E Commerce Reports

Introduction

You got a new assignment where you have to keep a track of how customers are liking the products of the company. You are keeping this track by capturing the number of likes against each Product by a customer

In this project the focus is on learning and writing different types of CQL (Cassandra Query Language) statements to insert and retrieve data from a database.
We will use the Cassandra database created in AstraDB.
For reference, feel free to access the weekly notebooks and sessions for hints, while working on CQL in this project.

Housekeeping points

- This is a minimal example and may not follow some standard practices.
- We focus on the main flow, and not much error handling.

Program Organization

The simple program is structured in various layers.

1. To give an insight we have three tables ***Customer***, ***Product*** *and* ***Product_Liked_By_Customer.*** Please find the table structures as below for more details.

   Customer Table (Primary Key = cust_id) - Captures basic information of a customer

cust_id	text
first_name	text
last_name	text
registered_on	timestamp

   Product Table (Primary Key = prdt_id) - Captures basic information of a product

prdt_id	text
title	text

Product_Liked_By_Customer Table (Primary Key = (cust_id, liked_prdt_id, liked_on)) - Captures information on various products customer(s) like

cust_id	text
first_name	text
last_name	text
liked_prdt_id	text
liked_on	timestamp
title	text

Shared Project folder statement

1. ***config*** : In this folder we have test data in3 .csv files. One for each table mentioned above. We will insert this data in the respective tables.

2. ***connect_bundle*** : In this folder we have a .zip file that has a secure connect bundle. We will use this to connect with the Cassandra database.

3. ***database.py*** : In this file we have implemented various methods and defined variables which are required to perform specific operations like creating a session, session key space and methods to execute a query. We also have defined variables like
    a. ASTRA_CLIENT_ID
    b. ASTRA_SECRET
    c. Cloud_config

These variables are used to initialize an authentication provider, create session with Cassandra databases and perform various operations.

4. ***setup.py***: This file implements the functionality of initializing a session with Cassandra database. Create the tables mentioned above. And insert the data in these tables.

5. ***main.py***: This file initializes a session object. This session object is used to perform various operations in the Cassandra database.

Problem Statement

In this project you have to do the following activities…

- Create an Account in AstraDB
- Create database within a keyspace
- Create the three tables
- Insert data in these tables

- Retrieve the data from these tables based on the requirements mentioned below

1. (**Easy**) Create an Account in AstraDB and create database with name ***‘e_commerce_db’***
    a. Create an Account in AstraDB and create a database named ***‘e_commerce_db’*** in keyspace ‘***e_commerce***’. Generate Token for connection and fetch ***‘Client ID’ and ‘Client Secret’*** for connection. Also fetch a ***secure connect bundle*** for the session creation.
    b. Implement a function ***create_session()*** in ***database.py*** method to initialize a session with Cassandra database to perform various operations.
    c. Implement a function ***set_session_keyspace()*** in ***database.py*** for setting keyspace in the initialized session.

2. (**Medium**) Create three tables as explained above, insert data into the tables and perform various operations on tables
    a. Implement the code for creating three tables as discussed before, in ***setup.py***.
    b. In ***setup.py***, Implement the code for inserting data in the config folder into the created tables.
    c. In ***main.py***, execute queries to…
        i. Count the number of Products and Customers added in the Product and customer Table respectively.
        ii. Count the numbers of likes received on the products(product wise).

Evaluation Rubric

Total Project Points: **20**

- Basic compilation without errors (**10%**) : **2 Points**
- Correctness:
  Correctness of implementation
    - Problem statement - point 1.a (**15%**) : **3 Points**
    - Problem statement - point 1.b (**15%**) : **3 Points**
    - Problem statement - point 1.b (**15%**) : **3 Points**
    - Problem statement - point 2.a (**15%**) : **3 Points**
    - Problem statement - point 2.b (**15%**) : **3 Points**
    - Problem statement - point 2.c (**15%**) : **3 Points**

Program Instructions

1. Download the zip file named **C04-Project02-02-Ecommerce-Reports.zip**, unzip it on your local machine, and save it. Go into the directory named **C04-Project02-02-Ecommerce-Reports.**

2. Make sure you have Python 3.6 or higher installed. At your command prompt, run:

```
$ python --version
Python 3.7.3
```

   If not installed, install the latest available version of Python 3.

3. You need an Account created on AstraDB and then create a database named '**e_commerce_db**' in the keyspace '**e_commerce'**.

4. First you need to write functionality in **setup.py**, to fill the data in the created database. Make sure that while running **setup.py** you are not getting any error.

5. You can now examine and run **main.py**. This will currently run various simple calls. As you solve the problems, you'll be frequently modifying and running this file. You can comment or modify the initial code as needed.

```
$ python3 main.py (On many Linux platforms)
                OR
$ python main.py  (On Windows platforms)
```

In any case, one of these two commands should work.

6. Alternatively, you could install a popular Python **IDE**, such as **PyCharm** or **Visual Studio Code**, and select a command to build the project from there.

7. Once the program is ready to submit, zip the folder **C04-Project02-02-Ecommerce-Reports** and upload the new zip file as **C04-Project02-02-Ecommerce-Reports.zip**. It is now ready for evaluation.