DATA EXPLORATION OF OLIST BRAZILIAN E-COMMERCE USING SQL & BIGQUERY

FATIMA EZZAHRA KABBA

DATA ANALYST | RESEARCH ANALYST | BUSINESS INTELLIGENCE | FREELANCE CONSULTANT

📧 **EMAIL:** FATIMA.EZZAHRA.KABBA.PRO@GMAIL.COM
🔗 **LINKEDIN:** LINKEDIN.COM/IN/DR-FATIMA-KABBA
💻 **GITHUB:** GITHUB.COM/FATIMAKABBA

PROJECT OVERVIEW

A STRUCTURED DATA EXPLORATION OF OLIST'S E-COMMERCE DATASET USING SQL & BIGQUERY

📦 What is the Olist Dataset?

- 🛒 **Brazilian e-commerce transactions** from a marketplace platform.
- 📊 Contains **multiple relational tables** (orders, products, customers, sellers).

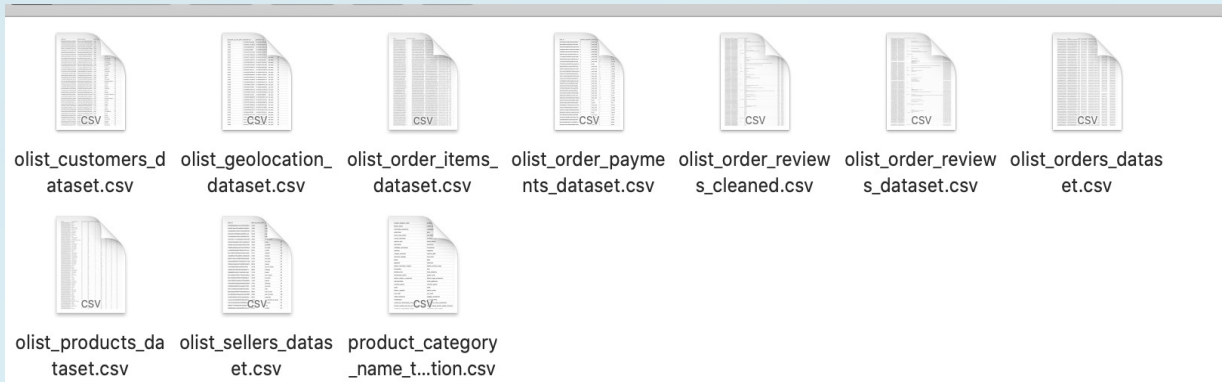
💡 Why This Dataset?

✅ **Real-world business data** – Used for e-commerce insights.

✅ **Great for SQL & data cleaning** – Perfect for **data exploration & transformation.**

✅ **Key project objectives:**

- 🔍 **Understand data structure & relationships.**
- 🔗 **Ensure data integrity & quality** before analysis.
- ⚠️ **Detect potential data issues (missing values, duplicates, errors).**



OLIST DATASET OVERVIEW: UNDERSTANDING THE DATA STRUCTURE

🔍 Key Details About the Dataset

✅ **Total Number of Tables:** 9 (covering various aspects of e-commerce).
✅ **Project Focus:** This project explores the orders table, ensuring data integrity before further analysis.
✅ **Dataset Context:** The Olist dataset represents real-world **Brazilian e-commerce transactions,** containing multiple relational tables.

🛒 Key Table Used in This Project

📦 **olist_orders_dataset** → Contains customer order details, timestamps, and transaction statuses.

💡 *Other tables such as products, customers, and sellers provide additional details but were not explored in this project.*

DATA EXPLORATION – OBJECTIVE & STEPS

🛠️ OBJECTIVE OF DATA EXPLORATION

- 💡 **Why explore the data?**

  ✅ Ensure **data integrity** before analysis.

  ✅ Identify **duplicates, missing values, and inconsistencies.**

  ✅ Validate **relationships between key columns.**

- 📌 **Key Steps in Data Exploration:**

  1️⃣ **Inspect Schema & Structure** – Understand column types & table structure.

  2️⃣ **Validate Primary Keys** – Ensure uniqueness of key identifiers.

  3️⃣ **Check Foreign Keys** – Confirm referential integrity between tables.

  4️⃣ **Identify Duplicates** – Detect and handle duplicate records.

  5️⃣ **Handle Missing Data** – Check for NULL values in key columns.

