**Hussien Tarek**

# Part 1:

## 3. Cassandra and MongoDB: ACID vs. BASE Transaction Management Comparison Briefing

### 1.1 Cassandra

*ACID Transaction Management:*

- Supports atomicity on row & partition level but not on multiple records operations level so, conflict in operations success on different replicas or subset of records processes won't result in instantaneous rollback.
- Provides flexible eventual consistency as it has two types & one of each is tunable: -

1. Tunable Consistency (in terms of CAP, most up-to-date transaction): Allow managing consistency & availability trade-offs by defining replicas policy to acknowledge a read or write operation successfully.
2. Linearizable Consistency (in terms of ACID, transaction doesn't corrupt database): Allow simulation of some degree of atomicity & serial isolation level for transactions by evaluating responses of different replica nodes to read or write operations & prioritize the latest timestamp response in case of inconsistency using a "compare-and-set" (CAS) request.

- Earlier versions of Cassandra allowed other users visibility of partial row update but now it allows full-row level isolation across different users such that partial updates are only visible for those users who performed the update operation.
- Provide data durability globally by replicating it across multiple nodes & locally by saving data in memory & through a commit log.

### *BASE Transaction Management:*

- Cassandra is primarily designed to prioritize high availability. This is achieved through its distributed storage system, which allows for linear scalability across multiple data centers or within the same node.
- To ensure this high availability, Cassandra employs eventual consistency as a compromise, although it does offer flexibility in other areas as mentioned before. For lightweight transactions that require strong consistency, Cassandra provides linearizable consistency. Additionally, users can define the number of replicas required to reach consensus when reading updates from the database, allowing for tunable consistency.
- Has a soft state due to the eventual consistency.

## 1.2 MongoDB

### *ACID Transaction Management:*

- Supports atomicity at a single document level and its subdocuments. MongoDB 4.2 versions support atomicity on multiple document level with a performance impact [3].
- By default, MongoDB offers strong consistency. The primary servers in a MongoDB cluster are highly consistent compared to the secondary servers which only provide eventual consistency. Consequently, accessing data from secondary nodes is not permitted by default However, this can be modified to prioritize availability over consistency.
- Allow for document update full isolation & rolling back the update in case of an error & provide isolation patterns & operators like "update if current" that prevent updates if the document changed after the last read & "$isolation" that ensures update operation isolated from other operations.
- Uses the same traditional database durability model that provides tuning of durability & performance trade-offs through configuring transactions committing to the database after saving updates to the journal files. by parameters like "syncdelay" & "journalCommitInterval".

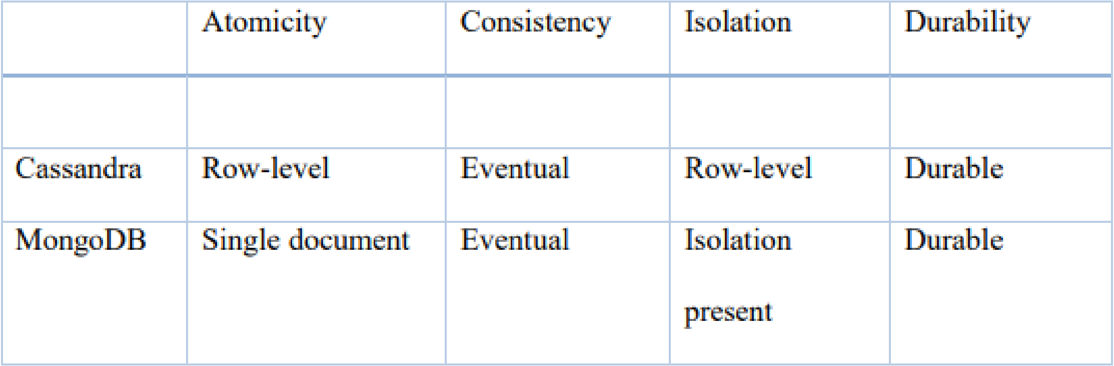
Based on the information available, it can be inferred that earlier versions of MongoDB only supported ACID compliance at the document level. However, starting from version 4.0, MongoDB introduced support for multi-document ACID transactions. Furthermore, in version 4.2, distributed multi-document ACID transactions were also implemented; although this came with a trade-off in terms of performance [4].

### *BASE Transaction Management:*

- MongoDB ensures availability by implementing replica sets in each data center. However, it prioritizes consistency over availability.
- As previously stated, MongoDB ensures strong consistency of primary server data and by default limits access to secondary servers for consistent reads.
- Has a soft state due to the eventual consistency of secondary MongoDB servers.

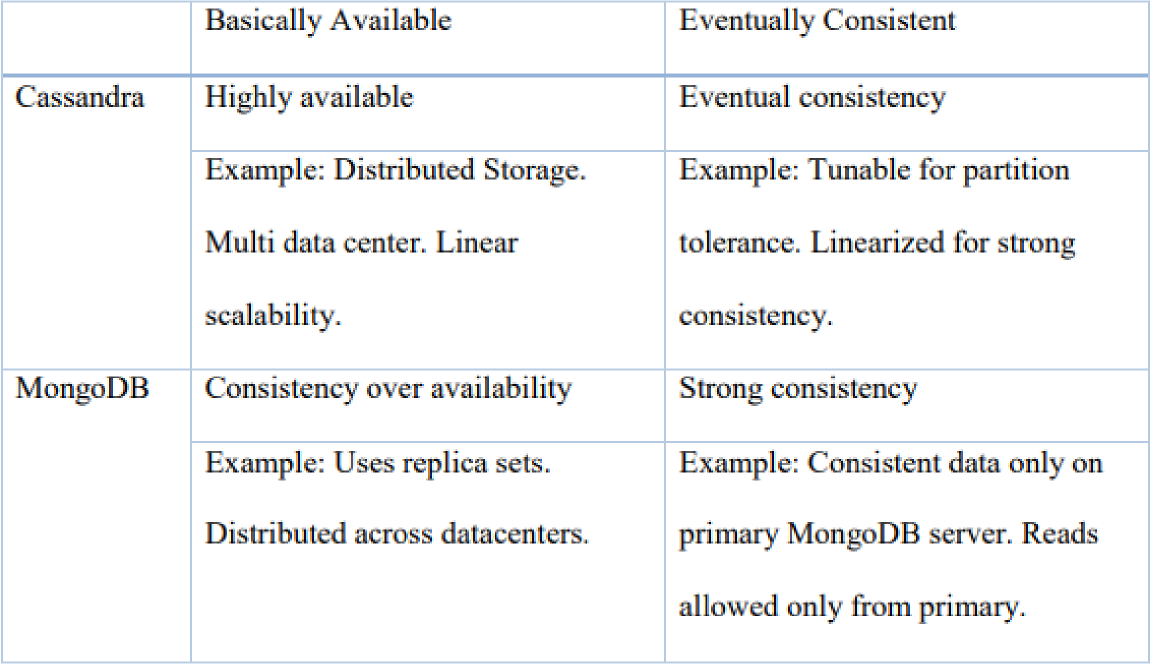
## 1.3 Conclusion

The transaction management capabilities of Cassandra and MongoDB showcase their distinctive design principles and priorities. Cassandra places emphasis on Basic Availability, Soft State, and Eventual Consistency rather than strong consistency. On the other hand, MongoDB offers comprehensive support for ACID transactions along with robust guarantees of strong consistency. Determining the most suitable database for your application depends on specific requirements and trade-offs that need to be considered between availability, partition tolerance, and consistency.



