**PROJECT REPORT**

**Moonmoon Samal (cds05\_062)**

Dataset: [Customer\_Purchase\_Behaviour](https://drive.google.com/file/d/18TRw86R911M4sTra7T4yQJEuoGB_HHWy/view?usp=drive_link)

SOFTWARE USED : -

* Database : My SQL Workbench
* Visualization: MS Power-BI
* Preprocessing & EDA: Jupyter Notebook

DATA DETAILS

This dataset containing customer purchase information for an online retail company. The dataset includes the following details:

* Transaction ID
* Customer ID
* Customer Name
* Product ID
* Product Name
* Product Category
* Purchase Quantity
* Purchase Price
* Purchase Date
* Country

**1. Data Extraction and Transformation (SQL):**

* **Setup a Database**: Create a database to store the customer purchase data.
* create database myproject ;
* **Data Ingestion**: Write SQL scripts to import the provided purchase data into the database.

**Steps :**

* + 1. Format of data is csv
    2. I used data ingestion manually by copying the script of the csv dataset using export method in sql.

**Normalization of the data.**

Data is normalized into 3 Tables:-

* 1. Customers
  2. Products
  3. Purchase

Data Manipulation :

Create table customers1 as with cte as (select customername, country, purchasedate from raw\_data)

Select row\_number()over(order by purchasedate)+99 as customerid, customername, country from cte;

alter table raw\_data add column customerid1 int;

- Update customer with the customerid1 values from customers1

update raw\_data r

join customers1 c1 on r.customername = c1.customername and r.country = c1.country

set r.customerid1 = c1.customerid;

-- Replacing the Old customerid Values with new

alter table raw\_data drop column customerid;

alter table raw\_data change column customerid1 customerid int;

-- We can see Now customerid have unique data and we can delete our customer1 table which we used for mapping

drop table customers1;

**Normalization of the data.**

Data is normalized into 3 Tables :-

1. Customers
2. Products
3. Purchase

create table purchases (transactionid int primary key,

customerid int,

productid int,

purchasequantity int,

purchaseprice double,

purchasedate date,

foreign key (customerid) references customers(customerid),

foreign key (productid) references products(productid)

);

insert into purchases (transactionid, customerid, productid,

purchasequantity, purchaseprice, purchasedate)

select transactionid, customerid, productid, purchasequantity,

purchaseprice, purchasedate

from raw\_data;

**Advanced queries to aggregate data**

* *Finding total purchase by each customer*

-- Total spent per customer

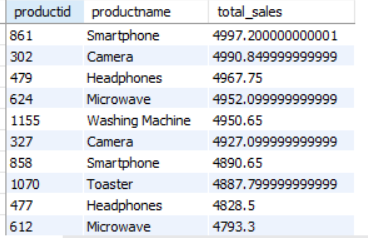
select c.customerid, c.customername, count(\*) as total\_purchases, sum(p.purchasequantity \* p.purchaseprice) as total\_spent

from customers c

join purchases p on c.customerid = p.customerid

group by c.customerid, c.customername

order by total\_spent ;



* *Total sales per product*

-- Total sales per product

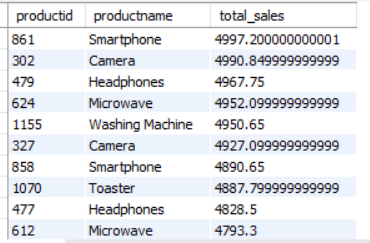
select p.productid, p.productname, sum(pur.purchasequantity \* pur.purchaseprice) as total\_sales

from products p

join purchases pur on p.productid = pur.productid

group by p.productid, p.productname

order by total\_sales desc;



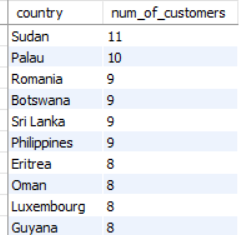
* *Number of customer per country*

select country, count(\*) as num\_of\_customers

from customers

group by country

order by num\_of\_customers desc;



* + *Top 5 product in each category*

select p.productname, sum(ps.purchasequantity) as total\_sales

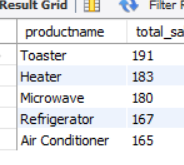
from products p

join purchases ps on p.productid = ps.productid

group by p.productname

order by total\_sales desc

limit 5;



