

# **Customer Sales Analysis**

## **A SUMMER INTERNSHIP REPORT**

*Submitted by*

**Name: J PRAVEEN DASS**

**Roll No:727722EUIT132**

**SRI KRISHNA COLLEGE OF ENGINEERING AND  
TECHNOLOGY**

*in partial fulfillment for the Completion of*

Summer Internship 2024

*In*

**PROJECT DEVELOPMENT CELL (PDC)**

**COMPUTER SCIENCE AND ENGINEERING**



**COIMBATORE INSTITUTE OF TECHNOLOGY**

*(Government-Aided Autonomous Institution Affiliated to Anna University)*

**COIMBATORE-641014**

**ANNA UNIVERSITY: CHENNAI 600 025**

**July 2024**

COIMBATORE INSTITUTE OF TECHNOLOGY  
(A Govt. Aided Autonomous Institution Affiliated to Anna University)  
COIMBATORE – 641014

**BONAFIDE CERTIFICATE**

Certified that this summer internship'2024 project “**Customer sales Analysis**” is the Bonafide work of **Praveen Dass J, 727722EUIT132, B. TECH Information Technology, Sri Krishna College of Engineering and Technology** under my mentorship during the period **28th June to 13th July 2024.**

Certified that the candidates were examined continuously by us during the summer internship held at our premises through PDC.

**Mrs. P. Anitha**

**Designation**

Department of CSE,  
Coimbatore Institute of Technology,  
Coimbatore – 641014.

**Dr. A. Kunthavai**

**Convener – PDC**

Department of CSE,  
Coimbatore Institute of Technology,  
Coimbatore – 641014.

Place: Coimbatore Institute of Technology (CIT)

Date: From 28.06.2024 to 15.07.2024

## ABSTRACT

### Background and Scope

The retail industry generates vast amounts of transactional data, which contain critical insights into customer behavior and sales trends. This project, Customer Sales Analysis, aims to analyze this data to provide actionable insights that guide strategic business decisions. The scope includes data collection and cleaning, such as importing the dataset, handling missing values and duplicates, and normalizing data for consistency. Additionally, it encompasses exploratory data analysis (EDA), which involves calculating summary statistics for sales quantity and revenue and performing correlation analysis between product price and sales quantity. Furthermore, data visualization is a key component, where line charts for sales trends over time, bar charts for top-selling products, and histograms to visualize the distribution of sales quantities are created.

### Algorithm and Results

The algorithm for this project involves several steps: importing the dataset into a Pandas DataFrame, cleaning the data by removing missing values and duplicates, and normalizing date formats. Summary statistics are then calculated using pandas' `describe()` method, and correlation analysis is performed using pandas' `corr()` method. Various visualizations are created using matplotlib, including line charts, bar charts, and histograms. The results of this analysis identify key sales trends and top-selling products, providing insights into customer behavior, optimizing inventory management, and improving marketing strategies.

## **TABLE OF CONTENTS**

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
	<b>ABSTRACT</b>	<b>I</b>
	<b>LIST OF TABLES</b>	<b>II</b>
	<b>LIST OF FIGURES</b>	<b>III</b>
	<b>LIST OF ABBREVIATIONS</b>	<b>IV</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Problem statement	1
	1.2 Scope and Objectives	1
<b>2</b>	<b>SYSTEM DESIGN AND IMPLEMENTATION</b>	<b>2</b>
	2.1 System Design	2
	2.2 System Architecture	2
	2.3 Module Description	3
<b>3</b>	<b>RESULTS AND DISCUSSIONS</b>	<b>4</b>
<b>4</b>	<b>CONCLUSION AND FUTURE WORK</b>	<b>11</b>
	<b>REFERENCES</b>	<b>12</b>

## **LIST OF FIGURES**

<b>FIGURE NO</b>	<b>TITLE</b>	<b>PAGE</b>
<b>1.1</b>	<b>Problem Statement</b>	<b>1</b>
<b>2.1</b>	<b>System Design</b>	<b>2</b>
<b>2.2</b>	<b>System Architecture</b>	<b>2</b>
<b>2.3</b>	<b>Module Description</b>	<b>3</b>
<b>3.1</b>	<b>Data Collection</b>	<b>4</b>
<b>3.2</b>	<b>EDA</b>	<b>6</b>
<b>3.3</b>	<b>Visualization</b>	<b>8</b>
<b>4.1</b>	<b>Conclusion</b>	<b>11</b>
<b>4.2</b>	<b>Future Work</b>	<b>12</b>

## LIST OF ABBREVIATIONS

- **CSV:** Comma-Separated Values
- **EDA:** Exploratory Data Analysis
- **CRM:** Customer Relationship Management
- **ARIMA:** Auto Regressive Integrated Moving Average
- **K-means:** A clustering algorithm
- **pandas:** Python Data Analysis Library
- **matplotlib:** A plotting library for Python
- **BI:** Business Intelligence

# CHAPTER 1

## INTRODUCTION

The retail industry is one of the most dynamic and data-rich sectors, where businesses continuously strive to understand customer behaviour and sales trends to stay competitive. Customer Sales Analysis involves the examination of sales data to uncover patterns and insights that can guide strategic decisions. By leveraging data analytics, companies can enhance their marketing efforts, optimize inventory management, and improve customer satisfaction.