| | Atomicity | Consistency | Isolation | Durability |
|---|---|---|---|---|
| | | | | |
| Cassandra | Row-level | Eventual | Row-level | Durable |
| MongoDB | Single document | Eventual | Isolation present | Durable |

*Table 1 ACID Properties on Cassandra & MongoDB [5].*



| | Basically Available | Eventually Consistent |
|---|---|---|
| Cassandra | Highly available | Eventual consistency |
| | Example: Distributed Storage. Multi data center. Linear scalability. | Example: Tunable for partition tolerance. Linearized for strong consistency. |
| MongoDB | Consistency over availability | Strong consistency |
| | Example: Uses replica sets. Distributed across datacenters. | Example: Consistent data only on primary MongoDB server. Reads allowed only from primary. |

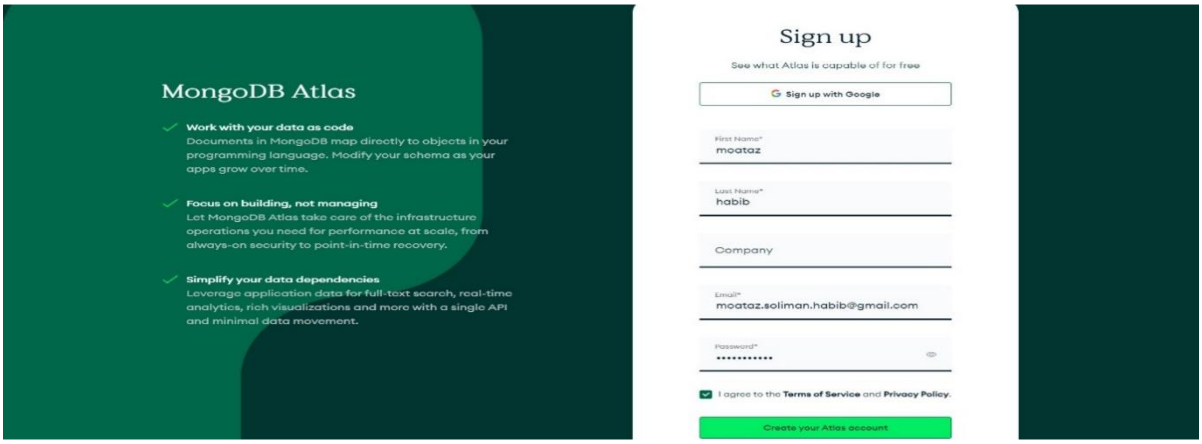
*Table 2 BASE Properties on Cassandra & MongoDB [5].*

# Part 2:

1) MongoDB Lab

Setup:

- Set an account on MongoDB Atlas - https://cloud.mongodb.com



MongoDB Atlas

**Work with your data as code**
Documents in MongoDB map directly to objects in your programming language. Modify your schema as your apps grow over time.

**Focus on building, not managing**
Let MongoDB Atlas take care of the infrastructure operations you need for performance at scale, from always-on security to point-in-time recovery.

**Simplify your data dependencies**
Leverage application data for full-text search, real-time analytics, rich visualizations and more with a single API and minimal data movement.

Sign up

See what Atlas is capable of for free

First Name* moataz

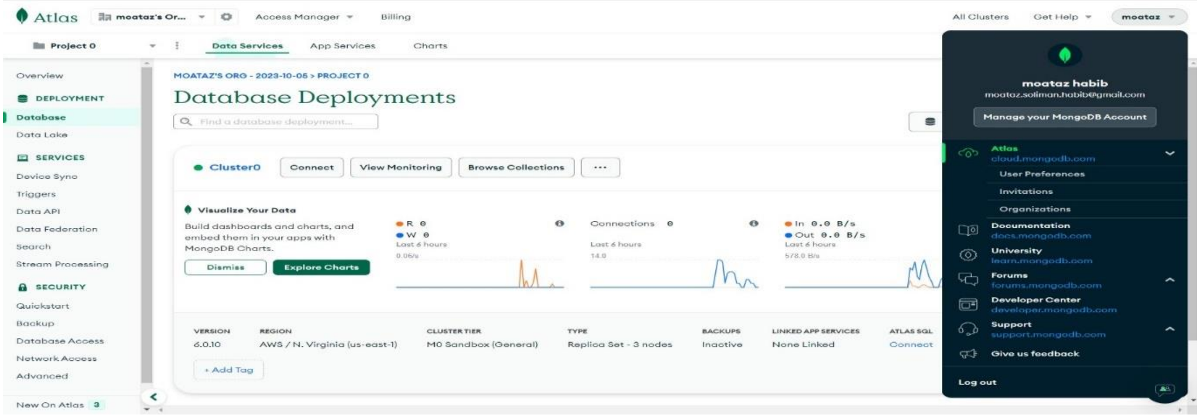
Last Name* habib

Company

Email* moataz.soliman.habib@gmail.com

Password* ••••••••••••

☑ I agree to the **Terms of Service** and **Privacy Policy**.



MOATAZ'S ORG - 2023-10-05 › PROJECT 0

Database Deployments

Cluster0

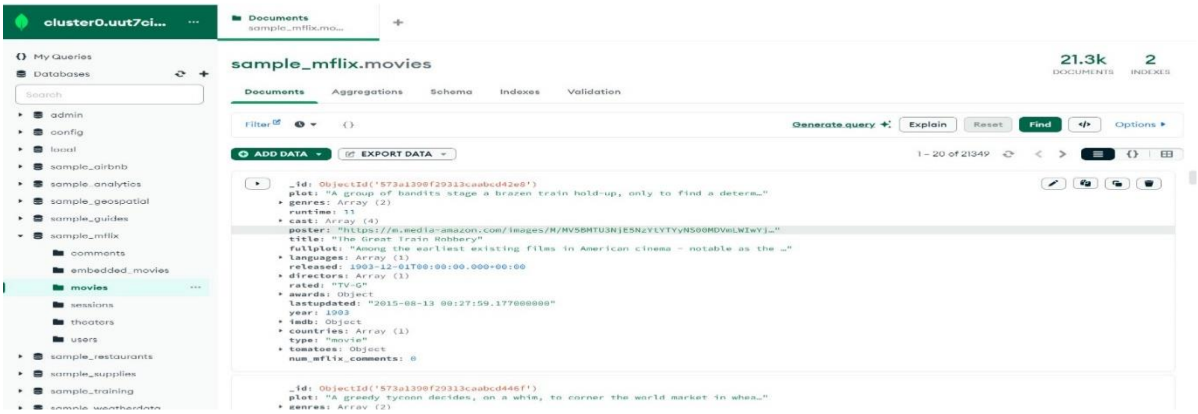
Visualize Your Data
Build dashboards and charts, and embed them in your apps with MongoDB Charts.

