A logo of a university

Description automatically generated

Business intelligence and Database Management System

Business Intelligence Project on

Golden Bloom Beans

Prepared by:

Tasnim Ouaer

Yassine Antit

Mohamed Aziz Khezami

Contents

1.Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

1.1.Objectives

1.2.Goals

1.3 Deliverables

2. Implementation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.1 Data Gathering . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.2 Data Preparation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.3 Data Storage . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

2.3.1 Storage . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

2.3.2 Facts & Measures. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

2.3.3 Dimensions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

2.4 Data Visualization . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

2.5 Machine learning Forecasting and prediction. . . . . . . . . . . . . . 11

3. Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

**1.Introduction**

Golden Bloom Beans, one of the leading coffee industry producers, uses Business Intelligence (BI) tools to analyze the performance of sales trends; buying behaviors of customers; and products. The project highlights required insight and strategic recommendations that will improve revenues and increase retention of customers while optimizing operation and addressing seasonal trends and loyalty programs.

**1.1. Objectives**

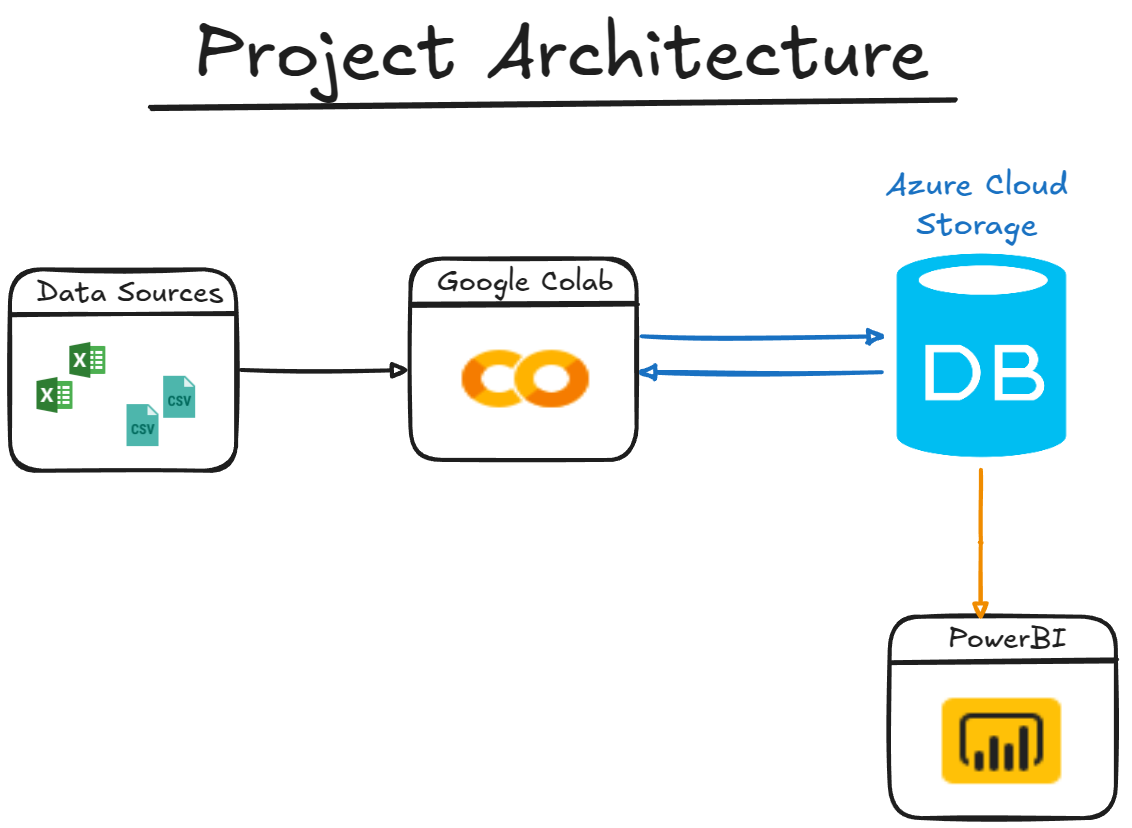
1. Identify the coffee products that generate the highest revenue and profit.
2. Analyze customer purchasing behavior by region, city, and loyalty status.
3. Detect trends in order frequency and quantity over time.
4. Develop a forecasting model for future sales and demand patterns

**1.3. Goals**

* Increase Revenue: Identify the most profitable products and focus on promoting them.
* Enhance Customer Retention: Analyze loyalty program data to improve customer satisfaction.
* Optimize Inventory Management: Understand product demand patterns to ensure efficient stock levels.
* Expanding the company's presence in the market.

