Task 4

```
CREATE TABLE sales (
    invoice_id VARCHAR(20),
    branch VARCHAR(5),
    city VARCHAR(50),
    customer_type VARCHAR(20),
    gender VARCHAR(10),
    product_line VARCHAR(50),
    unit_price NUMERIC(10, 2),
    quantity INT,
    tax_5 NUMERIC(10, 2),
    total NUMERIC(10, 2),
    date DATE,
    time TIME,
    payment VARCHAR(20),
    cogs NUMERIC(10, 2),
    gross_margin_percentage NUMERIC(5, 2),
    gross_income NUMERIC(10, 2),
    rating NUMERIC(3, 1)
);
COPY sales FROM 'C:/supermarket_sales.csv'
DELIMITER ',' CSV HEADER;

SELECT * FROM sales LIMIT 10;
```

```
SELECT SUM(total) AS total_revenue FROM sales;
```

```
SELECT city, SUM(total) AS total_sales
FROM sales
GROUP BY city
ORDER BY total_sales DESC;
```

```
SELECT gender, COUNT(*) AS total_customers
FROM sales
GROUP BY gender;
```

```
SELECT product_line, ROUND(AVG(rating), 2) AS avg_rating
FROM sales
GROUP BY product_line
ORDER BY avg_rating DESC;
```

```
SELECT payment, COUNT(*) AS total_transactions
FROM sales
```

```
GROUP BY payment
ORDER BY total_transactions DESC;


SELECT date, SUM(total) AS daily_sales
FROM sales
GROUP BY date
ORDER BY date;


SELECT EXTRACT(HOUR FROM time::time) AS hour, COUNT(*) AS sales_count
FROM sales
GROUP BY hour
ORDER BY sales_count DESC;


SELECT city, SUM(total) AS total_sales
FROM sales
GROUP BY city
HAVING SUM(total) > (
    SELECT AVG(total) FROM sales
);


SELECT city, SUM(total) AS total_sales,
```

```
    RANK() OVER (ORDER BY SUM(total) DESC) AS sales_rank
FROM sales
GROUP BY city;


SELECT date, SUM(total) AS daily_sales,
    SUM(SUM(total)) OVER (ORDER BY date) AS running_total
FROM sales
GROUP BY date
ORDER BY date;


SELECT product_line, SUM(total) AS total_sales
FROM sales
GROUP BY product_line
ORDER BY total_sales DESC
LIMIT 1;
```

```
pip install pandas matplotlib psycopg2


import pandas as pd
import matplotlib.pyplot as plt
import psycopg2

# Connect to your PostgreSQL database
conn = psycopg2.connect(
    dbname="supermarket_sales_db",  # Replace with your DB name
    user="postgres",                # Replace with your PostgreSQL username
    password="yourpassword",        # Replace with your PostgreSQL password
    host="localhost",
    port="5432"
)

# 1. Total Sales by City
query1 = """
SELECT city, SUM(total) AS total_sales
FROM sales
GROUP BY city
ORDER BY total_sales DESC;
"""
df1 = pd.read_sql(query1, conn)
df1.plot(kind='bar', x='city', y='total_sales', title='Total Sales by City', legend=False, color='skyblue')
plt.ylabel('Total Sales')
```

```
plt.tight_layout()

plt.show()


# 2. Average Rating by Product Line

query2 = """

SELECT product_line, ROUND(AVG(rating), 2) AS avg_rating

FROM sales

GROUP BY product_line

ORDER BY avg_rating DESC;

"""

df2 = pd.read_sql(query2, conn)

df2.plot(kind='barh', x='product_line', y='avg_rating', title='Average Rating by Product
Line', legend=False, color='orange')

plt.xlabel('Average Rating')

plt.tight_layout()

plt.show()


# 3. Sales by Payment Method

query3 = """

SELECT payment, COUNT(*) AS total_transactions

FROM sales

GROUP BY payment;

"""

df3 = pd.read_sql(query3, conn)

df3.set_index('payment').plot.pie(y='total_transactions', autopct='%1.1f%%',
title='Sales by Payment Method', legend=False)
```

```
plt.ylabel('')
plt.tight_layout()
plt.show()

# 4. Daily Sales Trend
query4 = """
SELECT date, SUM(total) AS daily_sales
FROM sales
GROUP BY date
ORDER BY date;
"""
df4 = pd.read_sql(query4, conn)
df4['date'] = pd.to_datetime(df4['date'])
df4.set_index('date', inplace=True)
df4.plot(title='Daily Sales Trend', linewidth=2, color='green')
plt.ylabel('Sales')
plt.tight_layout()
plt.show()

# 5. Revenue by Gender
query5 = """
SELECT gender, SUM(total) AS revenue
FROM sales
GROUP BY gender;
"""
```

```
df5 = pd.read_sql(query5, conn)

df5.plot(kind='bar', x='gender', y='revenue', title='Revenue by Gender', legend=False, color='purple')

plt.ylabel('Revenue')

plt.tight_layout()

plt.show()


# 6. Hourly Sales Activity

query6 = """

SELECT EXTRACT(HOUR FROM time::time) AS hour, COUNT(*) AS sales_count

FROM sales

GROUP BY hour

ORDER BY hour;

"""

df6 = pd.read_sql(query6, conn)

df6.plot(kind='line', x='hour', y='sales_count', title='Sales by Hour', marker='o', color='red')

plt.xlabel('Hour of Day')

plt.ylabel('Number of Sales')

plt.xticks(range(0, 24))

plt.tight_layout()

plt.show()


# 7. Top Product Line by Sales

query7 = """

SELECT product_line, SUM(total) AS total_sales
```

```
FROM sales

GROUP BY product_line

ORDER BY total_sales DESC

LIMIT 5;

"""

df7 = pd.read_sql(query7, conn)

df7.plot(kind='bar', x='product_line', y='total_sales', title='Top 5 Product Lines by
Sales', legend=False, color='teal')

plt.ylabel('Total Sales')

plt.tight_layout()

plt.show()


# Close the connection

conn.close()
```