**🔧 Data Science Project Steps**

**📥 Step 2: Data Collection**

**→ ডেটা কোথা থেকে, কীভাবে আনছো তা নির্ধারণ।**

**✅ কাজ:**

* ডেটা ফরম্যাট: .csv, .xlsx, .json, API বা SQL থেকে
* Source:
  + [Kaggle](https://www.kaggle.com/)
  + সরকারি ওপেন ডেটা (data.gov.bd)
  + Custom data manually তৈরি
  + Web scraping (যদি ওয়েব থেকে ডেটা আনা হয়)

**Describe about Data and column details**

**📄 উদাহরণ:**

**Dataset**: superstore.csv from Kaggle  
**Shape**: 10,000 rows, 15 columns  
**Fields**: Product Name, Sales, Profit, City, Category, Order Date

**🧹 Step 3: Data Cleaning**

**✅ Data Cleaning A to Z**

🔹 1. Load the Dataset

df = pd.read\_csv('your\_file.csv')

df.head()

আগে df\_cleaned = df.copy() করে নাও, যেনো মূল ডেটা overwrite না হয়।

🔹 2. Check Data Info & Shape

df.info() # Data types, non-null counts

df.shape # Rows x Columns

df.columns # Column names

df.dtypes # Data types only

🔹 3. Check for Missing Values

df.isnull().sum()

✅ Clean Missing Values:

* Drop rows:

df.dropna(inplace=True)

* Fill with mean/median/mode:

df['Age'].fillna(df['Age'].mean(), inplace=True)

* Fill with a fixed value:

df['Gender'].fillna('Unknown', inplace=True)

**✅ Tidiness মানে কী?**

ডেটা যেনো **গোছানো থাকে** — যেমন একটা পরিপাটি টেবিল।

**📋 সহজ নিয়ম (Tidy Data এর ৩টি শর্ত):**

1. **প্রতিটি Column = একটি ধরন (Variable)**  
   যেমনঃ name, subject, marks
2. **প্রতিটি Row = একটি তথ্য (Observation)**  
   যেমনঃ Rina, Math, 80
3. **প্রতিটি Cell = একটি মান (Value)**  
   যেমনঃ 80

**❌ উদাহরণ (Untidy Data):**

| **name** | **math\_marks** | **science\_marks** |
| --- | --- | --- |
| Rina | 80 | 90 |

এখানে subject-গুলো column হয়ে গেছে — **tidy না**।

**✅ উদাহরণ (Tidy Data):**

| **name** | **subject** | **marks** |
| --- | --- | --- |
| Rina | Math | 80 |
| Rina | Science | 90 |

এখন সব কিছু পরিপাটি — **tidy**।

🔹 4. Drop Unnecessary Columns

df.drop(['Postal Code', 'Customer ID'], axis=1, inplace=True)

🔹 5. Remove Duplicates

df.duplicated().sum()

df.drop\_duplicates(inplace=True)

🔹 6. Convert Data Types

df['quantity'] = df['quantity'].astype(int)

🔹 7. Rename Columns

df.rename(columns={'cust\_name': 'customer\_name'}, inplace=True)

🔹 8. Rearrange Column Order

df = df[['order\_id', 'order\_date', 'product\_name', 'sales', 'profit']]

🔹 9. Save the Cleaned Data

df.to\_csv('cleaned\_data.csv', index=False)

**📊 Step 4: Exploratory Data Analysis (EDA)**

**EDA: কিভাবে বুঝব ডেটাতে সমস্যা আছে?**

|  |  |  |
| --- | --- | --- |
| 5️⃣ Inconsistent Categories | df['col'].unique() | 'Tech', 'tech', 'Technology' আলাদা হিসেবে আছে |

|  |  |  |
| --- | --- | --- |
| 6️⃣ Unexpected Zeros or Negatives | df[df['sales'] < 0] | Sales বা Quantity নেগেটিভ যেটা সম্ভব না |

|  |  |  |
| --- | --- | --- |
| 7️⃣ Imbalanced Category | value\_counts(normalize=True) | একটা class অনেক বেশি, অন্যগুলো খুব কম |

|  |  |  |
| --- | --- | --- |
| 8️⃣ Mixed Format | Text-এ number/ID মিশে আছে | যেমন: cust\_01, tem\_10\_pat |

|  |  |  |
| --- | --- | --- |
| 9️⃣ No Variation | df['col'].nunique() | Column এ সব value একই / খুব কম variety |