**1.4. Deliverables**

* An interactive Power BI Dashboard.
* A comprehensive Recommendations Report.
* A machine learning model for sales forecasting.

**2. Implementations**

**2.1 Data Gathering**

We extracted our data from Kaggle datasets. All the raw data is stored in the Raw Data folder.

**2.2. Data Preparation**

For the Golden Bloom Beans project, we utilized Python (Google Collab) to manipulate data and prepare it for integration into our data warehouse.

Using Python's panda’s library and, we executed necessary transformations as part of the ETL process. We developed a Python script to check for invalid or not available information, as well as to verify the data types we were dealing with. This step ensured data quality and consistency before proceeding with further transformations.

|  |  |
| --- | --- |
| MDX | SQL |
| **1. Total Revenue by Coffee Type** SELECT     {[Measures].[Total Revenue]} ON COLUMNS,     {[Coffee Products].[Coffee Type].Children} ON ROWS  FROM [Coffee Orders] **2. Top 5 Products by Profit**SELECT   {[Measures].[Total Profit]} ON COLUMNS,   TOPCOUNT(       [Coffee Products].[Product ID].Members,       5,       [Measures].[Total Profit]   ) ON ROWSFROM [Coffee Orders]**3. Quantity Sold for Each Roast Type** SELECT     {[Measures].[Quantity]} ON COLUMNS,     {[Coffee Products].[Roast Type].Children} ON ROWS  FROM [Coffee Orders]   **4. Revenue and Profit for Loyalty Card Holders** SELECT     {[Measures].[Total Revenue], [Measures].[Total Profit]} ON COLUMNS,     {[Coffee Customers].[Loyalty Card].[True]} ON ROWS  FROM [Coffee Orders] | SELECT     p.coffee\_type,     SUM(o.quantity \* p.unit\_price) AS total\_revenue  FROM     orders o  JOIN     products p ON o.product\_id = p.product\_id  GROUP BY     p.coffee\_type  ORDER BY     total\_revenue DESC;    SELECT     p.product\_id,     p.coffee\_type,     SUM(o.quantity \* p.profit) AS total\_profit  FROM     orders o  JOIN     products p ON o.product\_id = p.product\_id  GROUP BY     p.product\_id, p.coffee\_type  ORDER BY     total\_profit DESC  LIMIT 5;  SELECT     p.roast\_type,     SUM(o.quantity) AS total\_quantity  FROM     orders o  JOIN     products p ON o.product\_id = p.product\_id  GROUP BY     p.roast\_type  ORDER BY     total\_quantity DESC;  SELECT     c.loyalty\_card,     SUM(o.quantity \* p.unit\_price) AS total\_revenue,     SUM(o.quantity \* p.profit) AS total\_profit  FROM     orders o  JOIN     customers c ON o.customer\_id = c.customer\_id  JOIN     products p ON o.product\_id = p.product\_id  WHERE     c.loyalty\_card = TRUE  GROUP BY     c.loyalty\_card; |

We dropped irrelevant columns by using the **SUM** function in order to identify the total number of missing values in the dataset. We also handled missing values by utilizing the **FILLNA** function, ensuring the dataset is more complete and better prepared. Finally, to ensure sequential and clean indexing; we reset the indexes by using the **RESET\_INDEX** function.