In this project, we focus on analysing sales data to gain insights into customer behavior and sales trends. The dataset consists of transactional data, including Customer ID, Date, Product, Quantity, and Price. This data will be used to perform various analyses, such as handling missing values, calculating summary statistics, conducting correlation analysis, and visualizing data through different charts.

### 1.1 Problem Statement

The main challenge in this project is to extract meaningful insights from raw sales data. This involves several tasks: cleaning the data, handling missing values and duplicates, normalizing the data, and performing exploratory data analysis (EDA). Additionally, visualizing the data is crucial to understanding trends and patterns.

#### Scope

The scope of this project includes the following tasks:

1. **Data Collection and Cleaning:** Importing the sales dataset, handling missing values and duplicates, and normalizing the data.
2. **Exploratory Data Analysis (EDA):** Calculating summary statistics for sales quantity and revenue, and performing correlation analysis between variables like product price and sales quantity.
3. **Data Visualization:** Creating line charts to show sales trends over time, generating bar charts to display top-selling products, and using histograms to visualize the distribution of sales quantities.

#### Objectives

1. To clean and preprocess the sales data for analysis.
2. To calculate and interpret summary statistics for sales quantity and revenue.
3. To analyze the correlation between product price and sales quantity.

#### Applications

1. **Marketing Strategies:** Identifying top-selling products and sales trends.
2. **Inventory Management:** Understanding sales patterns aids in optimizing inventory levels, reducing stockouts, and minimizing overstock situations.

## CHAPTER 2

### SYSTEM METHODOLOGY

## 2.1 System Design

1. **Data Collection and Cleaning:**
  - Importing the dataset.
  - Handling missing values and duplicates.
  - Normalizing the data for consistency.
2. **Exploratory Data Analysis (EDA):**
  - Calculating summary statistics.
  - Performing correlation analysis.
3. **Data Visualization:**
  - Creating line charts for sales trends.
  - Generating bar charts for top-selling products.
  - Using histograms to visualize sales quantities.

## 2.2 System Architecture

The system architecture outlines the logical flow of data and processes within the Customer Sales Analysis project. It consists of various components interacting with each other to achieve the project's objectives.

Components of the Architecture

1. **Data Source:** This is where the raw sales data is stored (e.g., CSV file).
2. **Data Preprocessing Module:** Responsible for data cleaning, handling missing values, and normalizing the data.
3. **Analysis Module:** Performs exploratory data analysis and calculates necessary statistics.
4. **Visualization Module:** Creates charts and graphs to visualize the results of the analysis.
5. **Output:** The final insights and visualizations that inform business decisions.

## 2.3 Module Description

Data Collection and Cleaning Module

**Functionality:**

- **Data Import:** Import the sales dataset from a CSV file.
- **Data Cleaning:** Handle missing values by either filling them with appropriate values or removing the rows/columns with missing data.
- **Duplicate Removal:** Identify and remove duplicate records to ensure data integrity.
- **Data Normalization:** Convert dates to a consistent format and standardize other data fields as necessary.



## Exploratory Data Analysis (EDA) Module

### Functionality:

- **Summary Statistics:** Calculate mean, median, mode, and other descriptive statistics for sales quantity and revenue.
- **Correlation Analysis:** Assess the relationship between variables, such as the correlation between product price and sales quantity.

## Data Visualization Module

### Functionality:

- **Line Charts:** Plot sales trends over time to identify patterns and seasonal variations.
- **Bar Charts:** Display top-selling products to highlight the most popular items.
- **Histograms:** Visualize the distribution of sales quantities to understand frequency and spread.

## Algorithm Explanation

The implemented project utilizes a straightforward algorithmic approach to achieve the objectives. The process can be summarized as follows:

