



University of Ottawa

Faculty of Engineering

School of Electrical Engineering and Computer Science

ELG 5166 – Cloud Analytics

Assignment #1

Personal Ethics & Academic Integrity Statement

Student name: Esraa Ahmed Abdelhakam Abo wadaa

Student ID: 300327225

Student Name: Abdallah Mohamed Mahmoud Mohamed Ragab

Student ID: 300327288

Student Name: Hosam Mahmoud Ibrahim Mahmoud

Student ID: 300327269

Student Name: Sondos Mohammed Hussein Ali

Student ID: 300327219

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 the nonparticipation 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

A. Part 1

I. Describe briefly what a NoSQL database means.

NoSQL (nonrelational database) is a database design approach that allows for data storage and querying outside of the traditional structures found in relational databases. It accesses and manages data using a variety of data models. These databases are designed specifically for applications that require high data volumes, low latency, and flexible data models, which are accomplished by relaxing some of the data consistency constraints of other databases.

NoSQL databases offer a number of data models, including key-value, document, and graph, that are optimized for performance and scalability.

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

HBase:

HBase is a distributed database, capable of hosting very large tables because it is layered on commodity hardware Hadoop clusters, it stores the data in HDFS-indexed files. It is highly configurable, allowing for a great deal of flexibility in dealing with massive amounts of data. Because HBase is a columnar database, all data is stored in tables with rows and columns, much like relational database management systems (RDBMSs), It stores the data in cells in decreasing order (using the timestamp) so that a read always returns the most recent values first.

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

Transportation:

Applications of Big Data in the Transportation Industry:

Big Data is used by governments for a variety of purposes, including traffic control, route planning, intelligent transportation systems, and congestion management by predicting traffic conditions. It also can be used for a variety of purposes, such as route planning to save on fuel and time, as well as for travel arrangements for tourism.

Challenges:

1. Location-based social network data and high-speed telecommunications data have an impact on travel behavior
- 2- Transport demand models continue to be built on the poorly understood new social media structure.

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

• Cassandra

- Cassandra does not provide ACID transactions but can be tuned to support “AID” transactions among “ACID” because it does not support foreign keys which means that data will be eventually consistent.
- But Cassandra let the users control their consistency Level by letting them decide whether some requests would be completed on only one node, or waiting until all nodes respond.
- and also, Cassandra supports BASE transactions.

• MongoDB

- MongoDB has always provided transactional guarantees on single-document operations, and nowadays with the release of MongoDB 4.0, it's now supporting Multi-document ACID transactions with snapshot isolation.
- and also, MongoDB supports BASE transactions.

- I. **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.**

- 1. Identify potential NoSQL databases that you can to capture data from the IoT devices**

The NoSQL database selected is MongoDB.

- 2. What are your design and analytics considerations and rationale behind your choice?**


We selected MongoDB for the following reasons:

- There is no need to define schemas first so it can receive data from different IoT devices with different data structures with no need to define a schema.
- MongoDB has the ability to scale out fast and inexpensive as we collect data from millions of IoT devices it will be there need to scale out.
- MongoDB provides the ability to perform analysis in real-time without affecting the availability of the database
- Also MongoDB provides replication for the data so it doesn't have a single point of failure.


B. MongoDB Lab


I. Create an Account

First, we have created an account via <https://account.mongodb.com/account/register>.



Create your account

 Google

 GitHub

or

Email Address

We recommend using your work email

araga093@uottawa.ca ✓

First Name

Abdallah ✓

Last Name

araga093@uottawa.ca ✓

Password

..... ✓

Your password must:

- Contain at least **8 characters**
- Contain unique characters, numbers, or symbols
- Not contain your email address

Company Name

Optional

- After verification of the email we created the cluster.

Deploy a cloud database

Experience the best of MongoDB on AWS, Azure, and Google Cloud. Choose a deployment option to get started.

NEW

Serverless

For application development and testing, or workloads with variable traffic. Minimal configuration required.

- ✓ Pay only for the operations you run
- ✓ Resources scale seamlessly to meet your workload
- ✓ Always-on security and backups

Create

Starting at
\$0.10/1M reads

ADVANCED

Dedicated

For production applications with sophisticated workload requirements. Advanced configuration controls.

- ✓ Network isolation and fine-grained access controls
- ✓ On-demand performance advice
- ✓ Multi-region and multi-cloud options available

Create

Starting at
\$0.08/hr*

*estimated cost \$56.94/month

FREE

Shared

For learning and exploring MongoDB in a cloud environment. Basic configuration options.

- ✓ No credit card required to start
- ✓ Explore with sample datasets
- ✓ Upgrade to dedicated clusters for full functionality

Create




Starting at
FREE

[I'll do this later](#) [Advanced Configuration Options](#)

- We selected AWS as the cloud provider and Bahrain region as shown below.

Cloud Provider & Region

AWS, Bahrain (me-south-1) ▼



★ Recommended region ⓘ 🏷️ Dedicated tier region ⓘ

NORTH AMERICA	EUROPE	AUSTRALIA
N. Virginia (us-east-1) ★	Frankfurt (eu-central-1)	Sydney (ap-southeast-2) ★
Oregon (us-west-2) ★	Ireland (eu-west-1) ★	ASIA
Ohio (us-east-2) ★ ⓘ	Paris (eu-west-3) ★	Tokyo (ap-northeast-1) ★
N. California (us-west-1) ⓘ	Stockholm (eu-north-1) ★	Singapore (ap-southeast-1) ★
Montreal (ca-central-1) ★ ⓘ	London (eu-west-2) ★ ⓘ	Hong Kong (ap-east-1) ★
SOUTH AMERICA	Milan (eu-south-1) ★ ⓘ	Seoul (ap-northeast-2) ★
Sao Paulo (sa-east-1) ★	MIDDLE EAST	Mumbai (ap-south-1) ★
	Bahrain (me-south-1) ★	Jakarta (ap-southeast-3) ★ ⓘ
	AFRICA	Osaka (ap-northeast-3) ★ ⓘ
	Cape Town (af-south-1) ★	

FREE Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.


[Back](#) [Create Cluster](#)

- Create a database username and password.


Create a database user using a username and password. Users will be given the *read and write to any database* [privilege](#) by default. You can update these permissions and/or create additional users later. Ensure these credentials are different to your MongoDB Cloud username and password.

Username

Abdallah

Password 

.....

 Autogenerate Secure Password

 Copy

Create User

- Edit database network access to allow access from anywhere.

Add IP Access List Entry

Atlas only allows client connections to a cluster from entries in the project's IP Access List. Each entry should either be a single IP address or a CIDR-notated range of addresses. [Learn more](#).

