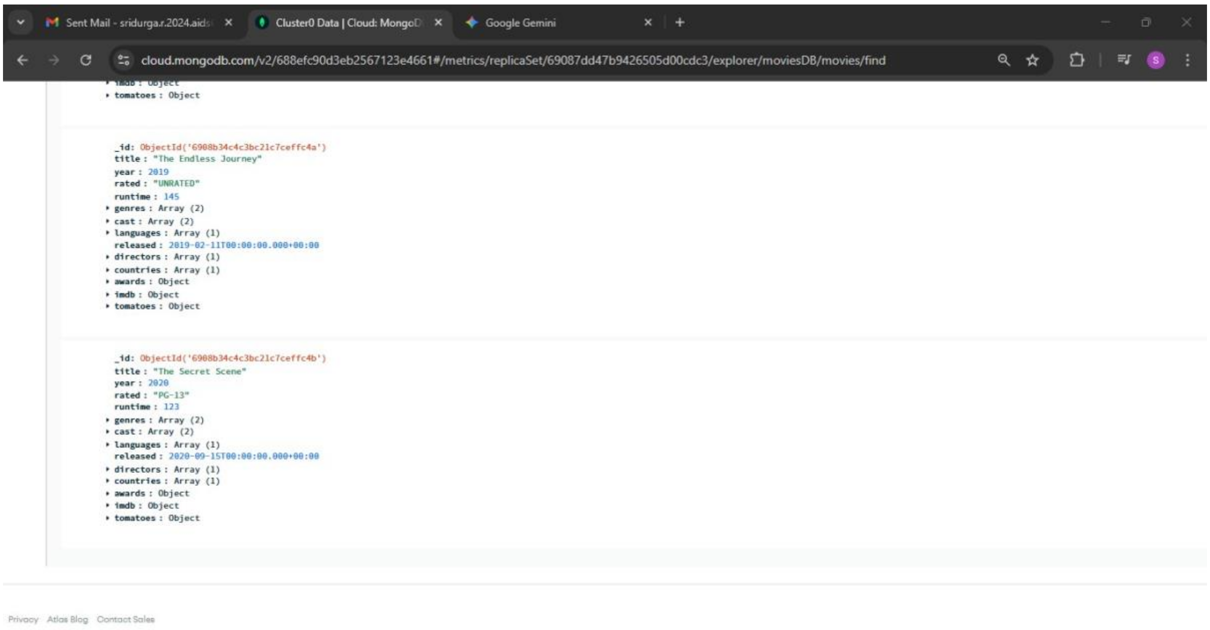
9. Retrieve all movies from the 'movies' collection that have complete information and have a viewer rating higher than 4 on Tomatoes.


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Filter

```
{ "tomatoes.viewer.rating": { "$gt": 4 } }
```