|  |  |  |
| --- | --- | --- |
| 🔟 Invalid Values | df['age'].min(), max() | বয়স যদি 500 দেখায় → স্পষ্ট ভুল |

| **ধাপ** | **কী চেক করব** | **সমস্যা বুঝবো কিভাবে?** | **সমাধান কী?** |
| --- | --- | --- | --- |
| **3. Univariate Analysis** | histplot, boxplot, value\_counts() | 🔹 Extreme value (outlier) 🔹 Skewed distribution 🔹 Spelling mistake in category ('Tech', 'tech') | 🔹 Category ঠিক করো: replace() 🔹 Distribution ঠিক করতে log/scale ব্যবহার 🔹 Outlier চিহ্নিত করো |
| **4. Bivariate Analysis** | scatterplot, boxplot, barplot | 🔹 Expected relation নেই (e.g. Sales vs Profit) 🔹 Category অনুযায়ী distribution mismatch | 🔹 নতুন feature বানাও 🔹 Group-based analysis করো |
| **5. Multivariate Analysis** | pairplot, groupby, pivot | 🔹 কোনো pattern নাই 🔹 Group-wise trend নেই | 🔹 Time features যোগ করো (month, year) 🔹 Aggregation ভালো করে করো |
| **6. Outlier Detection** | boxplot, IQR method | 🔹 অনেক dot বা line-এর বাইরে মান 🔹 মান সম্পূর্ণ realistic না (e.g., sales = 999999) | 🔹 IQR method দিয়ে বাদ 🔹 Capping বা replace |
| **7. Correlation Matrix** | df.corr(), heatmap | 🔹 কোনো feature এর সাথে কোনো সম্পর্ক নেই → useless 🔹 High correlation → multicollinearity | 🔹 Useless column drop 🔹 High-correlation column combine/one drop |

🔹 3. Understand Data Shape and Types

df.shape # (rows, columns)

df.columns # Column names

df.info() # Dtype, non-null values

df.dtypes # Just data types

df.describe() # Summary stats for numeric columns

**🔹 5. Univariate Analysis (একটি Column বিশ্লেষণ)**

→ যখন আমরা শুধুমাত্র **একটি Column** নিয়ে তার **distribution, frequency বা summary statistics** বিশ্লেষণ করি।

➤ Numerical Columns:

df['sales'].plot(kind=’hist’, bins=30)

sns.kdeplot(df['sales'])

df['profit'].skew()

df['profit'].plot(kind='box') for outlier

df['profit'] >2000

➤ Categorical Columns:

df['category'].value\_counts().plot(kind='bar/pie')

sns.countplot(x='region', data=df) look like bar chart

🔹 6. Bivariate Analysis (Two Columns)

➤ Numerical vs Numerical:

plt.scatter(df['sales'], df['profit'])

sns.regplot(x='sales', y='profit', data=df)

➤ Categorical vs Numerical:

sns.boxplot(x='category', y='sales', data=df)

kde

sns.barplot(x='region', y='profit', data=df, estimator=sum)

➤ Categorical vs Categorical:

pd.crosstab(df['region'], df['ship\_mode']).plot(kind='bar', stacked=True)

**7. Multivariate Analysis (তিন বা তার বেশি Column বিশ্লেষণ)**

**🎯 কি?**

→ যখন আমরা **তিন বা তার বেশি column একসাথে** বিশ্লেষণ করি।

**১: Pairplot (multiple numeric columns)**

sns.pairplot(df[['sales', 'profit', 'discount']])

**২: Category-wise trend over time**

df['order\_date'] = pd.to\_datetime(df['order\_date'])

df['month'] = df['order\_date'].dt.to\_period('M')

monthly\_sales = df.groupby(['month', 'category'])['sales'].sum().unstack()

monthly\_sales.plot()

🔹 8. Correlation Matrix

corr = df.corr()

corr=df.cor().[‘sales’]

sns.heatmap(corr, annot=True, cmap='coolwarm')

🔹 9. Groupby Aggregation

df.groupby('category')['sales'].sum().sort\_values(ascending=False)

df.groupby(['category', 'region'])['profit'].mean()

🔹 10. Time Series Analysis (if Date column exists)

df['order\_date'] = pd.to\_datetime(df['order\_date'])

df['month'] = df['order\_date'].dt.to\_period('M')

df.groupby('month')['sales'].sum().plot(kind='line')

