Olist Ecommerce Analytics: User Guide

1. Introduction

This project is about an online shopping platform. We got the data of Olist, an e-commerce platform in Brazil from Kaggle, and used part of the data to create our own database based on our needs. Our project could be used to analyze sales performance and manage the platform data.

1. Database

2.1 Tables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Olist\_customers\_dataset table | |  | Olist\_orders\_dataset table | |
| Primary Key | customer\_id (ID generated per order) |  | Primary Key | order\_id (unique per order) |
| customer\_unique\_id | unique ID for every customer |  | Foreign Key | customer\_id (links to Olist\_customers\_dataset) |
| Other variables | customer location information |  | Other variables | order status, timestamps for various stages, and estimated delivery dates. |
| Olist\_products\_dataset table | |  | Olist\_order\_items\_dataset table | |
| Primary Key | product\_id |  | Foreign Key | order\_id (links to Olist\_orders\_dataset) |
| Other variables | product descriptions |  | Foreign Key | product\_id (links to Olist\_products\_dataset) |
|  |  |  | Other variables | order\_item\_id (item sequence in an order), seller IDs, shipping limits, prices, and freight values. |

A screenshot of a computer

Description automatically generated

2.2 Queries

(1) Select TopX product by sales quantity

Merge [Olist\_products\_dataset], [Olist\_order\_items\_dataset] and [Olist\_orders\_dataset].Filter for delivered orders, count sales per product, sort descending, and display the top X results.

A close-up of a computer code

Description automatically generated(2) Select TopX product by sales revenue

Similar to the above, but calculate total sales revenue per product (excluding freight value) and display top X results.

A screen shot of a computer code

Description automatically generated

1. Sales by year and month

Merge [Olist\_order\_items\_dataset] and [Olist\_orders\_dataset]. Filter for delivered orders and exclude records with null delivery dates. Group by year and month based on delivery date.

A close-up of text

Description automatically generated

1. Check customer order

Merge [Olist\_orders\_dataset] and [Olist\_customers\_dataset] and only display the related records of a specific customer.