| VERSION | REGION | CLUSTER TIER | TYPE | BACKUPS | LINKED APP SERVICES | ATLAS SQL |
|---|---|---|---|---|---|---|
| 6.0.10 | AWS / N. Virginia (us-east-1) | M0 Sandbox (General) | Replica Set - 3 nodes | Inactive | None Linked | Connect |

moataz habib
moataz.soliman.habib@gmail.com

*Figure 2 MongoDB Setup*

- Load the Sample Netflix Movies Database to your Data Lake



sample_mflix.movies

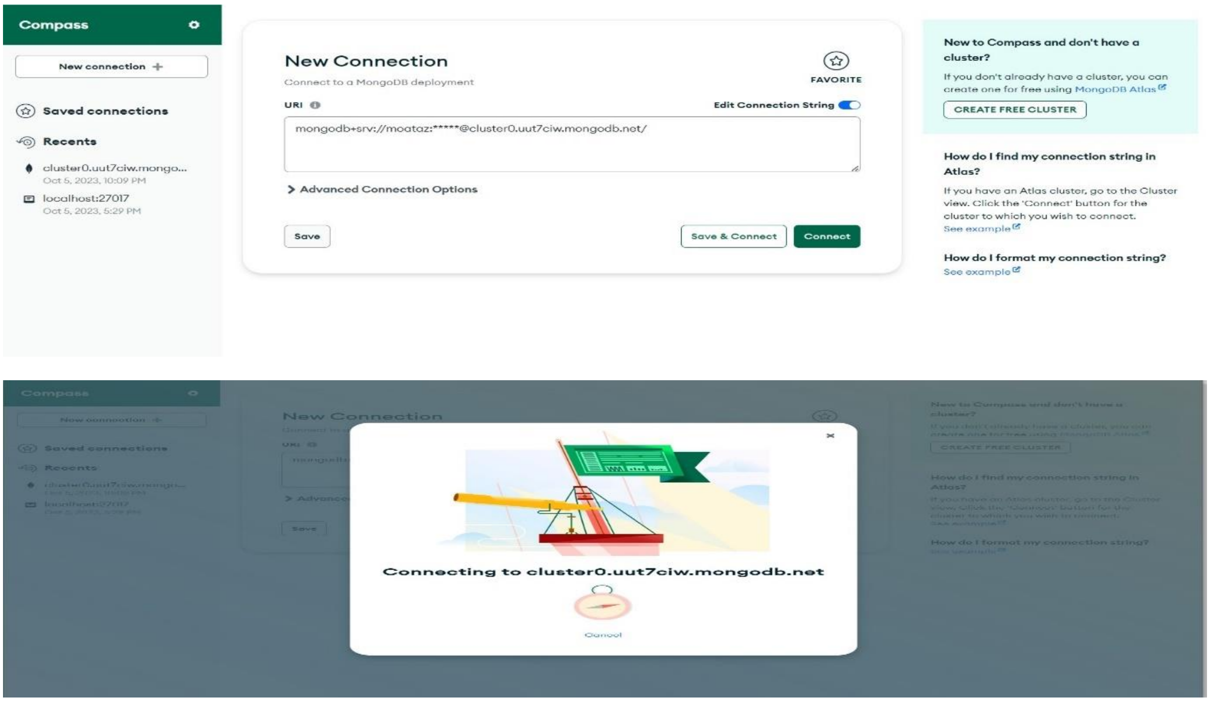
21.3k DOCUMENTS 2 INDEXES

1 – 20 of 21349

```
_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"
fullplot: "Among the earliest existing films in American cinema - notable as the …"
languages: Array (1)
released: 1903-12-01T00:00:00.000+00:00
directors: Array (1)
rated: "TV-G"
awards: Object
lastupdated: "2015-08-13 00:27:59.177000000"
year: 1903
imdb: Object
countries: Array (1)
type: "movie"
tomatoes: Object
num_mflix_comments: 0

_id: ObjectId('573a1390f29313caabcd446f')
plot: "A greedy tycoon decides, on a whim, to corner the world market in whea…"
genres: Array (2)
```

*Figure 3 Load Data*

- Set up a connection to this database instance from MongoDB compass or any other MongoDB client.



New Connection

Connect to a MongoDB deployment

URI

`mongodb+srv://moataz:*****@cluster0.uut7ciw.mongodb.net/`

New to Compass and don't have a cluster?

If you don't already have a cluster, you can create one for free using MongoDB Atlas

How do I find my connection string in Atlas?

If you have an Atlas cluster, go to the Cluster view. Click the 'Connect' button for the cluster to which you wish to connect.

How do I format my connection string?

Connecting to cluster0.uut7ciw.mongodb.net

*Figure 4 Connection Setup*

1. Briefly describe the movies database document model.

The movies database document model consists of a collection named "movies" that contains documents representing movies.
Each movie document has various fields.
The document model allows for flexible and dynamic data storage, as each movie document can have different fields and values.

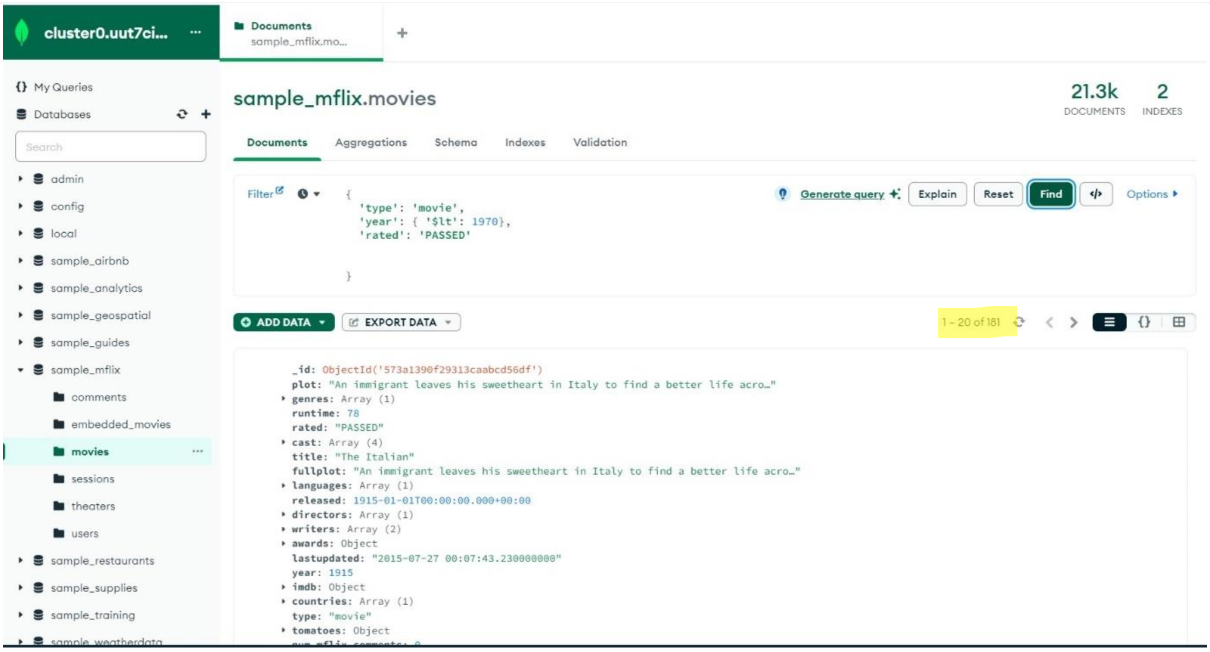
| Field | Description | Count | Categorization of Documents |
|---|---|---|---|
| **_id** | The unique identifier for this document in MongoDB. | 21349 | Required field |
| **plot** | A brief synopsis of the movie. | 20203 | Optional field |
| **genres** | An array listing the genres relevant to the movie. | 21237 | Optional field |
| **runtime** | The runtime of the movie in minutes. | 20910 | Optional field |
| **rated** | The rating for the movie. (e.g., `"TV-G"`,`"R"`, `"G"`, `TV-MA"`). | 11455 | Optional field |