**🔹 3. Outliers (অস্বাভাবিক মান)**

🔍 **কিভাবে বুঝবো?**

✅ যদি দেখি boxplot-এ অনেক "dots" বা "বাহিরে থাকা মান" → বুঝবো **Outlier** আছে

**✅ Feature Engineering A to Z**

**Feature Engineering** মানে হলো:

👉 ডেটার বিদ্যমান কলামগুলো থেকে নতুন ইনফরমেটিভ কলাম তৈরি করা  
👉 আগের ফিচারগুলো পরিবর্তন/ক্লিন করে বিশ্লেষণের উপযোগী করে তোলা  
👉 ডেটা থেকে যতটা সম্ভব অর্থপূর্ণ তথ্য বের করে আনা

**Feature = Column**  
**Engineering = Modify / Create / Transform**

**🔹 Step 1: Data বুঝে নাও**

import pandas as pd

df = pd.read\_csv("data.csv")

df.info()

df.describe()

df.head()

👉 এখানে তুমি datatype, missing values, range, type বুঝে নাও।

**🔹 Remove unwanted text from numeric string.**

Like 123sp =123

df['column\_name'] = df['column\_name'].str.extract('(\d+)')

Handle Inconsistent Categories

df['category'].unique()

**# Fix spelling:**

df['category'] = df['category'].replace({'Tech': 'Technology'})

**🔹 Step 2: টাইম-ভিত্তিক Feature Extraction**

(যদি Date column থাকে)

df['order\_date'] = pd.to\_datetime(df['order\_date'])

✅ Useful for time-trend analysis.

df['year'] = df['order\_date'].dt.year

df['month'] = df['order\_date'].dt.month

df['day'] = df['order\_date'].dt.day

df['weekday'] = df['order\_date'].dt.day\_name()

df['is\_weekend'] = df['weekday'].isin(['Saturday', 'Sunday'])

**✅ Text-Based Feature Engineering**

ধরো তোমার কলামের নাম হলো 'customer\_id'  
 যার ভ্যালু আছে এরকম: "cust\_01", "cust\_02", ...

df['customer\_id'] = df['customer\_id'].str.replace('cust\_', '', regex=False)  
 এখন "cust\_01" → "01" হয়ে যাবে।

**if you want to add more things**

df['name'] = df['name'].str.replace('suma', 'suman', regex=False)

**🔹 String Rearrangement using Regex (Text-Based Feature Engineering)**

ধরো তোমার কলামের নাম 'item\_code':

# উদাহরণ ডেটাসেট

df = pd.DataFrame({

'item\_code': ['tem\_10\_pat', 'tem\_25\_pat', 'tem\_99\_pat']

})

# Regex দিয়ে extract করে নতুন ফরম্যাট বানানো

df['formatted\_code'] = df['item\_code'].str.extract(r'tem\_(\d+)\_([a-zA-Z]+)')\

.apply(lambda x: f'{x[1]}-{x[0]}', axis=1)

print(df)

**Step 3: Mathematical Feature Creation**

✅ Sales dataset বা Finance dataset-এ খুব দরকারি।

df['profit\_margin'] = df['profit'] / df['sales']

df['total\_cost'] = df['sales'] - df['profit']

**🔹 Step 4: Text থেকে নতুন Feature**

(যদি text/string column থাকে)

df['name\_length'] = df['product\_name'].apply(lambda x: len(str(x)))

df['has\_technology'] = df['product\_name'].str.contains("Technology", case=False)

✅ NLP বা product dataset-এ দরকারি।

**🔹 5. Use Regular Expression to Separate**

import re

# Function to split phone and email

def split\_contact(text):

phone = re.search(r'\d{11}', text)

email = re.search(r'[\w\.-]+@[\w\.-]+', text)

return pd.Series([phone.group() if phone else None,

email.group() if email else None])

# Apply to DataFrame

df[['phone', 'email']] = df['contact\_info'].apply(split\_contact)

**✅ Output:**

| **contact\_info** | **phone** | **email** |
| --- | --- | --- |
| 01723445566senarul@gmail.com | 01723445566 | senarul@gmail.com |
| 01723445566, senarul@gmail.com | 01723445566 | senarul@gmail.com |
| 01723445566 senarul@gmail.com | 01723445566 | senarul@gmail.com |
| 01876543210 - someone@example.com | 01876543210 | someone@example.com |

**🔹 Step 6: Group-based Feature**

df['customer\_total\_sales'] = df.groupby('customer\_id')['sales'].transform('sum')

df['city\_avg\_profit'] = df.groupby('city')['profit'].transform('mean')

✅ Powerful for personalized or location-based analysis.

