Rushi Samirbhai Patel
B00886157
rs397441@dal.ca

Assignment 2 - Problem 2

Cleaning and Refining data models:

1. **Olist_order_reviews_dataset:**
    - Changed blank values to ‘ ‘ for columns review_comment_title and review_comment_message.

2. **Olist_orders_dataset:**
    - Changed blank values from columns order_delivered_carrier_date and order_delivered_customer_date columns to 0000-00-00 00:00:00.

3. **Olist_products_dataset:**
    - Changed blank values from product_category_name column to ‘ ‘.
    - Changed blank values from product_name_lenght, product_photos_qty, product_weight_g, product_length_cm, product_height_cm and product_width_cm columns to 0.

4. **olist_customers_dataset:**
    - I did not find any blank values or invalid data the rest of the datasets, so I kept this table unchanged.

5. **olist_geolocation_dataset:**
    - I did not find any blank values or invalid data the rest of the datasets, so I kept this table unchanged.

6. **olist_order_items_dataset:**
    - I did not find any blank values or invalid data the rest of the datasets, so I kept this table unchanged.

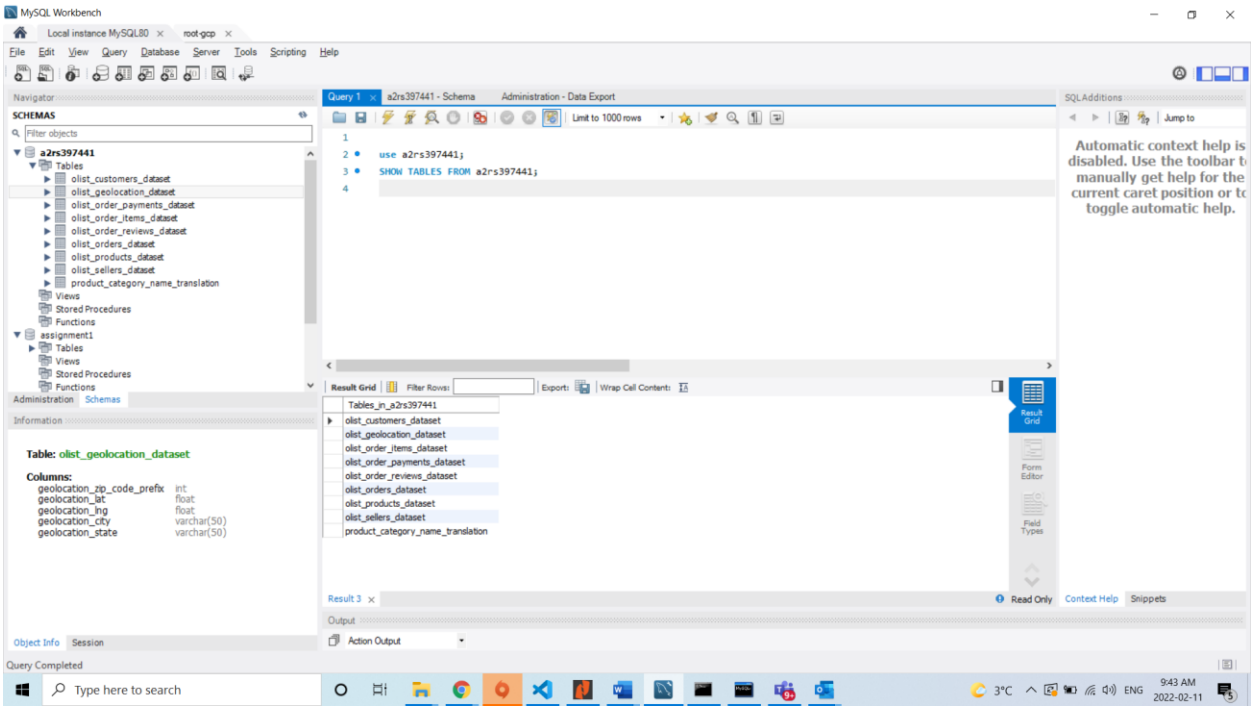
7. **olist_order_payments_dataset:**
    - I did not find any blank values or invalid data the rest of the datasets, so I kept this table unchanged.

8. **olist_sellers_dataset:**
    - I did not find any blank values or invalid data the rest of the datasets, so I kept this table unchanged.

