



ELG 5166 – Cloud Analytics

Assignment #1

Group – 9

Mohamed Elesawy	300327237
Eslam Khalaf	300327261
Ahmed Abdo Zaid	300327306
Kareem Waly	300327303

Important Note: Read the following academic integrity statement, type in your full name and student ID, and include a copy in your submission. Submitting this form electronically by one of the team members is considered the same as signing the document by all members of the team.

Personal Ethics & Academic Integrity Statement

Student name: Mohamed Elesawy Student ID: 300327237

Student Name: Eslam Khalaf Student ID: 300327261

Student Name: Ahmed Abdo Zaid Student ID: 300327306

Student Name: Kareem Waly Student ID: 300327303

By typing in my name and student ID on this form and submitting it electronically, I am attesting to the fact that I have reviewed not only my work but the work of my team member, in its entirety.

I attest to the fact that my work in this project adheres to the fraud policies as outlined in the Academic Regulations in the University's Graduate Studies Calendar. I further attest that I have knowledge of and have respected the "Beware of Plagiarism" brochure for the university. To the

best of my knowledge, I also believe that each of my group colleagues has also met the aforementioned requirements and regulations. I understand that if my group assignment is submitted without a completed copy of this Personal Work Statement from each group member, it will be interpreted by the school that the missing student(s) name is confirmation of non-participation of the aforementioned student(s) in the required work.

We, by typing in our names and student IDs on this form and submitting it electronically,

- warrant that the work submitted herein is our own group members' work and not the work of others
- acknowledge that we have read and understood the University Regulations on Academic Misconduct
- acknowledge that it is a breach of University Regulations to give or receive unauthorized and/or unacknowledged assistance on a graded piece of work

Note:

This assignment puts you in the position of a consultant/analyst who is using her/his knowledge of the course to address a real-world problem. There is no "unique" or "best" solution for this assignment question.

Additionally,

- Keep your answers short and succinct.
- Use a diagram if it helps demonstrate or illustrate your answer. Diagrams without appropriate content description and reference will not count as valid responses.
- Please cite all references properly and provide a bibliography or a reference section if needed.

Part 1: Questions

1. Describe briefly what a NoSQL database means. Select a NoSQL database (except MongoDB & Cassandra) and describe how this database can be used for the storage and management of big data.

A NoSQL database refers to 'not only SQL' that means it can store many types of data not only structured data but also semi structured data the availability is the most important factor for it. In relational databases, consistency and atomicity is the best important factor due to dealing with heavy inserting and updates queries then the performance of reading queries not good as needed because there are many joins that will slow down the performance, so in the big data needs massive reading queries that will not be suitable with relational databases, so NoSQL appears to solve massive reading big data problems.

Riak Database: is a particular kind of key-value store for a NoSQL database that can store any kind of data and does so over numerous nodes in what is referred to as a cluster. Data replication occurs in a cluster to support partition tolerance. This method of data replication ensures that this data is constantly accessible. Users of websites like social networking platforms or file-sharing programmers are an illustration of this. The data has been replicated among numerous nodes that extend across several servers, so it is safe in the case of a server crash, system failure, or other network anomaly. Once all nodes are successfully back online again, Riak starts to automatically allow reads and writes to and from the nodes that were not operational. In addition to extreme fault tolerance, Riak is highly scalable; it allows anyone to add new nodes with a few short commands and scale the size of the cluster to meet the data storage needs in a simple, linear, and predictable fashion, saving both time and money.

Or

AWS DynamoDB: is a fully managed key-value database offered by amazon web services, that delivers single-digit millisecond performance at any scale. you don't have to configure anything because it's a fully managed database, AWS manage scalability, configuration, replication, at rest encryption and hardware provisioning.

2. Investigate and describe one application of Big Data Analytics that was not described in

Class.

Big data analytics in organizational performance:

Large data samples can assist in identifying and leveraging business transformation and offer business-centric approaches and practices that give organizations or enterprises a competitive advantage by enabling them to enhance their current applications and daily operational decisions ad in creating new business values for the organization.

3. Briefly describe the transaction management features of Cassandra and MongoDB in the context of ACID vs. BASE properties.

Cassandra	MongoDB
<ul style="list-style-type: none"> Cassandra is a NoSQL DB that can be categorized as a BASE database because of the fundamental availability, a soft state, and eventual consistency. Cassandra is more efficient at the level of availability because it has multiple master nodes Cassandra is a soft state because of the data's temporary inconsistency and change over time, it has a commit log, all data is written to the commit log as part of this backup technique to prevent data loss, and the system will change state without user intervention due to eventual consistency. 	<ul style="list-style-type: none"> MongoDB is a NoSQL DB supports ACID compliance at the document level.[3] MongoDB supports multi-document ACID transactions on a single replica set.[3] MongoDB supports availability with only one master nodes what make it not a best choice in case of availability.[3] MongoDB isn't a soft state and eventual consistency because it's supported the consistency which mean that there will not be a lack of immediate consistency. [6]



- | | |
|--|--|
| <ul style="list-style-type: none"> • Cassandra does not immediately obligate consistency but updates data over time, and until it does, the data reads are possible even though they might not reflect reality. • It possesses atomicity, isolation, and long-term durability.[1] • It supports atomic(A), because inserting or modifying several columns in a row is considered a single write operation.[2] • Cassandra cannot be described as an ACID-compliant database because the ACID lacks consistency(C) the concepts of referential integrity, foreign keys, and joins.[2] • It supports isolation(I) Because it believes that writing to a row should be isolated to just the user who is writing and should not be visible to any other users until the current user has finished writing.[2] • It supports durability(D) since all writes to a replica node are stored in memory and committed to disc before being declared successful.[2] | <ul style="list-style-type: none"> • The operation on a single document is atomic(A), since by using embedded documents to obtain relationships between data in a single document structure rather than normalizing across several documents and collections, the need for multi-document transactions is removed. [3] • MongoDB supports consistency(C) the most recent value will be returned by any subsequent read after a write has finished.[3] • MongoDB is isolation(I) since Data changes implemented during a transaction will not be visible outside of that transaction, until each transaction has finished processing, the data from other transactions cannot be accessed when several transactions are being carried out.[3] • It supports durability(D) as it accommodates data from heterogeneous to homogeneous.[3] |
|--|--|

4. You are working on a project that requires you to capture data from millions of IoT devices in people's homes. Each IoT device uploads a JSON document with the data elements required for analytics.

a) Identify potential NoSQL databases that you can capture data from the IoT Devices

- o Document Database: such as MongoDB, it's a perfect solution for IOT app because it's power of storing data as JSON documents.[4]
- o Time-Series Database: such as Timestream, it designed to hold time-oriented data, which makes it ideal for time-series-dependent machine learning models, IoT applications, and audit logs.[4]
- o Columnar Database: such as Cassandra DB, considered as a good option because its power of ignoring the rows to directly query the columns. [4]

b) What are your design and analytics considerations and rationale behind your choice?

