Financial Analysis

Tools Used: Excel and Python

**Key Tasks for Preprocessing**

1. **Column Names and Errors:**
   * Inspect and correct column names if they have errors or are not descriptive.
2. **Data Cleaning:**
   * **Remove Special Characters:**
     + $ sign and - from numerical columns.
     + Commas from numbers to ensure they are correctly interpreted as numeric values.
   * **Change Data Types:**
     + Convert columns from object (string) type to numeric types (integer or float) after cleaning.
3. **Data Analysis:**
   * Plot **sales** and **profit** with respect to the timeline to understand trends and patterns.

**Suitability for Your Project**

* **Financial Analysis:** The dataset seems to contain financial metrics that are essential for performing financial analysis.
* **Data Exploration:** The need to clean and preprocess the data provides an excellent opportunity for hands-on experience with data wrangling and exploration.
* **Visualization:** The dataset allows for visualization of sales and profit over time, which is useful for understanding financial performance and trends.

**Next Steps**

1. **Download and Inspect the Dataset:**
   * Load the dataset into your preferred data analysis tool (Python with Pandas, Excel, etc.).
   * Check for column names and errors, and perform necessary cleaning.
2. **Data Cleaning and Transformation:**
   * Remove unwanted characters ($, -, ,).
   * Convert appropriate columns to numeric types.
3. **Exploratory Data Analysis (EDA):**
   * Conduct EDA to gain insights into the financial metrics.
   * Create visualizations (e.g., time series plots for sales and profit).
4. **Documentation and Reporting:**
   * Document your findings and preprocess steps.
   * Prepare visualizations for reporting or further analysis.

## **Project Overview: Financial Data Analysis**

### ****Objective****

The project involves analyzing financial data from a company's dataset to gain insights into sales, profits, and other financial metrics. The dataset was cleaned and prepared for analysis using Python and Excel, and the analysis was performed to generate meaningful visualizations.

### ****Steps Taken****

1. **Data Import and Cleaning**
   * **Import Data**: Loaded the dataset from a CSV file into Python using pandas.
   * **Clean Column Names**: Removed extra spaces and special characters from column names.
   * **Clean Data Values**: Removed '$' signs, commas, and other non-numeric characters from numerical columns.
2. **Data Analysis**
   * **Preprocessing**: Converted cleaned data to numeric values where necessary.
   * **Visualization**: Used libraries like matplotlib to create visualizations such as line charts and trend lines to understand relationships between variables.
3. **Export Cleaned Data**
   * Saved the cleaned dataset back to the desktop for further use.

### ****Challenges Faced****

1. **File Handling Issues**
   * **Error**: Encountered a PermissionError when attempting to read or write files.
   * **Resolution**: Verified file paths and permissions, ensuring that the correct file path was used.
2. **Missing Modules**
   * **Error**: Faced a ModuleNotFoundError for matplotlib.
   * **Resolution**: Installed the missing module using pip install matplotlib.
3. **Data Cleaning Difficulties**
   * **Issue**: Column names contained extra spaces and special characters.
   * **Resolution**: Used Python functions to clean column names and Excel formulas to verify the cleaned data.
4. **File Path Errors**
   * **Issue**: Errors due to incorrect file paths or missing file extensions.
   * **Resolution**: Ensured accurate file paths and verified the presence of file extensions.

### ****Important Codes and Libraries****

* **Libraries Used**:
  + pandas: For data manipulation and cleaning.
  + matplotlib: For creating visualizations.
  + numpy: For numerical operations.
* **Key Codes**:

import pandas as pd

import matplotlib.pyplot as plt

# Load the dataset

df = pd.read\_csv('C:/Users/DELL/Desktop/Company Data Financial Analysis/Financials.csv')

# Clean column names

df.columns = df.columns.str.strip().str.replace(' ', '\_')

# Clean data values

df.replace({'\$': '', ',': '', '-': '0'}, regex=True, inplace=True)

# Convert columns to numeric

df['Sales'] = pd.to\_numeric(df['Sales'], errors='coerce')

# Save cleaned data

df.to\_csv('C:/Users/DELL/Desktop/Company Data Financial Analysis/Cleaned\_Financials.csv', index=False)

# Create a plot

plt.plot(df['Date'], df['Sales'])

plt.title('Sales Over Time')

plt.xlabel('Date')

plt.ylabel('Sales')

plt.show()