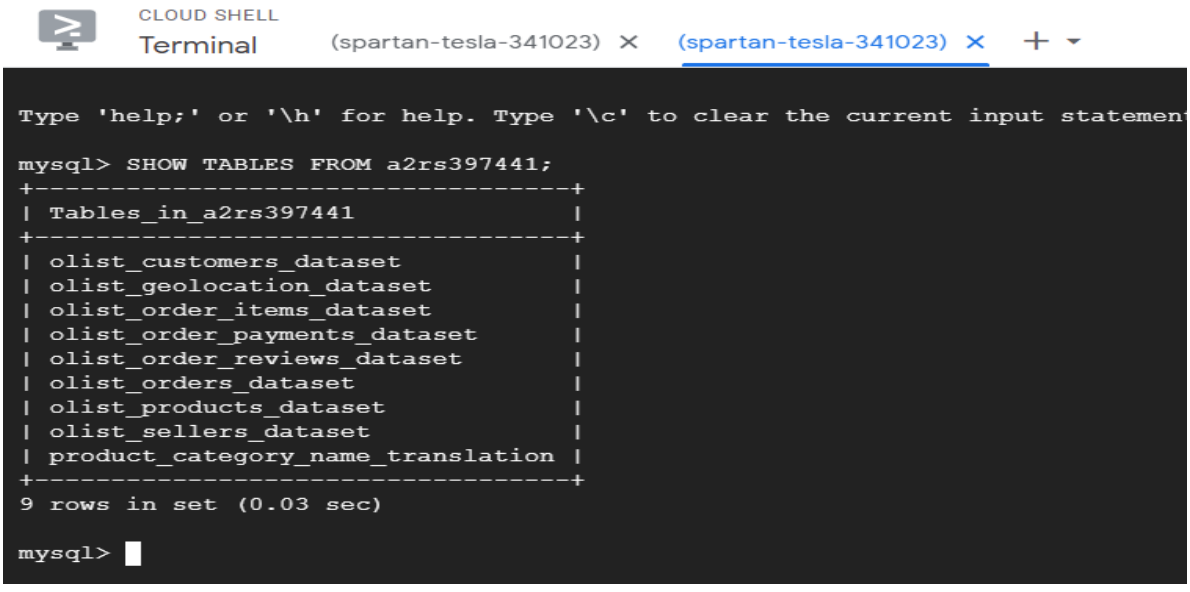
Rushi Samirbhai Patel
B00886157
rs397441@dal.ca

9. **product_category_name_translation:**
    - I did not find any blank values or invalid data the rest of the datasets, so I kept this table unchanged.

Local Instance Screenshot: (SHOW TABLES FROM a2rs397441;):



GCP Instance screenshot (SHOW TABLES FROM a2rs397441;):



Rushi Samirbhai Patel rs397441@dal.ca
B00886157

Table 2.1.1 Transaction Query based on task number

Tasks	Execution Time	Transaction Query	Your Observation

Task 2	1.962 s	DECLARE @start_time DATETIME, @end_time DATETIME
		SET @start_time1 = CURRENT_TIMESTAMP
		
		DELIMITER //
		
		CREATE PROCEDURE myTransaction()
		BEGIN
		Start transaction;
		insert into olist_customers_dataset values("dummy_customer_id","dummy_customer_unique_id",231,"dsaa","dfdf");
		select * from olist_customers_dataset where customer_id="dummy_customer_id";
		insert into olist_geolocation_dataset values(1,-95.32,231,"dsaa","dfdf");
		select * from olist_geolocation_dataset where geolocation_lat=-95.32;
		delete from olist_order_payments_dataset where order_id="b81ef226f3fe1789b1e8b2acac839d17";
		select * from olist_order_payments_dataset where order_id="b81ef226f3fe1789b1e8b2acac839d17";
		delete from olist_order_items_dataset where order_id="b81ef226f3fe1789b1e8b2acac839d17";
		select * from olist_order_items_dataset where order_id="b81ef226f3fe1789b1e8b2acac839d17";
		update olist_order_reviews_dataset set review_id = "updated_id" where review_id="7bc2406110b926393aa56f80a40eba40";
		select * from olist_order_reviews_dataset where review_id="updated_id";
		update olist_orders_dataset set order_id = "updated_id" where order_id="e481f51cbdc54678b7cc49136f2d6af7";
		select * from olist_orders_dataset where order_id="updated_id";
		update olist_products_dataset set product_id = "updated_id" where product_id="1e9e8ef04dbcff4541ed26657ea517e5";
		select * from olist_products_dataset where product_id="updated_id";
		update olist_sellers_dataset set seller_id = "updated_id" where seller_id="3442f8959a84dea7ee197c632cb2df15";
		select * from olist_sellers_dataset where seller_id="updated_id";
		insert into product_category_name_translation values("dummy_product_category_name","dummy_product_category_name_english");
			To start a transaction, I had to first set the auto-commit off. Secondly, I created a procedure for my transaction and later called the procedure. I declared a start time and end time by using datetime. Once the transaction is finished, I calculated the total execution time with the help of datediff function.
			