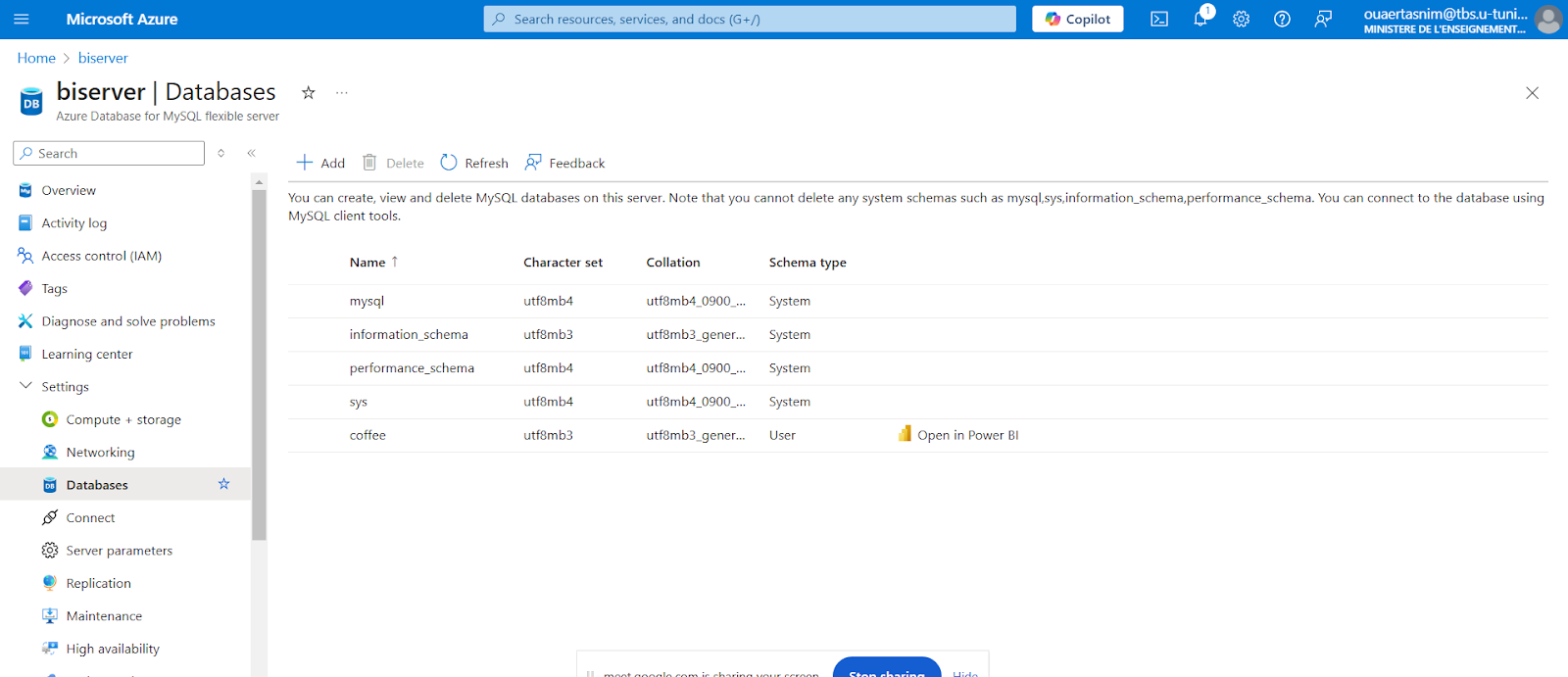
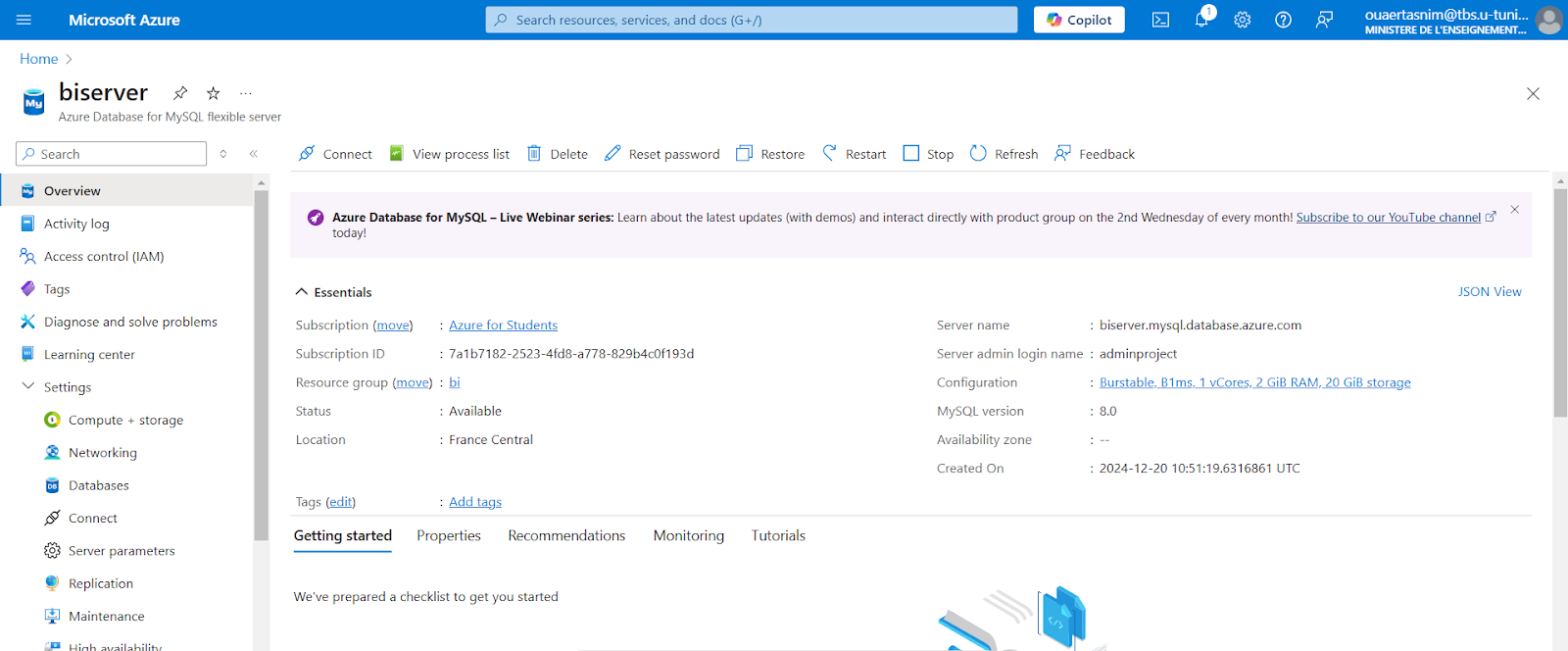
**2.3. Data Storage**