Since the IoT devices share huge streaming of data, a document database like MongoDB will be suitable because it does not require a schema, what make it more adaptable to changes, and it supports sharding to spread data across multiple servers to increase scalability and availability and can handle the flood of data from these devices, MongoDB scaled data either horizontally or vertically, and it natively supports time series data, which is typical of IoT architectures. In addition to the Column-oriented Database like Cassandra due to the fact that it may be used where analysis or grouping is being done on certain columns for the IoT applications, it's better than MongoDB in level of the availability because Cassandra Has multiple master nodes where MongoDB has only one master nodes, and the more master nodes there are in a cluster, the better the write speed (scalability), However, when it comes to consistency, MongoDB excels because you can query multiple nodes in a replica set and receive the same data, in contrast to Cassandra, which offers tunable consistency at the expense of performance.

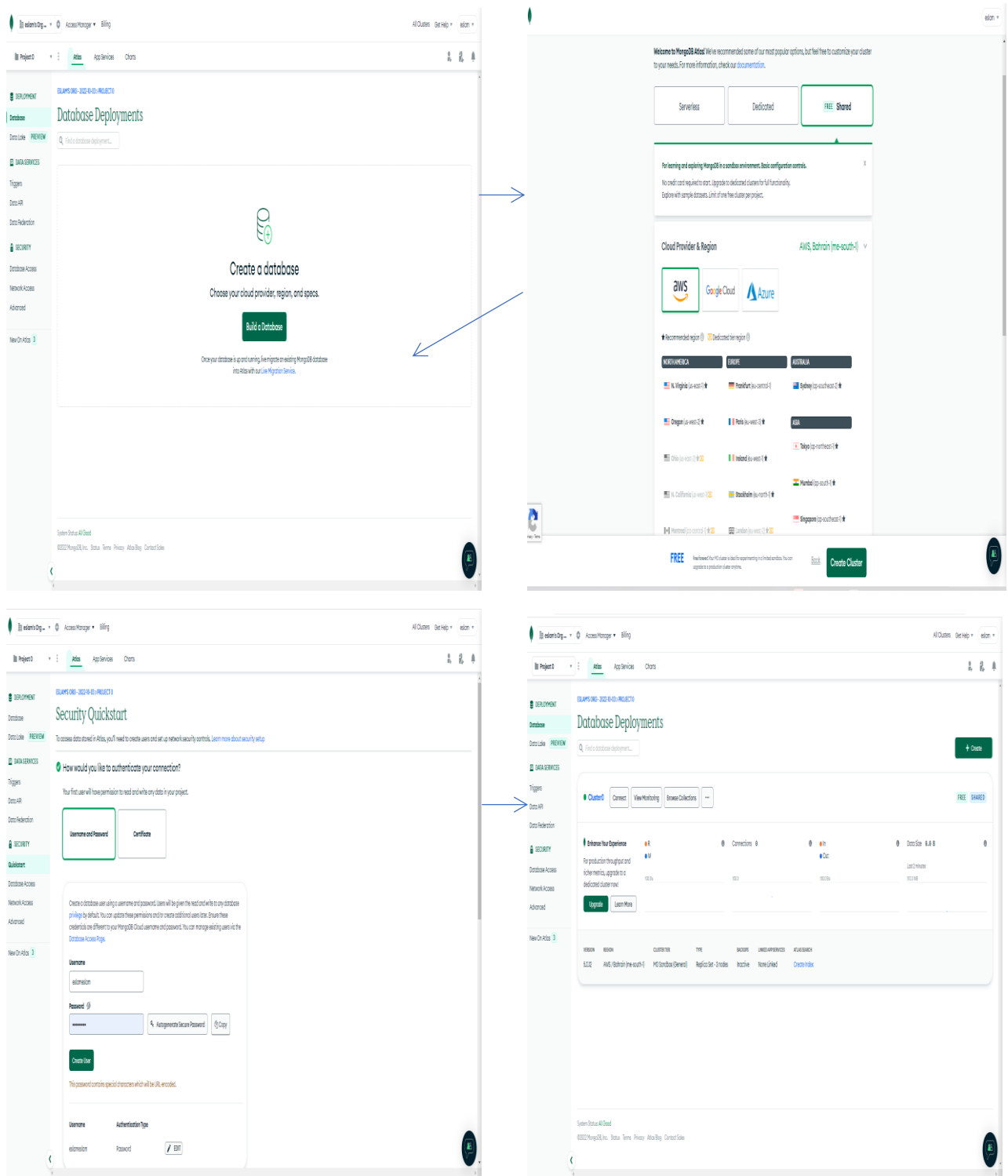
Finally, Time Series is an excellent choice too for collecting change events in the system, sensing, or storing device readings, each database of the above provides high scalability for IoT.[4] [5]

Part 2: NoSQL Labs

Setup (Please show evidence of your setup with screenshots)

1) MongoDB Lab

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



The screenshots illustrate the MongoDB Atlas setup process:

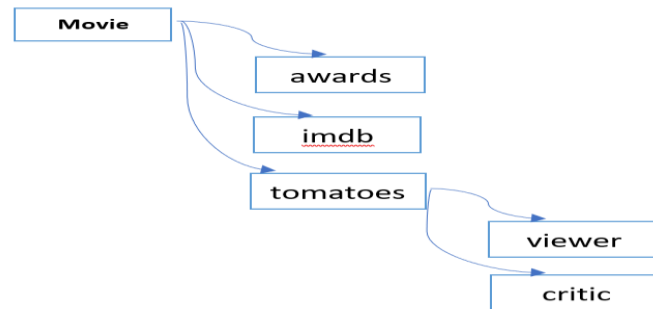
- Create a database:** The 'Build a Database' button is highlighted.
- Welcome to MongoDB Atlas:** The 'FREE Shared' option is selected under 'Serverless', 'Dedicated', and 'FREE Shared'.
- Security Quickstart:** The 'Username and Password' option is selected under 'How would you like to authenticate your connection?'. The 'Username' field is filled with 'edmondson' and the 'Password' field is filled with 'edmondson'.
- Database Deployments:** The 'Cluster' section shows a 'Cluster' with a 'Database' section.