ADD CURRENT IP ADDRESS

ALLOW ACCESS FROM ANYWHERE

Access List Entry:

0.0.0.0/0

Comment:

Optional comment describing this entry



This entry is temporary and will be deleted in

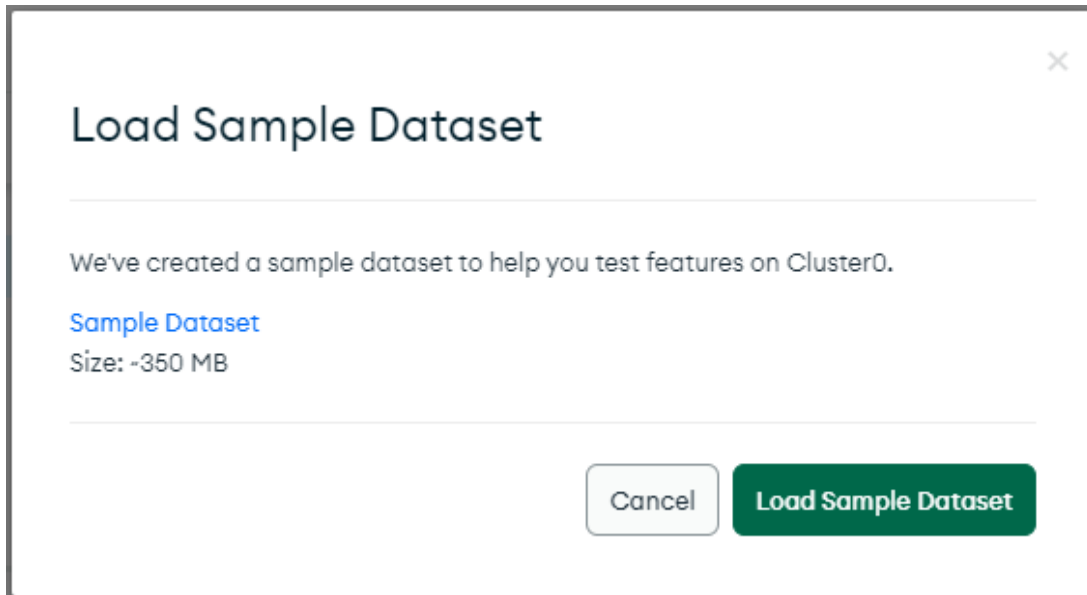
6 hours



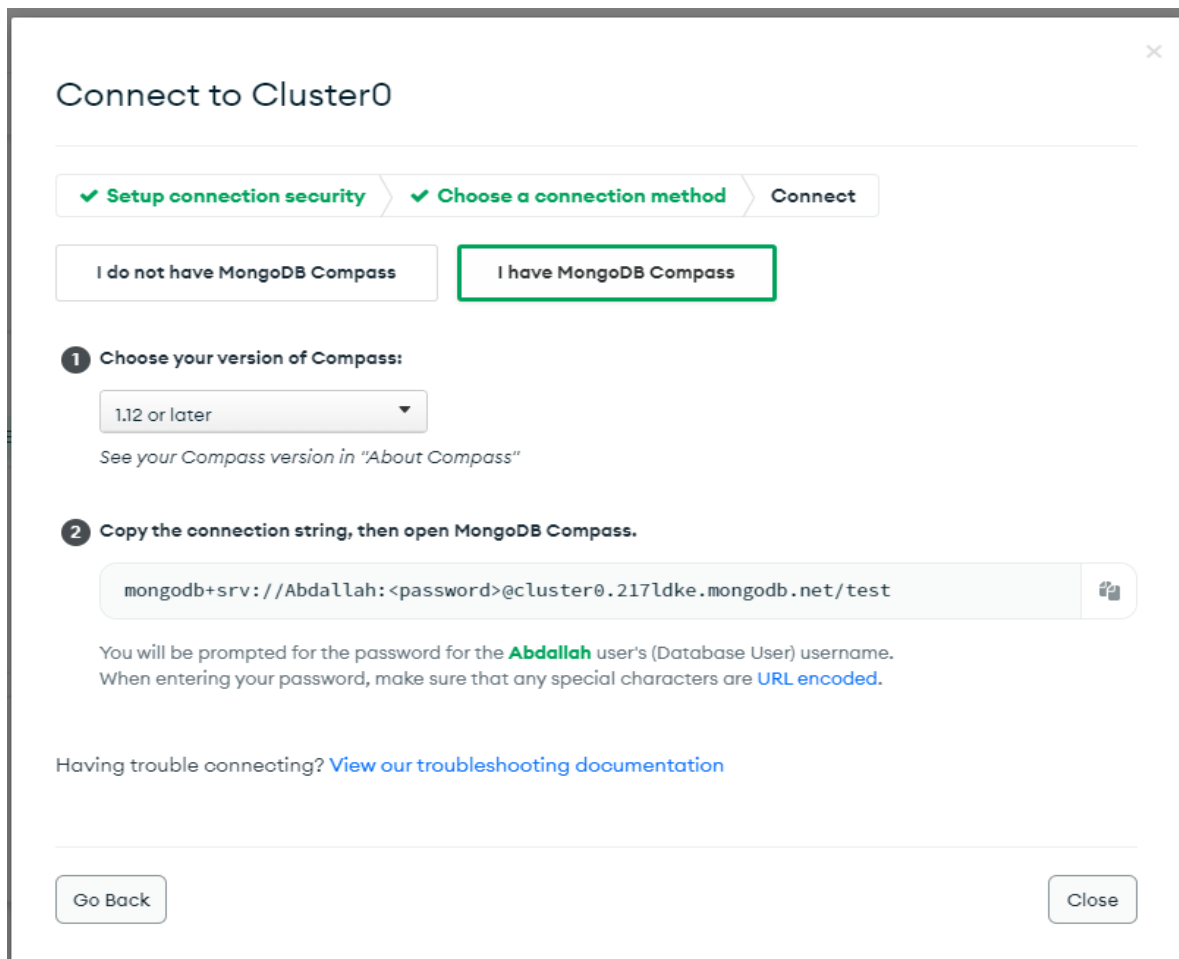
Cancel

Confirm

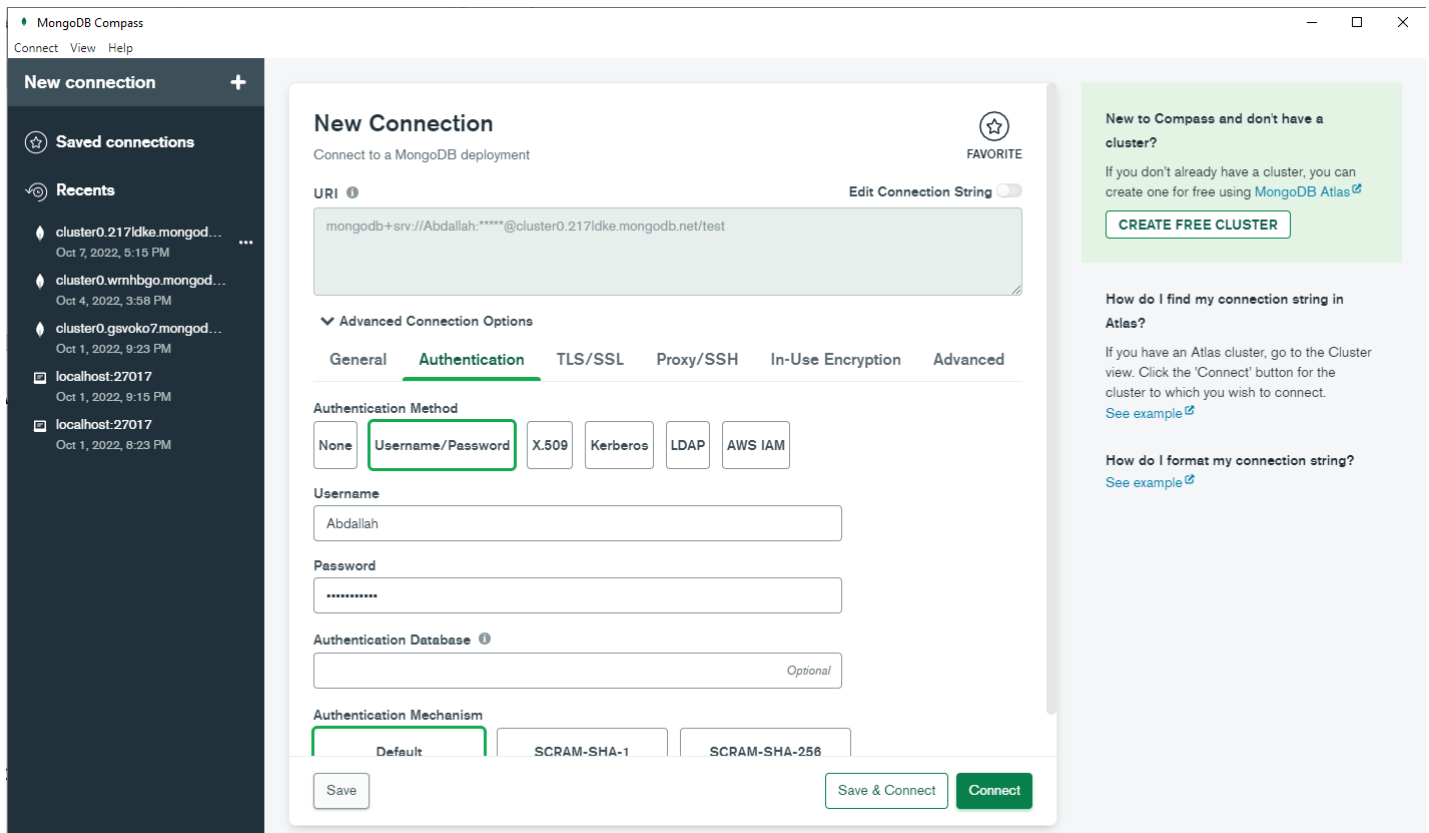
- After creating the cluster successfully, we loaded the sample database Mflix.