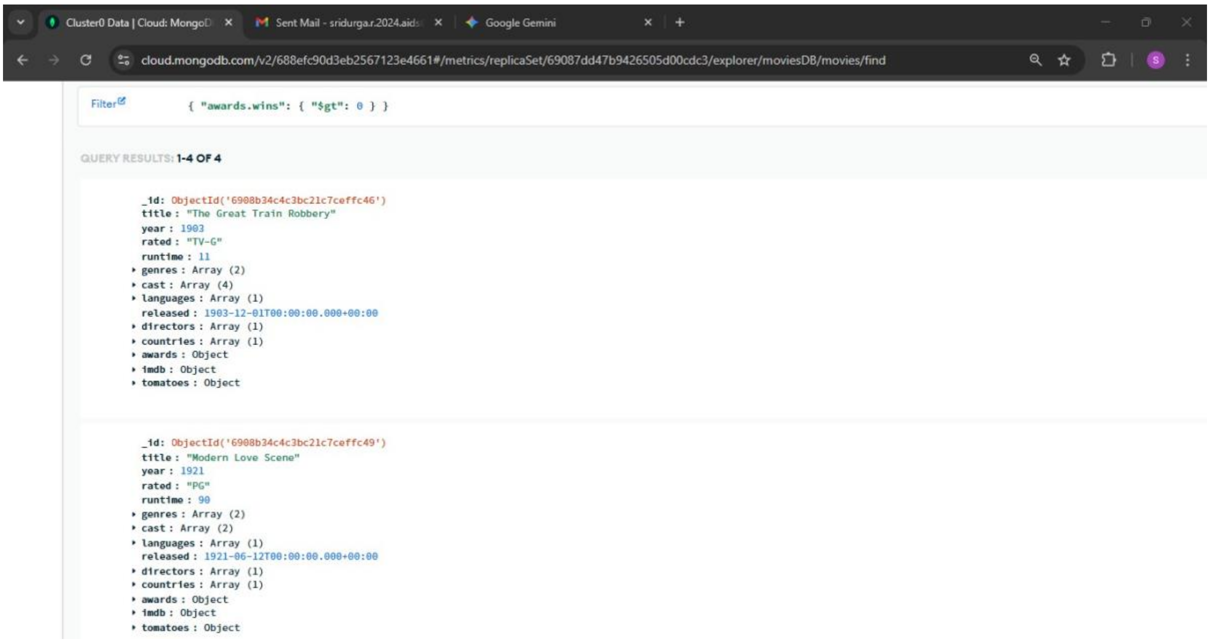
QUERY RESULTS: 1-4 OF 4

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc47')
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  rated : "UNRATED"
  runtime : 1
▸ genres : Array (2)
▸ cast : Array (1)
▸ languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc49')
  title : "Modern Love Scene"
  year : 1921
  rated : "PG"
  runtime : 90
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1921-06-12T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
```


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find

```
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4b')
  title : "The Secret Scene"
  year : 2020
  rated : "PG-13"
  runtime : 123
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

10. Retrieve all movies from the 'movies' collection that have received an award.

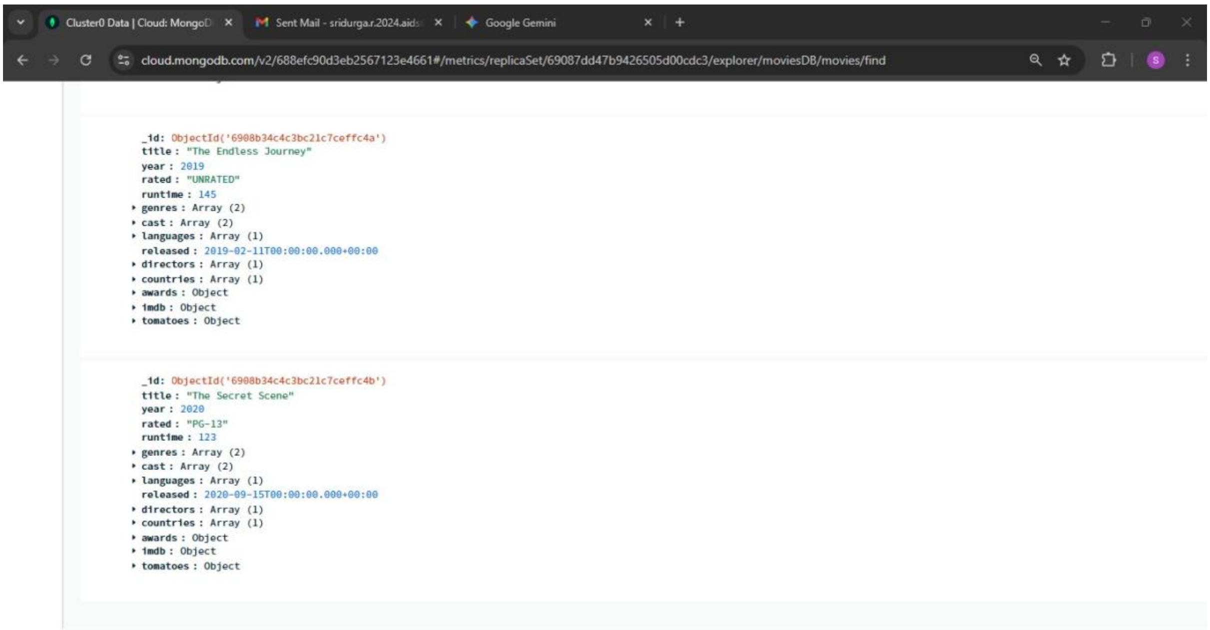


Cluster0 Data | Cloud: MongoD
Sent Mail - sridurga.r.2024.aids
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Filter

```
{ "awards.wins": { "$gt": 0 } }
```

