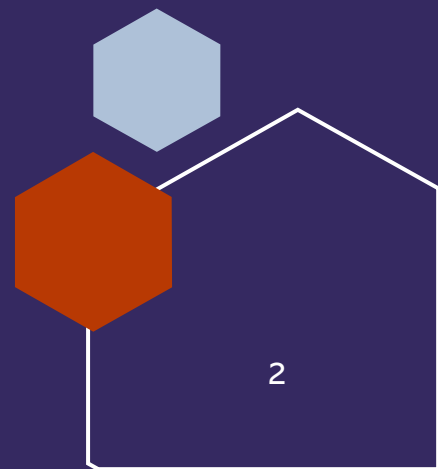
MENTORNESS SQL SALES DATABASE

Hanifa
Salihu

Introduction

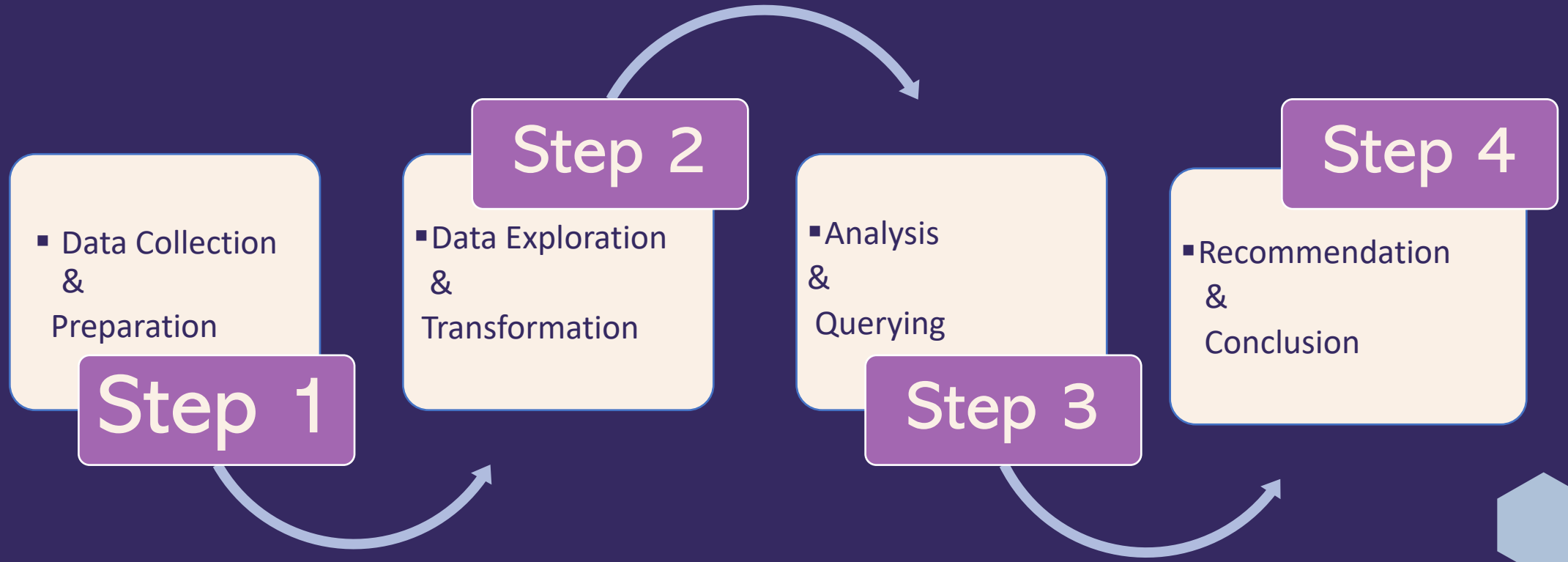
This sales data for a company plays a pivotal role in driving informed decisions and optimizing processes. The database system comprises five interconnected tables, which will be used to explore the relationships between various aspects of the business, show the data manipulated, analyzed, and formulate strategies for operational growth.
PostgreSQL & PowerBI was used for analysis and visualization.