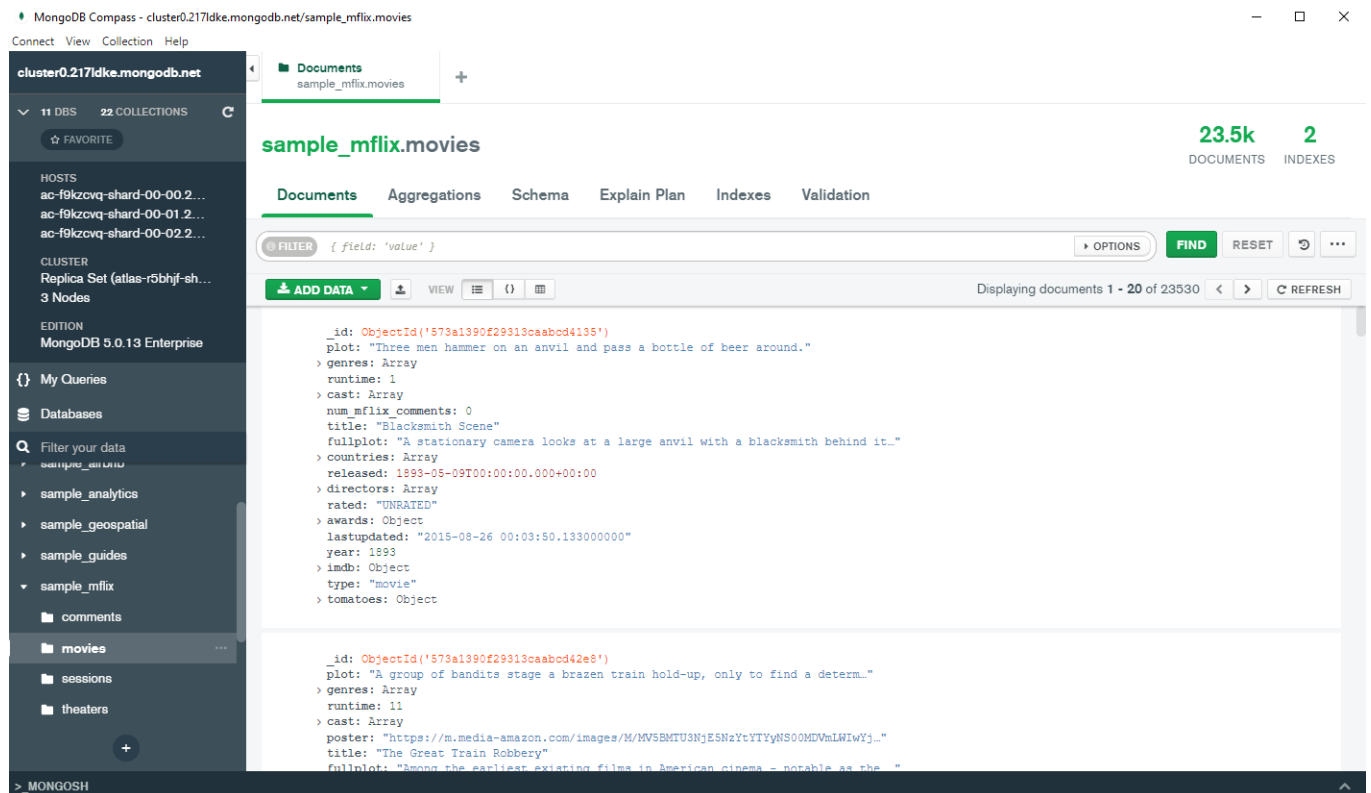
- Copy our database connection string to connect to it from MongoDB Compass.



- After that we downloaded and setup the MongoDB Compass program and then connect it to our database.



- Select the sample_mflix.movie to write our query



II. Briefly describe the movies database document model

Database attributes:

- **_id**: unique field for each file type objectId
- **Awards**: document type consist of 3 fields:
 - **nominations**: type int32
 - **text**: type string
 - **wins**: type int32
- **cast**: type array
- **countries**: type array
- **directors**: type array
- **fullplot**: type string
- **genres**: type array
- **imdb**: document type consist of 3 fields:
 - **id**: type int32
 - **rating**: type double
 - **votes**: type int32
- **languages**: type array
- **metacritic**: type int32
- **num_mflix_comments**: type int32
- **plot**: type string
- **poster**: type string
- **rated**: type string
- **released**: type string
- **runtime**: type string
- **title**: type string
- **tomatoes**: document type consist of 2 fields:
 - **boxOffice**: type string
 - **consensus**: type string
- **type**: type string
- **writers**: type array
- **year**: type int32

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

- Query:

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

The number of results: 180

The screenshot shows a MongoDB query interface. At the top, there's a FILTER section with the query: `{ $and : [{ type: "movie" }, { released : { $lt : ISODate('1970') } }, { rated: "PASSED" }] }`. Below it are PROJECT, SORT, and COLLATION sections. The PROJECT section has `{ field: 0 }`. The SORT section has `{ field: -1 } or [['field', -1]]`. The COLLATION section has `{ locale: 'simple' }`. There are buttons for SKIP (0) and LIMIT (0). On the right, there are buttons for FIND, RESET, and a refresh icon. Below the query fields, there's a toolbar with ADD DATA, VIEW, and other icons. The main area displays two documents. The first document is for a movie titled "The Italian" with a runtime of 78 minutes, released on 1915-01-01, and rated "PASSED". The second document is for a movie titled "Regeneration" with a runtime of 72 minutes, released on 1915-09-13, and rated "PASSED". Both documents have a plot, genres, cast, and other metadata. The interface also shows "Displaying documents 1 - 20 of 180" and a REFRESH button.

```
{
  "_id": ObjectId("573a1390f29313caabed56df"),
  "plot": "An immigrant leaves his sweetheart in Italy to find a better life acro...",
  "genres": Array,
  "runtime": 78,
  "rated": "PASSED",
  "cast": Array,
  "title": "The Italian",
  "fullplot": "An immigrant leaves his sweetheart in Italy to find a better life acro...",
  "languages": Array,
  "released": 1915-01-01T00:00:00.000+00:00,
  "directors": Array,
  "writers": Array,
  "awards": Object,
  "lastupdated": "2015-07-27 00:07:43.2300000000",
  "year": 1915,
  "imdb": Object,
  "countries": Array,
  "type": "movie",
  "tomatoes": Object,
  "num_mflix_comments": 0
}
```

```
{
  "_id": ObjectId("573a1390f29313caabed587d"),
  "plot": "At 10 years old, Owens becomes a ragged orphan when his sainted mother...",
  "genres": Array,
  "runtime": 72,
  "rated": "PASSED",
  "cast": Array,
  "num_mflix_comments": 1,
  "poster": "https://m.media-amazon.com/images/M/MV5BNDkxZGU4NmMtODJlNy00YzA2LTg4ZG...",
  "title": "Regeneration",
  "fullplot": "At 10 years old, Owens becomes a ragged orphan when his sainted mother...",
  "languages": Array,
  "released": 1915-09-13T00:00:00.000+00:00,
  "directors": Array
}
```