* *Greatest purchase Quantity*

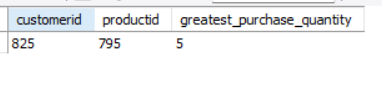
select customerid, productid, max(purchasequantity) as greatest\_purchase\_quantity

from purchases

group by customerid, productid

order by greatest\_purchase\_quantity desc

limit 1;



1. **Data Analysis (Python):**

Libraries used :-

import sqlalchemy

import pymysql

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

* **Data Extraction**:

engine = sqlalchemy.create\_engine('mysql+pymysql://root:12345@localhost:3306/myproject')

#Loading Tables from 'myproject' database

customers = pd.read\_sql\_table("customers", engine)

products = pd.read\_sql\_table("products", engine)

purchases = pd.read\_sql\_table("purchases", engine)

* **Data Analysis**: Perform the following analysis using basic Python:
  + *Calculate total purchases, total revenue, and average purchase value.*

total\_purchases total\_purchases = combined\_df['purchasequantity'].sum()

print(f"Total Purchase: {total\_purchases}")

print(total\_purchases)

total\_revenue = total\_revenue = (combined\_df['purchasequantity'] \* combined\_df['purchaseprice']).sum()

print(f"Total Revenue: {total\_revenue}") :

Avg\_purchase = #average\_purcahse\_value

Avg\_purchase = round(combined\_df['purchaseprice'].mean(),2)

print(f"Average Purchase Value: {Avg\_purchase}")



* ***Identify top customers and their purchasing behavior.***

top\_customers = purchases.groupby('customerid').agg(

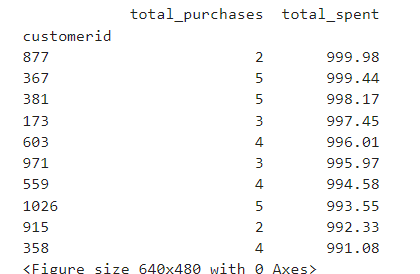
total\_purchases=('purchasequantity', 'sum'),

total\_spent=('purchaseprice', 'sum')

).nlargest(10, 'total\_spent')

plt.savefig('top\_customer.png')

print(top\_customers)



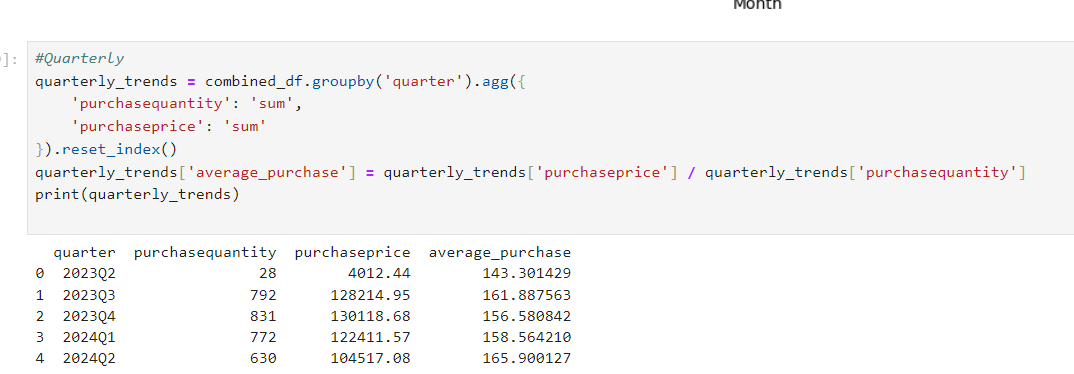
* ***Analyze purchase trends over time (monthly, quarterly, yearly).***