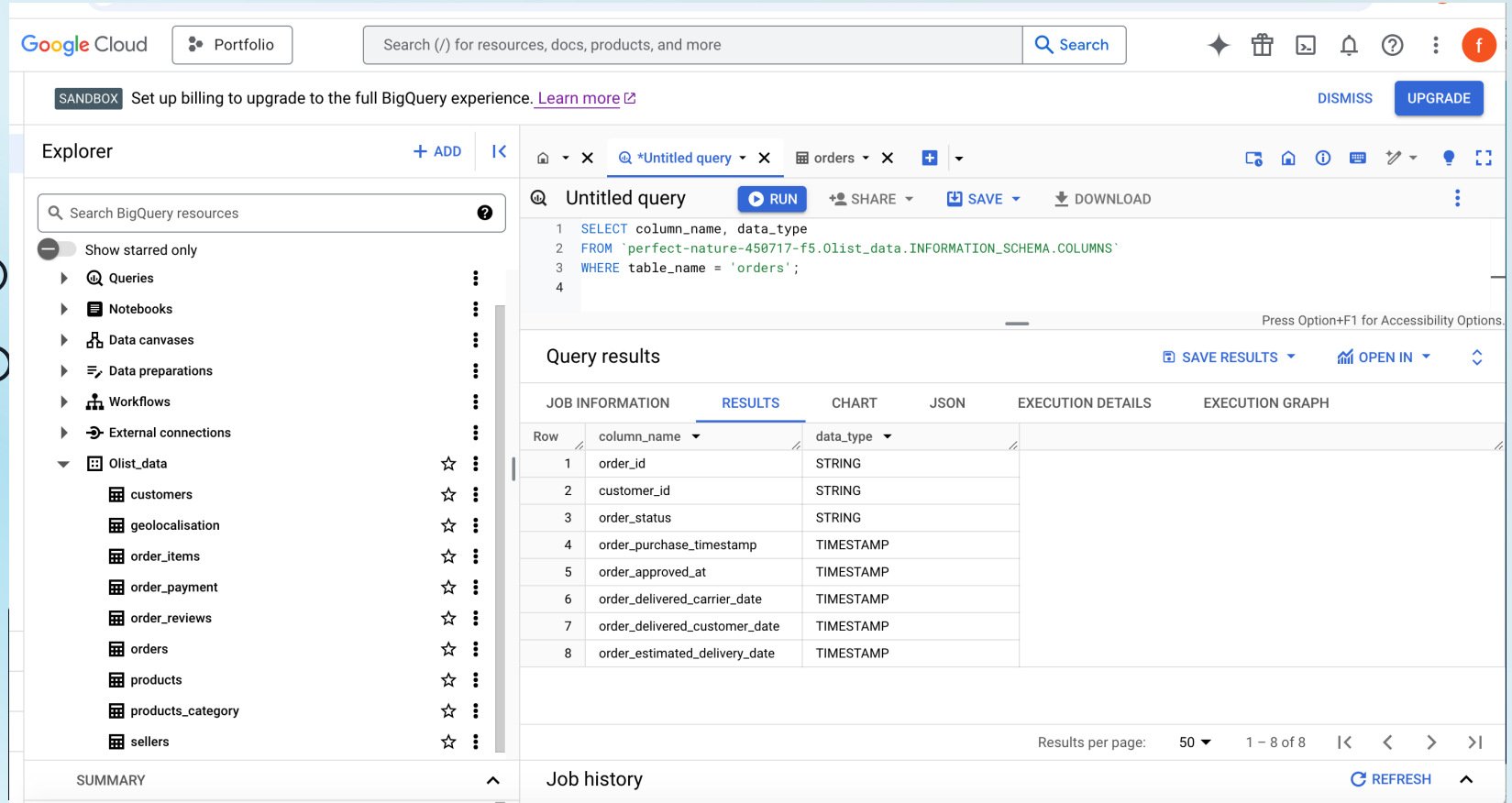
STEP 1: INSPECTING TABLE SCHEMA & UNDERSTANDING DATA TYPES

✅ **Why?**

•"Schema" alone may not be clear for all audiences; **mentioning "data types" adds clarity**.

✅ **Objective:** Identify column names, data types, and table structure for better data understanding.

```sql
SELECT column_name, data_type
FROM `perfect-nature-450717-f5.Olist_data.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'orders';
```



✅ *The orders table is a key dataset in the Olist e-commerce database.*

This step inspects its schema to identify column names, data types, and

potential data quality issues before analysis.”