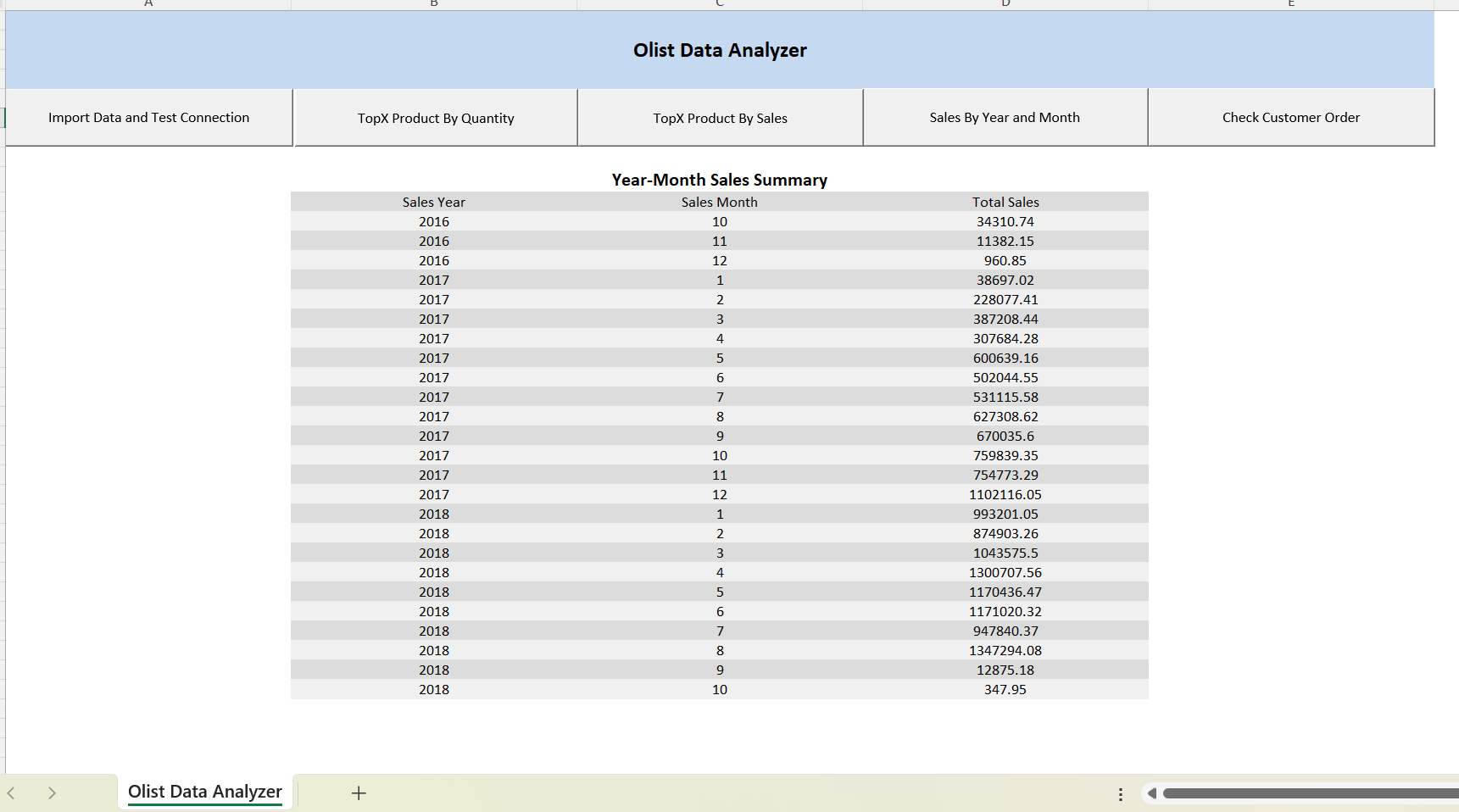
A screen shot of a computer code

Description automatically generated

1. Front-end

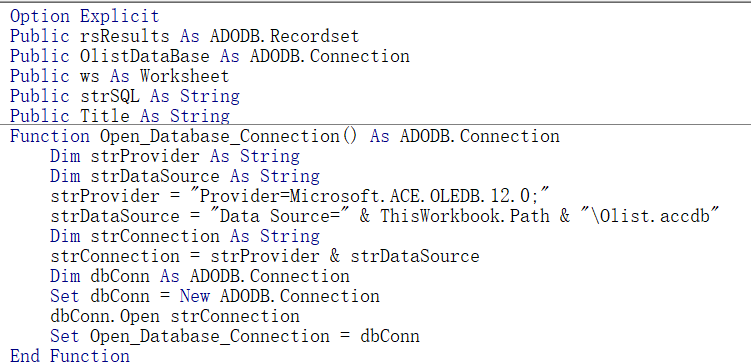
On front-end we have one sheet as our home page. All the functions can be called by clicking on the tabs, the corresponding result will be shown on the same sheet.

1. Import data and test connection: Establishes a connection between Excel VBA and the Access database. Displays success/failure message.
2. TopX product by sales quantity: Prompts user to input a number. Displays top X products by sales quantity. Invalid inputs trigger a warning.
3. TopX product by sales revenue: Prompts user to input a number. Displays top X products by sales revenue. Invalid inputs trigger a warning.
4. Sales by year and month: Displays a pivot table showing total sales by year and month.
5. Check customer order: Prompts user to input a customer\_unique\_id. Displays orders related to the given ID. Invalid inputs trigger a warning.



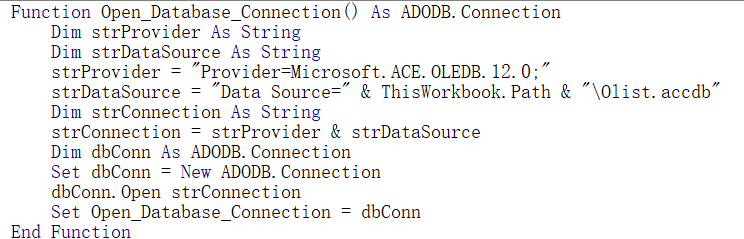
1. VBA Middleware
   1. **Database Connection**

We define some public variables in order to use them in several subroutines.Then we define a function to open database connection to our Access database Olist.