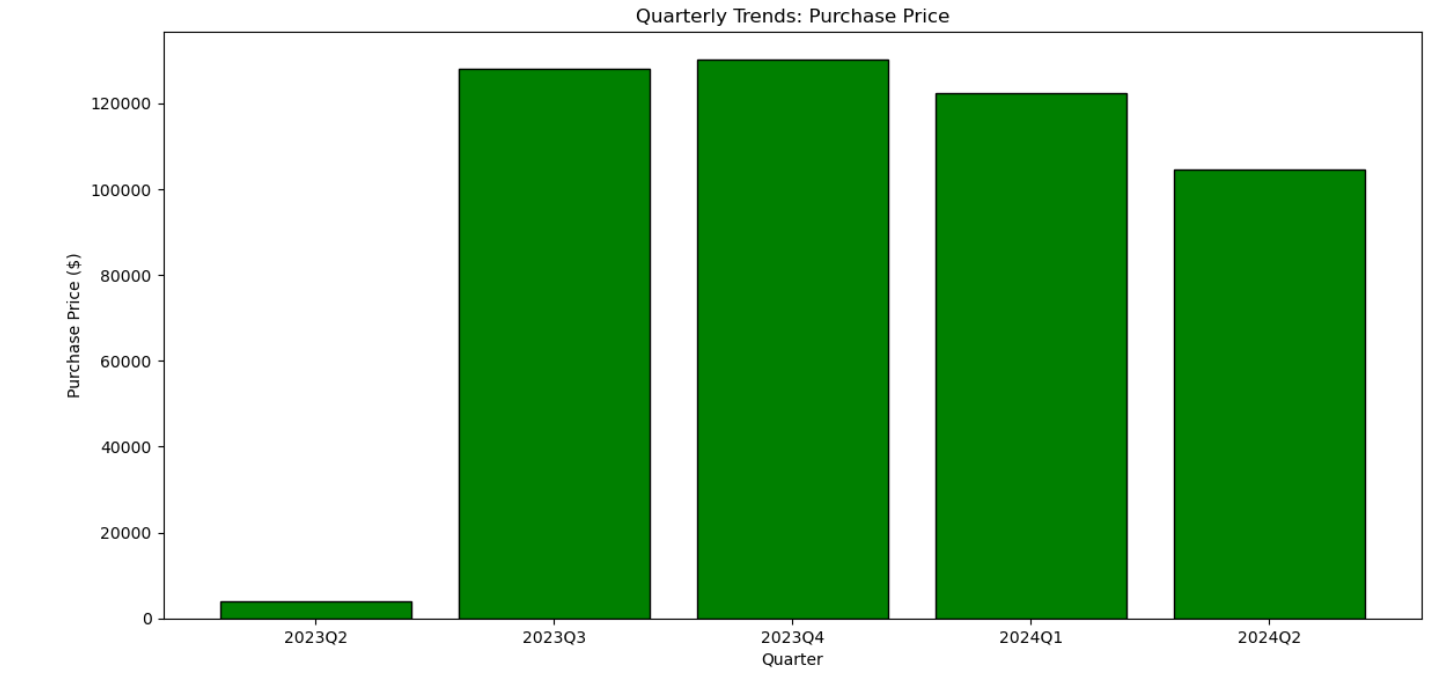
MONTHLY TRENDS



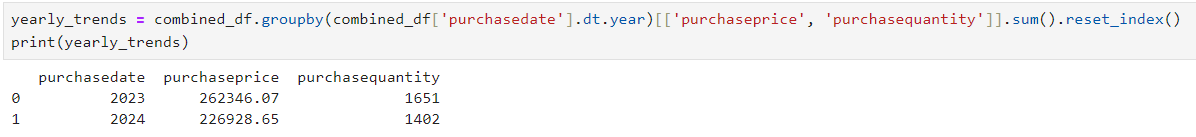


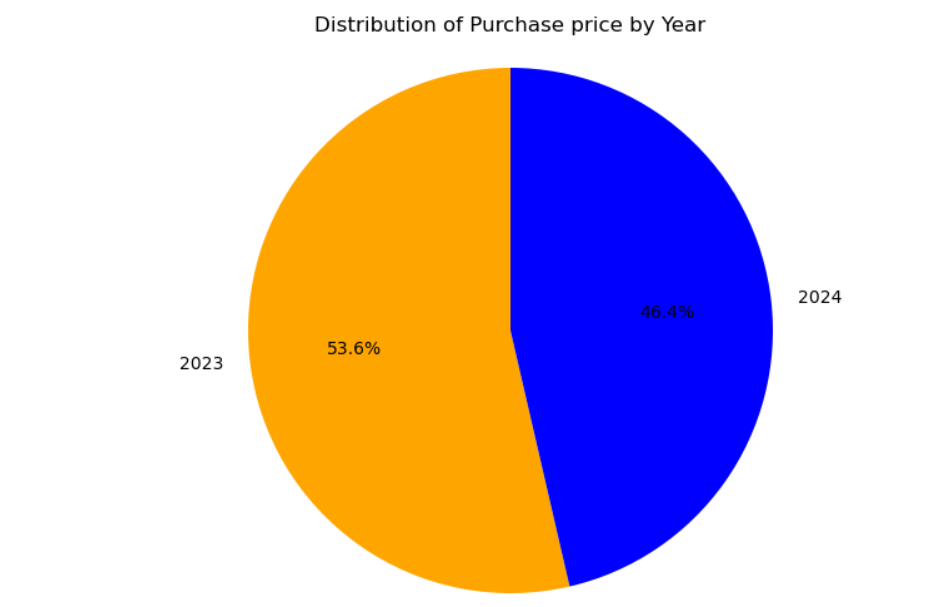
QUATERLY TREND





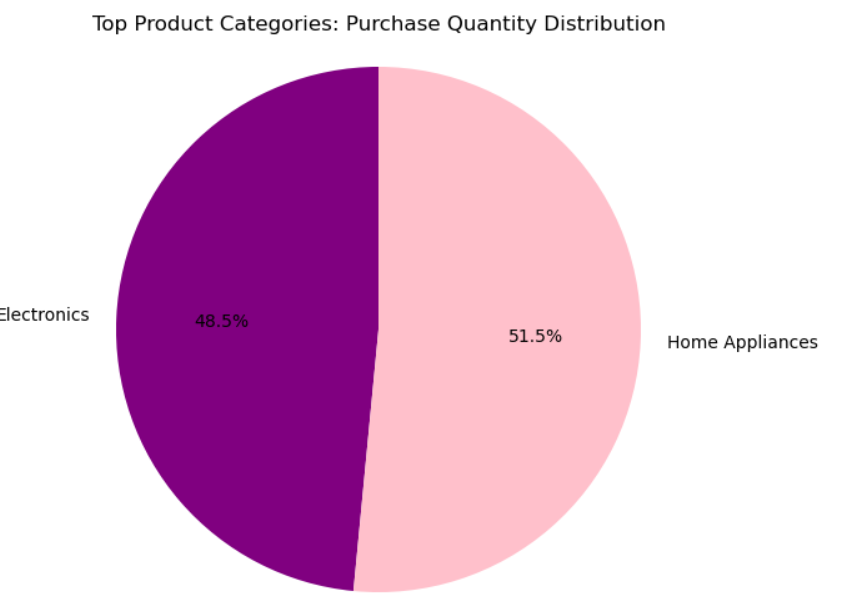
YEARLY TREND



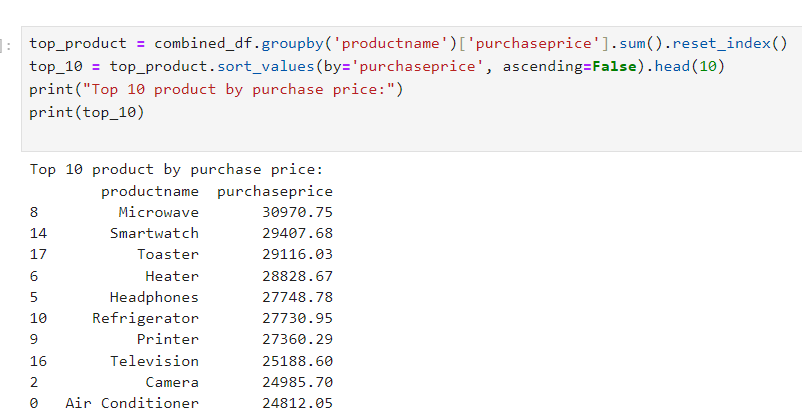


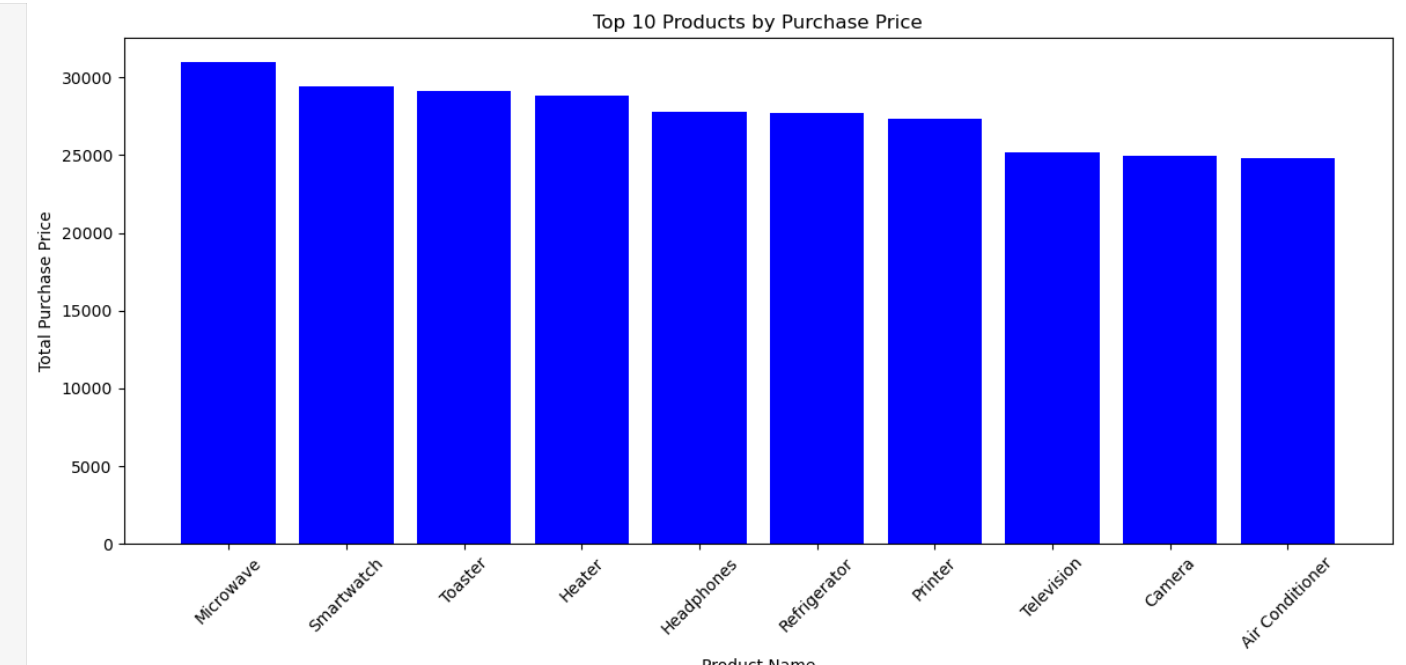
* ***Identify the top-performing product categories.***





* ***Identify the top-performing product categories.***





Key Insight Report;

1. Total Purchases: 3053

2. Total Revenue: 1,485,760.54

3. Average Purchase Value: 489.29

Purchase\_trend over time:

Monthly Trend in 2023:

December-2023 saw the highest purchase quantity (295 units) , purchase price of 51910.42, with an average purchase value of 175.96.

and lowest is the jun-2023 with the purchase quantity(28 unit) , purchase price of 4012.44 with an avg purchase value of 143.30

Purchases showed fluctuations across other months, indicating seasonal variations or promotional effects.

Quarterly Trend:

Q2 2023 had the lowest purchase quantity but a moderate average purchase value.

Q3 and Q4 2023 showed higher quantities and revenues

Q1 2024 maintained a steady purchase quantity and average purchase value.

Yearly Trend:

2023 saw the highest overall purchase quantity (1,651 units) and revenue ($262,346.07), with a consistent average purchase value.

2024 started slightly lower but maintained a strong average purchase value.