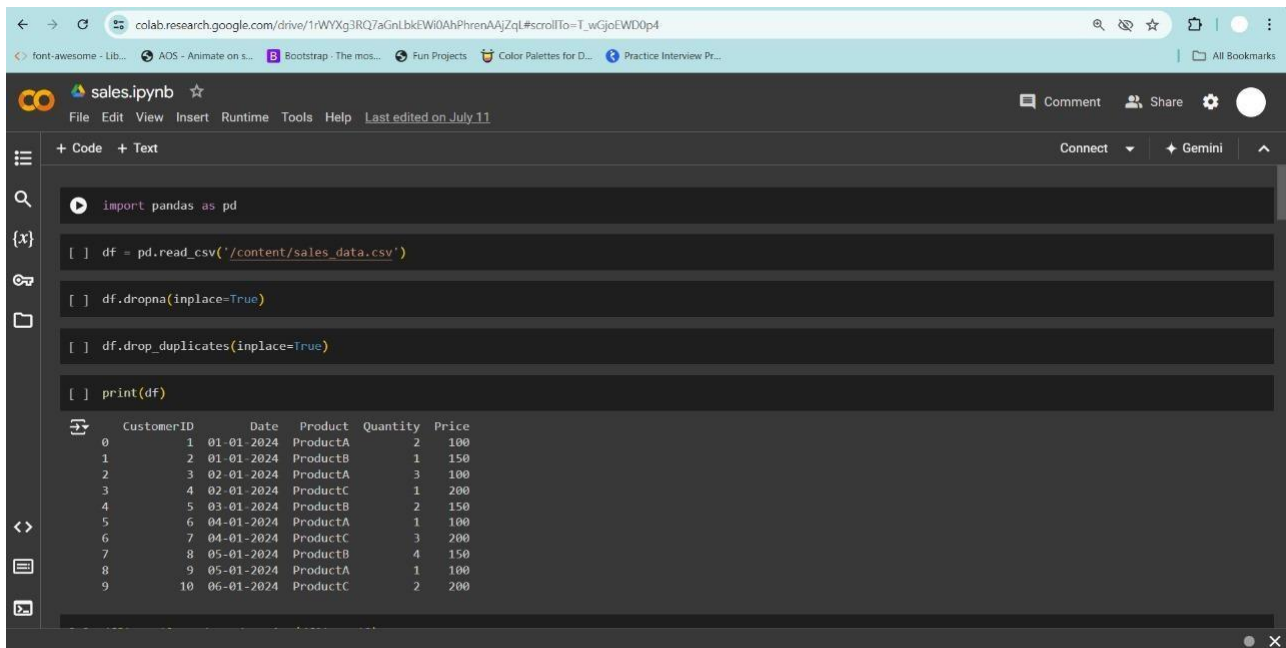
1. **Data Importation:**
  - Load the dataset into a Pandas Data Frame.
2. **Data Cleaning:**
  - Remove missing values and duplicates.
  - Normalize the data, especially date formats.
3. **Exploratory Data Analysis:**
  - Calculate summary statistics using pandas' describe() method.
  - Perform correlation analysis using pandas' corr() method.
4. **Data Visualization:**
  - Use matplotlib to create line charts for sales trends.
  - Generate bar charts for top-selling products.
  - Create histograms to visualize sales quantities distribution.

# CHAPTER 3

## RESULTS AND DISCUSSIONS

### 3.1 Data Collection and Cleaning

>Handle Missing Values and Duplicates



The screenshot shows a Google Colab notebook interface. The code cell contains the following Python code:

```
import pandas as pd

df = pd.read_csv('/content/sales_data.csv')

df.dropna(inplace=True)

df.drop_duplicates(inplace=True)

print(df)
```

The output of the code is a DataFrame with the following data:

	CustomerID	Date	Product	Quantity	Price
0	1	01-01-2024	ProductA	2	100
1	2	01-01-2024	ProductB	1	150
2	3	02-01-2024	ProductA	3	100
3	4	02-01-2024	ProductC	1	200
4	5	03-01-2024	ProductB	2	150
5	6	04-01-2024	ProductA	1	100
6	7	04-01-2024	ProductC	3	200
7	8	05-01-2024	ProductB	4	150
8	9	05-01-2024	ProductA	1	100
9	10	06-01-2024	ProductC	2	200