Answers:

1. Briefly describe the movies database document model.

After loading Sample Netflix Movies Database, we noticed documents contain the following attributes, but these attributes not found in all documents but these attributes we noticed them by showing some documents in our database:

This is the data model for our document that represents that the movie document has many (awards, imdb, tomatoes) and each tomatoes has many (viewer, critic)



- “_id” with type “ObjectId” that represent the document ID of each document
- “plot” with type “String”
- “genres” with type “Array”
- “runtime” with type “Integer”
- “cast” with type “Array”
- “num_mflix_comments” with type “Integer”
- “poster” with type “String”
- “title” with type “String”
- “fullplot” with type “String”
- “languages” with type “Array”
- “released” with type “Timestamp”
- “directors” with type “Array”
- “rated” with type “String”
- “awards” with type “Object” that object contain the following (“wins” with type “Integer”, “nominations” with type “Integer” and “text” with type “String”)
- “lastupdated” with type “String”
- “year” with type “Integer”
- “imdb” with type “Object” that object contain the following (“rating” with type “Double”, “votes” with type “Integer” and “id” with type “Integer”)
- “countries” with type “Array”
- “type” with type “String”

•“tomatoes” with type “Object” that object contain the following (“viewer” with type “Object” that object contain the following (“rating” with type “Double”, “numReviews” with type “Integer” and “meter” with type “Integer”)

“critic” with type “Object” that object contain the following (“rating” with type “Double”, “numReviews” with type “Integer” and “meter” with type “Integer”)

“rotten” with type “Integer” and

“fresh” with type “Integer” and

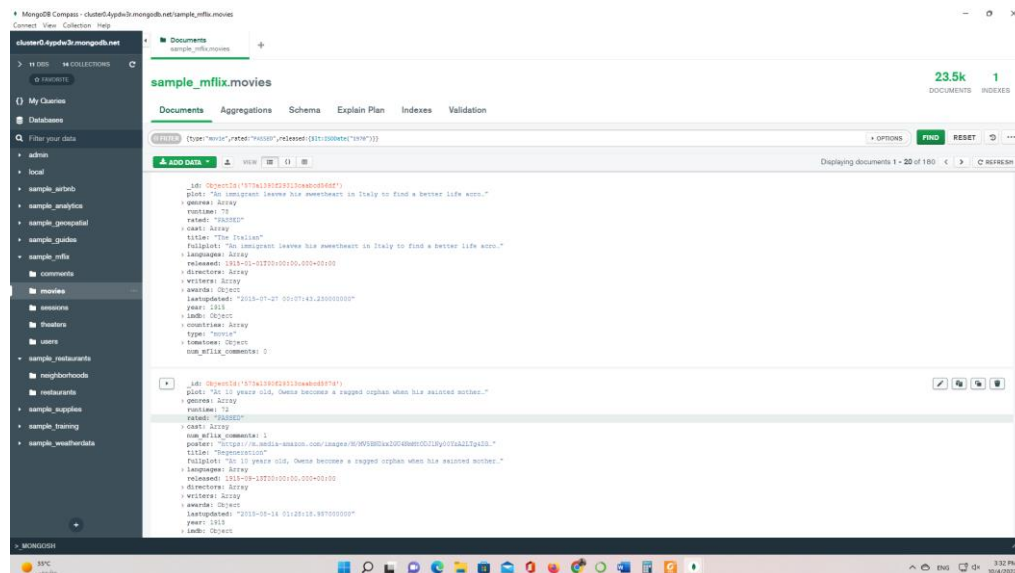
“dvd” with type “Timestamp” and

“lastUpdated” with type “Timestamp”).

All of this represents each document in Netflix Movies Database.

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

```
{type:"movie",rated:"PASSED",released:{$lt:ISODate("1970")}}
```



2. Build an Aggregation Pipeline that shows all movies (“type”) that have won at least one award and provide the results in release year aggregate count.

```
{
  $and:[
    { type:"movie"},
    {"awards.wins":{$gte:1}}
  ]
}
```

```
{_id: { $dateTrunc: { date: "$released", unit: "year" } },
Count: { "$sum": 1 }
}
```

MongoDB Compass - cluster0.1z72bx.mongodb.net/sample_mflix.movies

Connect View Collection Help

cluster0.1z72bx.mongodb.net

11 DBS 22 COLLECTIONS

HOSTS
ac-4uqug/7-shard-00-00.1...
ac-4uqug/7-shard-00-01.1...
ac-4uqug/7-shard-00-02.1...

CLUSTER
Replica Set (atlas-i5jpr-sh...)
3 Nodes

EDITION
MongoDB 5.0.13 Enterprise

My Queries

Databases

Filter your data

- local
- sample_airbnb
- sample_analytics
- sample_geospatial
- sample_guides
- sample_mflix
- comments
- movies
- sessions
- theaters
- users
- sample_restaurants
- sample_supplies
- sample_training
- sample_weatherdata

sample_mflix.movies

Documents Aggregations Schema Explain Plan Indexes Validation

Pipeline \$match \$group

Untitled - modified SAVE + CREATE NEW EXPORT TO LANGUAGE

AUTO PREVIEW

runtime: 11
> tomatoes: Object

runtime: 14
> cast: Array
> poster: "https://i.media-

runtime: 14
> rated: "UNRATED"
> cast: Array

> cast: Array
num_mflix_...
title: "A..."

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

```
1 {
2   "_id": {
3     "$dateFrom": {
4       "date": "1965-01-01T00:00:00.000+00:00",
5       "unit": "year"
6     },
7     "sum": 1
8   }
9 }
```

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

```
1 {
2   "_id": {
3     "$dateFrom": {
4       "date": "1965-01-01T00:00:00.000+00:00",
5       "unit": "year"
6     },
7     "sum": 1
8   }
9 }
```

ADD STAGE

MONGOSH

MongoDB Compass - cluster0.1z72bx.mongodb.net/sample_mflix.movies

Connect View Collection Help

cluster0.1z72bx.mongodb.net

11 DBS 22 COLLECTIONS

HOSTS
ac-4uqug/7-shard-00-00.1...
ac-4uqug/7-shard-00-01.1...
ac-4uqug/7-shard-00-02.1...