**2.3.1. Storage**

we insert data from three pandas DataFrames into MySQL tables (**customers**, **products** and **orders**)

We use a loop to execute INSERT queries for each row of the DataFrames, transferring the data into the respective MySQL tables. After inserting all rows, we commit the changes to the database and print the number of rows successfully inserted.

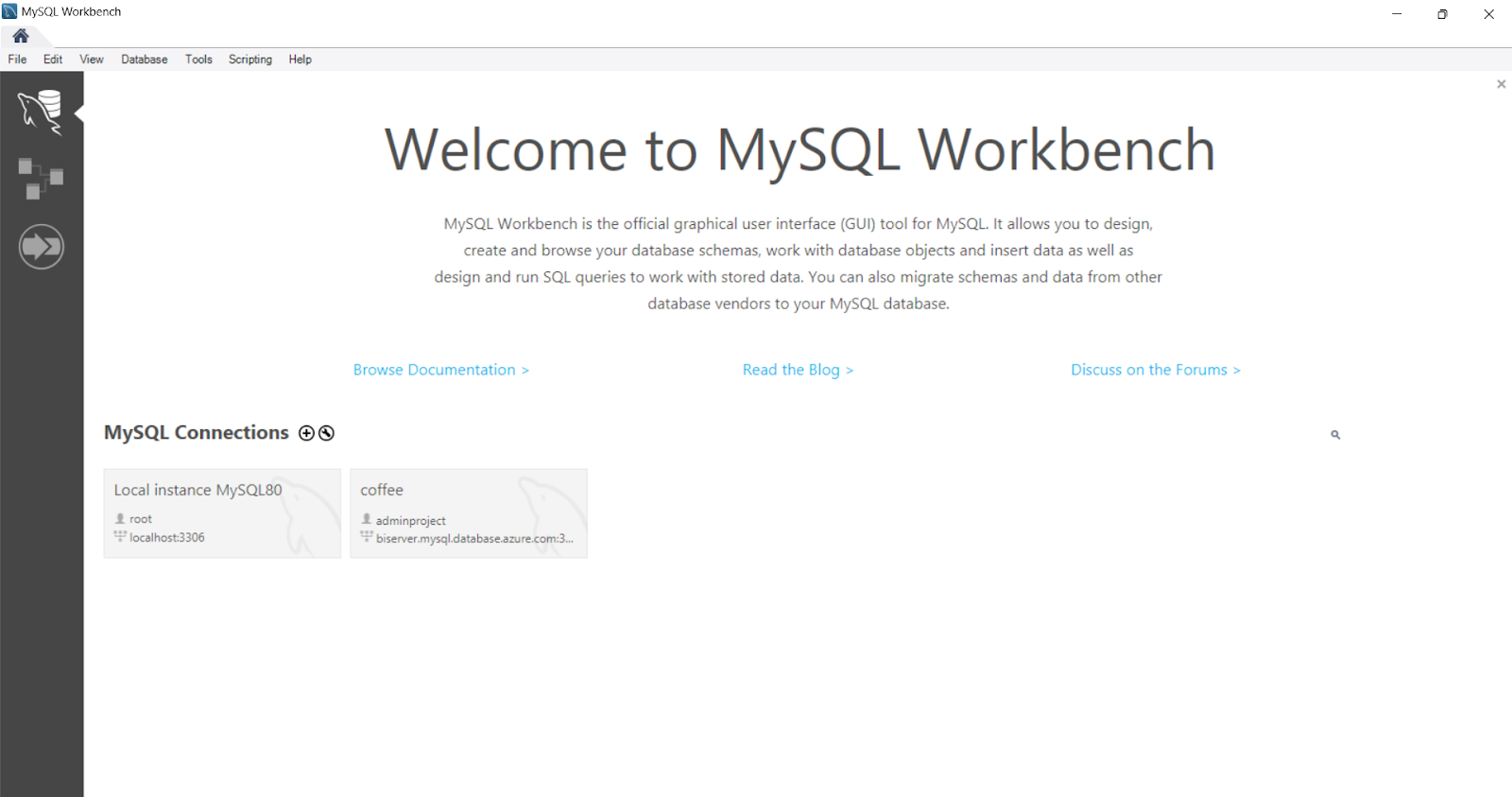
We created a Data Warehouse Server using Azure Cloud Service.