| | | | |
|---|---|---|---|
| **cast** | An array listing the actors featured in the movie. | 20987 | Optional field |
| **num_mflix_comments** | The number of comments made in the MFLIX system. | 21349 | Optional field |
| **poster** | A URL to the poster image of the movie. | 18044 | Optional field |
| **title** | The title of the movie. | 21349 | Required field |
| **fullplot** | A full synopsis of the movie plot. | 19852 | Optional field |
| **languages** | An array of languages in which the movie is available. | 21119 | Embedded fields |
| **released** | The release date of the movie. | 20878 | Optional field |
| **directors** | An array listing the directors of the movie. | 21107 | Optional field |
| **writers** | An array listing the writers of the movie. | 20256 | Optional field |
| **awards** | An object detailing the awards the movie has won. | 21349 | Embedded field |
| **lastupdated** | The timestamp of the last update made to this document. | 21349 | Optional field |
| **year** | The year the movie was released. | 21349 | Optional field |
| **imdb** | An object with details about the film's IMDB rating, votes, and ID. | 21349 | Embedded field |
| **countries** | An array of countries where the movie was released. | 21339 | Embedded field |
| **type** | Defines the type of the document, in this case, a "movie". | 21349 | Optional field |
| **tomatoes** | An object with ratings from Rotten Tomato reviews. | 18588 | Embedded field |
| **metacritic** | Metacritic score or rating for the movie. | 6964 | embedded field |

*Table 3 Table demonstrate movies database*

2. Filter the documents for type “movies” that are released before 1970 and rated as “PASSED.”

```
{
'type': 'movie',
 'year': { '$lt': 1970},
 'rated': 'PASSED'
}
```



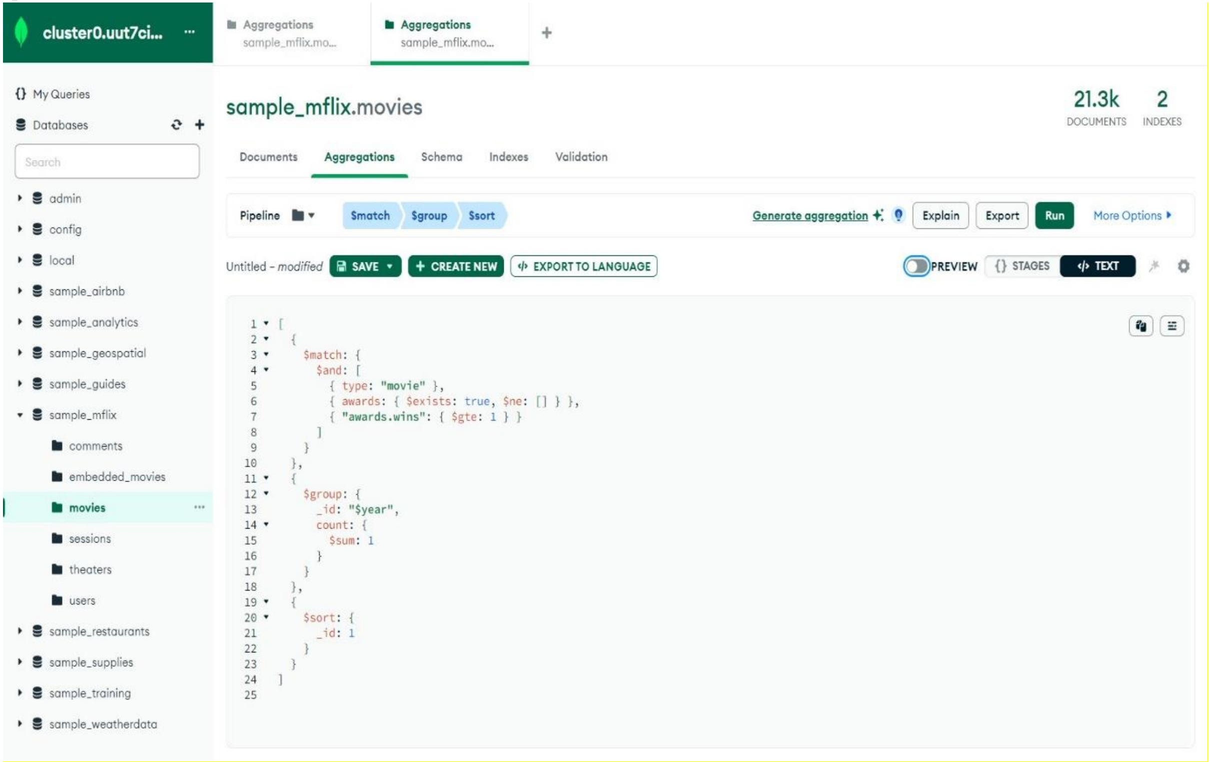
```
{
  'type': 'movie',
  'year': { '$lt': 1970},
  'rated': 'PASSED'


}
```

```
  _id: ObjectId('573a1390f29313caabcd56df')
  plot: "An immigrant leaves his sweetheart in Italy to find a better life acro…"
▸ genres: Array (1)
  runtime: 78
  rated: "PASSED"
▸ cast: Array (4)
  title: "The Italian"
  fullplot: "An immigrant leaves his sweetheart in Italy to find a better life acro…"
▸ languages: Array (1)
  released: 1915-01-01T00:00:00.000+00:00
▸ directors: Array (1)
▸ writers: Array (2)
▸ awards: Object
  lastupdated: "2015-07-27 00:07:43.230000000"
  year: 1915
▸ imdb: Object
▸ countries: Array (1)
  type: "movie"
▸ tomatoes: Object
```