CLUSTER
Replica Set (atlas-i5jpr-sh...)
3 Nodes

EDITION
MongoDB 5.0.13 Enterprise

My Queries

Databases

Filter your data

- local
- sample_airbnb
- sample_analytics
- sample_geospatial
- sample_guides
- sample_mflix
- comments
- movies
- sessions
- theaters
- users
- sample_restaurants
- sample_supplies
- sample_training
- sample_weatherdata

sample_mflix.movies

Documents Aggregations Schema Explain Plan Indexes Validation

Pipeline \$match \$group

Showing 1 - 20 of 110

1 {
 "_id": "1913-01-01T00:00:00.000+00:00",
 "Count": 1
}

2 {
 "_id": "1921-01-01T00:00:00.000+00:00",
 "Count": 3
}

3 {
 "_id": "1965-01-01T00:00:00.000+00:00",
 "Count": 75
}

4 {
 "_id": "1942-01-01T00:00:00.000+00:00",
 "Count": 23
}

5 {
 "_id": "1972-01-01T00:00:00.000+00:00",
 "Count": 95
}

6 {
 "_id": "2008-01-01T00:00:00.000+00:00",
 "Count": 712
}

7 {
 "_id": "1939-01-01T00:00:00.000+00:00",
 "Count": 25
}

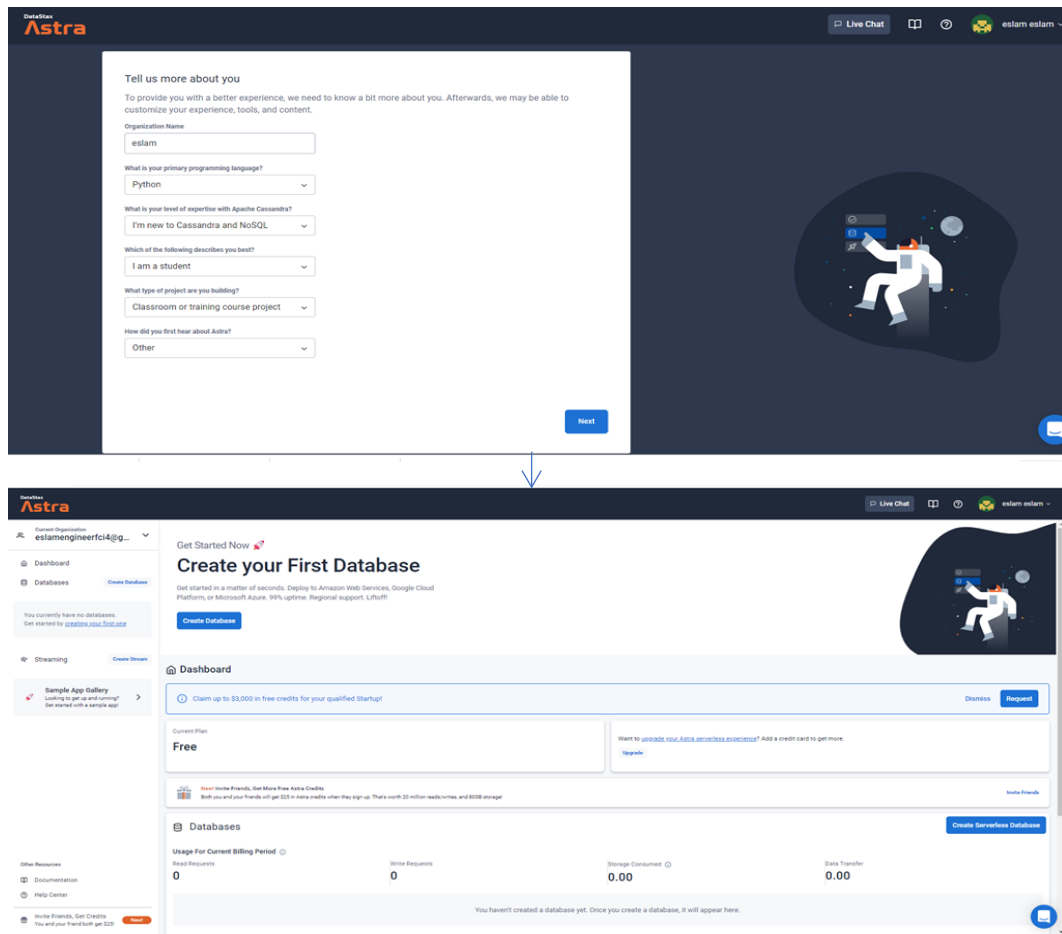
8 {
 "_id": "1983-01-01T00:00:00.000+00:00",
 "Count": 112
}

9 {
 "_id": "2010-01-01T00:00:00.000+00:00",
 "Count": 1
}

MONGOSH

2) Cassandra Lab

Using DataStax Astra Cassandra-as-a-Service (<https://astra.datastax.com/>)



Registration Form:

Tell us more about you

To provide you with a better experience, we need to know a bit more about you. Afterwards, we may be able to customize your experience, tools, and content.

Organization Name:

What is your primary programming language?:

What is your level of expertise with Apache Cassandra?:

Which of the following describes you best?:

What type of project are you building?:

How did you first hear about Astra?:

Dashboard:

Create your First Database

Get started in a matter of seconds. Deploy to Amazon Web Services, Google Cloud Platform, or Microsoft Azure. 99% uptime. Regional support. L1R24/7.

Dashboard

Claim up to \$3,000 in free credits for your qualified startup!

Current Plan: **Free**

Want to upgrade your Astra serverless experience? Add a credit card to get more.

Databases

Usage For Current Billing Period

Read Requests	Write Requests	Storage Consumed	Data Transfer
0	0	0.00	0.00

You haven't created a database yet. Once you create a database, it will appear here.

Create a Database

1 Enter the Basic Details

Database Name

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

Keyspace Name

Want to know more? [Database Docs](#) to learn more about keyspaces.

2 Select a Provider and Region

You've got access to 3 free regions in GCP. Unlock all regions by [upgrading to the Pro or the Enterprise plan](#).

Google Cloud

Amazon Web Services

Microsoft Azure

Select an Area

North America: 0 of 6 regions selected

Europe, Middle East, and Africa: 1 of 2 regions selected

Asia Pacific: 3 of 4 regions selected

South America: 0 of 1 region selected

Select a Region

10: Stockholm, Belgium

Europe-west1

West Europe (London, England, ...)

Want to use the region you're after? [Request a new region](#).

Current Plan

Free

You're currently on our free plan, which gives you free credits monthly. That recurring credit should be more than sufficient for your development needs, running sample code or apps, building proof-of-concepts, bootstrap participation - even running small production workloads.

[Learn more](#) about our free accounts.

Create Database

Database Created!

Woohoo! You just created a new database. While it's deploying, grab your token so you can access it later.

Save your secure token details

Here's your auto-generated token for this database. It's got the default permissions assigned to it. Store it somewhere safe - it's your key to accessing this database.

1 Save your auto-generated token and keep it somewhere safe. You can always [generate a new token](#), but you won't be able to access this one later.

Your Token

```
{
  "clientId": "...",
  "clientSecret": "...",
  "token": "..."
}
```

Download Token Details

eslam-token.json (1KB)

Get started with your database

New to Astra and looking for more help? Check out our getting started guide to learn how to load sample data, sample apps, and more.

Get Instructions

eslam

Your database is now active!

Database Details

Name	eslam
Keyspace	northwind
Provider	Google Cloud
Region	Europe-west1

Go To Database

No thanks, [go to the dashboard](#)

Create the customer tables (attached SQLite definition will serve as a guide) Review the questions in the queries section below and create one or more tables that partition and clusters data so these queries will execute without using Cassandra “ALLOW FILTERING” that scans all partitions.