#### Interpretation:

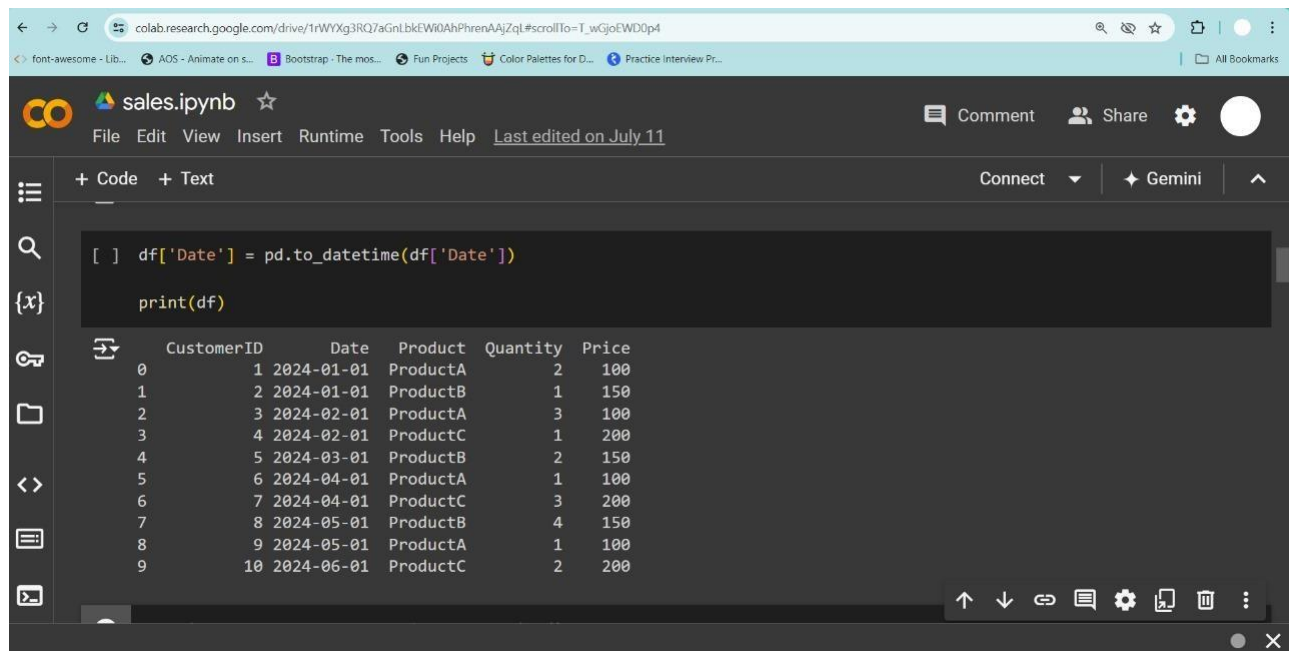
**Handling Missing Values:** The first step in data cleaning is to address any missing values within the dataset. Missing values can arise due to various reasons, such as incomplete data entry or data corruption. By using the `dropna()` method, we ensure that any rows containing missing values are removed from the dataset. This step is crucial to prevent any inaccuracies or biases in subsequent analysis.

**Removing Duplicates:** The presence of duplicate records can skew the analysis and lead to incorrect conclusions. The `drop_duplicates()` method identifies and removes any duplicate rows from the dataset. This step ensures that each sales transaction is unique, maintaining the integrity of the data.

#### Implications:

- **Data Quality:** Handling missing values and removing duplicates enhances the quality of the dataset, ensuring that the analysis is based on accurate and reliable data.
- **Accuracy:** By removing incomplete and duplicate records, we minimize the risk of errors and biases, leading to more precise and valid analytical results.
- **Efficiency:** Clean data enables more efficient and effective analysis, saving time and resources in the long run.

## >Normalize Data



The screenshot shows a Google Colab notebook interface. The top bar includes the Google Colab logo, the name 'sales.ipynb', and various icons for file management, comments, and sharing. Below the top bar, there are tabs for 'Code' and 'Text'. The main area displays a code cell with the following Python code:

```
[ ] df['Date'] = pd.to_datetime(df['Date'])  
print(df)
```

Below the code cell, the output of the code is displayed as a DataFrame. The DataFrame has five columns: 'CustomerID', 'Date', 'Product', 'Quantity', and 'Price'. The data is as follows:

	CustomerID	Date	Product	Quantity	Price
0	1	2024-01-01	ProductA	2	100
1	2	2024-01-01	ProductB	1	150
2	3	2024-02-01	ProductA	3	100
3	4	2024-02-01	ProductC	1	200
4	5	2024-03-01	ProductB	2	150
5	6	2024-04-01	ProductA	1	100
6	7	2024-04-01	ProductC	3	200
7	8	2024-05-01	ProductB	4	150
8	9	2024-05-01	ProductA	1	100
9	10	2024-06-01	ProductC	2	200

### Interpretation:

**Normalizing Dates:** Data normalization involves converting data into a consistent and standardized format. In this case, normalizing the 'Date' column ensures that all dates are in a uniform datetime format. This step is essential for accurate time-based analysis, as inconsistent date formats can lead to errors in interpreting and analyzing temporal data.