*Figure 5 Filter documents*

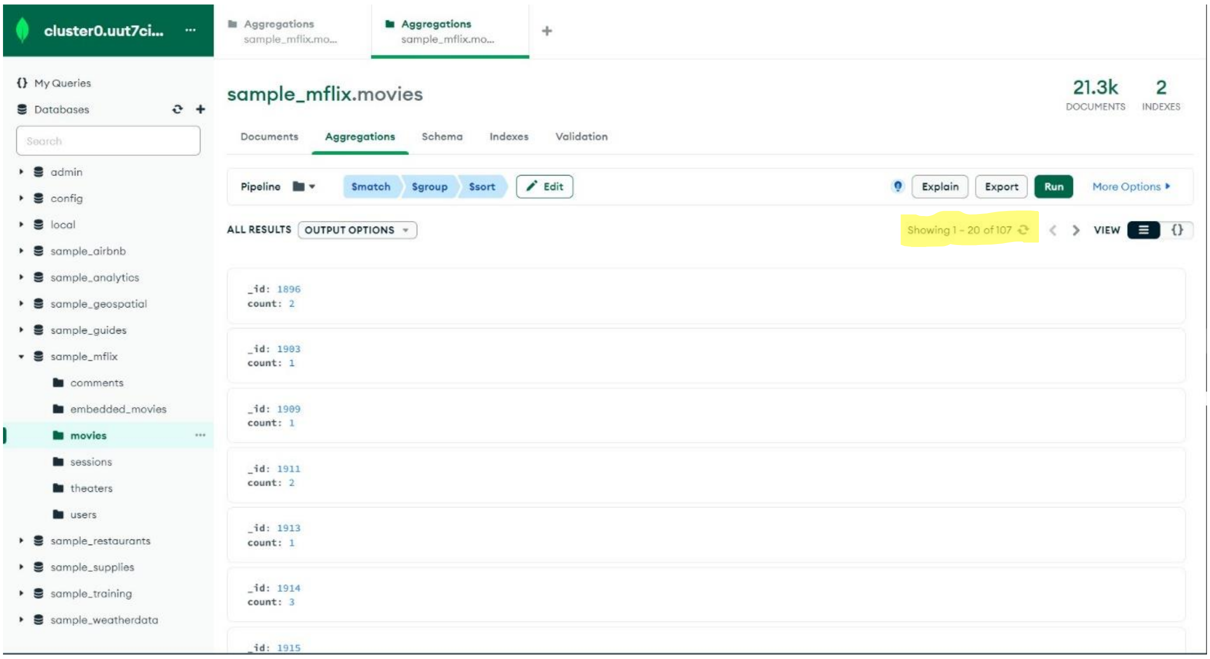
3. Build an Aggregation Pipeline that shows all entries of type movie that have won at least one award and return the release year aggregate counts.

```
[
 {
 $match: {
     $and: [
         { type: "movie" },
         { awards: { $exists: true, $ne: [] } },
         { "awards.wins": { $gte: 1 } }
     ]
   }
},
{
 $group: {
  _id: "$year",
  count: {
  $sum: 1
    }
  }
},
{
   $sort: {
  _id: 1
   }
 }
]
```



```
[
  {
    $match: {
      $and: [
        { type: "movie" },
        { awards: { $exists: true, $ne: [] } },
        { "awards.wins": { $gte: 1 } }
      ]
    }
  },
  {
    $group: {
      _id: "$year",
      count: {
        $sum: 1
      }
    }
  },
  {
    $sort: {
      _id: 1
    }
  }
]
```

*Figure 6 Aggregation Pipeline*

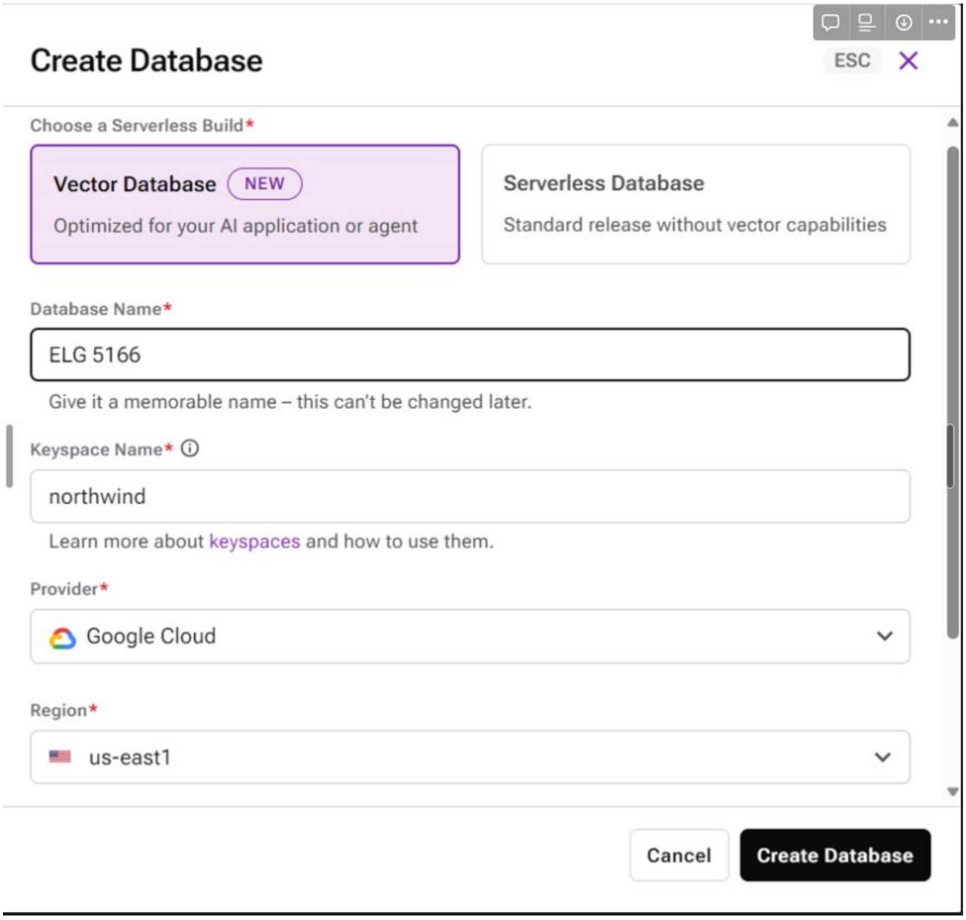


```
_id: 1896
count: 2

_id: 1903
count: 1

_id: 1909
count: 1

_id: 1911
count: 2

_id: 1913
count: 1

_id: 1914
count: 3

_id: 1915
```

*Figure 7 Aggregation Pipeline con.*

2) Cassandra Lab:

*Figure 8 Cassandra Setup*



Choose a Serverless Build*: Vector Database

Database Name*: ELG 5166

Give it a memorable name – this can't be changed later.

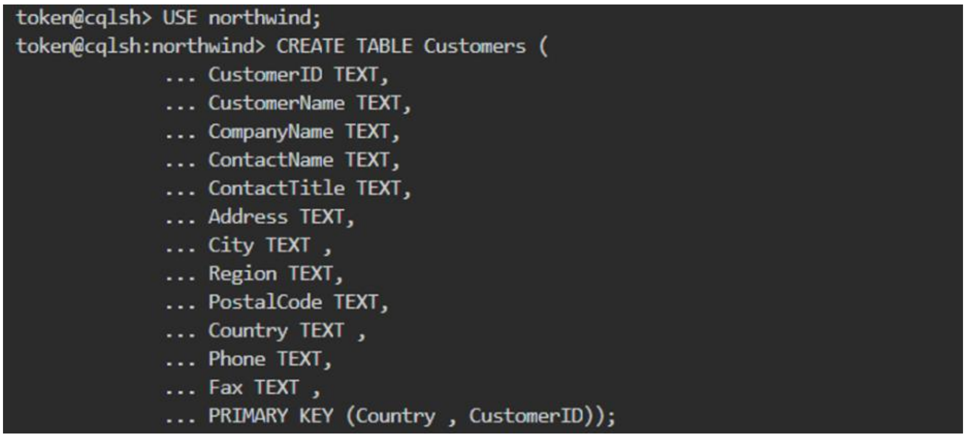
Keyspace Name*: northwind

Learn more about keyspaces and how to use them.