Current Organization

eslamengineerfci4@g...

Dashboard

Databases

Streaming

Sample App Gallery

Create Database

Create Stream

Dashboard / Serverless Databases

eslam Active

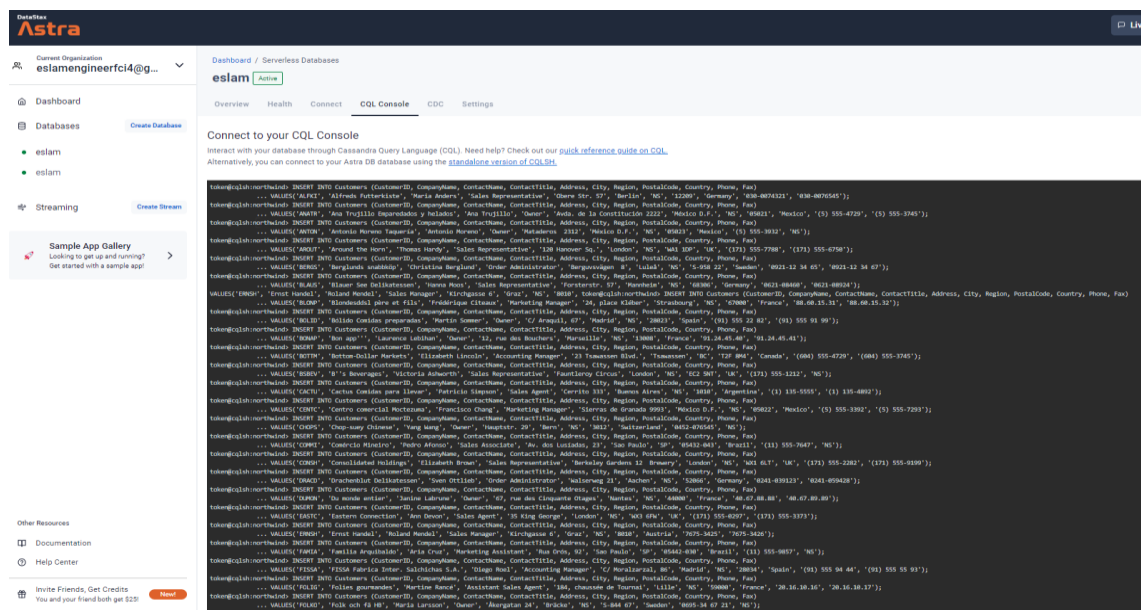
Overview Health Connect CQL Console CDC Settings

Connect to your CQL Console

Interact with your database through Cassandra Query Language (CQL). Need help? Check out our [quick reference guide on CQL](#). Alternatively, you can connect to your Astra DB database using the [standalone version of CQLSH](#).

```
Connected as eslamengineerfci4@gmail.com.
Connected to cndb at cassandra.ingress:9042.
[cqlsh 6.8.0 | Cassandra 4.0.0-6816 | CQL spec 3.4.5 | Native protocol v4]
Use HELP for help.
token@cqlsh> use "northwind";
token@cqlsh:northwind> DROP TABLE IF EXISTS Customers;
token@cqlsh:northwind> CREATE TABLE Customers
... (
... CustomerID TEXT,
... CompanyName TEXT,
... ContactName TEXT,
... ContactTitle TEXT,
... Address TEXT,
... City TEXT,
... Region TEXT,
... PostalCode TEXT,
... Country TEXT,
... Phone TEXT,
... Fax TEXT,
... PRIMARY KEY ((CustomerID,Country,City),Address)
... );
token@cqlsh:northwind> []
```

Load the attached data into your table(s) using the insert statements (minor modifications may be needed if your definitions include multiple tables). Please include screens counts after loading your data.



The screenshot shows the Astra database management interface. On the left, there's a sidebar with navigation options like Dashboard, Databases, Streaming, and Other Resources. The main area displays the CQL Console with a large block of INSERT statements for loading data into the Customers table. The statements are organized into groups, each starting with a comment like `token@cqlsh:northwind: INSERT INTO Customers (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)`. The data includes various companies from different countries, such as LINO in Venezuela, DUMON in France, HUNGO in Ireland, BOLID in Spain, PRINTI in Portugal, ANATR in Mexico, LONEP in USA, VINET in France, LETSS in USA, and QUEDE in Brazil.

Select * from Customers limit 10; (show 10 records from customers table)

```
token@cqlsh:northwind: select * from Customers limit 10;
```

customerid	country	city	address	companyname	contactname	contacttitle	fax	phone	postalcode	region
LINO	Venezuela	I. de Margarita	Ave. 5 de Mayo Porlamar	LINO-Delicatessen	Felipe Izquierdo	Owner	(8) 34-93-93	(8) 34-56-12	4988	Nueva Esparta
DUMON	France	Nantes	67, rue des Cinquante Otages	Du monde entier	Janine Labruno	Owner	40.67.89.89	40.67.88.88	44000	NS
HUNGO	Ireland	Cork	8 Johnstown Road	Hungry Owl All-Night Grocers	Patricia McKenna	Sales Associate	2967 3333	2967 542	NS	Co. Cork
BOLID	Spain	Madrid	C/ Araquil, 67	Bólidio Comidas preparadas	Martín Sommer	Owner	(91) 555 91 99	(91) 555 22 82	28023	NS
PRINTI	Portugal	Lisboa	Estrada da saúde n. 58	Princesa Isabel Vinhos	Isabel de Castro	Sales Representative	NS	(1) 356-5634	1756	NS
ANATR	Mexico	México D.F.	Avda. de la Constitución 2222	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	(5) 555-3745	(5) 555-4729	05021	NS
LONEP	USA	Portland	89 Chiaroscuro Rd.	Lonesome Pine Restaurant	Fran Wilson	Sales Manager	(503) 555-9646	(503) 555-9573	97219	OR
VINET	France	Reims	59 rue de l'Abbaye	Vins et alcools Chevalier	Paul Henriot	Accounting Manager	26.47.15.11	26.47.15.18	51100	NS
LETSS	USA	San Francisco	87 Polk St. Suite 5	Let's Stop N Shop	Jaime Yorres	Owner	NS	(415) 555-5938	94117	CA
QUEDE	Brazil	Rio de Janeiro	Rua da Panificadora, 12	Que Pasa	Bernardo Batista	Accounting Manager	(21) 555-4545	(21) 555-4522	02389-673	RJ

(18 rows)
token@cqlsh:northwind: [

Select count (*) from Customers;

```
token@cqlsh:northwind> select count(*) from Customers;
```

count
93

(1 rows)

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.