IV. 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.

- First stage
\$match

```
{
  type:"movie"
}
```

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

Document 1	Document 2	Document 3	Document 4
{ "_id": "573a1390f29313caabod4135", "plot": "Three men hammer on an anvil and pass a bottle of beer around.", "genres": "Array", "runtime": 1, "cast": "Array", "num_mflix_comments": 0, "title": "Blacksmith Scene" }	{ "_id": "573a1390f29313caabod42e8", "plot": "A group of bandits stage a brazen train hold-up, only to find a determ...", "genres": "Array", "runtime": 11, "cast": "Array", "poster": "https://m.media-" }	{ "_id": "573a1390f29313caabod4323", "plot": "A young boy, oppressed by his mother, goes on an outing in the country ..." }	{ "_id": "573a1390f29313caabod4329", "plot": "A young boy, oppressed by his mother, goes on an outing in the country ..." }

- Second stage
\$match

```
{
  'awards.wins' : { $gte : 1 }
}
```

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

Document 1	Document 2	Document 3	Document 4
{ "_id": "573a1390f29313caabod4135", "plot": "Three men hammer on an anvil and pass a bottle of beer around.", "genres": "Array", "runtime": 1, "cast": "Array", "num_mflix_comments": 0, "title": "Blacksmith Scene" }	{ "_id": "573a1390f29313caabod42e8", "plot": "A group of bandits stage a brazen train hold-up, only to find a determ...", "genres": "Array", "runtime": 11, "cast": "Array", "poster": "https://m.media-" }	{ "_id": "573a1390f29313caabod4323", "plot": "A young boy, oppressed by his mother, goes on an outing in the country ..." }	{ "_id": "573a1390f29313caabod4329", "plot": "A young boy, oppressed by his mother, goes on an outing in the country ..." }

- Third stage
\$group

```
{
  _id: "$year",
  Count: { $count: {} }
}
```

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

Document 1	Document 2	Document 3	Document 4
{ "_id": "1925", "Count": 11 }	{ "_id": "1990", "Count": 169 }	{ "_id": "1928", "Count": 15 }	{ "_id": "1922", "Count": 7 }

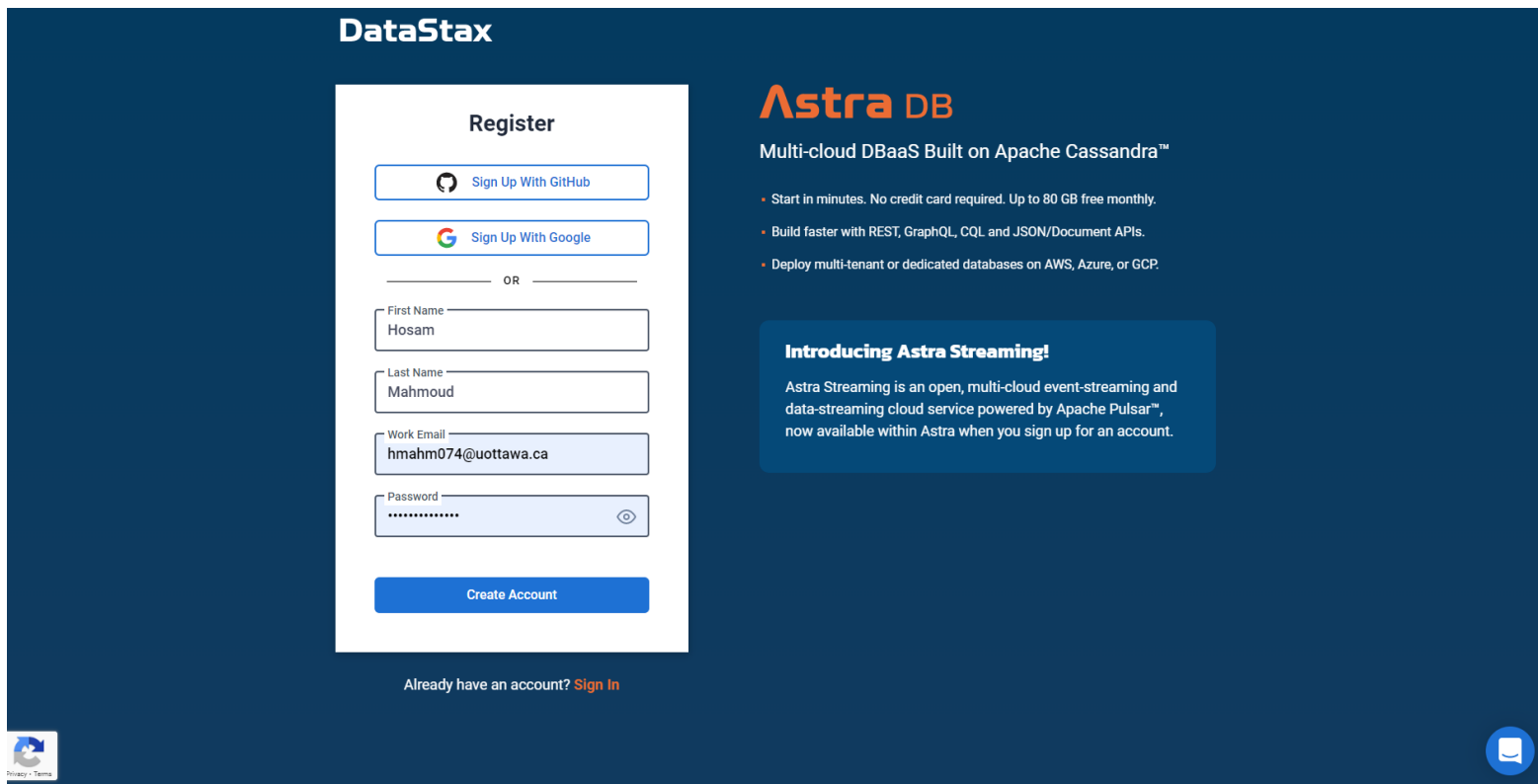
- After Running the aggregation pipeline

Showing 1 – 20 of 112		VIEW	
_id: 1981	Count: 137		
_id: 1977	Count: 92		
_id: 1971	Count: 99		
_id: 1997	Count: 360		
_id: 1928	Count: 15		
_id: 1922	Count: 7		
_id: 1925	Count: 11		
_id: 1948	Count: 40		
_id: 1967			

II. Cassandra Lab

i. Setup

- First, we have created an account on <https://astra.datastax.com/>.



The screenshot shows the Astra DB registration page. On the left, there is a 'Register' form with fields for First Name (Hosam), Last Name (Mahmoud), Work Email (hmahm074@uottawa.ca), and Password. There are also buttons for 'Sign Up With GitHub' and 'Sign Up With Google'. Below the form is a 'Create Account' button. On the right, there is a section for 'Astra DB' with the tagline 'Multi-cloud DBaaS Built on Apache Cassandra™'. It lists three bullet points: 'Start in minutes. No credit card required. Up to 80 GB free monthly.', 'Build faster with REST, GraphQL, CQL and JSON/Document APIs.', and 'Deploy multi-tenant or dedicated databases on AWS, Azure, or GCP.' Below this is a blue box titled 'Introducing Astra Streaming!' with text about Astra Streaming being an open, multi-cloud event-streaming and data-streaming cloud service powered by Apache Pulsar™, now available within Astra when you sign up for an account. At the bottom, there is a link 'Already have an account? Sign In'.