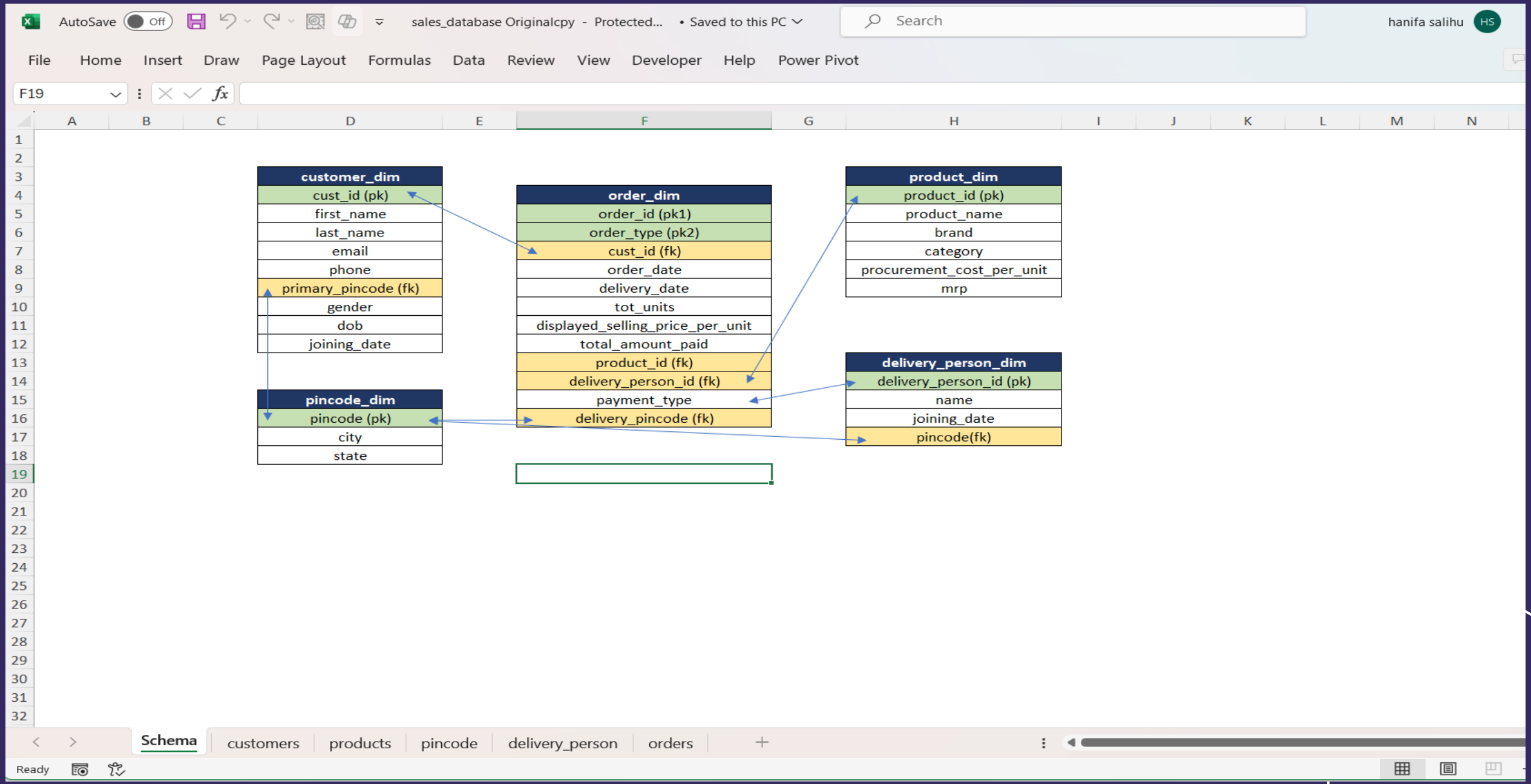
Problem Statement

This analysis aims to understand the sales data from customers, products, orders, and delivery personnel to uncover valuable insights, identify patterns and provide data-driven recommendations.