Provider*: Google Cloud

Region*: us-east1

- Create a Keyspace called northwind
- Create the customer tables



```
token@cqlsh> USE northwind;
token@cqlsh:northwind> CREATE TABLE Customers (
            ... CustomerID TEXT,
            ... CustomerName TEXT,
            ... CompanyName TEXT,
            ... ContactName TEXT,
            ... ContactTitle TEXT,
            ... Address TEXT,
            ... City TEXT ,
            ... Region TEXT,
            ... PostalCode TEXT,
            ... Country TEXT ,
            ... Phone TEXT,
            ... Fax TEXT ,
            ... PRIMARY KEY (Country , CustomerID));
```

*Figure 9 customer tables*

## Load the attached data:

```
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('ALFKI', 'Alfreds Futterkiste', 'Maria Anders', 'Sales Representative', 'Obere Str. 57', 'Berlin', 'NS', '12209', 'Germany', '030-0074321', '
030-0076545');
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('ANATR', 'Ana Trujillo Emparedados y helados', 'Ana Trujillo', 'Owner', 'Avda. de la Constitución 2222', 'México D.F.', 'NS', '05021', 'Mexic
o', '(5) 555-4729', '(5) 555-3745');
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('ANTON', 'Antonio Moreno Taquería', 'Antonio Moreno', 'Owner', 'Mataderos  2312', 'México D.F.', 'NS', '05023', 'Mexico', '(5) 555-3932', 'NS
');
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('AROUT', 'Around the Horn', 'Thomas Hardy', 'Sales Representative', '120 Hanover Sq.', 'London', 'NS', 'WA1 1DP', 'UK', '(171) 555-7788', '(1
71) 555-6750');
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('BERGS', 'Berglunds snabbköp', 'Christina Berglund', 'Order Administrator', 'Berguvsvägen  8', 'Luleå', 'NS', 'S-958 22', 'Sweden', '0921-12
34 65', '0921-12 34 67');
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('BLAUS', 'Blauer See Delikatessen', 'Hanna Moos', 'Sales Representative', 'Forsterstr. 57', 'Mannheim', 'NS', '68306', 'Germany', '0621-08460
', '0621-08924');
token@cqlsh:northwind> INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
          ... VALUES('BLONP', 'Blondesddsl père et fils', 'Frédérique Citeaux', 'Marketing Manager', '24, place Kléber', 'Strasbourg', 'NS', '67000', 'France', '8
8.60.15.31', '88.60.15.32');
) 642-7012');
token@cqlsh:northwind> SELECT * FROM Customers;

 country    | customerid | address                       | city      | companyname                    | contactname          | conta
cttitle                 | customername | fax             | phone           | postalcode | region
------------+------------+-------------------------------+-----------+--------------------------------+----------------------+------
------------------------+--------------+-----------------+-----------------+------------+--------------
      Spain |      BOLID |               C/ Araquil, 67 |    Madrid |      Bólido Comidas preparadas |        Martín Sommer |
                  Owner |         null |   (91) 555 91 99 |  (91) 555 22 82 |      28023 |            NS
      Spain |      FISSA |            C/ Moralzarzal, 86 |    Madrid | FISSA Fabrica Inter. Salchichas S.A. |       Diego Roel |
     Accounting Manager |         null |   (91) 555 55 93 |  (91) 555 94 44 |      28034 |            NS
      Spain |      GALED |        Rambla de Cataluña, 23 | Barcelona |         Galería del gastrónomo |     Eduardo Saavedra |
      Marketing Manager |         null |   (93) 203 4561 |   (93) 203 4560 |      08022 |            NS
      Spain |      GODOS |                C/ Romero, 33 |   Sevilla |             Godos Cocina Típica |    José Pedro Freyre |
          Sales Manager |         null |              NS |  (95) 555 82 82 |      41101 |            NS
      Spain |      ROMEY |                  Gran Vía, 1 |    Madrid |               Romero y tomillo |     Alejandra Camino |
     Accounting Manager |         null |   (91) 745 6210 |   (91) 745 6200 |      28001 |            NS
    Austria |      ERNSH |                  Kirchgasse 6 |      Graz |                    Ernst Handel |        Roland Mendel |
          Sales Manager |         null |       7675-3426 |       7675-3425 |       8010 |            NS
    Austria |      PICCO |                   Geislweg 14 |  Salzburg |               Piccolo und mehr |          Georg Pipps |
          Sales Manager |         null |       6562-9723 |       6562-9722 |       5020 |            NS
         NS |      VALON |                            NS |        NS |                              IT |           Valon Hoti |
                     IT |         null |              NS |              NS |         NS |            NS
         NS |       Val2 |                            NS |        NS |                              IT |                 Val2 |
                     IT |         null |              NS |              NS |         NS |            NS
    Denmark |      SIMOB |                  Vinbæltet 34 | Kobenhavn |                  Simons bistro |       Jytte Petersen |
                  Owner |         null |     31 13 35 57 |     31 12 34 56 |       1734 |            NS
    Denmark |      VAFFE |                 Smagsloget 45 |     Århus |                    Vaffeljernet |          Palle Ibsen |
    Germany |      TOMSP |                 Luisenstr. 48 |   Münster |             Toms Spezialitäten |        Karin Josephs |
      Marketing Manager |         null |     0251-035695 |     0251-031259 |      44087 |            NS
    Germany |      WANDK |              Adenauerallee 900 | Stuttgart |             Die Wandernde Kuh |          Rita Müller |
   Sales Representative |         null |     0711-035428 |     0711-020361 |      70563 |            NS
         UK |      AROUT |               120 Hanover Sq. |    London |                Around the Horn |         Thomas Hardy |
   Sales Representative |         null |  (171) 555-6750 |  (171) 555-7788 |    WA1 1DP |            NS
         UK |      BSBEV |              Fauntleroy Circus |    London |                  B's Beverages |    Victoria Ashworth |
   Sales Representative |         null |              NS |  (171) 555-1212 |    EC2 5NT |            NS
         UK |      CONSH |   Berkeley Gardens 12  Brewery |    London |           Consolidated Holdings |      Elizabeth Brown |
   Sales Representative |         null |  (171) 555-9199 |  (171) 555-2282 |    WX1 6LT |            NS
         UK |      EASTC |                 35 King George |    London |             Eastern Connection |            Ann Devon |
            Sales Agent |         null |  (171) 555-3373 |  (171) 555-0297 |    WX3 6FW |            NS
         UK |      ISLAT |        Garden House Crowther Way |     Cowes |                  Island Trading |        Helen Bennett |
      Marketing Manager |         null |              NS |  (198) 555-8888 |   PO31 7PJ | Isle of Wight
         UK |      NORTS |   South House 300 Queensbridge |    London |                     North/South |       Simon Crowther |
        Sales Associate |         null |  (171) 555-2530 |  (171) 555-7733 |    SW7 1RZ |            NS
         UK |      SEVES |               90 Wadhurst Rd. |    London |              Seven Seas Imports |            Hari Kumar |
          Sales Manager |         null |  (171) 555-5646 |  (171) 555-1717 |   OX15 4NB |            NS
     Poland |      WOLZA |               ul. Filtrowa 68 |  Warszawa |                  Wolski  Zajazd | Zbyszek Piestrzeniewicz |
                  Owner |         null |   (26) 642-7012 |   (26) 642-7012 |     01-012 |            NS
  Argentina |      CACTU |                   Cerrito 333 | Buenos Aires |     Cactus Comidas para llevar |     Patricio Simpson |
            Sales Agent |         null |    (1) 135-4892 |    (1) 135-5555 |       1010 |            NS
  Argentina |      OCEAN | Ing. Gustavo Moncada 8585 Piso 20-A | Buenos Aires |   Océano Atlántico Ltda. |       Yvonne Moncada |
            Sales Agent |         null |    (1) 135-5535 |    (1) 135-5333 |       1010 |            NS
  Argentina |      RANCH |        Av. del Libertador 900 | Buenos Aires |                 Rancho grande |      Sergio Gutiérrez |
   Sales Representative |         null |    (1) 123-5556 |    (1) 123-5555 |       1010 |            NS
      Italy |      FRANS |           Via Monte Bianco 34 |    Torino |                  Franchi S.p.A. |         Paolo Accorti |
   Sales Representative |         null |     011-4988261 |     011-4988260 |      10100 |            NS
      Italy |      MAGAA |       Via Ludovico il Moro 22 |   Bergamo |    Magazzini Alimentari Riuniti |      Giovanni Rovelli |
      Marketing Manager |         null |      035-640231 |      035-640230 |      24100 |            NS
      Italy |      REGGC |       Strada Provinciale 124 | Reggio Emilia |            Reggiani Caseifici |       Maurizio Moroni |
        Sales Associate |         null |     0522-556722 |     0522-556721 |      42100 |            NS

(93 rows)
token@cqlsh:northwind>
```