DataStax

Register

Sign Up With GitHub

Sign Up With Google

OR

First Name
Hosam

Last Name
Mahmoud

Work Email
hmahm074@uottawa.ca

Password

Create Account

Already have an account? [Sign In](#)

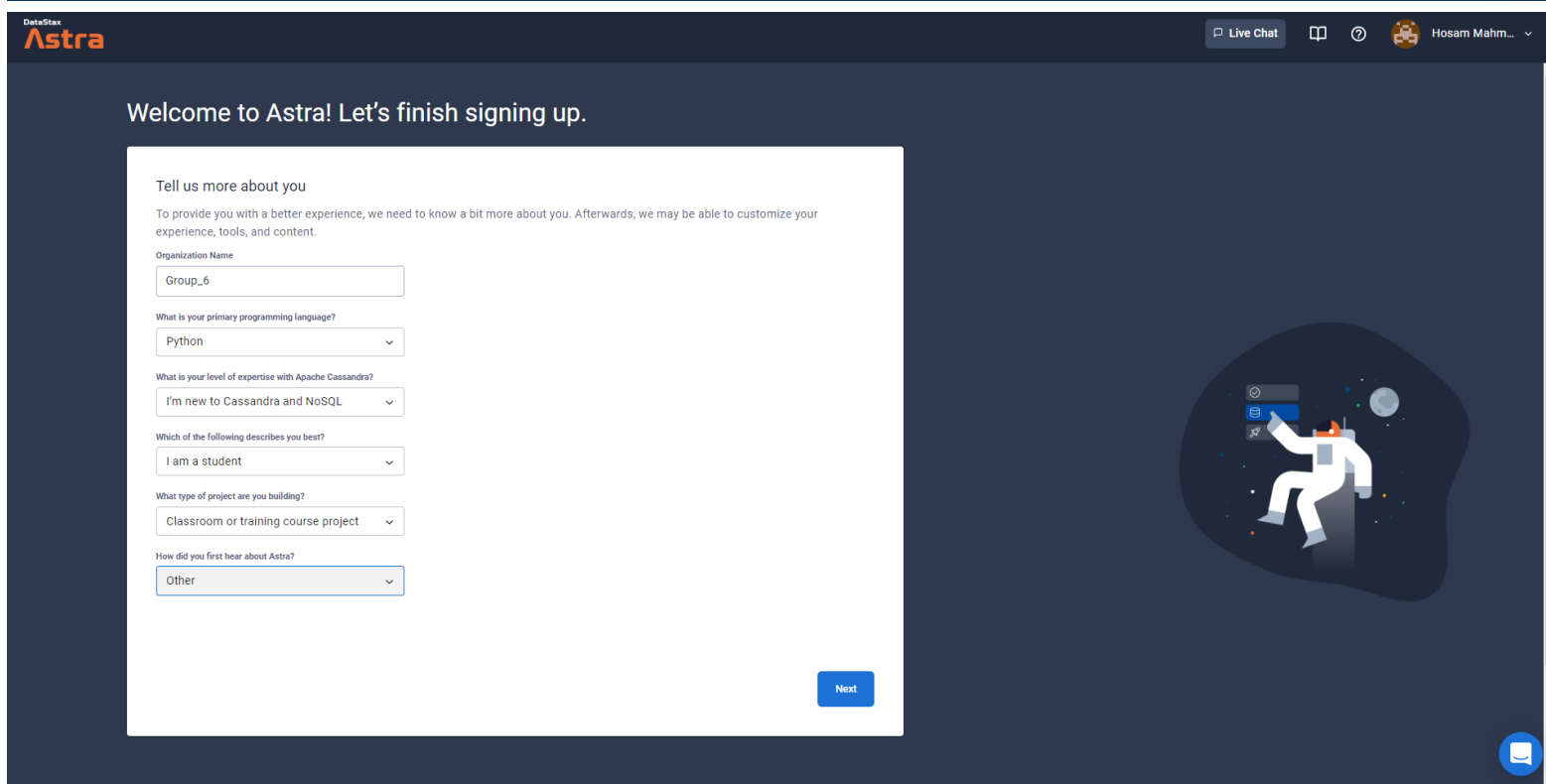
Astra DB

Multi-cloud DBaaS Built on Apache Cassandra™

- Start in minutes. No credit card required. Up to 80 GB free monthly.
- Build faster with REST, GraphQL, CQL and JSON/Document APIs.
- Deploy multi-tenant or dedicated databases on AWS, Azure, or GCP.

Introducing Astra Streaming!

Astra Streaming is an open, multi-cloud event-streaming and data-streaming cloud service powered by Apache Pulsar™, now available within Astra when you sign up for an account.



The screenshot shows the Astra DB onboarding page. The header includes the DataStax Astra logo, a 'Live Chat' button, and a user profile for 'Hosam Mahm...'. The main heading is 'Welcome to Astra! Let's finish signing up.' Below this is a form titled 'Tell us more about you' with a subtext: '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.' The form contains several dropdown menus: 'Organization Name' (Group_6), 'What is your primary programming language?' (Python), 'What is your level of expertise with Apache Cassandra?' (I'm new to Cassandra and NoSQL), 'Which of the following describes you best?' (I am a student), 'What type of project are you building?' (Classroom or training course project), and 'How did you first hear about Astra?' (Other). A 'Next' button is at the bottom right of the form. On the right side of the page, there is an illustration of an astronaut in space.

DataStax Astra

Live Chat

Hosam Mahm...

Welcome to Astra! Let's finish signing up.

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
Group_6

What is your primary programming language?
Python

What is your level of expertise with Apache Cassandra?
I'm new to Cassandra and NoSQL

Which of the following describes you best?
I am a student

What type of project are you building?
Classroom or training course project

How did you first hear about Astra?
Other

Next

- Then, we created our database by clicking here.

The screenshot shows the DataStax Astra dashboard. On the left sidebar, there are links for Dashboard, Databases, Streaming, and Sample App Gallery. The main content area is titled 'Get Started Now' and 'Create your First Database'. A red arrow points to the 'Create Database' button. Below this, there's a 'Dashboard' section with a 'Claim up to \$3,000 in free credits' banner. The 'Current Plan' is 'Free'. There's a 'Databases' section showing usage for the current billing period: Read Requests (0), Write Requests (0), Storage Consumed (0.00), and Data Transfer (0.00). A 'Create Serverless Database' button is visible. At the bottom, there's a 'Streaming' section and a 'Create Stream' button.

- After that, we have chosen our **database** name, **Keyspace** name, the **cloud provider**, and the **region**.

The screenshot shows the 'Create a Database' form. It has two main steps: 1. Enter the Basic Details, and 2. Select a Provider and Region. In step 1, the 'Database Name' is 'ELG5166_Group_6' and the 'Keyspace Name' is 'northwind'. In step 2, the 'Google Cloud' provider is selected, and the region 'St. Ghislain, Belgium' (europa-west1) is chosen. A 'Create Database' button is at the bottom right. A user profile dropdown menu is open on the right, showing the user's name 'Hosam Mahmoud' and email 'hmahm074@uottawa.ca', with options for Account Settings, Billing, and Logout.

- Database created successfully.

