UDACITY

Return to "Data Engineering Nanodegree" in the classroom

Data Modeling with Cassandra

REVIEW

HISTORY

Meets Specifications

Keen Learner,
Congratulations on completing the project!🎉
You should be very proud of your accomplishments in building a Data Model with Cassandra. You'll find feedback and some tips to help you continue learning.
Keep up the good work and all the best to you with the next project. Kudos!

Extra Materials

You can check out the following for further learning

- Understanding Cassandra's data model and Cassandra Query Language (CQL)
- Apache Cassandra: Compound Primary Key
- Apache Cassandra Data Modeling and Query Best Practices
- Designing a Cassandra Data Model

- Cassandra data modeling: The primary key
- CQL data types

ETL Pipeline Processing

Student creates `event_data_new.csv` file.

Nice job! I see the `event_datafile_new.csv` file, which indicates you followed the ETL pipeline to create the csv file.

Student uses the appropriate datatype within the `CREATE` statement.

Nice work!

You are using the correct datatype, such as `int` for the `itemInsession` column. You are also using the correct string datatype for the relevant columns as all of your INSERT statements are processing the data correctly.

Suggestions

Here are some links about CQL Datatypes:

- Datastax CQL data types
- Cassandra - CQL Datatypes

Data Modeling

Student creates the correct Apache Cassandra tables for each of the three queries. The `CREATE TABLE` statement should include the appropriate table.

Excellent!

Your data is modeled correctly to generate the results requested in the questions.

Student demonstrates good understanding of data modeling by generating correct SELECT statements to generate the result being asked for in the question.

The SELECT statement should NOT use `ALLOW FILTERING` to generate the results.

Nice work! The specific fields are queried from each table as asked. 👍

Student should use table names that reflect the query and the result it will generate. Table names should include alphanumeric characters and underscores, and table names must start with a letter.

Nice work using table names that reflect the query and the result it will generate. This is important because this describes the data model. 🎉

The sequence in which columns appear should reflect how the data is partitioned and the order of the data within the partitions.

Excellent work!
This is probably one of the most important learnings that I want you to walk away with from this lesson.

Tips

Apache Cassandra is a partition row store, which means the partition key determines which node a particular row is stored on. With the Primary key (which includes the Partition Key and any clustering columns), the partitions are distributed across the nodes of the cluster. It determines how data are chunked for write purposes. Any clustering column(s) would determine the order in which the data is sorted within the partition.

PRIMARY KEYS

The combination of the PARTITION KEY alone or with the addition of CLUSTERING COLUMNS should be used appropriately to uniquely identify each row.

Nice work understanding and implementing the PRIMARY KEY with a COMPOSITE Partition for both the CREATE and INSERT statements.

Suggestions

- You may check this resource for additional information.
- Here is a nice discussion about the Difference between partition key, composite key and clustering key in Cassandra.

Presentation

The notebooks should include a description of the query the data is modeled after.

Good work including headers to denote descriptions of the query! It was clear how each query was being addressed.

Code should be organized well into the different queries. Any in-line comments that were clearly part of the project instructions should be removed so the notebook provides a professional look.

Good work taking out the inline instructions and comments that were part of the project instructions. Your notebook looks professional.

DOWNLOAD PROJECT