Our first table is Customers__country__city__address that represent our first query


In our first table we used (country, city) as partition keys and (address, customerid) as cluster keys because these four values will make a unique value for each record



uOttawa

Faculty of Engineering





Current Organization
eslamengineerfc14@g...

Dashboard / Serverless Databases
eslam Active

Dashboard

Databases

Streaming

Create Database

Create Stream

Sample App Gallery

Overview

Health

Connect

CQL Console

CDC

Settings

Connect to your CQL Console

Interact with your database through Cassandra Query Language (CQL). Need help? Check out our [quick reference guide on CQL](#). Alternatively, you can connect to your Astra DB database using the [standalone version of CQLSH](#).

```

token@cqlsh:northwind: DROP TABLE IF EXISTS Customers__country_city__address;
token@cqlsh:northwind: CREATE TABLE Customers__country_city__address
... (
... CustomerID TEXT,
... CompanyName TEXT,
... ContactName TEXT,
... ContactTitle TEXT,
... Address TEXT,
... City TEXT,
... Region TEXT,
... PostalCode TEXT,
... Country TEXT,
... Phone TEXT,
... Fax TEXT,
... PRIMARY KEY ((Country,City),Address,CustomerID));
... WITH CLUSTERING ORDER BY (Address DESC, CustomerID DESC);
token@cqlsh:northwind:

```

There aren't any data yet so we will insert our data to our table that will be used in first query

```

... VALUES('ARQUIT', 'Around the Horn', 'Thomas Hardy', 'Sales Representative', '128 Hanover Sq.', 'London', 'NS', 'M1 2DP', 'UK', '(171) 555-7788', '(171) 555-4798');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BERNS', 'Berglunds snabbköp', 'Christina Berglund', 'Order Administrator', 'Bergsgatan 4', 'Lund', 'NS', 'S-208 22', 'Sweden', '(0621) 12 34 67', '(0621) 12 34 67');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (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) 49408', '(0621) 49408');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BOLCOM', 'Börendesliljé päre et fölls', 'Frédérique Citeaux', 'Marketing Manager', '28, place Kléber', 'Strasbourg', 'NS', '67000', 'France', '(88) 68 15 11', '(88) 68 15 11');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BOLID', 'Bólido Condas preparadas', 'Martín Sommer', 'Owner', 'C/ Araquil, 67', 'Madrid', 'NS', '28023', 'Spain', '(91) 555 22 82', '(91) 555 91 99');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BONAP', 'Bon app''', 'Laurence Labihan', 'Owner', '12, rue des Bouchers', 'Marseille', 'NS', '13008', 'France', '(91) 34 45 48', '(91) 34 45 41');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BOTAN', 'Botan-O-Garden Markets', 'Elizabeth Lincoln', 'Accounting Manager', '123 Yawston Blvd.', 'Tampa', 'NS', '33604', 'USA', '(884) 555-4329', '(884) 555-3745');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BOVES', 'B''s Beverages', 'Victoria Ashworth', 'Sales Representative', 'Fauntleroy Circus', 'London', 'NS', 'EC2 9HT', 'UK', '(171) 555-1212', 'NS');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('CENIC', 'Centro comercial Mactozuma', 'Francisco Chang', 'Marketing Manager', 'Sierras de Granada 9993', 'México D.F.', 'NS', '05022', 'Mexico', '(5) 555-3392', '(5) 555-7293');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('CHOP', 'Chop-ney Chinese', 'Yang Wang', 'Owner', 'Hauptstr. 29', 'Bern', 'NS', '3011', 'Switzerland', '0452-876545', 'NS');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('COMPT', 'Comércio Mineiro', 'Pedro Afonso', 'Sales Associate', 'Av. dos Lusitãos, 23', 'São Paulo', 'SP', '05432-043', 'Brazil', '(11) 555-7647', 'NS');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('CONSH', 'Consolidated Holdings', 'Elizabeth Brown', 'Sales Representative', 'Berkeley Gardens 12 Brewery', 'London', 'NS', 'W3 6LT', 'UK', '(171) 555-2282', '(171) 555-9199');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('DARAC', 'Drachenhaut Delikatessen', 'Sven Ottlieb', 'Order Administrator', 'Walserweg 21', 'Aachen', 'NS', '52060', 'Germany', '(041) 430121', '(041) 430428');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('DUMK', 'Du monde entier', 'Janine Labrousse', 'Owner', '67, rue des Cinquante Otages', 'Nantes', 'NS', '44000', 'France', '(48) 67 88 88', '(48) 67 89 89');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('EASTC', 'Eastern Connection', 'Ann Devere', 'Sales Agent', '28 King George', 'London', 'NS', 'W3 0PA', 'UK', '(171) 555-8097', '(171) 555-1373');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('ERONG', 'Ernst Handel', 'Roland Mendel', 'Sales Manager', 'Kirchgasse 6', 'Graz', 'NS', '8010', 'Austria', '7675-3425', '7675-3426');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FABRIS', 'Família Arquibaldo', 'Maria Cruz', 'Marketing Assistant', 'Rua Orós, 92', 'São Paulo', 'SP', '05442-080', 'Brazil', '(11) 555-9857', 'NS');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FISSA', 'FISSA Fabrica Inter. Salchichas S.A.', 'Diego Roel', 'Accounting Manager', 'C/ Moralzarzal, 86', 'Madrid', 'NS', '28034', 'Spain', '(91) 555 94 44', '(91) 555 55 93');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FOLIG', 'Folies gourmandes', 'Martine Rance', 'Assistant Sales Agent', '184, chaussée de Tournai', 'Lille', 'NS', '59000', 'France', '(20) 16 10 16', '(20) 16 10 17');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FOLAR', 'Folk och färb', 'Maria Larsson', 'Owner', 'Margatan 44', 'Brisel', 'NS', 'S-104 67', 'Sweden', '(069) 34 67 21', 'NS');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FRANK', 'Frankenversand', 'Peter Franken', 'Marketing Manager', 'Berliner Platz 43', 'München', 'NS', '80005', 'Germany', '089-487718', '089-4877451');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FRANK', 'France restauration', 'Catherine Schmitt', 'Marketing Manager', '54, rue Royale', 'Nantes', 'NS', '44000', 'France', '(48) 32 21 21', '(48) 32 21 28');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FRANS', 'Franchi s.p.a.', 'Paolo Accorti', 'Sales Representative', 'Via Monte Bianco 34', 'Torino', 'NS', '10100', 'Italy', '011-4988368', '011-4988261');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('FURIS', 'Furia Bacalhau e Frutos do Mar', 'Lino Rodriguez', 'Sales Manager', 'Lardia das rosas n. 32', 'Lisboa', 'NS', '1675', 'Portugal', '(1) 354-2534', '(1) 354-2535');
token@cqlsh:northwind: INSERT INTO Customers__country_city__address (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)