The screenshot shows the 'Database Created!' success message in the DataStax Astra interface. The user is Hosam Mahmoud (hmahm074@uottawa.ca). The main message says: 'Wohoo! You just created a new database. While it's deploying, grab your token so you can access it later.'

There are three main sections on the screen:

- Save your secure token details:** This section explains that the user has an auto-generated token and provides instructions on how to save it. It includes a 'Your Token' section with a JSON object containing 'clientId', 'clientSecret', and 'token' fields, all masked with dots. A 'Download Token Details' button is also present.
- Database Details:** This section shows the details for the newly created database 'ELG5166_Group_6'. It includes a 'Finishing up...' status bar and a table with the following information:

Database Details	
Name	ELG5166_Group_6
Keyspace	northwind
Provider	Google Cloud
Region	eu-west-1

A 'Go To Database' button is located at the bottom of this section.
- Get started with your database:** This section provides a brief introduction and a link to 'Get Instructions'.

A 'No thanks, go to the dashboard' link is also visible at the bottom right of the main content area.

- Now to be able to access Cassandra Query Language we clicked here.

The screenshot shows the 'ELG5166_Group_6' database page in the DataStax Astra dashboard. The user is Hosam Mahmoud (hmahm074@uottawa.ca). The page is titled 'Dashboard / Serverless Databases' and 'ELG5166_Group_6 [Active]'. A red arrow points to the 'CQL Console' tab in the navigation menu.

The page displays usage statistics for the current billing period:

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

The 'Regions' section shows the current region configuration:

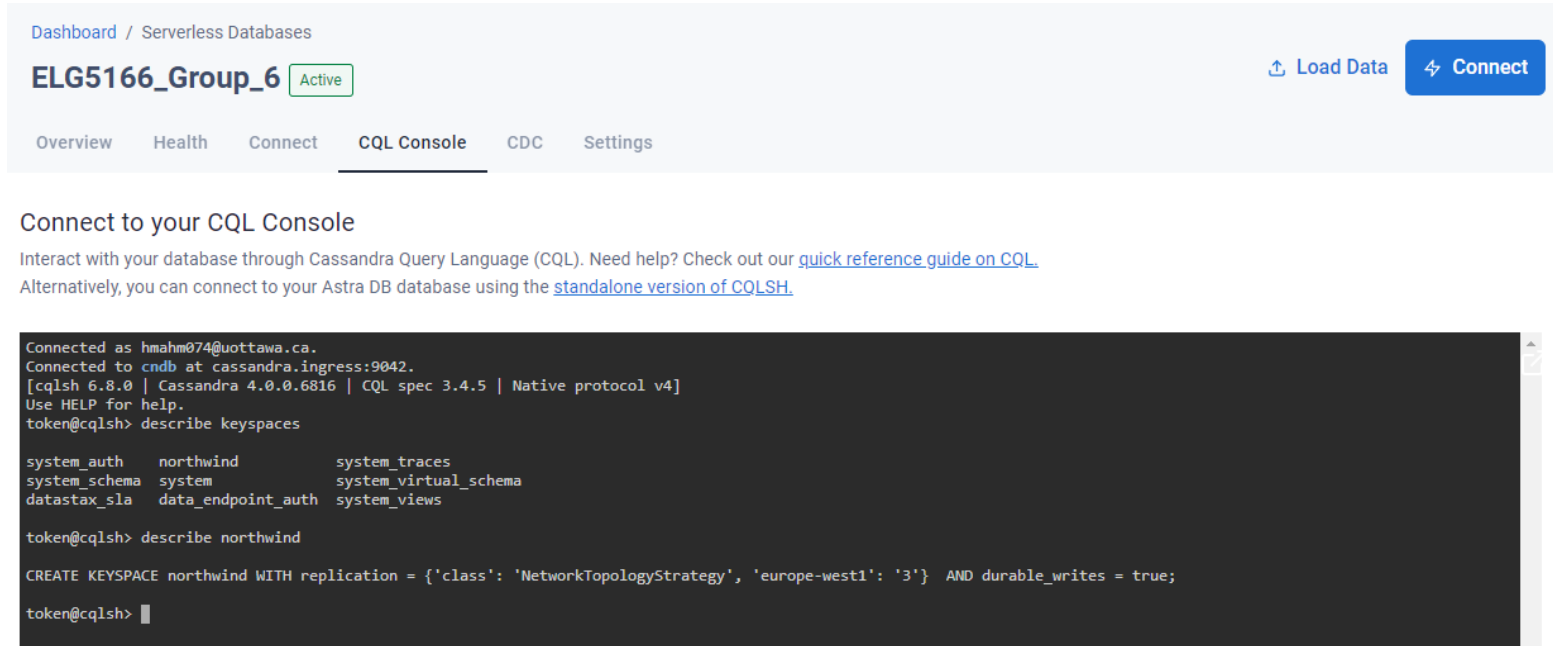
Provider	Area	Region	Region Name	Datacenter ID	Region Availability
Google Cloud	Europe, Middle East, and Africa	eu-west-1	St. Ghislain, Belgium	962e580f-5f57-4be8-8088-407748d6a1f7-1	Online

The 'Keyspaces' section shows the current keyspace configuration:

Keyspace
northwind

ii. Queries

- To see what is the keyspaces that we have created, we used the following commands:
 - describe keyspaces
 - describe northwind



The screenshot shows the Astra DB CQL Console interface. At the top, there's a header with 'Dashboard / Serverless Databases' and a status 'Active' for 'ELG5166_Group_6'. Navigation tabs include Overview, Health, Connect, CQL Console (selected), CDC, and Settings. Below the tabs, a message says 'Connect to your CQL Console' and provides links for help and standalone version. The main area is a terminal window showing the following commands and output:

```
Connected as hmahm074@uottawa.ca.
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> describe keyspaces

system_auth      northwind         system_traces
system_schema    system            system_virtual_schema
datastax_sla     data_endpoint_auth system_views

token@cqlsh> describe northwind

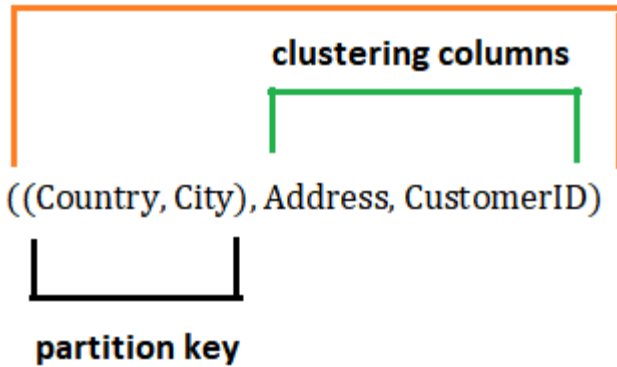
CREATE KEYSPACE northwind WITH replication = {'class': 'NetworkTopologyStrategy', 'europa-west1': '3'} AND durable_writes = true;

token@cqlsh>
```

- here we have used the following commands:
 - use northwind;
 - DROP TABLE IF EXISTS Customers_By_Country;
 - CREATE TABLE Customers_By_Country
(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)
);

1. We have used the first command so that we don't have to write northwind each time we would like to create a table or drop a table.
2. And for the second command is to ensure that there is no table named “Customers_By_Country” in our keyspace.
3. And for the third command is to create our first table which is called “Customers_By_Country”.

primary key



- here we have set “Country” as the main partition and “City” as a sub partition to be able to retrieve the required data fast, without having to use “ALLOW FILTERING”.
- And we have set “Address” as a clustering column to sort the customer’s addresses by their country and city (partition key).
- And also, we have set “CustomerID” as a clustering column to ensure that this table primary key’s is unique.