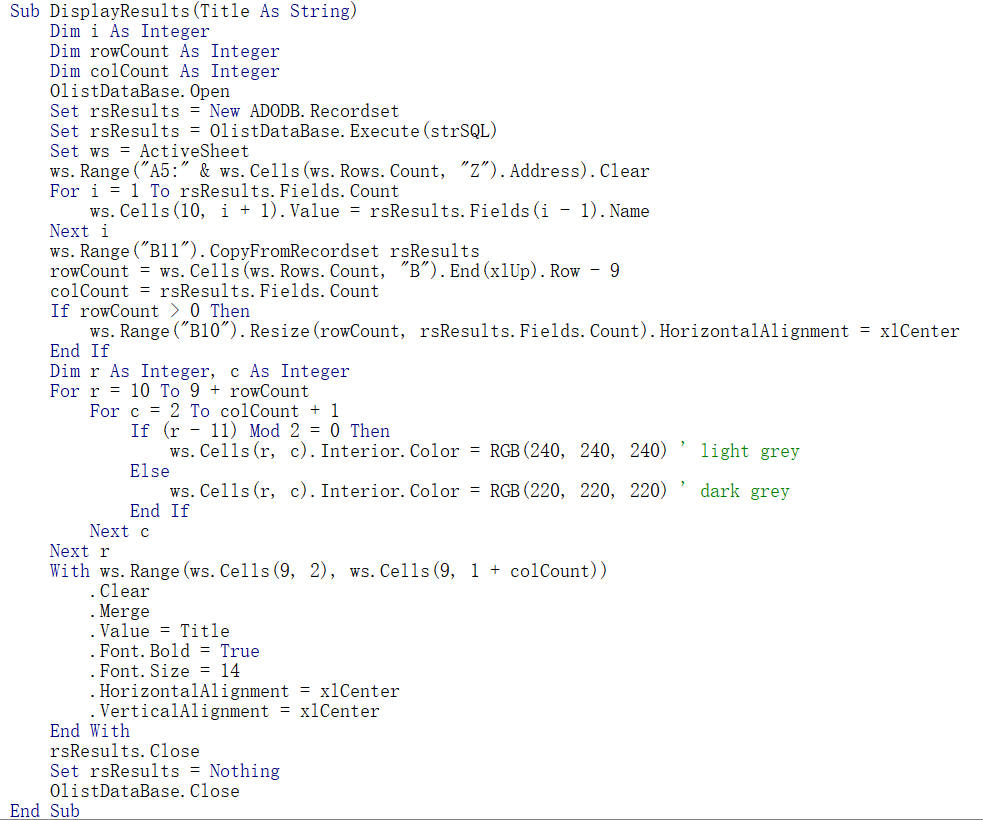


* 1. **Query Execution**

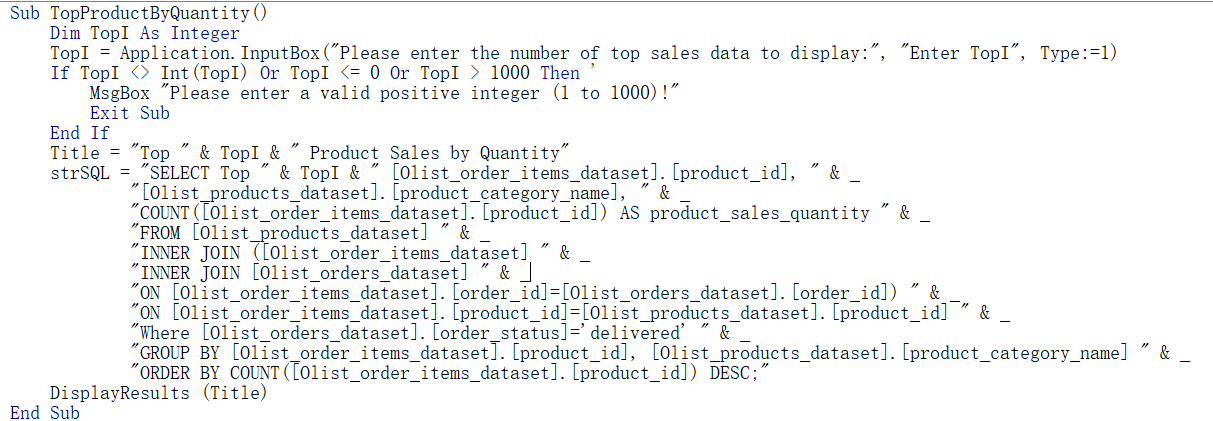
This subroutine opens connection, stores it for reuse, and verifies its success.



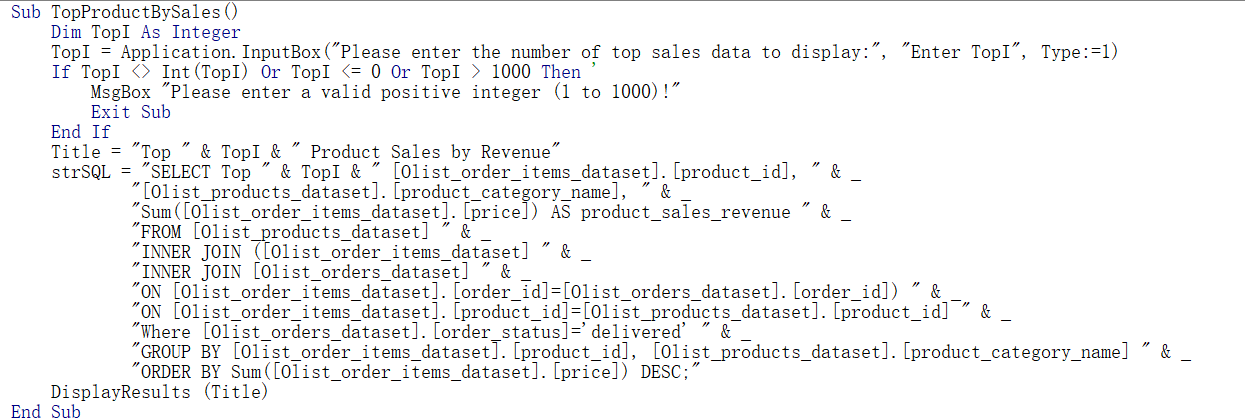
We have designed a subroutine that selects data from a recordset and displays