📊 **Key Findings: Schema & Structure Overview**

✅ **Total Columns:** 8

✅ **Primary Identifiers:**

•order_id, customer_id (**Both are STRING data types**)

✅ **Timestamps Available:**

•order_purchase_timestamp

•order_approved_at

•order_delivered_carrier_date *(and more)*

✅ **Potential Data Cleaning Needs:**

•Check for **NULL values** in date fields.

📌 Step 2: Identifying Primary Keys in the orders Table

✅ **Objective:**

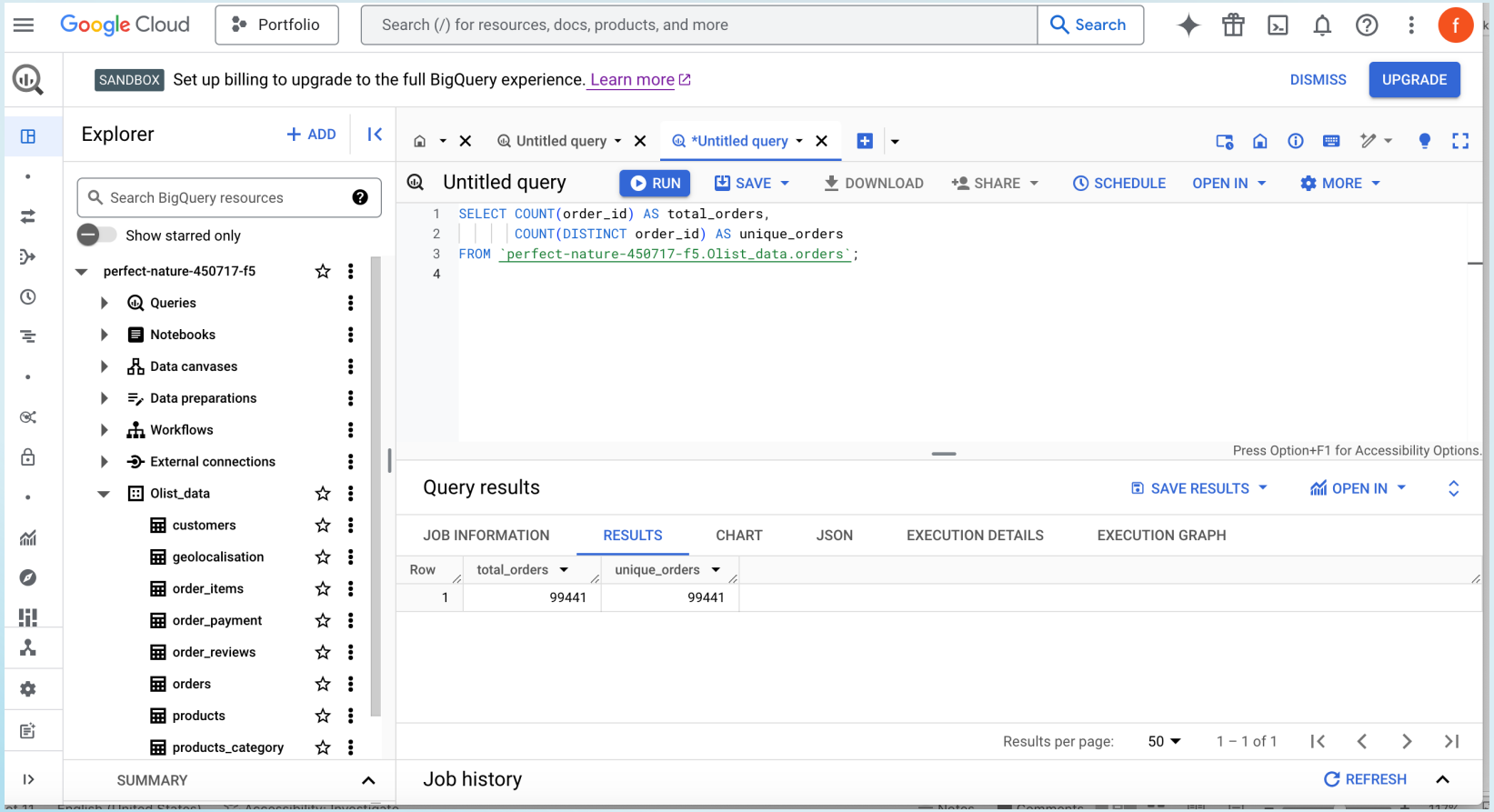
•Verify if order_id is a **Primary Key** (i.e., each order_id is unique).

