Liz Kraisinger

DS2002

11/27/22

**Midterm Project Overview**

1. **Repository Contents:**
   1. **classicdata:** Folder that contains the 4 JSON source files for creating the dimension tables.
   2. **01-mysqlsampledatabase:** SQL source code for sample “classicmodels” database.
   3. **02-dim\_date:** Contains SQL code for creating the date dimension table, based on a previous lab.
   4. **03-JSON Queries:** SQL queries used to export data into JSON files (see classicdata folder).
   5. **04-Midterm:** Python notebook with ETL process for classicmodels data.
2. **Project Documentation:**
   1. **Loading data:** 
      1. Created the classicmodels schema in mySQL Workbench using 01-mysqlsampledatabase.
      2. Ran 02-dim\_date to create date dimension table in classicmodels. There was also a line of code that creates the “destination” database, classicmodels\_dw3
      3. Used 03-JSON Queries to select the necessary tables from classicmodels, then exported them as JSON files to be used for the dimension tables.
   2. **Code:**
      1. 0.0: Setup
         1. Import necessary packages, define variables and SQL/Mongo functions, and push JSON files in from classicdata folder into MongoDB cluster
      2. 1.0: Dimension Tables
         1. Extracted data from source MongoDB collections into dataframes using queries for customers, payments, products, and date tables.
         2. Performed transformations to dataframes (dropped unnecessary columns, inserted primary keys where necessary).
         3. Loaded new tables into classicmodels\_dw3 as the dimension tables.
         4. Used SQL select statements to verify that the dimension tables were created and loaded correctly.
      3. 2.0: Fact Table
         1. Extracted orders and order details tables from classicmodels in mySQL.
         2. Merged orders/orderdetails into the fact orders table, then performed joins for the 4 dimension tables to add the primary key from each dimension table as a surrogate key in the fact orders table. Performed any necessary transformations (e.g., renaming or dropping columns).
         3. Loaded new fact orders table into classicmodels\_dw3.
         4. Used query to validate that fact orders table was correctly loaded.
         5. Used 3 SQL select statements to verify that the data warehouse is set up correctly.
   3. **Deployment Strategy:** 
      1. After creating the necessary data and loading it in SQL and MongoDB, the code extracted data from MongoDB databases for the dimension tables as pandas dataframes to perform necessary transformations. It then loaded the transformed tables into classicmodels\_dw3 in SQL.
      2. The fact table was created by extracting data from the source database, classicmodels, and the surrogate keys were added using table merges. The transformed data was loaded back into classicmodels\_DW3.