**🔹 Step 7: Binning (Continuous → Category)**

df['sales\_level'] = pd.cut(df['sales'],

bins=[0, 200, 500, 1000, float('inf')],

labels=['Low', 'Medium', 'High', 'Very High'])

✅ Helps with visualization or segmentation.

**🔹 Step 9: Interaction Features (combine two+ features)**

df['sales\_discount\_interaction'] = df['sales'] \* df['discount']

✅ Sometimes hidden patterns emerge this way.

**📋 Step 5: Insight Report & Visualization**

**→ সব findings report আকারে সাজানো।**

**✅ কাজ:**

* Key Findings (Bullet points)
* Charts + explanation
* Recommendations (কী সিদ্ধান্ত নেওয়া যায়)
* Export as PDF or Markdown for GitHub

**📄 Report Example (English):**

* 📌 Sales are highest in California and New York.
* 📌 Technology products bring the highest profit.
* 📌 Standard Class shipping is used 60% of the time.  
  ✅ Recommendation: Focus on tech products and optimize shipping time.

**📂 Project Folder Structure: for github**

📁 Superstore-Sales-Analysis/

├── 📄 superstore\_analysis.ipynb

├── 📁 dataset/

│ └── superstore.csv

├── 📊 images/ (charts, screenshots)

├── 📄 final\_report.pdf

├── 📄 README.md

└── 📄 requirements.txt

**📝 README.md Summary Example:**

# 📊 Superstore Sales Data Analysis

## 🧠 Project Objective

Analyze sales data to extract business insights and provide actionable recommendations.

## 📁 Dataset

- Source: Kaggle

- Records: 9994

- Features: Product, Region, Sales, Profit, Category, etc.

## 🛠 Tools

- Python (Pandas, Matplotlib, Seaborn)

- Jupyter Notebook

## 🔍 Key Insights

- California had the highest sales.

- Technology category gave highest profit.

- High discounts usually reduce profit.

## 📋 Recommendations

- Focus on technology products.

- Limit discount to maintain profit.

**📄 PDF Report-এ যা থাকবে (Standard Structure)/ Report লিখো Word save as pdf**

**1. Title Page**

📊 Superstore Sales Data Analysis

By: [Your Name]

Date: July 2025

**2. Table of Contents**

1. Introduction

2. Objective

**✅ প্রফেশনাল রিপোর্ট ফর্ম্যাটে এটা হয় এমনভাবে:**

**📘 2. Objective**

The goal of this project is to analyze the sales data to discover trends, patterns, and make actionable business decisions.

**Key Business Questions (Problem Statements)**

Here are the core questions we aim to answer in this project:

* **Q1:** Which product categories generate the most profit?
* **Q2:** Which region has the highest sales?
* **Q3:** How does discount affect profit?
* **Q4:** What is the monthly trend of sales and profit?
* **Q5:** Who are the top 5 customers by revenue?
* **Q6:** Is there any seasonality in the sales data?
* **Q7:** Which shipping mode is most used?

3. Dataset Description

4. Data Cleaning

**✅ 1. Missing Value Check**

df.isnull().sum()

**🖨 Output (Example):**

Order ID 0

Order Date 0

Ship Date 0

Customer Name 0

Sales 0

Discount 2

Profit 0

dtype: int64

➡️ এই Output থেকে বোঝা যাচ্ছে, **Discount** column-এ ২টা missing value আছে।

**✅ 2. Convert Date Format**

df['Order Date'] = pd.to\_datetime(df['Order Date'])

🔍 Output দেখাতে পারো df.dtypes দিয়ে:

df.dtypes

**🖨 Output:**

Order ID object

Order Date datetime64[ns]

Ship Date object

Customer Name object

Sales float64

➡️ এতে বোঝা যায় Order Date সফলভাবে datetime এ convert হয়েছে।

**✅ 3. Remove Duplicates**

df.drop\_duplicates(inplace=True)

✅ Output দেখানোর জন্য আগে-পরে row count দেখাও:

print("Before:", df.shape)

df.drop\_duplicates(inplace=True)

print("After :", df.shape)

**🖨 Output:**

Before: (9994, 21)

After : (9978, 21)