•Ensure there are **no duplicate orders** in the dataset.

•Confirm order_id can be used for **joins with other tables.**

Objectif:Verify if order_id is a **Primary Key** (i.e., each order_id is unique).

```
SELECT COUNT(order_id) AS total_orders,
COUNT(DISTINCT order_id) AS unique_orders
FROM `perfect-nature-450717-f5.Olist_data.orders`;
```



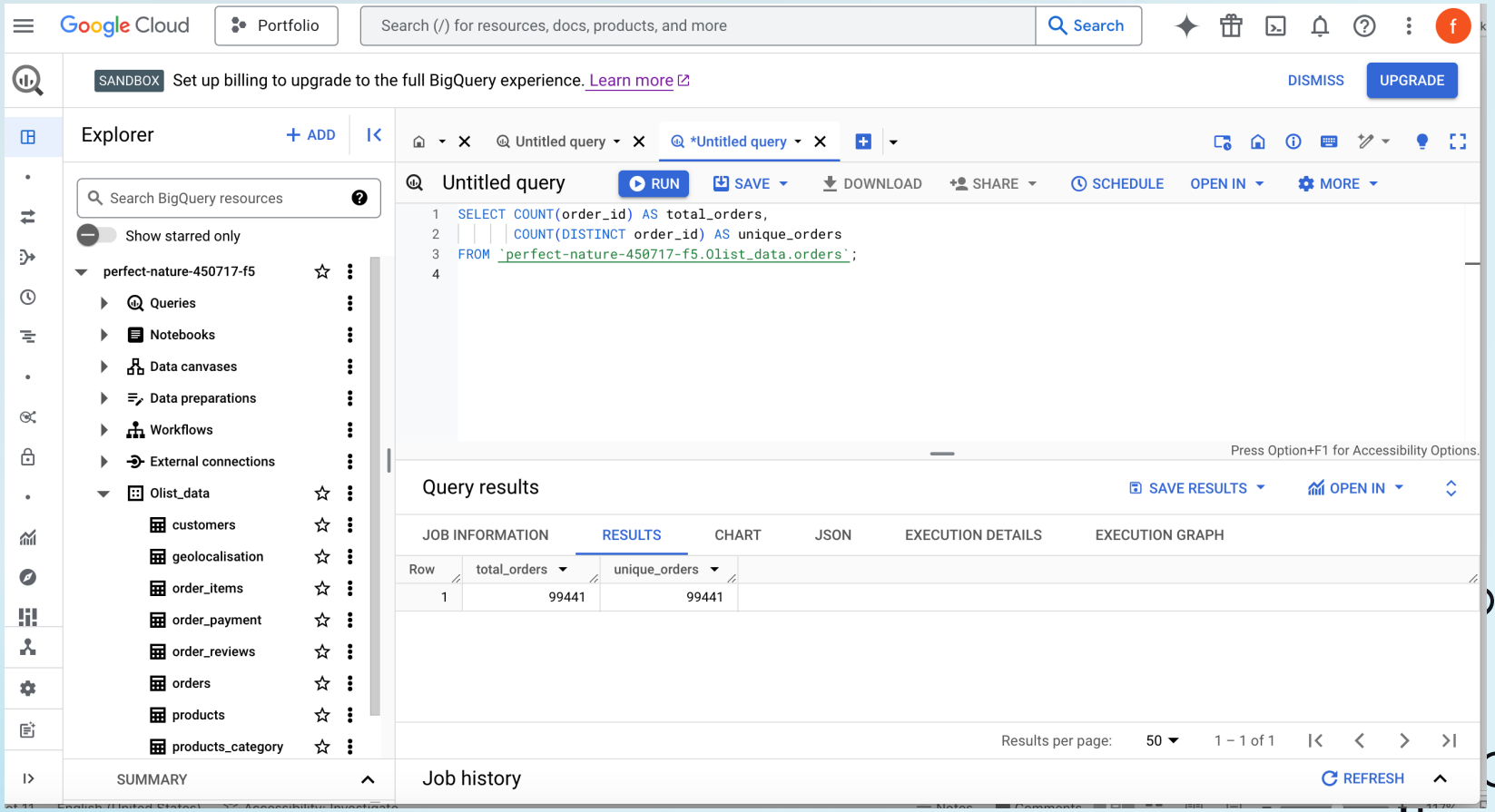
🏆 **Key Findings (Bullet Points):**

✅ **Total Orders: 99,441**

✅ **Unique Orders: 99,441**

✅ **Primary Key Status:** ✅ **Valid** (No duplicate order IDs found)