QUERY RESULTS: 1-4 OF 4

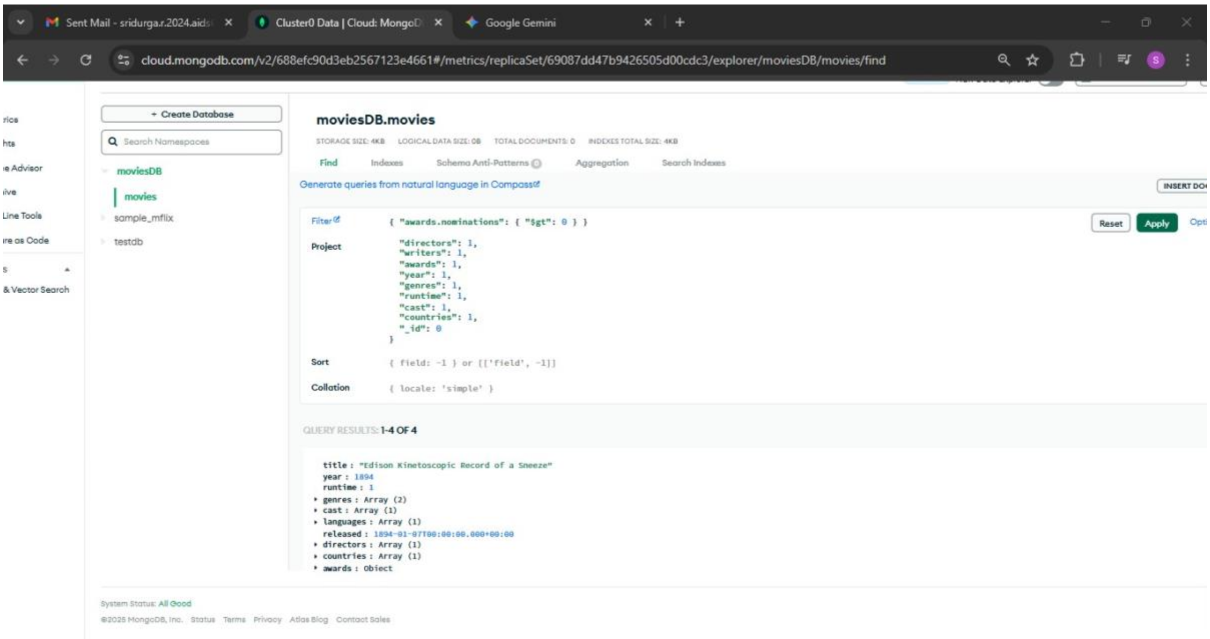
```
  _id: ObjectId('6908b34c4c3bc21c7ceffc46')
  title : "The Great Train Robbery"
  year : 1903
  rated : "TV-G"
  runtime : 11
▸ genres : Array (2)
▸ cast : Array (4)
▸ languages : Array (1)
  released : 1903-12-01T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc49')
  title : "Modern Love Scene"
  year : 1921
  rated : "PG"
  runtime : 90
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1921-06-12T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```



Cluster0 Data | Cloud: MongoD
Sent Mail - sridurga.r.2024.aids
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find

```
  _id: ObjectId('6908b34c4c3bc21c7ceffc4a')
  title : "The Endless Journey"
  year : 2019
  rated : "UNRATED"
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object