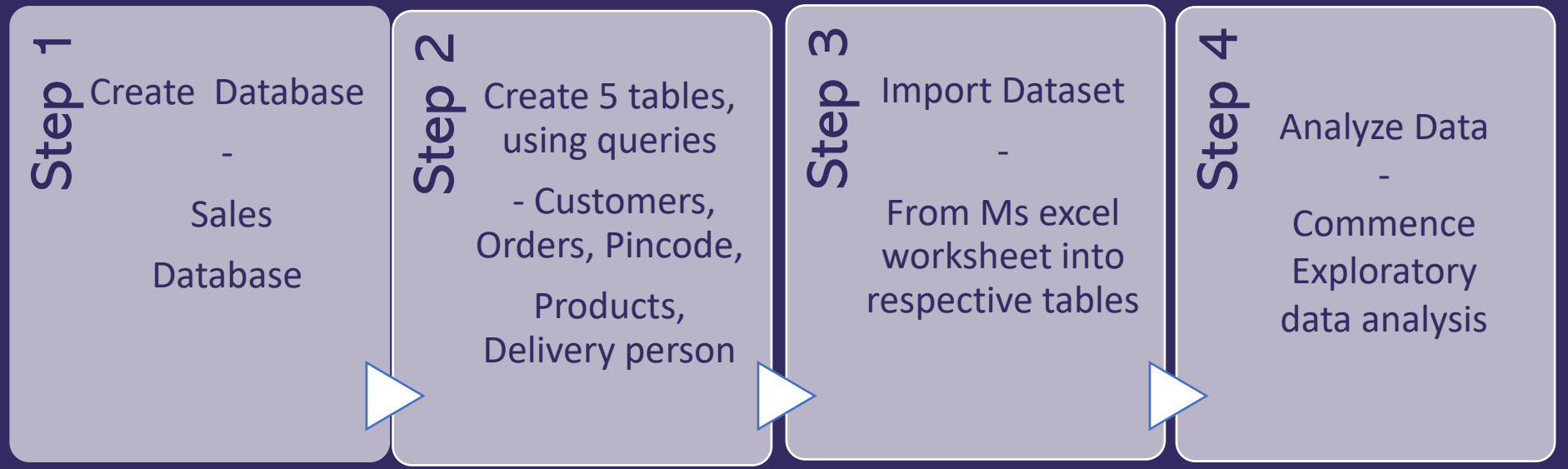
Process Workflow



Unclean Data



Exploratory Data Process



Sales Database Source Code

Query | Query History

```sql
---CREATE 5 TABLES---

CREATE TABLE customers (
    cust_id INT PRIMARY KEY,
    first_name VARCHAR,
    last_name VARCHAR,
    email VARCHAR,
    phone BIGINT,
    primary_pincode INT,
    gender VARCHAR,
    dob DATE,
    joining_date DATE)

SELECT * FROM customers


CREATE TABLE products (
    product_id INT PRIMARY KEY,
    product_name VARCHAR,
    brand VARCHAR,
    category VARCHAR,
    procurement_cost_per_unit INT,
    mrp INT)

SELECT * FROM products


CREATE TABLE pincode (
    pincode INT PRIMARY KEY,
    city VARCHAR,
    state VARCHAR)

SELECT * FROM pincode
```

Total rows: 0 of 0

Query | Query History

```sql
----- 2. How many customers are there in each pincode and gender combination?

SELECT primary_pincode, gender,
COUNT(*) AS customer_count
FROM customers
GROUP BY primary_pincode, gender


----- 3. Print product name and mrp for products which have more than 50000 MRP?

SELECT product_name, mrp
FROM products
WHERE mrp > 50000


----- 4. How many delivery personal are there in each pincode?

SELECT pincode,
COUNT(*) AS total_delivery_personnels
FROM delivery_person
GROUP BY pincode


----- 5. For each Pin code, print the count of orders, sum of total amount paid, average amount paid,
-- maximum amount paid, minimum amount paid for the transactions which were paid by 'cash'. Take only 'buy' order types

SELECT delivery_pincode,
    COUNT(*) AS order_count,
    SUM(total_amount_paid) AS total_amount_paid,
    AVG(total_amount_paid) AS avg_amount_paid,
    MAX(total_amount_paid) AS max_amount_paid,
    MIN(total_amount_paid) AS min_amount_paid
FROM orders
WHERE order_type = 'buy'
```