➡️ বোঝা গেল, **16টি duplicate row** remove হয়েছে।

**✅ কেন Output দরকার?**

| **কারণ** | **ব্যাখ্যা** |
| --- | --- |
| 📊 প্রমাণ | রিপোর্টে দেখাতে হবে তুমি সত্যিই কাজ করেছো |
| 📉 Before vs After | কতটুকু missing/duplicate ছিল, বোঝা যায় |
| 👨‍🏫 Instructor/Interviewer Friendly | যারা রিপোর্ট পড়বে, তারা স্পষ্ট বুঝবে |

5. Exploratory Data Analysis

**📊 Sales Distribution**

df['Sales'].hist(bins=30)

(Insert Histogram Image Here)

**📊 Category vs Profit**

import seaborn as sns

sns.barplot(x='Category', y='Profit', data=df)

(Insert Barplot Image Here)

**📊 Discount vs Profit (Scatter)**

sns.scatterplot(x='Discount', y='Profit', data=df)

6. Key Insights

7. Visualization

**Monthly Sales Trend**

df.groupby(df['Order Date'].dt.to\_period('M'))['Sales'].sum().plot()

(Insert Line Chart Here)

……………………………………………………………………………………………………………………………………………

8. Conclusion & Recommendation

9. Tools Used

**3. Sample Report Content:**

**🧠 Introduction**

This project aims to analyze Superstore sales data and generate business insights regarding products, sales regions, and customer trends.

**🎯 Objective**

Identify which products and regions are the most profitable, and understand how discounts affect profit.

**📊 Dataset Description**

* Rows: 9,994
* Columns: 13
* Source: Kaggle
* Fields: Order Date, Category, City, Sales, Profit, Discount, etc.

**🧹 Data Cleaning**

* Removed 119 missing records
* Converted date column to datetime format
* Dropped unnecessary columns like Postal Code

**📊 Exploratory Data Analysis (EDA)**

* California is the top-performing state in terms of sales.
* Technology is the most profitable category.
* Discounts >20% lead to a sharp drop in profit.

**📈 Charts (Add Screenshots)**

* Bar Chart: City vs Sales
* Heatmap: Correlation between Sales, Profit, Discount
* Line Chart: Monthly Sales Trend

**📋 Conclusion & Recommendation**

* Focus more on California & Technology Category
* Optimize discount rates to prevent profit loss

**✅** **Tools Used**

* Python (Pandas, Seaborn, Matplotlib)
* Jupyter Notebook
* Dataset: Kaggle – Superstore

**Project Report Structure with Placement of 16 Questions**

**🟨 1. Title Page**

* Project Name
* Your Name / Team
* Institution / Company
* Date

**🟨 2. Abstract / Executive Summary**

* সংক্ষিপ্ত বিবরণ: এই প্রজেক্টে কি হয়েছে, কেন হয়েছে, এবং কি ফলাফল এসেছে।

**🟨 3. Objectives / Goals**

* আপনি যে 16টি প্রশ্ন সলভ করবেন, তার **সংক্ষিপ্ত উদ্দেশ্য** এখানে bullet আকারে দিন:

**Example:**

* + To identify top-performing product categories
  + To analyze discount effect on sales
  + To detect seasonal sales patterns
  + To understand payment method usage trends

**🟨 4. Dataset Description**

* Dataset Source
* Column Names + Descriptions
* Sample Data Table (optional)

**🟨 5. Data Cleaning & Preprocessing**

* কীভাবে missing values handle করেছেন
* কী কী column change/modification হয়েছে

**🟨 6. Exploratory Data Analysis (EDA)**

📍 এখানে শুধু basic statistics, distribution, correlation matrix ইত্যাদি থাকবে।

**🟨 🔴 7. Business Questions & Visual Analysis (Main Section)**

🟢 **এই সেকশনেই আপনি আপনার 16টি প্রশ্ন রাখবেন। প্রতিটি প্রশ্ন আলাদা সাবসেকশন হবে।**

**🧱 Structure:**

markdown

CopyEdit

7.1 Question 1: Which product categories generate the highest total sales?

- Graph: Bar Chart

- Insight: Furniture leads with 45% share, followed by Technology.

- Recommendation: Increase stock for top categories.

7.2 Question 2: What are the top 10 most sold items by quantity?

- Graph: Horizontal Bar Chart

- Insight...

- Recommendation...

...

7.16 Question 16: What is the trend of different payment method usage over time?

- Graph: Line Chart

- Insight...

- Recommendation...