  _id: ObjectId('6908b34c4c3bc21c7ceffc4b')
  title : "The Secret Scene"
  year : 2020
  rated : "PG-13"
  runtime : 123
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
▸ imdb : Object
▸ tomatoes : Object
```

11. Find all movies with title, languages, released, directors, writers, awards, year, genres, runtime, cast, countries from the 'movies' collection in MongoDB that have at least one nomination.

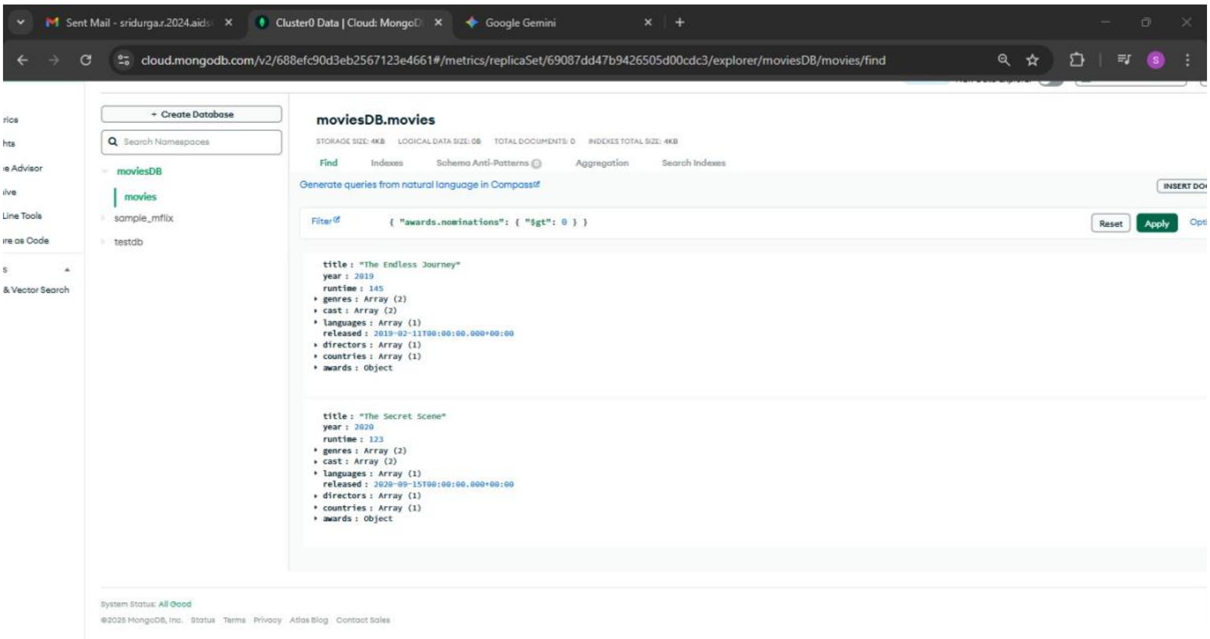


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
+ Create Database
Search Namespaces
moviesDB
movies
sample_mflix
testdb
moviesDB.movies
STORAGE SIZE: 4KB LOGICAL DATA SIZE: 0B TOTAL DOCUMENTS: 0 INDEXES TOTAL SIZE: 4KB
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass
INSERT DO
Reset
Apply
Opti
System Status: All Good
©2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

```
Filter     { "awards.nominations": { "$gt": 0 } }
Project      "directors": 1,
             "writers": 1,
             "awards": 1,
             "year": 1,
             "genres": 1,
             "runtime": 1,
             "cast": 1,
             "countries": 1,
             "_id": 0
           }
Sort       { field: -1 } or [['field', -1]]
Collation  { locale: 'simple' }
```

QUERY RESULTS: 1-4 OF 4