📌 *This confirms that order_id is unique and can be used as a primary key for further analysis.*



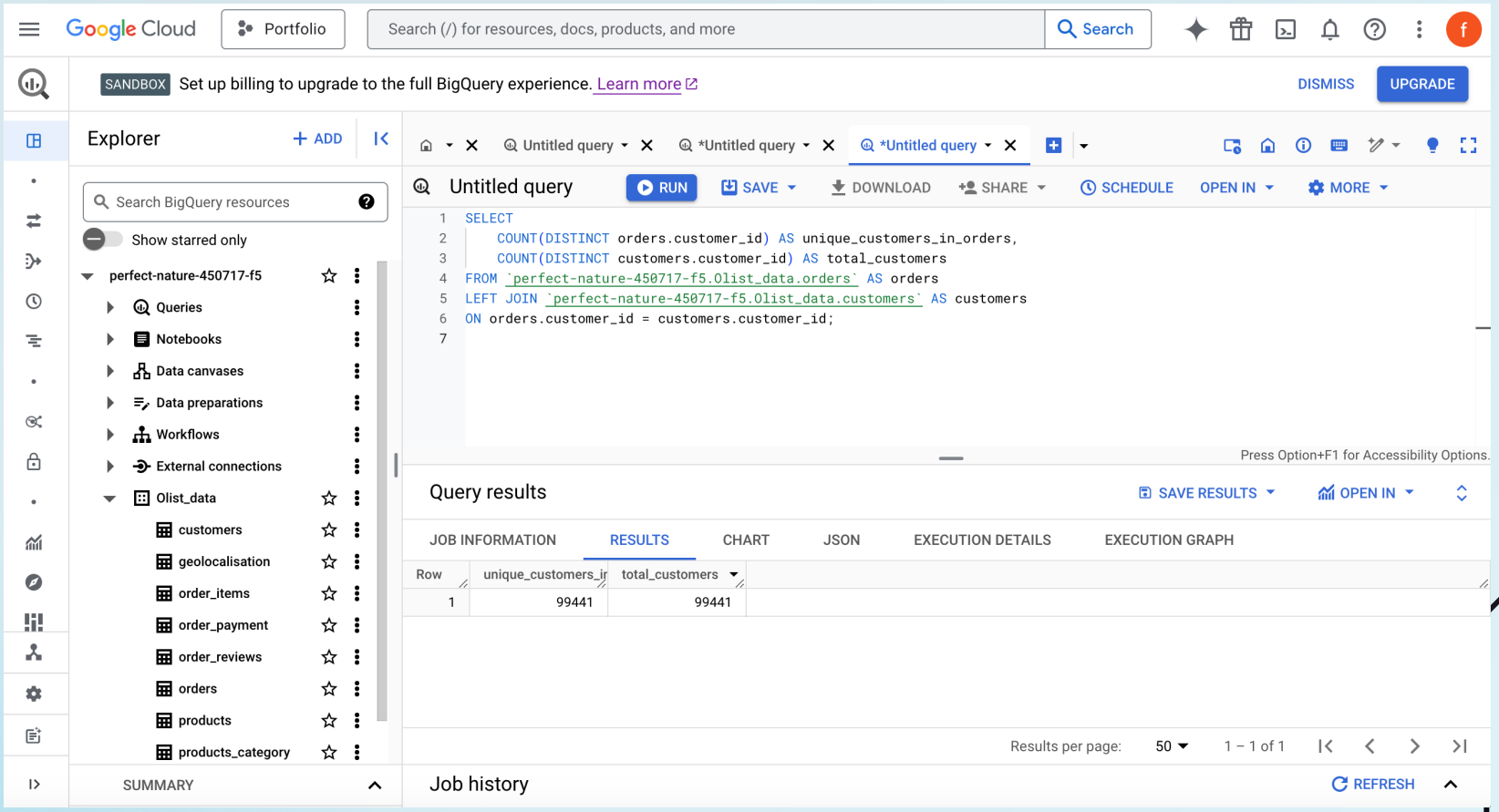
📌 Step 3: Foreign Key Validation (customer_id in orders vs. customers)