Total rows: 0 of 0

Query | Query History

```sql
SELECT
    c.cust_id,
    SUM(o.tot_units) AS total_units_ordered,
    SUM(CASE WHEN o.delivery_pincode = c.primary_pincode THEN o.tot_units ELSE 0 END) AS units_ordered_primary_pincode,
    SUM(CASE WHEN o.delivery_pincode <> c.primary_pincode THEN o.tot_units ELSE 0 END) AS units_ordered_not_primary_pincode,
    (100.0 * SUM(CASE WHEN o.delivery_pincode = c.primary_pincode THEN o.tot_units ELSE 0 END)) /
    SUM(o.tot_units) AS percentage_ordered_primary_pincode
FROM customers c
JOIN  orders o ON c.cust_id = o.cust_id
GROUP BY c.cust_id
ORDER BY percentage_ordered_primary_pincode DESC


----- 14. For each product name, print the sum of number of units, total amount paid, total displayed selling price, total mrp of these units,
and finally the net discount from selling price.(i.e. 100.0 - 100.0 * total amount paid / total displayed selling price) &
the net discount from mrp (i.e. 100.0 - 100.0 * total amount paid / total mrp)

SELECT p.product_name,
    SUM(o.tot_units) AS total_units,
    SUM(o.total_amount_paid) AS total_amount_paid,
    SUM(o.displayed_selling_price_per_unit * o.tot_units) AS total_displayed_selling_price,
    SUM(p.mrp * o.tot_units) AS total_mrp,
    (100.0 - 100.0 * SUM(o.total_amount_paid) / SUM(o.displayed_selling_price_per_unit * o.tot_units)) AS net_discount_selling_price,
    (100.0 - 100.0 * SUM(o.total_amount_paid) / SUM(p.mrp * o.tot_units)) AS net_discount_mrp
FROM orders o
JOIN products p ON o.product_id = p.product_id
GROUP BY p.product_name


----- 15. For every order_id (exclude returns), get the product name and calculate the discount percentage from selling price.
Sort by highest discount and print only those rows where discount percentage was above 10.10%.

WITH order_details AS (
  SELECT
```

Total rows: 0 of 0

Insights & Analysis

15
customers &
2 without
dob

6 cities
&
states

10
delivery
men

Payment
Types

Netbanking

Cash

Debit

3 products
&
2 brands

1,000 buys
&
50 returns

Data Visualization 1

SALES ANALYSIS DASHBOARD

Total Amount Paid

113M

Products

- Dell AX420
- HP 241H
- HP XYZ Mouse
- Dell ABC Mouse
- HP 8GB Pendrive
- Dell 8GB Pendrive