			With respect to transaction, I noticed that all the Data Definition Language (DDL) were not supported in the transaction however, Data Manipulation Languages (DML) were executed successfully in the transaction.

		select * from product_category_name_translation where product_category_name="dummy_product_category_name"; commit; END // call myTransaction; SET @end_time = CURRENT_TIMESTAMP SELECT DATEDIFF(ms, @start_time, @end_time)	
Task 3	2.358 s	Start transaction; update product_category_name_translation set product_category_name_english="updated_name" where product_category_name="beleza_saude"; select * from product_category_name_translation where product_category_name_english="updated_name"; insert into olist_sellers_dataset values ("dummy_seller_id",2,"dummy_city", "dummy_state"); select * from olist_sellers_dataset where seller_id="dummy_seller_id"; update olist_products_dataset set product_category_name="updated_category_name" where product_id="3aa071139cb16b67ca9e5dea641aaa2f"; select * from olist_products_dataset where product_category_name="updated_category_name"; update olist_orders_dataset set order_id="updated_order_id_2" where product_id="53cdb2fc8bc7dce0b6741e2150273451"; select * from olist_orders_dataset where order_id="updated_order_id_2"; delete from olist_order_reviews_dataset where review_id="228ce5500dc1d8e020d8d1322874b6f0"; select * from olist_order_reviews_dataset where review_id="228ce5500dc1d8e020d8d1322874b6f0"; insert into olist_order_payments_dataset values ("dummy_order_id_inserted",2,"dummy_payment_type", 2,3); select * from olist_order_reviews_dataset where order_id="dummy_order_id_inserted"; delete from olist_order_items_dataset where order_id="a9810da82917af2d9aefd1278f1dcfa0"; select * from olist_order_items_dataset where order_id="a9810da82917af2d9aefd1278f1dcfa0"; update olist_geolocation_dataset set geolocation_lat=10 where geolocation_city="sao paula"; select * from olist_geolocation_dataset where geolocation_lat=10; delete from olist_customers_dataset where customer_id="06b8999e2fba1a1fbc88172c00ba8bc7"; select * from olist_customers_dataset where customer_id="06b8999e2fba1a1fbc88172c00ba8bc7"; commit;	Firstly, I noticed that when I fire any query on my GCP instance it took more time to execute when compared to the local execution of query. When I started the transaction I noticed that the behavior remained the same as compared to the local instance query that I fired in Task 2. The DDL queries were not performed in the transaction, however the DML queries were executed with comparatively larger execution time.
Task 4	4 s	Starting Local Instance: SET AUTOCOMMIT=0;	Here to perform distributed transaction, I first

		insert into olist_order_payments_dataset values('dummy_order_id_task4', 2, 'dummy_payment_type_task4', 2, 56.23);
		insert into product_category_name_translation values('dummy_product_category_name_task4','dummy_product_category_name_english_task4');
		select * from olist_orders_dataset where order_id='53cdb2fc8bc7dce0b6741e2150273451';
		select * from olist_products_dataset where product_id='3aa071139cb16b67ca9e5dea641aaa2f';
		delete from olist_sellers_dataset where seller_id='d1b65fc7debc3361ea86b5f14c68d2e2';
		
		Starting GCP Instance:
		
		SET AUTOCOMMIT=0;
		update olist_customers_dataset set customer_id='dummy_customer_id_task4' where customer_unique_id='861eff4711a542e4b93843c6dd7febb0';
		select * from olist_geolocation_dataset where geolocation_city='sao paulo';
		update olist_order_reviews_dataset set review_id='dummy_review_id_task4' where order_id='73fc7af87114b39712e6da79b0a377eb';
		delete from olist_order_payments_dataset where order_id='ba78997921bbcdc1373bb41e913ab953';
		
		Local instance: commit;
		GCP instance: commit;
			started my local instance where I first disabled auto-commit. I then performed all the queries related to the 5 tables considered for the local instance. Before committing these queries, I started the GCP instance where I again disabled the auto-commit and performed the queries related to the rest 4 tables considered for the GCP instance. After performing all the queries, I committed both the instances to their respective databases and enacted a distributed transaction acting both locally and on GCP. I observed that it took more time to execute maybe because I used java.time to capture the start time and end time of the operations.

Task 4 Repository link:

https://git.cs.dal.ca/rspatel/csci-5408-w2022-b00886157-rushi_patel/-/tree/main/Assignment-2

SSH: git@git.cs.dal.ca:rspatel/csci-5408-w2022-b00886157-rushi_patel.git

HTTPS: https://git.cs.dal.ca/rspatel/csci-5408-w2022-b00886157-rushi_patel.git

Rushi Samirbhai Patel rs397441@dal.ca
B00886157

References:

[1] M. work, T. Tomov and D. Azamar, "MySQL begin end from documentation doesn't work", Database Administrators Stack Exchange, 2022. [Online]. Available: https://dba.stackexchange.com/questions/207758/mysql-begin-end-from-documentation-doesnt-work. [Accessed: 12- Feb- 2022].

[2] "XA-implemented distributed transactions based in mysql - Programmer All", Programmerall.com, 2022. [Online]. Available: https://programmerall.com/article/62061617774/. [Accessed: 12- Feb- 2022].

[3] "BEGIN...END (Transact-SQL) - SQL Server", Docs.microsoft.com, 2022. [Online]. Available: https://docs.microsoft.com/en-us/sql/t-sql/language-elements/begin-end-transact-sql?view=sql-server-ver15. [Accessed: 12- Feb- 2022].