✅ **Objective:**

- Ensure that every customer_id in the orders table **exists** in the customers table.
- Validate **referential integrity** for correct **table relationships.**
- Identify any **missing or orphaned customer IDs.**

```sql
SELECT
COUNT(DISTINCT orders.customer_id) AS
unique_customers_in_orders,
COUNT(DISTINCT customers.customer_id) AS total_customers
FROM `perfect-nature-450717-f5.Olist_data.orders` AS
orders
LEFT JOIN `perfect-nature-450717-f5.Olist_data.customers`
AS customers
ON orders.customer_id = customers.customer_id;
```

✅ **Unique Customers in Orders: 99,441**
✅ **Total Customers in Dataset: 99,441**
✅ **Foreign Key Status:** ✅ **Valid** (No missing customer_id values)
📌 *This confirms that every customer_id in the orders table exists in the customers table, ensuring accurate relationships.*



📌 Step 4: Checking for Duplicates in orders Table

✅ **Objective** *The goal of this step is to ensure that each order_id in the orders table is unique and there are no duplicate records. Duplicate orders can lead to incorrect analysis, inflated sales numbers, and misleading insights. Detecting and removing them is crucial for maintaining data integrity.*

```sql
SELECT order_id, COUNT(*) AS order_count
FROM `perfect-nature-450717-f5.Olist_data.orders`
GROUP BY order_id
HAVING COUNT(*) > 1;
```

This step verifies whether duplicate order_id values exist in the orders table. Duplicates can cause inaccurate sales calculations, reporting errors, and inconsistencies in data analysis