Dashboard / Serverless Databases

ELG5166_Group_6 Active Load Data Connect

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> use northwind;
token@cqlsh:northwind> DROP TABLE IF EXISTS Customers_By_Country;
token@cqlsh:northwind> CREATE TABLE Customers_By_Country
... (
...     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)
... );
token@cqlsh:northwind> []
```

- To ensure that our table is created we used this command:
 - Describe northwind;

ELG5166_Group_6 Active Load Data Connect

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> describe northwind;

CREATE KEYSPACE northwind WITH replication = {'class': 'NetworkTopologyStrategy', 'eu-west-1': '3'} AND durable_writes = true;

CREATE TABLE northwind.customers_by_country (
  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 = '99PERCENTILE'
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': '64', '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'
```

- To insert some data in that table we used commands like this:
 - INSERT INTO Customers_By_Country (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');

Dashboard / Serverless Databases

ELG5166_Group_6 Active

[Load Data](#)

[Connect](#)

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>
Dtoken@cqlsh:northwind> INSERT INTO Customers_By_Country (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_By_Country (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 Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('ANTON', 'Antonio Moreno Taqueria', 'Antonio Moreno', 'Owner', 'Mataderos 2312', 'México D.F.', 'NS', '05023', 'Mexico', '(5) 555-3932', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (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 Customers_By_Country (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_By_Country (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_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BLOMP', 'Blondesdds1 père et fils', 'Frédérique Citeaux', 'Marketing Manager', '24, place Kléber', 'Strasbourg', 'NS', '67000', 'France', '88.6
0.15.31', '88.60.15.32');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BOLID', 'Bóldo Comidas preparadas', 'Martin Sommer', 'Owner', 'C/ Araquil, 67', 'Madrid', 'NS', '28023', 'Spain', '(91) 555 22 82', '(91) 555
91 99');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (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 Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('BOTTM', 'Bottom-Dollar Markets', 'Elizabeth Lincoln', 'Accounting Manager', '23 Tsawassen Blvd.', 'Tsawassen', 'BC', 'T2F 8M4', 'Canada', '(604
```

- To get the number of records in this table we have used this command:
 - SELECT COUNT(*) FROM Customers_By_Country;

Dashboard / Serverless Databases

ELG5166_Group_6 Active

[Load Data](#)

[Connect](#)

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> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VAFFE', 'Vaffeljernet', 'Palle Ibsen', 'Sales Manager', 'Smagsloget 45', 'Århus', 'NS', '8200', 'Denmark', '86 21 32 43', '86 22 33 44');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VAL2', 'IT', 'Val2', 'IT', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VALON', 'IT', 'Valon Hoti', 'IT', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VICTE', 'Victuailles en stock', 'Mary Savelley', 'Sales Agent', '2, rue du Commerce', 'Lyon', 'NS', '69004', 'France', '78.32.54.86', '78.32.54.87');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VINET', 'Vins et alcools Chevalier', 'Paul Henriot', 'Accounting Manager', '59 rue de l'Abbaye', 'Reims', 'NS', '51100', 'France', '26.47.15.10', '26.47.15.11');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WAMOK', 'Die Wandernde Kuh', 'Rita Müller', 'Sales Representative', 'Adenauerallee 900', 'Stuttgart', 'NS', '70563', 'Germany', '0711-620361', '0711-835428');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WARTH', 'Martian Herku', 'Pirkko Koskitalo', 'Accounting Manager', 'Torikatu 38', 'Oulu', 'NS', '90110', 'Finland', '081-443655', '081-443655');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WELLI', 'Wellington Importadora', 'Paula Parente', 'Sales Manager', 'Rua do Mercado, 12', 'Resende', 'SP', '08737-363', 'Brazil', '(14) 555-8122', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WHITC', 'White Clover Markets', 'Karl Jablonski', 'Owner', '305 - 14th Ave. S. Suite 3B', 'Seattle', 'WA', '98128', 'USA', '(206) 555-4112', '(206) 555-4115');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WILMK', 'Wilman Kala', 'Matti Karttunen', 'Owner/Marketing Assistant', 'Keskuskatu 45', 'Helsinki', 'NS', '21240', 'Finland', '90-224 8858', '90-224 8858');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WOLZA', 'Wolski Zajazd', 'Zbyszek Piestrzeniewicz', 'Owner', 'ul. Filtrowa 68', 'Warszawa', 'NS', '01-012', 'Poland', '(26) 642-7012', '(26) 642-7012');
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind> SELECT COUNT(*) FROM Customers_By_Country;

count
-----
93

(1 rows)
```

- To get the customers that are from Rio de Janeiro, Brazil, and ordered by their addresses, we have used this command.
 - `select * from northwind.Customers_By_Country where Country = 'Brazil' and City = 'Rio de Janeiro';`
- we did not need to use the “ALLOW FILTERING” command, because of calling the partition key.

ELG5166_Group_6 Active
Load Data Connect

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](#).

```
... VALUES('VINET', 'Vins et alcools Chevalier', 'Paul Henriot', 'Accounting Manager', '59 rue de l''Abbaye', 'Reims', 'NS', '51100', 'France', '26.47.15.10', '26.47.15.11');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WANDK', 'Die Wandernde Kuh', 'Rita Müller', 'Sales Representative', 'Adenauerallee 900', 'Stuttgart', 'NS', '70563', 'Germany', '0711-020361', '0711-035428');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WARTH', 'Wartian Herkku', 'Pirkko Koskitalo', 'Accounting Manager', 'Torikatu 38', 'Oulu', 'NS', '90110', 'Finland', '981-443655', '981-443655');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WELLI', 'Wellington Importadora', 'Paula Parente', 'Sales Manager', 'Rua do Mercado, 12', 'Resende', 'SP', '08737-363', 'Brazil', '(14) 555-8122', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WHITC', 'White Clover Markets', 'Karl Jablonski', 'Owner', '385 - 14th Ave. S. Suite 38', 'Seattle', 'WA', '98128', 'USA', '(206) 555-4112', '(206) 555-4115');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WILMK', 'Wilman Kala', 'Matti Karttunen', 'Owner/Marketing Assistant', 'Keskuskatu 45', 'Helsinki', 'NS', '21240', 'Finland', '90-224 8858', '90-224 8858');
token@cqlsh:northwind> INSERT INTO Customers_By_Country (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('MOLZA', 'Wolski Zajazd', 'Zbyszek Piestrzeniewicz', 'Owner', 'ul. Filtrowa 68', 'Warszawa', 'NS', '01-012', 'Poland', '(26) 642-7012', '(26) 642-7012');
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind> SELECT COUNT(*) FROM Customers_By_Country;
```

```
count
-----
  93
(1 rows)
```

Warnings :
Aggregation query used without partition key

```
token@cqlsh:northwind> select * from northwind.Customers_By_Country where Country = 'Brazil' and City = 'Rio de Janeiro';
```

country	city	address	customerid	companyname	contactname	contacttitle	fax	phone	postalcode	region
Brazil	Rio de Janeiro	Av. Copacabana, 267	RICAR	Ricardo Adocicados	Janete Limeira	Assistant Sales Agent	NS	(21) 555-3412	02389-890	RJ
Brazil	Rio de Janeiro	Rua da Panificadora, 12	QUEDE	Que Delícia	Bernardo Batista	Accounting Manager	(21) 555-4545	(21) 555-4252	02389-673	RJ
Brazil	Rio de Janeiro	Rua do Paço, 67	HANAR	Hanari Carnes	Mario Pontes	Accounting Manager	(21) 555-8765	(21) 555-0091	05454-876	RJ

```
token@cqlsh:northwind> select * from northwind.Customers_By_Country where Country = 'Brazil' and City = 'Rio de Janeiro';
```

country	city	address	customerid	companyname	contactname	contacttitle	fax	phone	postalcode	region
Brazil	Rio de Janeiro	Av. Copacabana, 267	RICAR	Ricardo Adocicados	Janete Limeira	Assistant Sales Agent	NS	(21) 555-3412	02389-890	RJ
Brazil	Rio de Janeiro	Rua da Panificadora, 12	QUEDE	Que Delícia	Bernardo Batista	Accounting Manager	(21) 555-4545	(21) 555-4252	02389-673	RJ
Brazil	Rio de Janeiro	Rua do Paço, 67	HANAR	Hanari Carnes	Mario Pontes	Accounting Manager	(21) 555-8765	(21) 555-0091	05454-876	RJ