it in Excel in a predefined format, making it easier for future use.



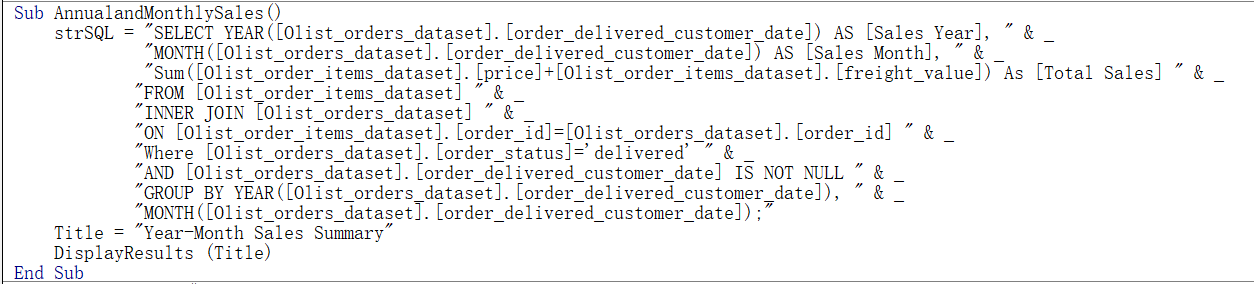
This subroutine prompts the user for a number(if input is invalid a warning will pop up), validates the input, executes the SQL to select the Top X products by sales quantity, and displays the records.