```
  title : "Edison Kinetoscopic Record of a Sneeze"
  year : 1894
  runtime : 1
▸ genres : Array (2)
▸ cast : Array (1)
▸ languages : Array (1)
  released : 1894-01-07T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
```

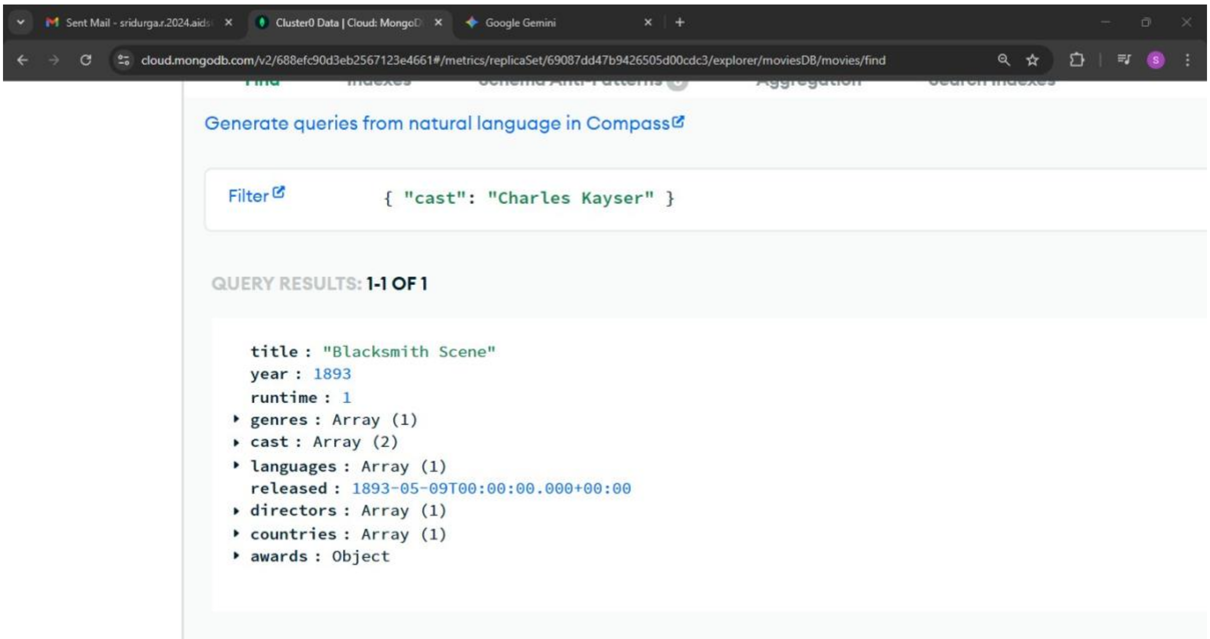


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
+ Create Database
Search Namespaces
moviesDB
movies
sample_mflix
testdb
moviesDB.movies
STORAGE SIZE: 4KB LOGICAL DATA SIZE: 0B TOTAL DOCUMENTS: 0 INDEXES TOTAL SIZE: 4KB
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass
INSERT DO
Reset
Apply
Opti
System Status: All Good
©2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

```
Filter     { "awards.nominations": { "$gt": 0 } }
```

```
  title : "The Endless Journey"
  year : 2019
  runtime : 145
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2019-02-11T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object

  title : "The Secret Scene"
  year : 2020
  runtime : 123
▸ genres : Array (2)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
```

12. Find all movies with title, languages, released, directors, writers, awards, year, genres, runtime, cast, countries from the 'movies' collection in MongoDB with cast including "Charles Kayser".

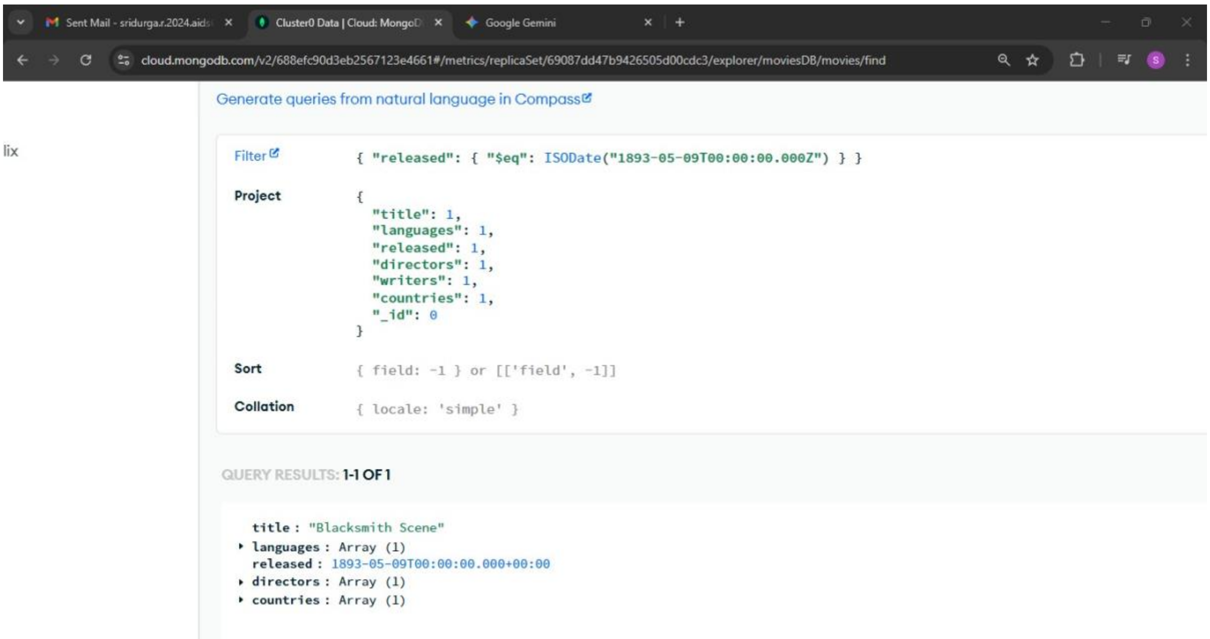

Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Generate queries from natural language in Compass

Filter `{ "cast": "Charles Kayser" }`

QUERY RESULTS: 1-1 OF 1