*Figure 10 Insertion statements*

**Queries:**

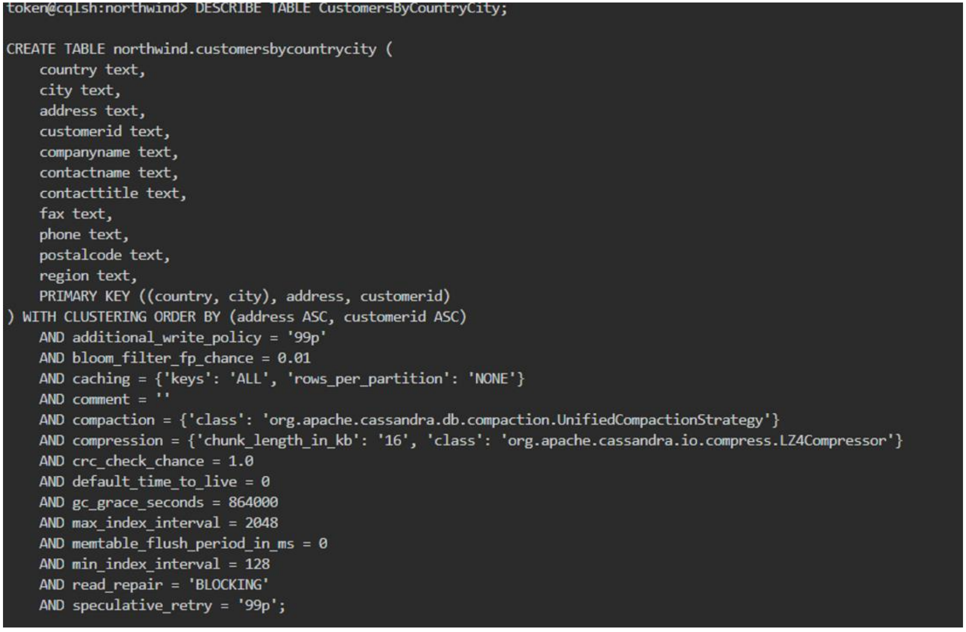
1. Provide the query and the results (screenshots and a copy of your query) that show the customers from Rio de Janeiro, Brazil ordered by their addresses.

The answer:

```
CREATE TABLE CustomersByCountryCity (
  Country TEXT,
  City TEXT,
  CustomerID TEXT,
  CompanyName TEXT,
  ContactName TEXT,
  ContactTitle TEXT,
  Address TEXT,
  Region TEXT,
  PostalCode TEXT,
  Phone TEXT,
  Fax TEXT,
  PRIMARY KEY ((Country, City), Address, CustomerID)
);
```

The query:

```
SELECT *
FROM CustomersByCountryCity
WHERE Country = 'Brazil' AND City = 'Rio de Janeiro';
```



```
token@cqlsh:northwind> DESCRIBE TABLE CustomersByCountryCity;

CREATE TABLE northwind.customersbycountrycity (
    country text,
    city text,
    address text,
    customerid text,
    companyname text,
    contactname text,
    contacttitle text,
    fax text,
    phone text,
    postalcode text,
    region text,
    PRIMARY KEY ((country, city), address, customerid)
) WITH CLUSTERING ORDER BY (address ASC, customerid ASC)
    AND additional_write_policy = '99p'
    AND bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.UnifiedCompactionStrategy'}
    AND compression = {'chunk_length_in_kb': '16', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
    AND crc_check_chance = 1.0
    AND default_time_to_live = 0
    AND gc_grace_seconds = 864000
    AND max_index_interval = 2048
    AND memtable_flush_period_in_ms = 0
    AND min_index_interval = 128
    AND read_repair = 'BLOCKING'
    AND speculative_retry = '99p';
```

*Figure 11 Table_1 creation*

```
token@cqlsh:northwind> SELECT *
          ... FROM CustomersByCountryCity
          ... WHERE Country = 'Brazil' AND City = 'Rio de Janeiro';

 country | city           | address                  | customerid | companyname        | contactname      | contacttitle          | fax           | phone         | po
stalcode | region
---------+----------------+--------------------------+------------+--------------------+------------------+-----------------------+---------------+---------------+---
---------+--------
  Brazil | Rio de Janeiro |      Av. Copacabana, 267 |      RICAR | Ricardo Adocicados |   Janete Limeira | Assistant Sales Agent |            NS | (21) 555-3412 |  0
2389-890 |     RJ
  Brazil | Rio de Janeiro | Rua da Panificadora, 12 |      QUEDE |        Que Delícia | Bernardo Batista |    Accounting Manager | (21) 555-4545 | (21) 555-4252 |  0
2389-673 |     RJ
  Brazil | Rio de Janeiro |        Rua do Paço, 67 |      HANAR |      Hanari Carnes |     Mario Pontes |    Accounting Manager | (21) 555-8765 | (21) 555-0091 |  0
5454-876 |     RJ
```