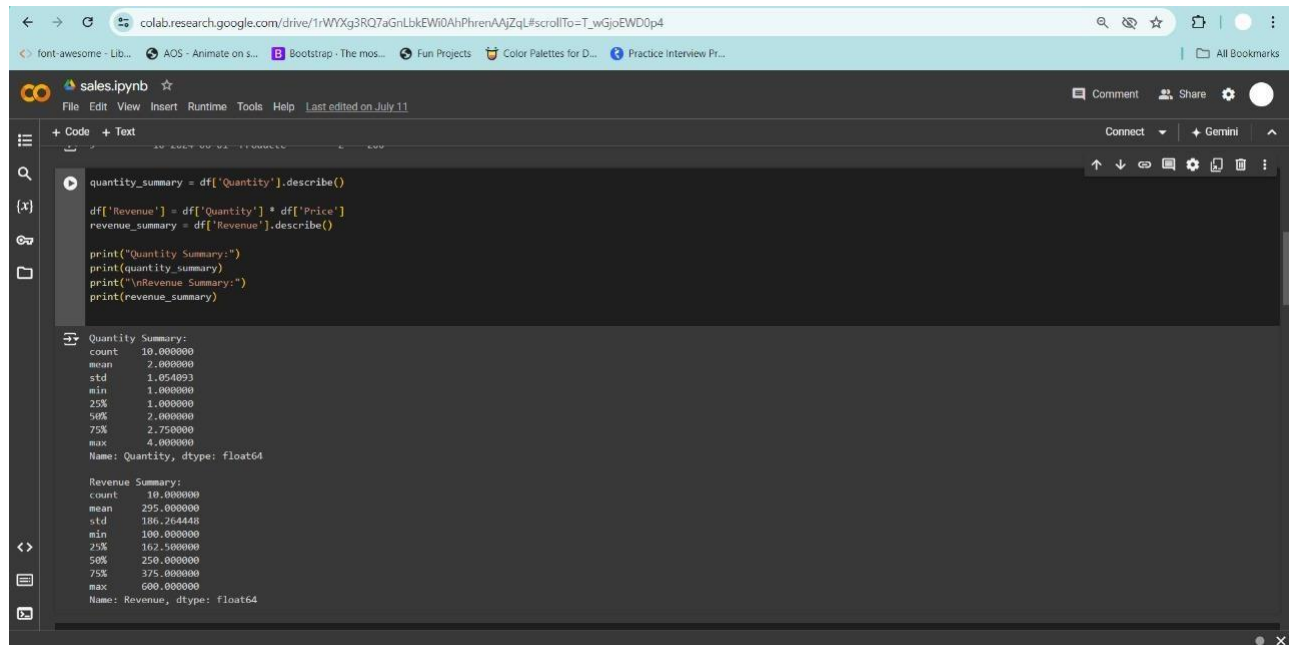
### Implications:

- **Consistency:** Normalizing the date format ensures consistency across the dataset, making it easier to compare and analyze time-based trends.
- **Accuracy:** Accurate date formats allow for precise time-series analysis, enabling businesses to identify seasonal patterns, trends, and anomalies in sales data.
- **Data Integration:** Consistent date formats facilitate the integration of the dataset with other data sources, enhancing the scope and depth of the analysis.

## 3.2 Exploratory Data Analysis (EDA)

### >Calculate summary statistics

#### Interpretation:



```
quantity_summary = df['Quantity'].describe()

df['Revenue'] = df['Quantity'] * df['Price']
revenue_summary = df['Revenue'].describe()

print("Quantity Summary:")
print(quantity_summary)
print("\nRevenue Summary:")
print(revenue_summary)
```

Quantity Summary:

count	10.000000
mean	2.000000
std	1.054093
min	1.000000
25%	1.000000
50%	2.000000
75%	2.750000
max	4.000000
Name: Quantity, dtype: float64	

Revenue Summary:

count	10.000000
mean	295.000000
std	186.264448
min	100.000000
25%	102.500000
50%	250.000000
75%	375.000000
max	600.000000
Name: Revenue, dtype: float64	

**Quantity Summary:** The summary statistics for sales quantity provide a comprehensive overview of the distribution and central tendency of the quantity of products sold. Key statistics include:

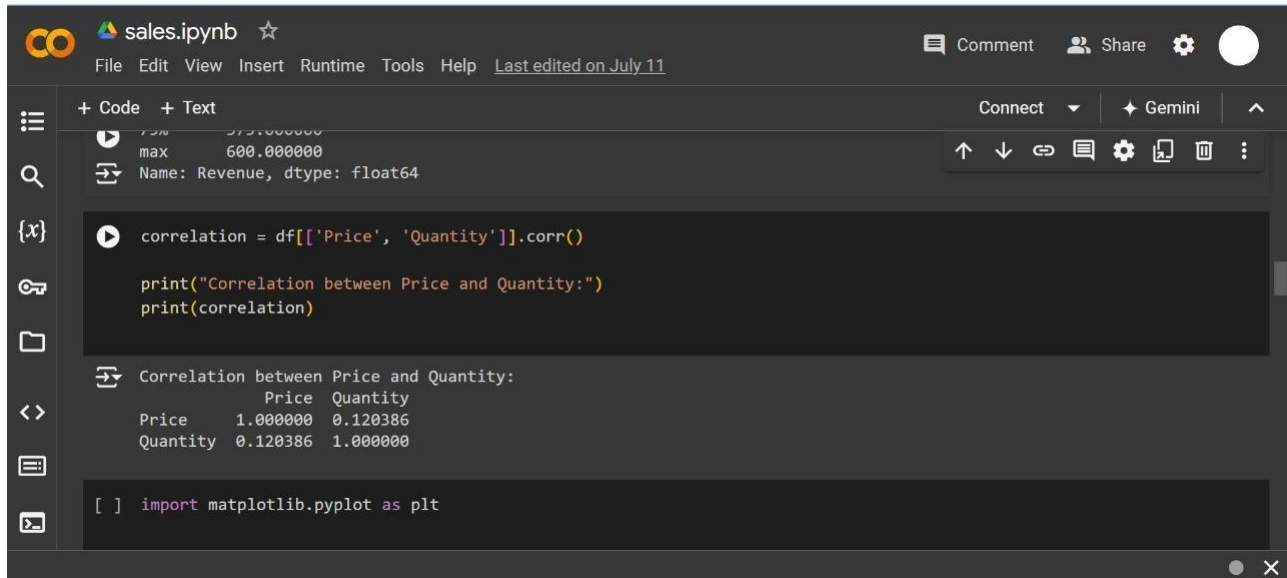
- **Count:** The total number of sales transactions.
- **Mean:** The average quantity of products sold per transaction.
- **Standard Deviation:** The variation in the quantity of products sold.
- **Min, 25%, 50%, 75%, Max:** These values represent the minimum, first quartile, median, third quartile, and maximum quantity sold, respectively.

