



Athens University of Economics and Business

Department of Management Science and Technology

MSc in Business Analytics—Big Data Systems

Cassandra Assignment

Associate Professor P. Louridas

Instructions

Your submission will consist of zip archive that will contain the documentation of the solution to the assignment and a CQL file that can be directly read by Cassandra.

The Problem

You will design a Cassandra database for a music streaming service. The service stores information on the following:

- Songs
- Users
- Playlists

In particular, for each song the service needs to store:

- Song name
- Artist
- Album
- Genre
- Year
- Play count
- The song binary file

For each user, the service needs to store:

- Name
- Address

- Payment information
- Complete history of songs played
- Users that the specific user follows; a user may follow other users so that they can easily find their playlists etc.

For each playlist, the service needs to store:

- Name
- Description
- Genre
- User that has created the playlist
- Songs in the playlist; the songs must be in a particular order
- Users that follow the playlist

After analysing the requirements, it was found that the service will need to implement the following queries:

- Q1: Find user by name
- Q2: Find song by name
- Q3: Find songs played by a user, arranged in reverse chronological order (more recent first)
- Q4: Find playlist by name
- Q5: Find playlist by genre
- Q6: Find playlist by creator
- Q7: Find the followers of a playlist
- Q8: Find the followers of a user
- Q9: Find the songs contained in a playlist
- Q10: Find how many times a playlist has been played
- Q11: Find how many times a song has been played
- Q12: List playlists in decreasing popularity; the popularity of a playlist is defined by the number of times it has been played
- Q13: List users in decreasing popularity: the popularity of a user is defined by the number of followers

Notes

Your solution must cover both the database tables and the CQL queries that correspond to each of the queries Q1–Q12. As mentioned above, your solution will consist of two files, which you will put them in a single zip archive.

The first file will explain the database design: how you decided on the particular tables and queries. The second file will be straight CQL code for the table creation statements and the queries. It must be possible to run the code by entering `SOURCE 'your_file.cql'` from inside `cqlsh`.

You are well advised to populate your database with some dummy data, so that you will be able to run your queries; you should definitely check that your queries work. Remember that your tables must be designed so that they can answer the queries: the queries may look nice on paper (or on screen), but when you try to run them you may discover that you get errors because the tables have not been designed to meet their requirements!

You know that Cassandra tables are generally denormalized. However, you can still save space with the right choices, for example using static columns; keep this in mind.

Go for it!