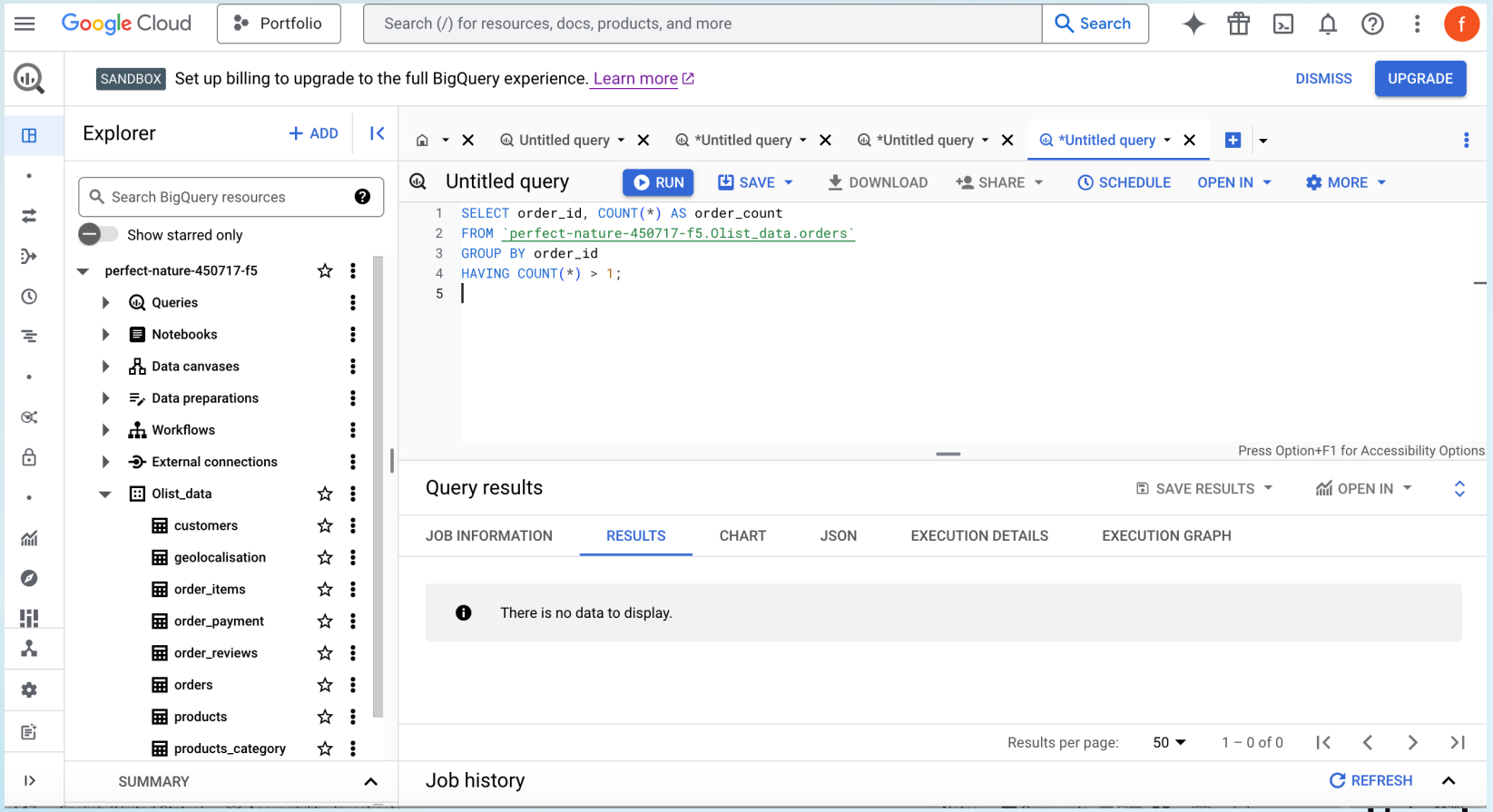
🏆 **Key Findings:**

✅ **No duplicate order_id values found** (Query returned zero results).

✅ The order_id column is **unique,** confirming its validity as a **Primary Key.**

✅ No data cleaning is required for duplicate orders.

📌 *This ensures that each order is recorded only once, preventing potential double counting in further analysis.*



📌 Step 5: Checking for Missing Data in the orders Table

🛠️ **Objective:**

"The goal of this step is to identify missing (NULL) values in key columns. Missing data can impact analysis, lead to incomplete reports, and cause errors in business decision-making. Detecting and handling them ensures accurate insights

```sql
SELECT
COUNTIF(order_id IS NULL) AS missing_order_id,
COUNTIF(customer_id IS NULL) AS missing_customer_id,
COUNTIF(order_status IS NULL) AS missing_order_status,
COUNTIF(order_purchase_timestamp IS NULL) AS
missing_order_purchase_date
FROM `perfect-nature-450717-f5.Olist_data.orders`;
```