**2.3.2. Facts & Measures**

1. **Facts:**

* Coffee Orders

****

1. **Measures:**

* Total Profit
* Total Revenue
* Quantity
* Loyalty Card Holders
* Number of Inactive Customers

**2.3.3. Dimensions**

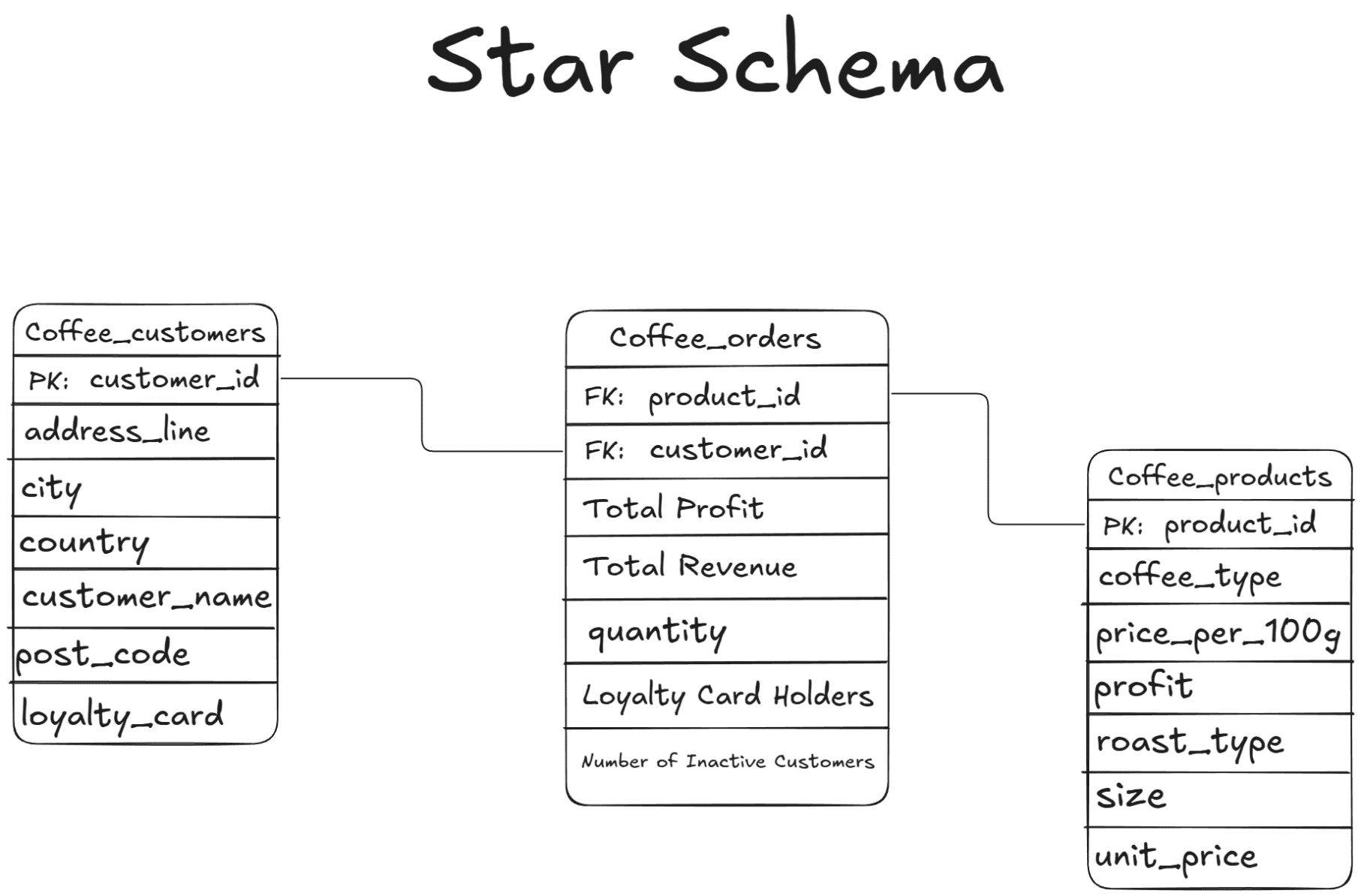
1. **Coffee Products**