**Revenue Summary:** Similarly, the summary statistics for revenue offer insights into the financial performance of sales transactions. Revenue is calculated by multiplying the quantity of products sold by their respective prices.

#### Implications:

- **Business Performance:** These statistics provide a clear picture of overall sales performance and help identify trends and patterns in sales quantity and revenue.
- **Resource Allocation:** Understanding the distribution of sales quantities and revenue can assist in better resource allocation, inventory management, and sales forecasting.

## >Correlation Analysis



The screenshot shows a Jupyter Notebook titled 'sales.ipynb'. The code cell contains the following Python code:

```
correlation = df[['Price', 'Quantity']].corr()

print("Correlation between Price and Quantity:")
print(correlation)
```

The output of the code is displayed below the code cell:

```
Correlation between Price and Quantity:
      Price  Quantity
Price  1.000000  0.120386
Quantity 0.120386  1.000000
```

Below the output, there is a code cell with the following code:

```
[ ] import matplotlib.pyplot as plt
```

### Interpretation:

**Correlation between Price and Quantity:** The correlation analysis examines the relationship between product price and sales quantity. The resulting correlation coefficient indicates the strength and direction of this relationship:

- **Positive Correlation:** A positive coefficient suggests that as the price increases, the quantity sold also increases.
- **Negative Correlation:** A negative coefficient suggests that as the price increases, the quantity sold decreases.
- **Zero Correlation:** A coefficient close to zero indicates no significant relationship between price and quantity.

### Implications:

- **Pricing Strategy:** If the correlation is negative, it implies that higher prices might lead to a decrease in sales quantity. This insight can inform pricing strategies to optimize revenue.
- **Sales Planning:** Understanding the relationship between price and quantity helps in making data-driven decisions on product pricing and sales planning.

## 3.3 Data Visualization

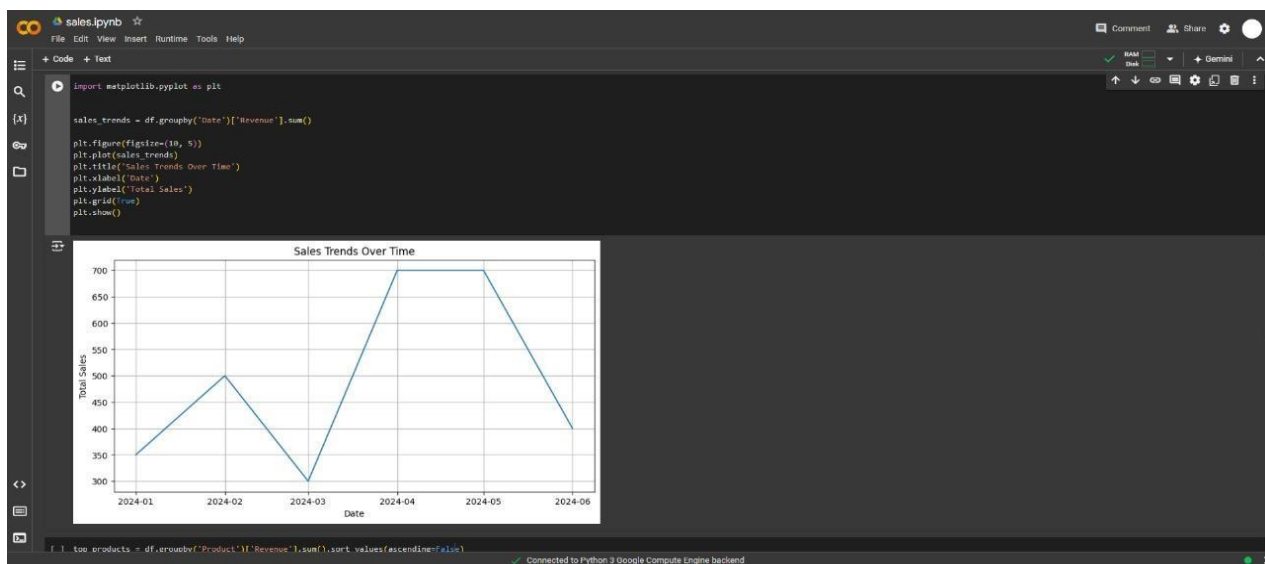
### >Sales Trends Over Time

**Interpretation:** The line chart depicting sales trends over time reveals significant fluctuations in sales activity. Notably, there is a pronounced peak in sales during the early part of January. This surge is likely due to New Year promotions or holiday season sales, which typically drive higher consumer spending.