Total Units

5624

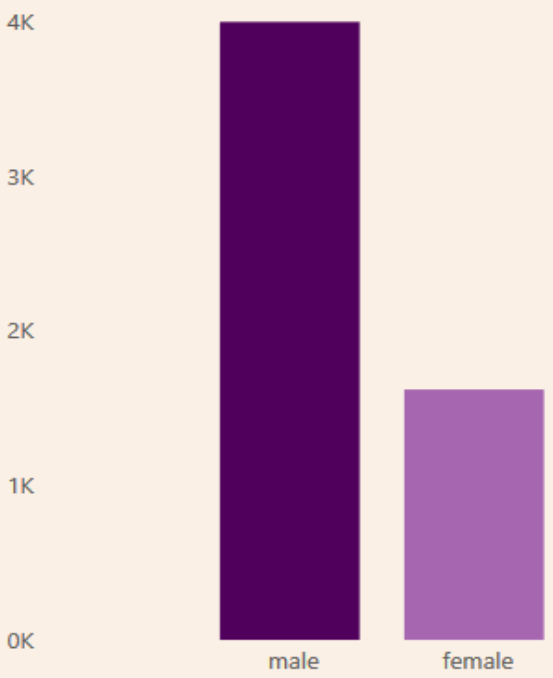
Total Customers

14

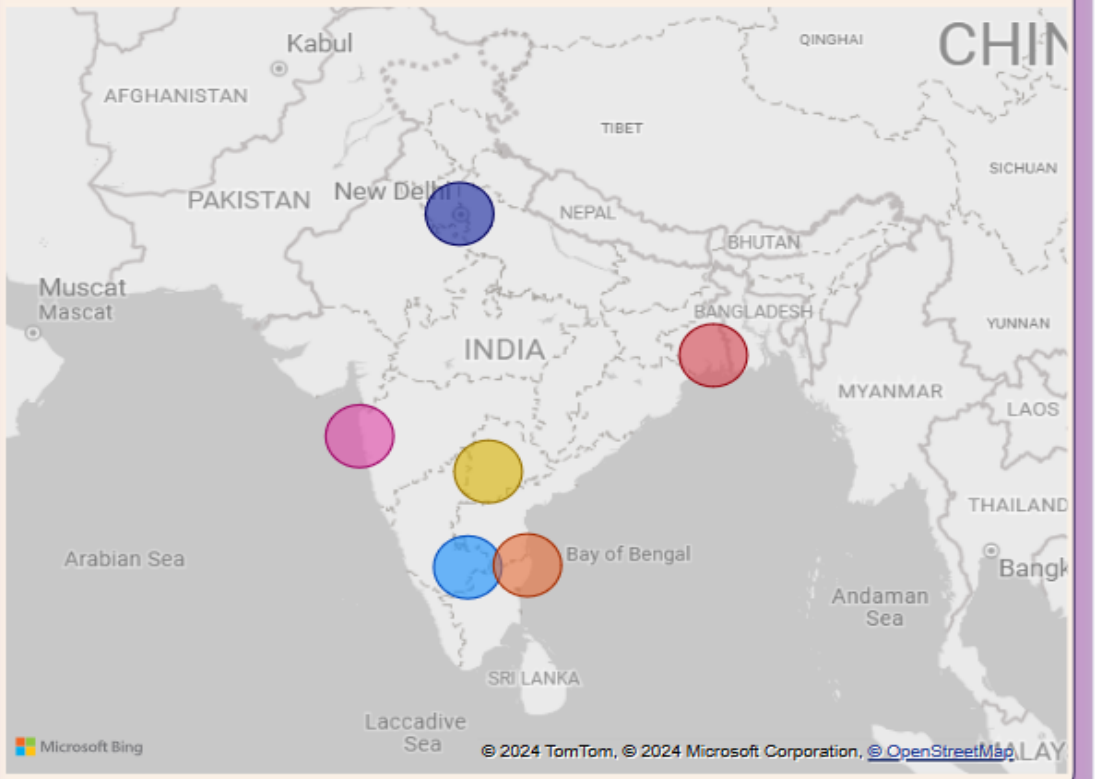
No. of Delivery Persons

10

Total Orders by Gender



Cities & States



Recommendation

- Review product profitability data and identify most/least profitable products or categories.
- Analyze delivery personnel order volumes by month. Reallocate resources, adjust routes for efficiency, and plan staffing for peak periods.
- Identify high-value customers ordering from their primary Pin codes. Use them for loyalty programs, personalized promotions.

Limitation

- Limited customer information like demographics, purchase history.
- Absence of external factors like economic conditions, seasonality, and market trends and purchasing behavior.
- Limited time period analysis, preventing identification of trends, patterns, and anomalies over different time frames.

Conclusion

- Additional information to the dataset to enable personalized promotions and targeted marketing strategies.
- Incorporate competitor data, including product offerings, pricing, and market share information, to facilitate benchmarking and develop effective competitive strategies.
- Introduce time-based analysis to identify trends, patterns, over different time periods, enabling more informed decision-making and strategic planning.

Thank you