*Figure 12 first query*

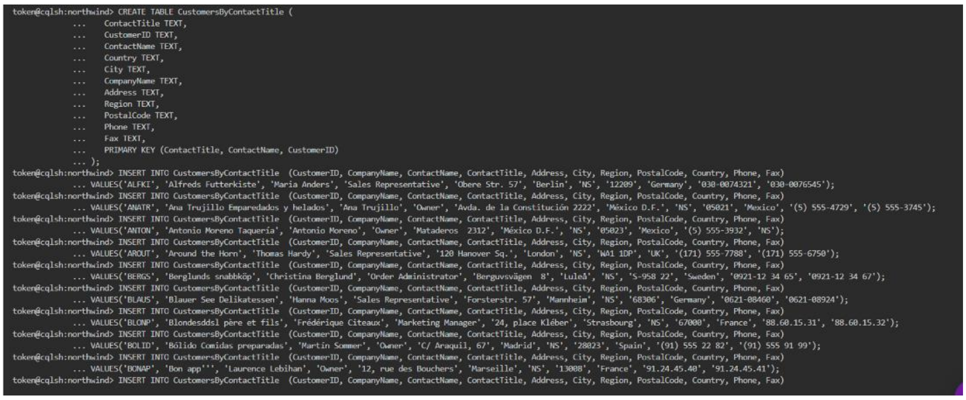
2. Provide a list of customers that are in the Sales Manager role without forcing the scan of all partitions across all databases. The result should be ordered by their names

The answer:

```
CREATE TABLE CustomersByContactTitle (
  ContactTitle TEXT,
  CustomerID TEXT,
  ContactName TEXT,
  Country TEXT,
  City TEXT,
  CompanyName TEXT,
  Address TEXT,
  Region TEXT,
  PostalCode TEXT,
  Phone TEXT,
  Fax TEXT,
  PRIMARY KEY (ContactTitle, ContactName, CustomerID)
);
```

The query:

```
SELECT * FROM CustomersByContactTitle
WHERE ContactTitle = 'Sales Manager'
ORDER BY ContactName;
```



```
token@cqlsh:northwind> CREATE TABLE CustomersByContactTitle (
           ...   ContactTitle TEXT,
           ...   CustomerID TEXT,
           ...   ContactName TEXT,
           ...   Country TEXT,
           ...   City TEXT,
           ...   CompanyName TEXT,
           ...   Address TEXT,
           ...   Region TEXT,
           ...   PostalCode TEXT,
           ...   Phone TEXT,
           ...   Fax TEXT,
           ...   PRIMARY KEY (ContactTitle, ContactName, CustomerID)
           ... );
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('ALFKI', 'Alfreds Futterkiste', 'Maria Anders', 'Sales Representative', 'Obere Str. 57', 'Berlin', 'NS', '12209', 'Germany', '030-0074321', '030-0076545');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('ANATR', 'Ana Trujillo Emparedados y helados', 'Ana Trujillo', 'Owner', 'Avda. de la Constitución 2222', 'México D.F.', 'NS', '05021', 'Mexico', '(5) 555-4729', '(5) 555-3745');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('ANTON', 'Antonio Moreno Taquería', 'Antonio Moreno', 'Owner', 'Mataderos  2312', 'México D.F.', 'NS', '05023', 'Mexico', '(5) 555-3932', 'NS');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('AROUT', 'Around the Horn', 'Thomas Hardy', 'Sales Representative', '120 Hanover Sq.', 'London', 'NS', 'WA1 1DP', 'UK', '(171) 555-7788', '(171) 555-6750');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('BERGS', 'Berglunds snabbköp', 'Christina Berglund', 'Order Administrator', 'Berguvsvägen  8', 'Luleå', 'NS', 'S-958 22', 'Sweden', '0921-12 34 65', '0921-12 34 67');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('BLAUS', 'Blauer See Delikatessen', 'Hanna Moos', 'Sales Representative', 'Forsterstr. 57', 'Mannheim', 'NS', '68306', 'Germany', '0621-08460', '0621-08924');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('BLONP', 'Blondesddsl père et fils', 'Frédérique Citeaux', 'Marketing Manager', '24, place Kléber', 'Strasbourg', 'NS', '67000', 'France', '88.60.15.31', '88.60.15.32');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('BOLID', 'Bólido Comidas preparadas', 'Martín Sommer', 'Owner', 'C/ Araquil, 67', 'Madrid', 'NS', '28023', 'Spain', '(91) 555 22 82', '(91) 555 91 99');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
           ... VALUES('BONAP', 'Bon app''', 'Laurence Lebihan', 'Owner', '12, rue des Bouchers', 'Marseille', 'NS', '13008', 'France', '91.24.45.40', '91.24.45.41');
token@cqlsh:northwind> INSERT INTO CustomersByContactTitle  (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
```

*Figure 13 Table_ creation*

```
token@cqlsh:northwind> SELECT * FROM CustomersByContactTitle
           ... WHERE ContactTitle = 'Sales Manager'
           ... ORDER BY ContactName;
```

```
 contacttitle  | contactname        | customerid | address                  | city     | companyname                    | country     | fax            | phone          | postalcode | region
---------------+--------------------+------------+--------------------------+----------+--------------------------------+-------------+----------------+----------------+------------+--------
 Sales Manager |     Annette Roulet |      LAMAI | 1 rue Alsace-Lorraine    | Toulouse |                La maison d'Asie |      France |    61.77.61.11 |    61.77.61.10 |      31000 |     NS
 Sales Manager | Art Braunschweiger |      SPLIR |            P.O. Box 555 |   Lander |            Split Rail Beer & Ale |         USA | (307) 555-6525 | (307) 555-4680 |      82520 |     WY
 Sales Manager |        Fran Wilson |      LONEP |     89 Chiaroscuro Rd. | Portland |        Lonesome Pine Restaurant |         USA | (503) 555-9646 | (503) 555-9573 |      97219 |     OR
 Sales Manager |        Georg Pipps |      PICCO |           Geislweg 14 | Salzburg |             Piccolo und mehr |     Austria |      6562-9723 |      6562-9722 |       5020 |     NS
 Sales Manager |         Hari Kumar |      SEVES |        90 Wadhurst Rd. |   London |              Seven Seas Imports |          UK | (171) 555-5646 | (171) 555-1717 |   OX15 4NB |     NS
 Sales Manager |  José Pedro Freyre |      GODOS |         C/ Romero, 33 |  Sevilla |            Godos Cocina Típica |       Spain |             NS |  (95) 555 82 82 |      41101 |     NS
 Sales Manager |     Lino Rodriguez |      FURIB | Jardim das rosas n. 32 |   Lisboa | Furia Bacalhau e Frutos do Mar |    Portugal |  (1) 354-2535 |  (1) 354-2534 |       1675 |     NS
 Sales Manager |       Michael Holz |      RICSU |     Grenzacherweg 237 |   Genève |            Richter Supermarkt | Switzerland |             NS |    0897-034214 |       1203 |     NS
 Sales Manager |        Palle Ibsen |      VAFFE |         Smagsloget 45 |    Århus |                 Vaffeljernet |     Denmark |    86 22 33 44 |    86 21 32 43 |       8200 |     NS
 Sales Manager |      Paula Parente |      WELLI |   Rua do Mercado, 12 |  Resende |         Wellington Importadora |      Brazil |             NS |  (14) 555-8122 |  08737-363 |     SP
 Sales Manager |      Roland Mendel |      ERNSH |          Kirchgasse 6 |     Graz |                  Ernst Handel |     Austria |      7675-3426 |      7675-3425 |       8010 |     NS

(11 rows)
token@cqlsh:northwind> 
```

*Figure 14 Second query*

# List of Tables