```

Select * from Customers__country_city__address where Country = 'Brazil' and City = 'Rio de Janeiro';

```

token@cqlsh:northwind: Select * from Customers__country_city__address where Country = 'Brazil' and City = 'Rio de Janeiro';

```

country	city	address	companyname	contactname	contacttitle	customerid	fax	phone	postalcode	region
Brazil	Rio de Janeiro	Rua do Paço, 67	Hanari Carnes	Mario Pontes	Accounting Manager	HANAR	(21) 555-8765	(21) 555-0891	05454-876	RJ
Brazil	Rio de Janeiro	Rua da Panificadora, 12	Que Delícia	Bernardo Batista	Accounting Manager	QUEDE	(21) 555-4545	(21) 555-4252	02389-673	RJ
Brazil	Rio de Janeiro	Av. Copacabana, 267	Ricardo Adiciados	Janete Limeira	Assistant Sales Agent	RICAR	NS	(21) 555-3412	02389-890	RJ

(3 rows)

```

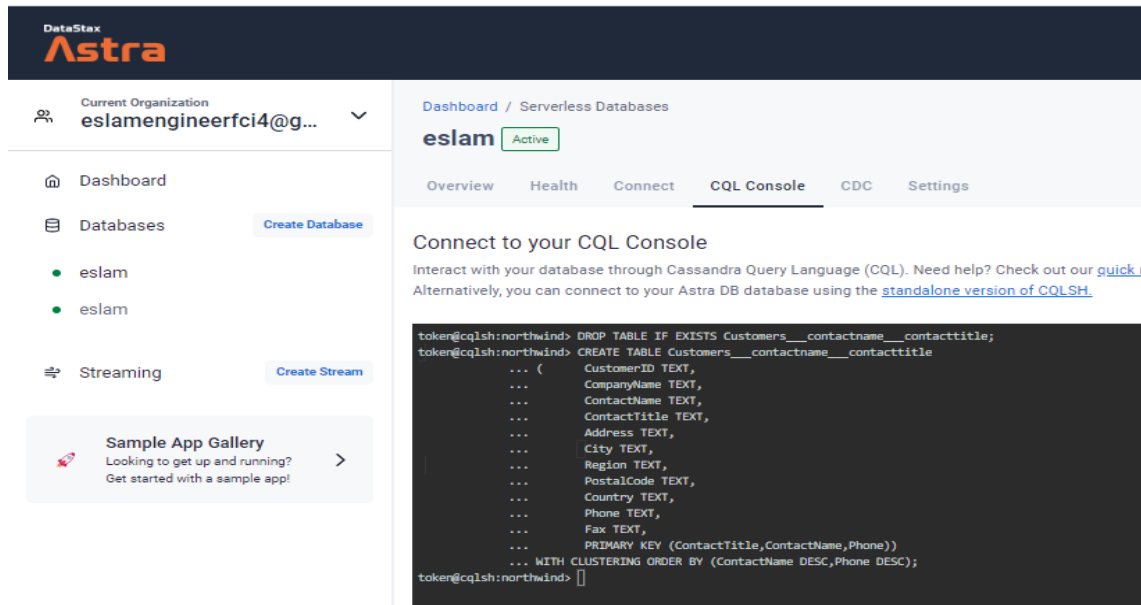
token@cqlsh:northwind:

```

- 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.

Our second table is Customers__contactname__contacttitle that represent our second query and insert data to it

In our second table we used (ContactTitle) as partition key and (ContactName, Phone) as cluster keys because these three values will make a unique value for each record