**Attributes:** product\_id(pk)/coffee\_type/price\_per\_100g/profit/roast\_type/size/ unit\_price

1. **Coffee Customers**

**Attributes:**

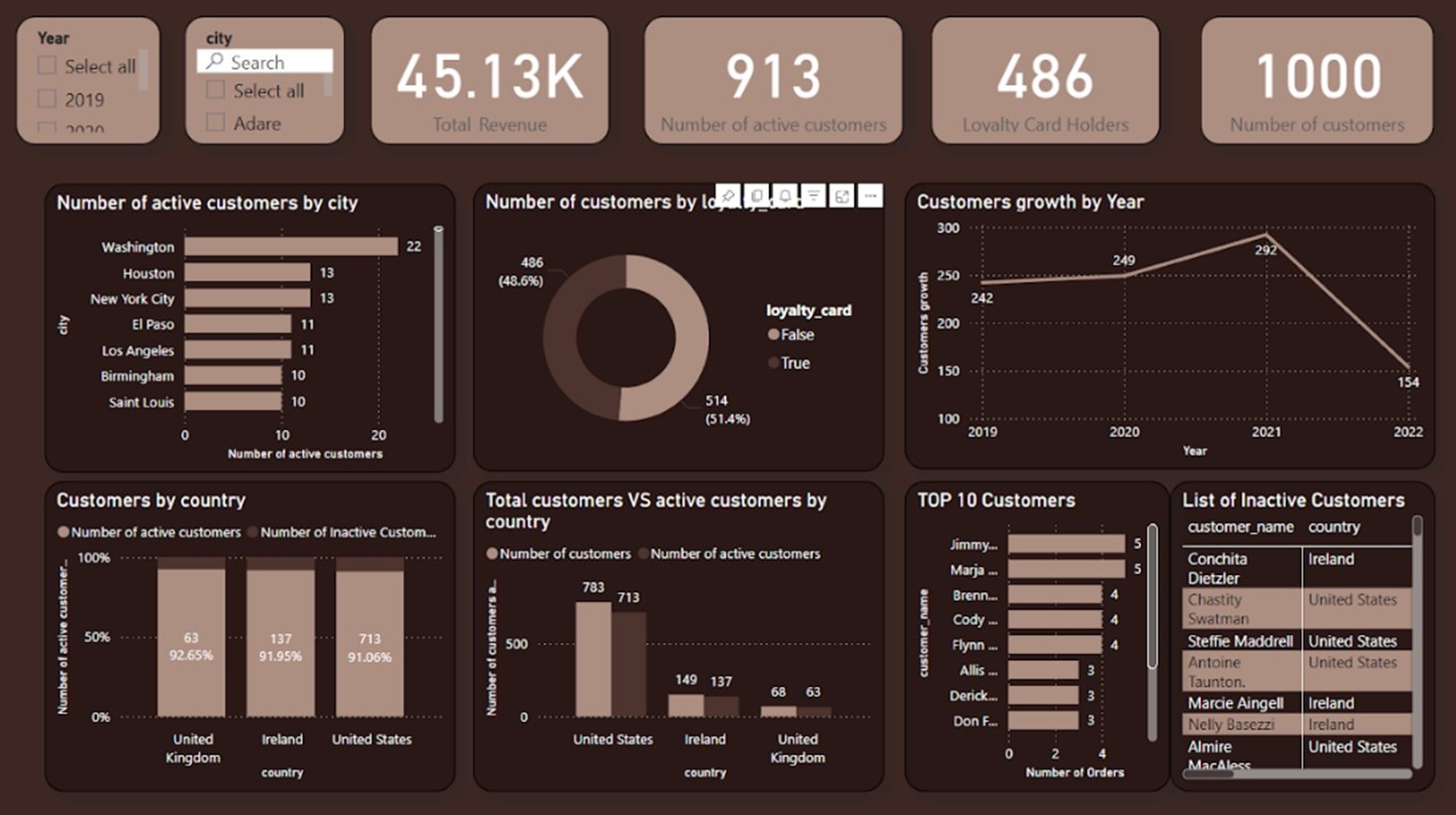
customer\_id(pk)/address\_line/city/country/customer\_name/loyalty\_card/postcode