📌 **Graph/Image add করবেন প্রতিটি প্রশ্নের নিচে।**

**🟨 8. Key Business Insights (Summary)**

* 16টি প্রশ্ন থেকে পাওয়া মূল বিষয়গুলো **সংক্ষেপে এক পাতায় লিখবেন**
* Bullet points এ দেওয়া ভালো

**🟨 9. Recommendations**

* কি করলে business performance improve করবে
* যে data থেকে যা শিখলেন তার actionable ideas

**🟨 10. Limitations**

* Data এর সীমাবদ্ধতা
* Analysis এ কি সমস্যা হয়েছে, future scope কি

**🟨 11. Conclusion**

* পুরো প্রজেক্টের সারাংশ
* ভবিষ্যতে কি করা যেতে পারে

**🟨 12. Appendix (Optional)**

* SQL Queries / Python Code
* Extra charts
* Raw data screenshot (if needed)

**Folder Structure (Professional & Clean)**

bash

CopyEdit

Retail-Sales-Analysis-Project/

│

├── 📁 data/ # Raw and cleaned datasets

│ ├── raw\_data.csv

│ └── cleaned\_data.csv

│

├── 📁 notebooks/ # Jupyter notebooks or analysis scripts

│ ├── 01\_data\_cleaning.ipynb

│ └── 02\_visual\_analysis.ipynb

│

├── 📁 images/ # Graphs, charts used in report or README

│ ├── q1\_sales\_by\_category.png

│ ├── q2\_top10\_items.png

│ └── ...

│

├── 📁 report/ # Final report, PDF or Word

│ ├── Retail\_Sales\_Report.pdf

│ └── Retail\_Sales\_Report.docx

│

├── 📁 presentation/ # PPT for presentation

│ └── Retail\_Sales\_Slides.pptx

│

├── 📁 dashboard/ # Power BI (.pbix) or Tableau (.twbx) files

│ └── Retail\_Sales.pbix

│

├── README.md # Project Overview (VERY IMPORTANT)

├── LICENSE # Optional, like MIT

└── requirements.txt # Python packages used (if Python used)

**🪄 README.md — “চমক লাগানোর জায়গা” 💥**

একটি দারুন README হ’ল আপনার GitHub project এর **ভূমিকা, portfolio, portfolio piece, এবং personal brand**।

**✅ Ideal README Structure:**

markdown

CopyEdit

# 🛒 Retail Sales Analysis Project

📊 A complete end-to-end analysis of retail sales to uncover trends, patterns, and business opportunities using clean data visualizations and actionable insights.

---

## 🎯 Objectives

- Analyze product performance and category-wise revenue

- Understand discount impacts on quantity and profit

- Visualize time-based sales trends

- Identify most used payment methods

- Provide data-driven recommendations to improve business

---

## 📁 Project Structure

| Folder | Description |

|----------------|-------------------------------------|

| `data/` | Raw and cleaned dataset files |

| `notebooks/` | Jupyter notebooks for analysis |

| `images/` | Saved graphs and charts |

| `report/` | Final project report (PDF/DOCX) |

| `presentation/`| PowerPoint presentation |

| `dashboard/` | Power BI dashboard (.pbix) |

---

## 📊 Key Questions Answered (16 Total)

1. Which product categories generate the highest total sales? → 📈 Bar Chart

2. What are the top 10 most sold items? → 📊 Horizontal Bar Chart

3. ... \*(List all 16)\*

---

## 📷 Sample Visualizations

| Category Sales | Monthly Revenue Trend |

|-----------------------|--------------------------|

| ![q1](images/q1\_sales\_by\_category.png) | ![q7](images/q7\_monthly\_trend.png) |

---

## 📌 Tools Used

- \*\*Power BI\*\* for interactive dashboard

- \*\*Excel\*\* for data cleanup

- \*\*Python (Pandas, Matplotlib)\*\* for custom visualizations

- \*\*MS Word & PowerPoint\*\* for reporting

---

## ✅ Results & Insights

- 💡 Discounts boost quantity but reduce profit in some categories

- 📈 December has the highest revenue spike

- 💳 Credit card is the most preferred payment method

---

## 📎 Report & Resources

- 📄 [Final Report PDF](report/Retail\_Sales\_Report.pdf)

- 📊 [Power BI Dashboard](dashboard/Retail\_Sales.pbix)

- 🖥️ [Presentation Slides](presentation/Retail\_Sales\_Slides.pptx)

---

## 🙋‍♂️ About Me

I'm a passionate data analyst skilled in turning raw data into business gold.

🔗 [LinkedIn Profile](https://linkedin.com/in/yourname)

📫 Email: yourname@email.com