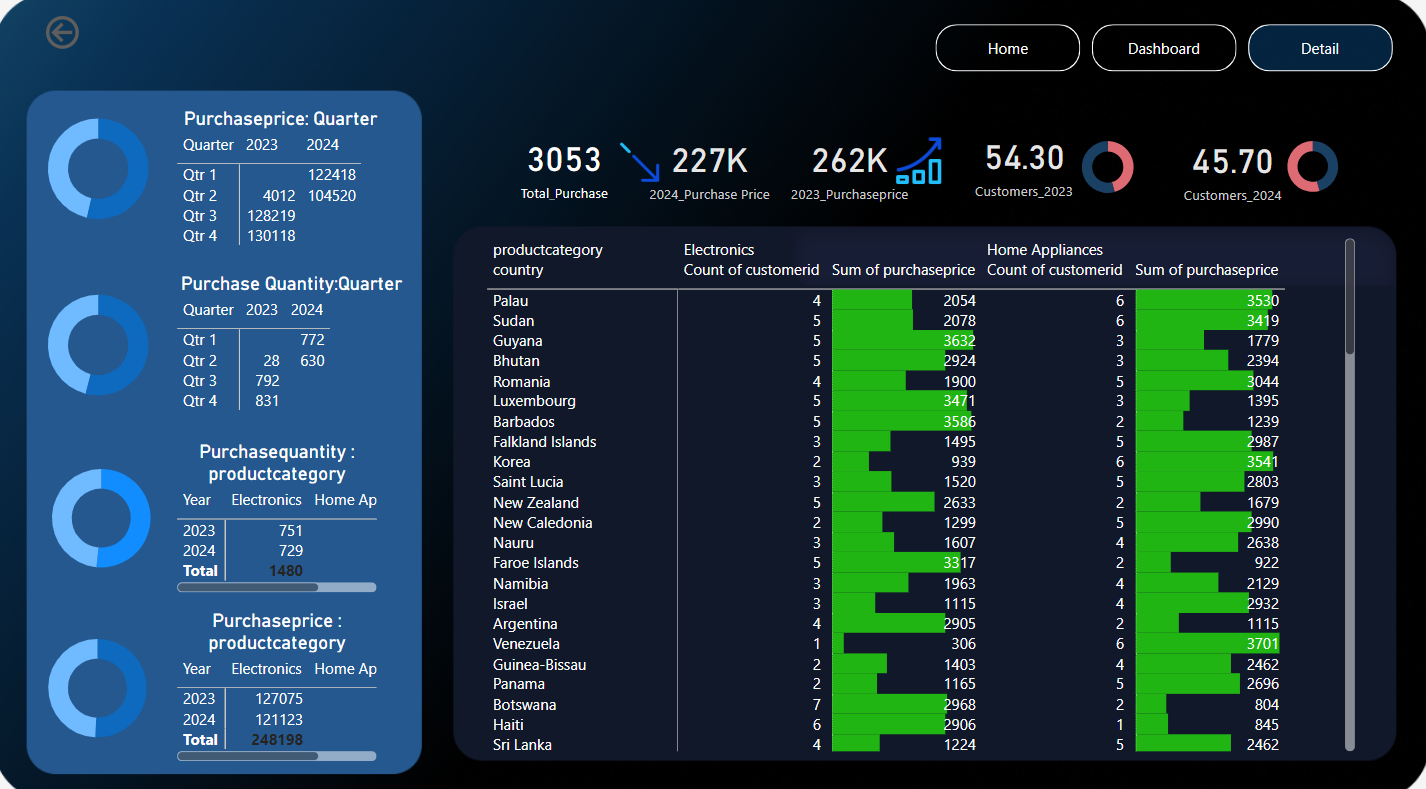
Top performing product Performance:

Electronics: Customers made 1,480 purchases, generating 248,194.56 in revenue. On average, each purchase in this category was about 167.70.

Home Appliances: This category saw 1573 purchases, totaling 241,080.16 in revenue. The average purchase here was around 153.26.

Data Visualization and Reporting (Power BI):





G drive link: [Link to myproject](https://drive.google.com/drive/folders/10zC5bYRB3k1OcJ_yFi_ht_fP-_607Ka2?usp=sharing)

Thank You.