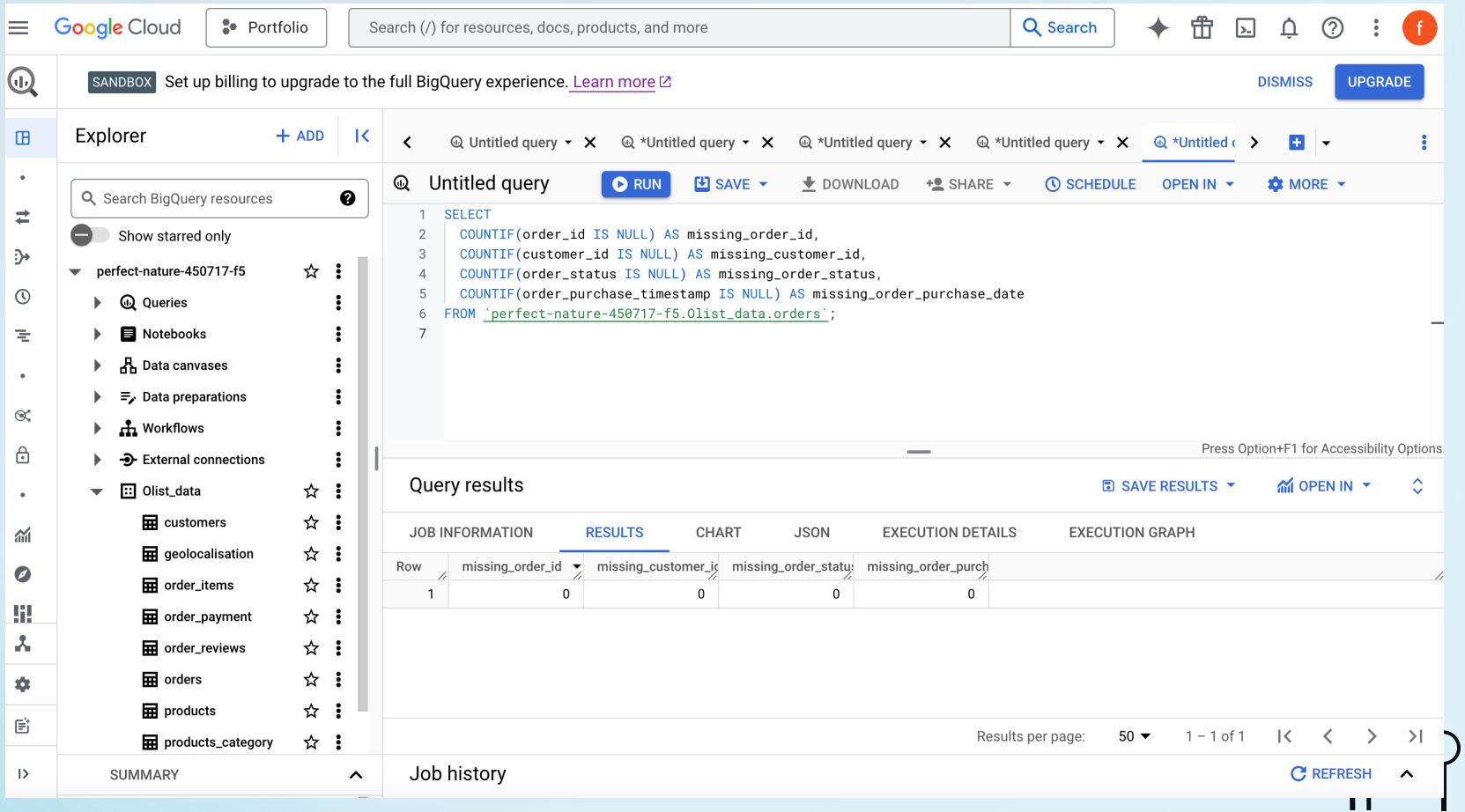
🏆 Key Findings

✅ **No missing values detected** in key columns.

✅ **Data is complete,** ensuring accuracy for further analysis.

✅ No immediate need for data cleaning related to missing values.

📌 *This confirms that all essential fields (order_id, customer_id, order_status, order_purchase_timestamp) are properly recorded, maintaining data integrity.*



SUMMARY INSIGHTS

Key Insights from Data Exploration

🔍*This data exploration process validated the integrity, completeness, and structure of the orders table in the Olist dataset. The results ensure that the dataset is clean, well-structured, and ready for deeper analysis or visualization.*

🏆 **Key Findings:**

✅ **Schema Inspection:** The orders table contains **8 key columns,** including order_id, customer_id, and multiple timestamps.

✅ **Primary Key Check:** order_id is **unique and valid,** confirming its role as the **primary key.**

✅ **Foreign Key Validation:** customer_id exists in both orders and customers tables, ensuring **referential integrity.**

✅ **Duplicate Check: No duplicate order IDs found,** confirming **data consistency.**

✅ **Missing Data Check: No missing values** in key columns, ensuring **data completeness.**

📌 *These results confirm that the dataset is reliable for further analysis, such as customer behavior insights, order fulfillment trends, and sales analysis.*

🔍 Step	🎯 Key Findings	✅ Insights
1️⃣ **Schema Inspection**	The orders table contains **8 columns,** including order_id, customer_id, order_status, and multiple timestamps.	The dataset is structured well, making it ready for further analysis.
2️⃣ **Primary Key Validation**	order_id is unique (**99,441 total orders = 99,441 unique orders**).	✅ order_id is a **valid Primary Key,** ensuring integrity.
3️⃣ **Foreign Key Validation**	Every customer_id in orders exists in customers (**99,441 matches 99,441**).	✅ **No missing customer records,** ensuring relational integrity.
4️⃣ **Duplicate Check**	**No duplicate orders** found in the dataset.	✅ Data is **clean & reliable,** preventing double-counting issues.
5️⃣ **Missing Data Check**	**No missing values** in key columns (order_id, customer_id, order_status, order_purchase_timestamp).	✅ **Complete dataset,** no need for data imputation.

CONCLUSION

This project showcases my ability to conduct structured SQL-based data exploration using BigQuery. The clean dataset can now be leveraged for further analysis, such as customer segmentation, sales trends, and dashboard visualization

LET'S CONNECT & COLLABORATE! 🚀

Looking for a data analyst who can transform raw data into actionable insights?

Let's connect!

📩 **Ways to Reach Me:**

- 💼 **LinkedIn:** https://www.linkedin.com/in/dr-fatima-kabba/
- 📂 **GitHub:** https://github.com/FatimaKabba
- 👋 **Tableau:https://public.tableau.com/app/profile/kabba.fatima.ezzahra/vizzes**
- 📧 **Email:** *fatima.ezzahra.kabba.pro@gmail.com*

📌 *I'm open to opportunities in **data analysis, business intelligence, and freelancing projects.***

Let's discuss how I can add value to your team!