- for the second query we have created another table to partition the customers by their titles, we used these commands:
 - `DROP TABLE IF EXISTS Customers_By_title;`

- CREATE TABLE Customers_By_title
 (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, CustomerID)
);

primary key

clustering columns

((ContactTitle), ContactName, CustomerID)

partition key

- here we have set “ContactTitle” as the only main partition to be able to retrieve the required data fast, without having to use “ALLOW FILTERING”.
 - And we have set “ContactName” as a clustering column to sort the customer’s names by their title (partition key).
- And also, we have set “CustomerID” as a clustering column to ensure that this table primary key’s is unique.

Dashboard / Serverless Databases

ELG5166_Group_6

Active

Load Data

Connect

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 hmahm074@ottawa.ca.
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_By_title;
token@cqlsh:northwind> CREATE TABLE Customers_By_title
... (
...   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, CustomerID)
... );
token@cqlsh:northwind> S

```


- To insert some data in that table we used commands like this:
 - INSERT INTO Customers_By_title (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');

Dashboard / Serverless Databases

ELG5166_Group_6 Active

[Load Data](#) [Connect](#)

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 COLSH](#).

```
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('TORTU', 'Tortuga Restaurante', 'Miguel Angel Paolino', 'Owner', 'Avda. Arceca 123', 'México D.F.', 'NS', '05033', 'Mexico', '(5) 555-2933', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('TRADH', 'Tradição Hipermercados', 'Anabela Domingues', 'Sales Representative', 'Av. Inês de Castro, 414', 'Sao Paulo', 'SP', '05634-030', 'Brazil', '(11) 555-2167', '(11) 555-2168');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('TRAIH', 'Trail's Head Gourmet Provisioners', 'Helvetius Nagy', 'Sales Associate', '722 DaVinci Blvd.', 'Kirkland', 'WA', '98034', 'USA', '(206) 555-8257', '(206) 555-2174');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VAFFE', 'Vaffeljernet', 'Palle Ibsen', 'Sales Manager', 'Smagsloget 45', 'Arhus', 'NS', '8200', 'Denmark', '86 21 32 43', '86 22 33 44');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('Val2', 'IT', 'Val2', 'IT', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VALON', 'IT', 'Valon Hoti', 'IT', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VICTE', 'Victuailles en stock', 'Mary Saveley', 'Sales Agent', '2, rue du Commerce', 'Lyon', 'NS', '69004', 'France', '78.32.54.86', '78.32.54.87');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VINET', 'Vins et alcools Chevalier', 'Paul Henriot', 'Accounting Manager', '59 rue de l'Abbaye', 'Reims', 'NS', '51100', 'France', '26.47.15.10', '26.47.15.11');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WANDK', 'Die Wandernde Kuh', 'Rita Müller', 'Sales Representative', 'Adenauerallee 900', 'Stuttgart', 'NS', '70563', 'Germany', '0711-020361', '0711-035428');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WARTH', 'Wartian Herkku', 'Pirkko Koskitalo', 'Accounting Manager', 'Torikatu 38', 'Oulu', 'NS', '90110', 'Finland', '981-443655', '981-443655');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WELLI', 'Wellington Importadora', 'Paula Parente', 'Sales Manager', 'Rua do Mercado, 12', 'Resende', 'SP', '08737-363', 'Brazil', '(14) 555-8122', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WHITC', 'White Clover Markets', 'Karl Jablonski', 'Owner', '305 - 14th Ave. S. Suite 38', 'Seattle', 'WA', '98128', 'USA', '(206) 555-4112', '(206) 555-4115');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WILMK', 'Wilman Kala', 'Matti Karttunen', 'Owner/Marketing Assistant', 'Keskuskatu 45', 'Helsinki', 'NS', '21240', 'Finland', '90-224 8858', '90-224 8858');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WOLZA', 'Wolski Zajazd', 'Zbyszek Piestrzeniewicz', 'Owner', 'ul. Filtrowa 68', 'Warszawa', 'NS', '01-012', 'Poland', '(26) 642-7012', '(26) 642-7012');
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind>
```

- To get the number of records in this table we have used this command:
 - SELECT COUNT(*) FROM Customers_By_title;

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](#).

```
... VALUES('VALON', 'IT', 'Valon Hoti', 'IT', 'NS', 'NS', 'NS', 'NS', 'NS', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VICTE', 'Victuailles en stock', 'Mary Saveley', 'Sales Agent', '2, rue du Commerce', 'Lyon', 'NS', '69004', 'France', '78.32.54.86', '78.32.54.87');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('VINET', 'Vins et alcools Chevalier', 'Paul Henriot', 'Accounting Manager', '59 rue de l'Abbaye', 'Reims', 'NS', '51100', 'France', '26.47.15.10', '26.47.15.11');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WANDK', 'Die Wandernde Kuh', 'Rita Müller', 'Sales Representative', 'Adenauerallee 900', 'Stuttgart', 'NS', '70563', 'Germany', '0711-020361', '0711-035428');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WARTH', 'Martian Herkku', 'Pirkko Koskitalo', 'Accounting Manager', 'Torikatu 38', 'Oulu', 'NS', '90110', 'Finland', '981-443655', '981-443655');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WELLI', 'Wellington Importadora', 'Paula Parente', 'Sales Manager', 'Rua do Mercado, 12', 'Resende', 'SP', '08737-363', 'Brazil', '(14) 555-8122', 'NS');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WHITC', 'White Clover Markets', 'Karl Jablonski', 'Owner', '305 - 14th Ave. S. Suite 3B', 'Seattle', 'WA', '98128', 'USA', '(206) 555-4112', '(206) 555-4115');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WILMK', 'Wilman Kala', 'Matti Karttunen', 'Owner/Marketing Assistant', 'Keskuskatu 45', 'Helsinki', 'NS', '21240', 'Finland', '90-224 8858', '90-224 8858');
token@cqlsh:northwind> INSERT INTO Customers_By_title (CustomerID, CompanyName, ContactName, ContactTitle, Address, City, Region, PostalCode, Country, Phone, Fax)
... VALUES('WOLZA', 'Wolski Zajazd', 'Zbyszek Piestrzeniewicz', 'Owner', 'ul. Filtrowa 68', 'Warszawa', 'NS', '01-012', 'Poland', '(26) 642-7012', '(26) 642-7012');
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind>
token@cqlsh:northwind> SELECT COUNT(*) FROM Customers_By_title;

count
-----
93

(1 rows)
```

- To get the customers that are in the Sales Manager role and ordered by their names, we have used this command.
 - select * from northwind.Customers_By_title where contacttitle = 'Sales Manager';
- we did not need to use the “ALLOW FILTERING” command, because of calling the partition key.

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> select * from northwind.Customers_By_title where contacttitle = 'Sales Manager';
```

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	NY
Sales Manager	Fran Wilson	LONEP	89 Chiaroscuro Rd.	Portland	Lonesome Pine Restaurant	USA	(503) 555-9646	(503) 555-9573	97219	OR
Sales Manager	Georg Pippes	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>
```