The screenshot shows the DataStax Astra dashboard for the organization 'eslamengineerfc4@g...'. The 'CQL Console' tab is active, displaying a CQL script to create a table named 'Customers__contactname__contacttitle'. The table has columns: CustomerID (TEXT), CompanyName (TEXT), ContactName (TEXT), ContactTitle (TEXT), Address (TEXT), City (TEXT), Region (TEXT), PostalCode (TEXT), Country (TEXT), Phone (TEXT), and Fax (TEXT). The primary key is defined as (ContactTitle, ContactName, Phone) and the clustering order is by (ContactName DESC, Phone DESC).

```
token@cqlsh:northwind> DROP TABLE IF EXISTS Customers__contactname__contacttitle;
token@cqlsh:northwind> CREATE TABLE Customers__contactname__contacttitle
... (
... CustomerID TEXT,
... CompanyName TEXT,
... ContactName TEXT,
... ContactTitle TEXT,
... Address TEXT,
... City TEXT,
... Region TEXT,
... PostalCode TEXT,
... Country TEXT,
... Phone TEXT,
... Fax TEXT,
... PRIMARY KEY (ContactTitle,ContactName,Phone))
... WITH CLUSTERING ORDER BY (ContactName DESC,Phone DESC);
token@cqlsh:northwind> []
```

```
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('ALFKI', 'Alfreds Futterkiste', 'Maria Anders', 'Sales Representative', 'Obere Str. 57', 'Berlin', 'DE', '12209', 'Germany', '030-8074311', '030-8076545');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('ANAT', 'Ana Trujillo Emparedados y helados', 'Ana Trujillo', 'Owner', 'Avda. de la Constitución 2222', 'Mexico D.F.', 'MX', '06921', 'Mexico', '(5) 555-4729', '(5) 555-3745');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('ANTON', 'Antonio Moreno Taquería', 'Antonio Moreno', 'Owner', 'Metadros 2312', 'Mexico D.F.', 'MX', '06923', 'Mexico', '(5) 555-9932', 'N/A');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('ARNDT', 'Arndt's Old German Sausages', 'Thomas Hardy', 'Sales Representative', '1208 Hansen St.', 'London', 'UK', 'W4 3DP', 'UK', '(171) 555-7083', '(171) 555-4796');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('BERGS', 'Berglunds snabbköp', 'Christina Berglund', 'Order Administrator', 'Bergsgatan 8', 'Luleå', 'SE', 'S-981 22', 'Sweden', '0921-12 34 65', '0921-12 34 67');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('BLAUS', 'Blauer See Delikatessen', 'Hanna Koch', 'Sales Representative', 'Poststr. 57', 'Münster', 'DE', '48186', 'Germany', '0251-89466', '0251-89464');
... VALUES ('BOLID', 'Blondiesdeli pere et fils', 'Frédérique Citeaux', 'Marketing Manager', '24, place Kléber', 'Strasbourg', 'FR', '67000', 'France', '88.68.15.31', '88.68.15.32');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('BONAP', 'Bon app'', 'Laurence Labarre', 'Owner', '12, rue des Bouchers', 'Marseille', 'FR', '13008', 'France', '91.24.45.48', '91.24.45.41');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('BOTTI', 'Botticelli Winery', 'Elizabeth Lincoln', 'Accounting Manager', '23 Thomas Blvd.', 'Tampa', 'FL', '33604', 'USA', '(804) 555-4702', '(804) 555-3745');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('BSBEV', 'B's Beverages', 'Victoria Ashworth', 'Sales Representative', 'Fountain City', 'London', 'UK', 'EC2 9HT', 'UK', '(171) 555-1212', 'N/A');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('CACTI', 'Cactus Gansas para Llevar', 'Patricia Simpson', 'Sales Agent', 'Corrillo 333', 'Buenos Aires', 'AR', '1018', 'Argentina', '(1) 135-5555', '(1) 135-4892');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('CENT', 'Centro comercial Mochamas', 'Francisco Chang', 'Marketing Manager', 'Sierritas de Granada 9993', 'Mexico D.F.', 'MX', '06922', 'Mexico', '(5) 555-3392', '(5) 555-7293');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('COPR', 'Copa-sury Olenese', 'Yong Wang', 'Owner', 'Hauptstr. 29', 'Bern', 'CH', '3011', 'Switzerland', '0432-676545', 'N/A');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('COMT', 'Comércio Mineiro', 'Pedro Afonso', 'Sales Associate', 'Av. dos Lusitâneos, 25', 'São Paulo', 'SP', '05412-043', 'Brazil', '(11) 555-7647', 'N/A');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('CONSO', 'Consolidated Holdings', 'Elizabeth Brown', 'Sales Representative', 'Barley Gardens 12 Brewery', 'London', 'UK', 'M3 6L1', 'UK', '(171) 555-2322', '(171) 555-9199');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('DRAO', 'Drachenhit Delikatessen', 'Sven Ottlieb', 'Order Administrator', 'Walserweg 21', 'Aachen', 'DE', '52066', 'Germany', '0241-891213', '0241-894234');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('DUMK', 'Du monde entier', 'Jérôme Labruno', 'Owner', '67, rue des Cinquante Otages', 'Nantes', 'FR', '44000', 'France', '40.67.88.88', '40.67.89.89');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('EAST', 'Eastern Connection', 'Ann Devon', 'Sales Agent', '35 King George', 'London', 'UK', 'W3 6PA', 'UK', '(171) 555-4229', '(171) 555-1373');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('ERNS', 'Ernst Handel', 'Roland Mendel', 'Sales Manager', 'Kirchgasse 6', 'Graz', 'AT', '8010', 'Austria', '7675-3425', '7675-3428');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('FAPA', 'Famila Arquibald', 'Aria Cruz', 'Marketing Assistant', 'Rua do, 92', 'São Paulo', 'SP', '05442-080', 'Brazil', '(11) 555-8837', 'N/A');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('FISAT', 'FISAT Fabric Inter. Salsichas S.A.', 'Ulrich Roel', 'Accounting Manager', 'C/ Moralzarzal, 86', 'Madrid', 'ES', '28034', 'Spain', '(91) 555 94 44', '(91) 555 55 93');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('FOLIO', 'Folies gourmandes', 'Martine Rance', 'Assistant Sales Agent', '184, chaussée de Tournai', 'Lille', 'FR', '59600', 'France', '20.16.18.15', '20.16.18.17');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES ('FOLIO', 'Folk och fä HB', 'Maria Larsson', 'Owner', 'Årsgatan 24', 'Bräcke', 'SE', 'S-844 67', 'Sweden', '0805-34 67 21', 'N/A');
token@cqlsh:northwind> INSERT INTO Customers__contactname__contacttitle (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
```



select * from Customers___contactname___contacttitle where ContactTitle ='Sales Manager';

The screenshot shows the Astra database management interface. On the left, there's a sidebar with 'Databases' and 'Streaming' sections. The main area displays a 'CQL Console' with a query: `select * from Customers___contactname___contacttitle where ContactTitle ='Sales Manager';`. Below the query, the results are shown in a table format with columns: contacttitle, contactname, address, city, companyname, country, customerid, fax, phone, postalcode, and region. The results list various sales managers from different countries like Austria, Brazil, Denmark, etc.

contacttitle	contactname	address	city	companyname	country	customerid	fax	phone	postalcode	region
Sales Manager	Roland Penzel	Kirchgasse 6	Graz	Erent Handel	Austria	EM006	7075-3420	7075-3425	8018	NS
Sales Manager	Paula Pereira	Rua do Paracatu, 52	Reusende	Wellington Importadora	Brazil	BR112	90	(141) 555-8122	08737-363	SP
Sales Manager	Palle Skovm	Søagbladet 45	Århus	Vaffeljernet	Denmark	DA998	86 22 33 44	86 21 32 43	8208	NS
Sales Manager	Michael Holz	Grossschmiede 207	Gießen	Richter Supermarkt	Germany	DE100	90	0997-814244	52043	NS
Sales Manager	Lino Rodriguez	Sardes das ruas n. 52	Lisboa	Faria Bacalho e Frutos do Mar	Portugal	PT018	(1) 354-2135	(1) 354-2134	1475	NS
Sales Manager	José Pedro Freyre	C/ Romero, 33	Sevilla	Godas Cocina Típica	Spain	ES005	90	(95) 555 82 82	41011	NS
Sales Manager	Hart Kauer	90 Bedford Sq., 1	London	Seven Seas Imports	UK	UK001	(171) 555-5666	(171) 555-1217	GA15 408	NS
Sales Manager	Georg Pipps	Geisweg 14	Salzburg	Piccola und mehr	Austria	AT000	6962-9723	6962-9722	5028	NS
Sales Manager	Fred Wilson	80 Oldenhouse Rd.	Purcell	Longhorn Fine Restaurant	USA	LO00P	(987) 555-5666	(987) 555-9573	97219	OR
Sales Manager	Art Braunschweiler	P.O. Box 555	Lander	Split Rail Beer & Ale	USA	SP11R	(907) 555-4525	(907) 555-4568	82526	WV
Sales Manager	Annette Roulet	1 rue Alsace-Lorraine	Toulouse	La maison d'Ale	France	FR001	61-77-61.11	61-77-61.16	31008	NS

References

- [1] <https://www.yugabyte.com/blog/apache-cassandra-lightweight-transactions-secondary-indexes-tunable-consistency/>
- [2] <https://docs.datastax.com/en/cassandraoss/2.2/cassandra/dml/dmlTransactionsDiffer.html>
- [3] <https://studio3t.com/whats-new/mongodb-acid-properties/>
- [4] <https://www.verypossible.com/insights/nosql-for-iot-development-how-to-choose-the-right-database>
- [5] <https://www.openlogic.com/blog/cassandra-vs-mongodb>
- [6] <https://www.geeksforgeeks.org/acid-model-vs-base-model-for-database/>