```
  title : "Blacksmith Scene"
  year : 1893
  runtime : 1
▸ genres : Array (1)
▸ cast : Array (2)
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
▸ awards : Object
```

13. Retrieve all movies with title, languages, released, directors, writers, countries from the 'movies' collection in MongoDB that released on May 9, 1893.


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
Generate queries from natural language in Compass
lix

Filter `{ "released": { "$eq": ISODate("1893-05-09T00:00:00.000Z") } }`

Project
```
{
  "title": 1,
  "languages": 1,
  "released": 1,
  "directors": 1,
  "writers": 1,
  "countries": 1,
  "_id": 0
}
```

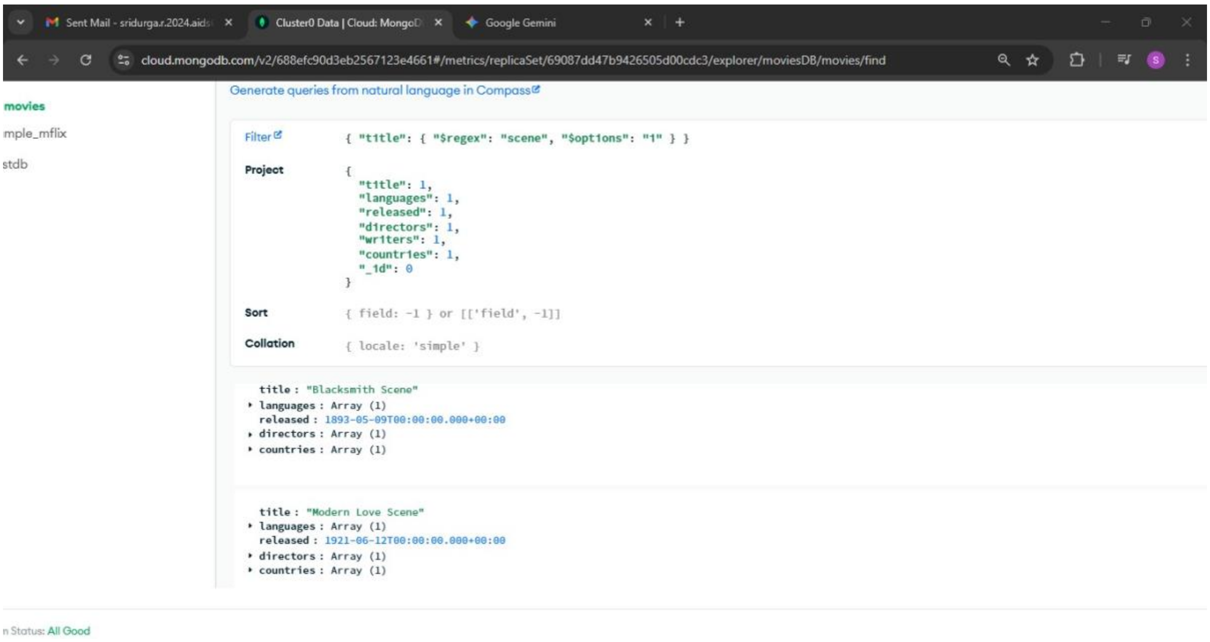
Sort `{ field: -1 } or [['field', -1]]`

Collation `{ locale: 'simple' }`

QUERY RESULTS: 1-1 OF 1