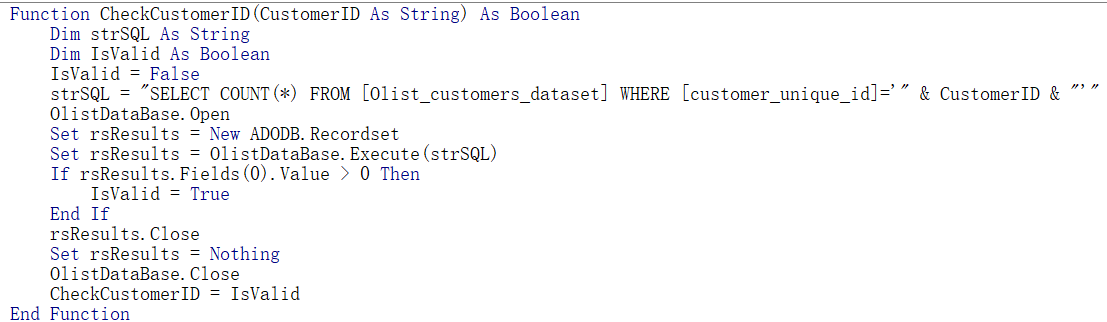


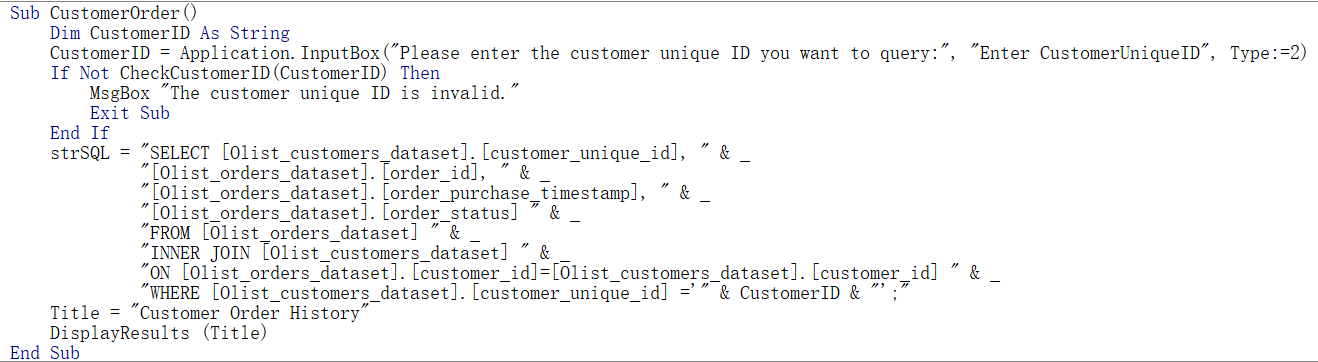
This is pretty much like the above one, except that we display the TopX product by sales revenue.



In this subroutine, we display total sales by year and month.

This function can check whether a given ID exists in our current customer unique ID records.

This subroutine prompts the user for a customer unique ID, validates it, executes the SQL, and displays the corresponding records.



1. Conclusion

Although our project uses data from a real-world e-commerce dataset, the selected subset limits the scope of analysis. To scale this application for a real business, the following improvements could be implemented:

(1)**Enhanced Data Integration:** Include real-time data feeds, customer behavior analytics, and inventory tracking for deeper insights.

(2)**Improved Performance:** Replace Excel VBA with a modern tech stack, such as Python for back-end, and Tableau for front-end.

(3)**Interactive Visualizations:** Integrate tools like Power BI or Tableau to provide dynamic charts and drill-down analysis.

For more details, you can access the project repository on [GitHub](Insert your GitHub link here).

References:

[Brazilian E-Commerce Public Dataset by Olist](https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce/data)