Lab Homework 9: Cassandra and CQL

# Learning Objectives

In this exercise, you will practice programming in Cassandra’s CQL language:

* Design a table suitable for the Cassandra database
* Create the table, Insert data
* Query the data, creating indexes and materialized views.

# Setup

To complete this homework or follow along with the instructor’s class demos from your coursework, you must first complete the steps in this setup section. For this exercise we require the SQL Server instance in the Docker environment. For more details on the how’s and why’s of managing the database environments used in this course, consult Lab 1. Instructions for this assignment:

1. Open the PowerShell prompt on Windows or the Terminal on MacOS.
2. Type:  
   cd adv-db-labs  
   to change the working directory to the repository folder. If you are in the correct spot, your command prompt should have adv-db-labs in it, for instance: PS adv-db-labs>
3. Change into the cassandra folder, type:  
   cd cassandra  
   if you are in the correct folder, your command prompt should have cassandra in it.
4. Bring up the Cassandra environment, type:  
   docker-compose up -d
5. Check to make sure the environment is running, type:  
   docker-compose ps

You should see 3 Cassandra nodes all **Up**.

1. You are now ready to connect to the **cql shell**:   
   docker-compose exec cassandra0 cqlsh

When you are connected you should be at the cqlsh> prompt, and you are now ready to

# Exercises

Complete each of the following exercises. If you are unsure how to accomplish the task, please consult the coursework videos where there are explanations and demos.

1. Design your own scenario for which a Cassandra table would be a good solution. Make sure to explain the scenario and the specific characteristics of the scenario which would make Cassandra a good fit. Make sure to follow a query first approach and justify how the partition and cluster keys should be setup.
2. Create your Cassandra table in CQL based on your scenario from the previous exercise. You should define the columns and data types to suit your scenario in addition to configuring the partition and cluster keys.
3. Write CQL statements to add data to your table. Add at least 9 records consisting of 3 different partition and cluster keys  
   .
4. Write a CQL statement to create an index or materialized view on your table so that you can set a different partition key to prevent ALLOW FILTERING. Then write a CQL SELECT statement to demonstrate it works as designed.
5. Write a CQL statement to create an index or materialized view on your table so that you can set a different cluster key to prevent ALLOW FILTERING. Then write a CQL SELECT statement to demonstrate it works as designed.

# Turning it in

Take your copy and paste each of the solutions to the exercises into the submission template file included with this assignment. Make sure your name and SU email are at the top and turn in your work through the course learning management system.

# Tear-Down

When you are finished with the homework you should stop the environment:

1. From the terminal window where you typed docker-compose up -d type in the following:  
   docker-compose stop