### Implications:

- **Seasonal Planning:** Recognizing these peak periods allows businesses to plan and allocate resources effectively, ensuring sufficient inventory and staffing levels during high-demand times.
- **Marketing Strategies:** Marketing efforts can be intensified during these peak periods to maximize sales. For instance, special promotions, discounts, and advertising campaigns can be strategically launched to capitalize on increased consumer activity.

*Line Chart Showing Sales Trends Over Time*



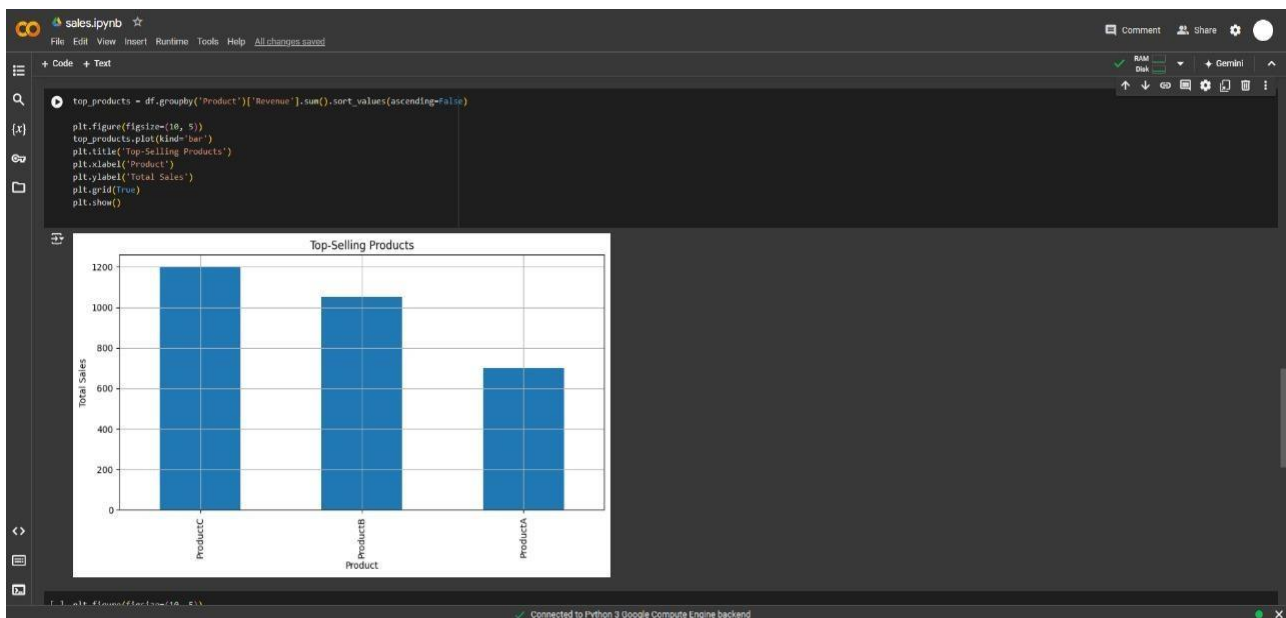
## >Top-Selling Products

**Interpretation:** The bar chart identifies the top-selling products, with Product A and Product C standing out as the most popular items. This indicates a strong customer preference for these products, which contribute significantly to the overall revenue.

### Implications:

- **Inventory Management:** Stocking levels for Product A and Product C should be carefully monitored and maintained to prevent stockouts. Ensuring the availability of these high-demand products is crucial for sustaining sales.
- **Promotional Focus:** Marketing campaigns can prioritize these top-selling products to further boost their sales. Highlighting these popular items in advertisements and promotions can attract more customers and increase revenue.
- **Product Analysis:** Analysing the features and benefits of these top-selling products can provide insights into what drives customer preferences. This information can guide the development and enhancement of other products.