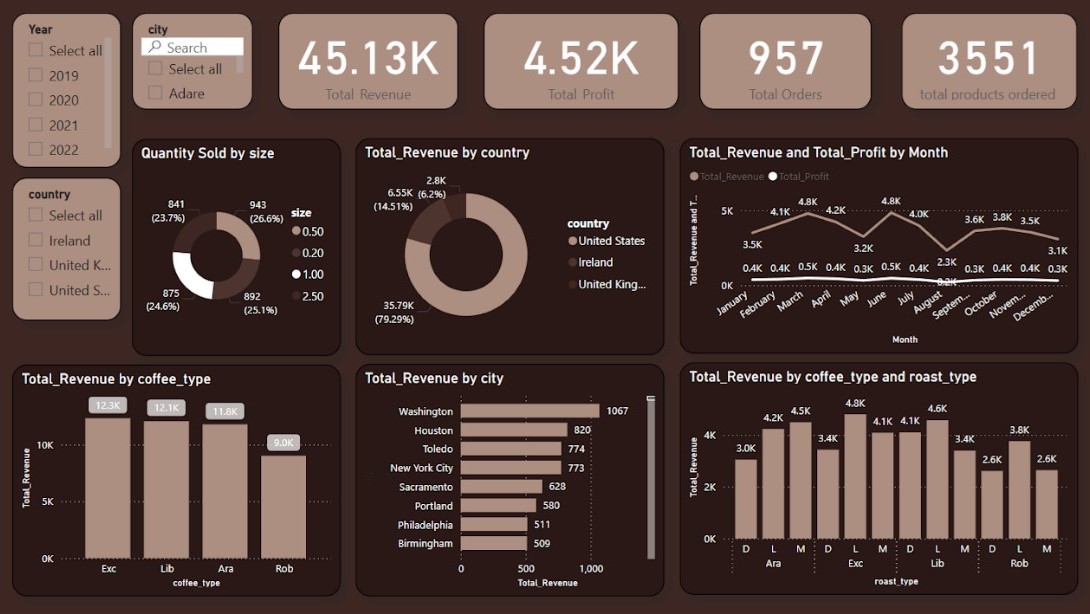


**2.4. Data Visualization**

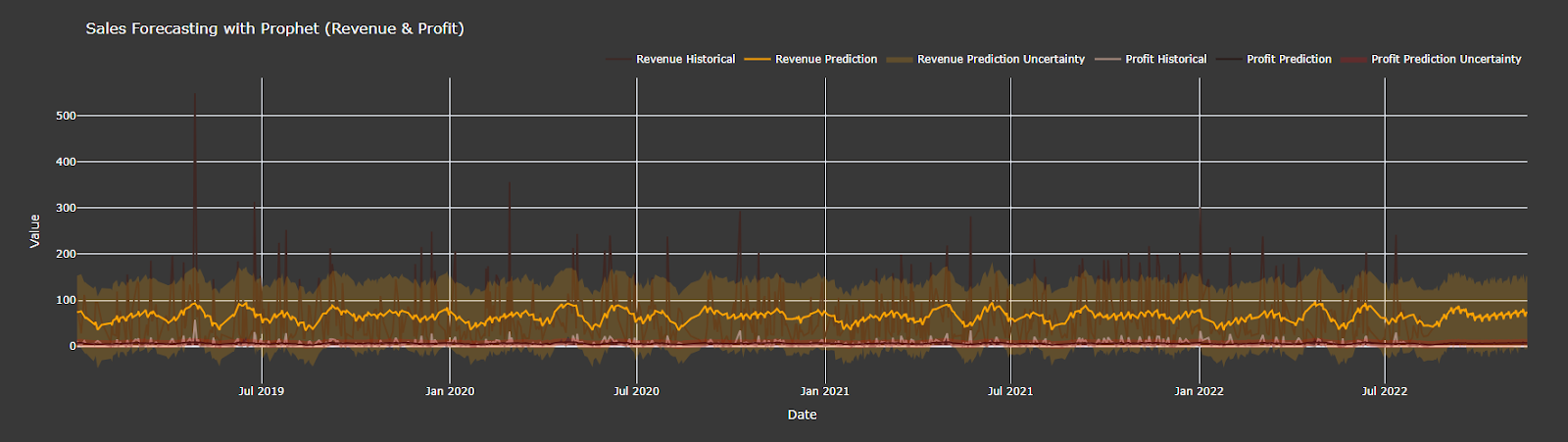
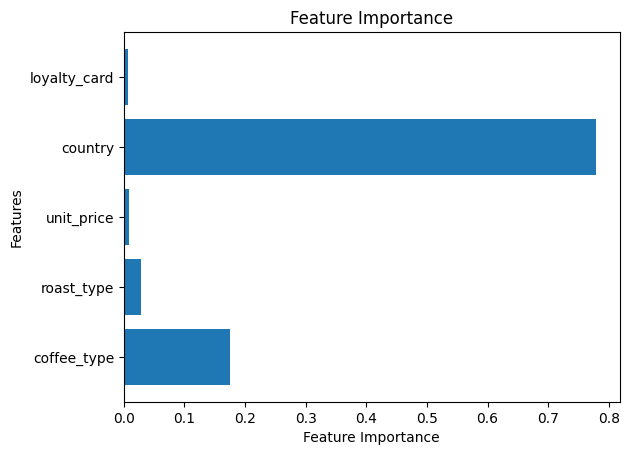
We used Power BI to analyze customer behavior and satisfaction with Golden Bloom Beans offerings. This provided insights into areas and channels Golden Bloom Beans can improve to attract and better serve customers.