```
  title : "Blacksmith Scene"
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
```

14. Retrieve all movies with title, languages, released, directors, writers, countries from the 'movies' collection in MongoDB that have a word "scene" in the title.


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
movies
mple_mflix
stdb
Generate queries from natural language in Compass

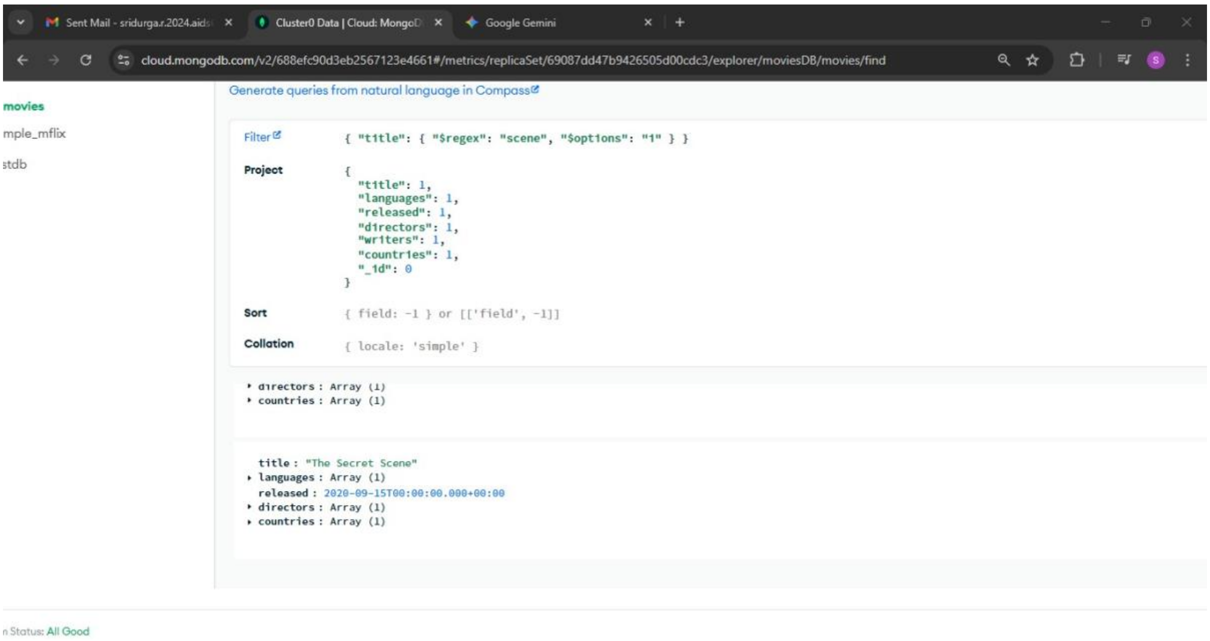
Filter `{ "title": { "$regex": "scene", "$options": "i" } }`

Project
```
{
  "title": 1,
  "languages": 1,
  "released": 1,
  "directors": 1,
  "writers": 1,
  "countries": 1,
  "_id": 0
}
```

Sort `{ field: -1 } or [['field', -1]]`

Collation `{ locale: 'simple' }`

```
  title : "Blacksmith Scene"
▸ languages : Array (1)
  released : 1893-05-09T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)

  title : "Modern Love Scene"
▸ languages : Array (1)
  released : 1921-06-12T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
```


Sent Mail - sridurga.r.2024.aids
Cluster0 Data | Cloud: MongoD
Google Gemini
cloud.mongodb.com/v2/688efc90d3eb2567123e4661#/metrics/replicaSet/69087dd47b9426505d00cdc3/explorer/moviesDB/movies/find
movies
mple_mflix
stdb
Generate queries from natural language in Compass

Filter `{ "title": { "$regex": "scene", "$options": "i" } }`

Project
```
{
  "title": 1,
  "languages": 1,
  "released": 1,
  "directors": 1,
  "writers": 1,
  "countries": 1,
  "_id": 0
}
```

Sort `{ field: -1 } or [['field', -1]]`

Collation `{ locale: 'simple' }`

```
▸ directors : Array (1)
▸ countries : Array (1)

  title : "The Secret Scene"
▸ languages : Array (1)
  released : 2020-09-15T00:00:00.000+00:00
▸ directors : Array (1)
▸ countries : Array (1)
```