### *Bar Chart Displaying Top-Selling Products*



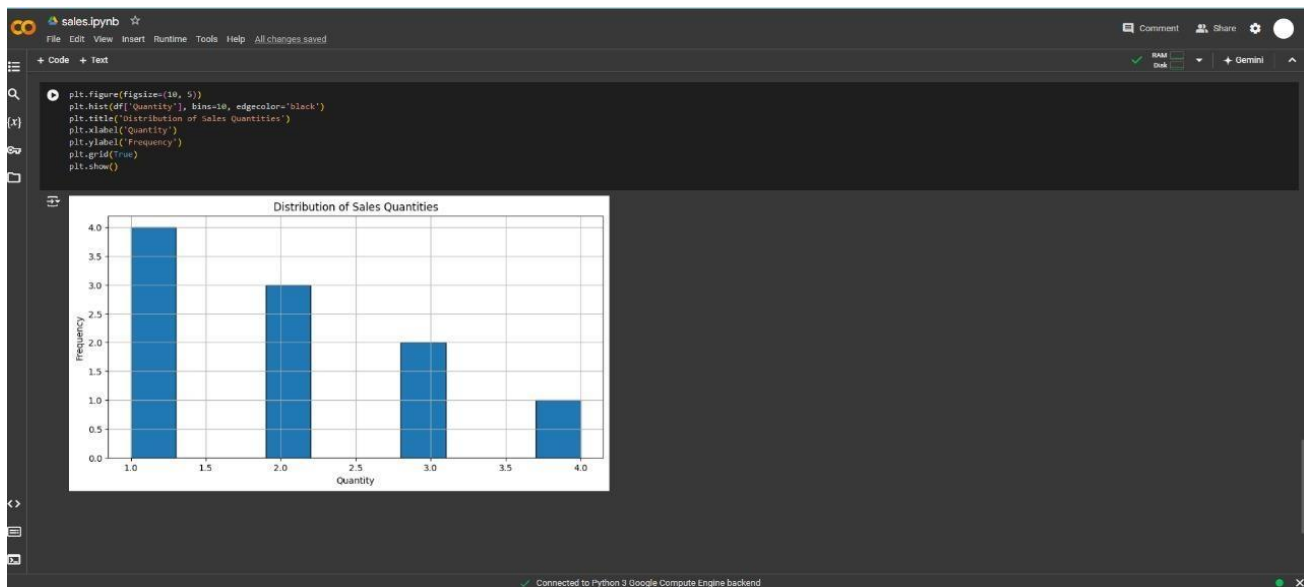
## >Distribution of Sales Quantities

**Interpretation:** The histogram illustrates the distribution of sales quantities, showing that most transactions involve small quantities. The frequency of sales decreases as the quantity increases, suggesting that customers typically purchase in lower volumes.

### Implications:

- **Customer Purchasing Behaviour:** Understanding that most customers buy in small quantities can help businesses tailor their sales strategies. For instance, offering bulk purchase discounts or bundle deals might encourage customers to buy more in a single transaction.
- **Pricing Strategies:** Pricing strategies can be adjusted to incentivize larger purchases. For example, tiered pricing or volume discounts can be introduced to make larger quantities more attractive to customers.
- **Inventory Planning:** Knowing the common purchase quantities can aid in inventory planning. Stock levels can be optimized based on typical sales volumes, reducing the risk of overstocking or understocking.

## Histogram Visualizing the Distribution of Sales Quantities





# CHAPTER 4

## CONCLUSION AND FUTURE WORK

### 4.1 CONCLUSION

The Customer Sales Analysis project provides valuable insights into customer behavior and sales trends, aiding businesses in making informed decisions. Here are the key takeaways from the project:

1. **Data Collection and Cleaning:** The sales data was successfully imported, cleaned, and normalized. This step ensured the dataset was free from missing values and duplicates, and all dates were in a consistent format, which is crucial for accurate analysis.
2. **Exploratory Data Analysis (EDA):** The EDA provided comprehensive summary statistics for sales quantity and revenue. It also highlighted the correlation between product price and sales quantity, revealing patterns and relationships within the data.
3. **Data Visualization:** Visualizations such as line charts, bar charts, and histograms offered clear and insightful representations of sales trends over time, top-selling products, and the distribution of sales quantities.

### 4.2 Future Work

1. **Advanced Predictive Analytics:**
  - **Sales Forecasting:** Implement advanced predictive models such as ARIMA, Prophet, or machine learning algorithms to forecast future sales trends
  - **Customer Segmentation:** Use clustering algorithms like K-means or hierarchical clustering to segment customers based on their purchasing behavior
2. **Real-time Analytics:**
  - **Real-time Data Processing:** Implement real-time data processing to provide up-to-date insights and enable businesses to respond quickly to changing market conditions.
  - **Dashboards and Reports:** Develop interactive dashboards and automated reports for real-time monitoring of sales performance. Tools like Power BI or Tableau can be utilized for this purpose.
3. **Advanced Visualization Techniques:**
  - **Geospatial Analysis:** Use geospatial visualization to analyze sales data by location. This will help in understanding regional sales performance and identifying potential market opportunities.
4. **Machine Learning Applications:**
  - **Recommendation Systems:** Develop recommendation systems to suggest products to customers based on their purchase history and preferences.

## REFERENCES

1. Aripnammal, S. and Natarajan, S. (1994) 'Transport Phenomena of Sm Sel – X Asx', Pramana – Journal of Physics Vol.42, No.1, pp.421-425.
2. Barnard, R.W. and Kellogg, C. (1980) 'Applications of Convolution Operators to Problems in Univalent Function Theory', Michigan Mach, J., Vol.27, pp.81–94.
3. **Keep editing with the same format**