[Power BI Dashboard Link](https://app.powerbi.com/view?r=eyJrIjoiYTNjMjVkZmEtMDE5OC00NjUzLWIxODMtODg4ZjdkMDA1MzhmIiwidCI6ImRiZDY2NjRkLTRlYjktNDZlYi05OWQ4LTVjNDNiYTE1M2M2MSIsImMiOjl9)



****

**2.5. Machine learning Forecasting and prediction**

This code leverages **Prophet** and **Plotly** to forecast and visualize revenue and profit trends over time. The datasets for revenue and profit are prepared by renaming columns to ds (date) and y (target variable) and converting dates to a datetime format. Separate Prophet models are then trained for revenue and profit, and future dataframes are created to generate 90-day forecasts, including predicted values (yhat) and confidence intervals (yhat\_upper and yhat\_lower). Using Plotly, historical data is visualized as line charts, while predictions are plotted with separate lines and shaded confidence intervals to reflect uncertainty. The visualization features a dark theme with custom colors and hover templates, presenting a clear and professional overview of historical performance and future forecasts.

**Expected Profit From customer**

This process involves merging three datasets into a single table to ensure all relevant information is consolidated. Next, the total profit for each order is calculated, providing a clear measure of individual order performance. Multiple regression models are then tested to analyze the impact of various features on the expected customer profit, enabling the identification of key drivers of profitability. Finally, the selected regression model is integrated into the workflow for practical application and decision-making.

**Conclusion**:

1. Top Products:
   * Excelsa generated the highest revenue, led by Excelsa Light Roast (12.3k), followed by Liberica Light Roast (12.1k).
   * Most popular size: 0.5Kg (26.6%), followed by 0.2Kg (21.1%).
2. Customer Behavior:
   * 51.4% of customers use loyalty cards.
   * Active customers account for 91.3% (713 in the U.S., notably in Washington, Houston, and New York).
3. Revenue Trends:
   * The U.S. drives 79.29% of revenue, followed by Ireland (14.51%).
   * Peak revenue and profit occur in June (4.8k); lowest in December (3.1k).
4. Customer Trends:
   * Customer count peaked in 2021 (292) and dropped significantly in 2022 (154).

**Strategic Recommendations**

1. Boost Revenue and Margins:
   * Promote high-performing coffee types (Excelsa, Liberica) and sizes (0.5Kg, 0.2Kg).
   * Reevaluate underperforming products like Robusta.
2. Enhance Customer Retention:
   * Reconstruct the loyalty programs to reengage inactive customers, particularly in key U.S. regions.
3. Expand Market Reach:
   * Strengthen presence in The US and explore growth opportunities in the United Kingdom, Ireland and possibly other parts in Europe.
4. Optimize Seasonal Performance:
   * Develop targeted campaigns to increase sales during off-peak months, especially December.
5. Inventory Management:
   * Focus on maintaining higher inventory space for high performing products such as Excelsa and Liberica and prioritize light roasts in order to avoid stockouts.
   * Increase inventory levels to meet sales peak in June and Reduce excess inventory in December to avoid overstocking.

**Project code**

[Notebook.ipynb](https://colab.research.google.com/drive/1L46o88hhM5JEGelktbuB7CEHoL0un_XH#scrollTo=Kuht-mG_r6yt)

[Github](https://github.com/MohamedAziz-Khezami/Coffee_Beans_Analysis)

[Power BI Dashboard](https://drive.google.com/file/d/1jRgWd44P1_xAy7vZ-MsBeyprrPR6f8On/view?usp=sharing)