

Super Store

Project





Super Store Project

- Project Title: *Super Store Project*
- Group: *Gharbia Governorate – G1 Data 1 Graduates*
- Company Name: *ITC Tanta*
- Instructor: *Ahmed Ayman EL-Saey*
- Group Name: *G2*
- Members: *Zekrayat Wael*

Amira Al-Banna

Radwa Adel

Abdullah Muhammad

Abdul Hamid Muhammad



Week 1

1. Importing Required Libraries

▶ `import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns`

✓ *We import these to prepare the environment for data cleaning, analysis, and visualization.*

2. Loading the Dataset

▶ `df = pd.read_csv("Superstore Sales Dataset.csv")`

✓ *This converts our Excel-like dataset into a Data Frame called df for analysis.*

3. Display First 10 Rows

▶ `df.head(10)`

✓ *Used to understand how the data looks in the beginning.*

4. Display Last 10 Rows

▶ `df.tail(10)`

✓ *Ensures dataset is properly loaded and complete.*



Week 1

5. Automated Data Profiling Report

```
▶ !pip install ydata-profiling  
df = pd.read_csv("Superstore Sales Dataset.csv")  
from ydata_profiling import ProfileReport  
  
profile = ProfileReport(df, title="data Report", explorative=True)  
  
profile.to_notebook_iframe()
```

- This helps you understand our dataset deeply and detect issues early.*
-

6. Checking Data Information

```
▶ df.info()
```

- Used for initial inspection of dataset structure.*

7. Checking Number of Unique Values

```
▶ df.nunique()
```

- Helps detect duplicated IDs or columns with very few unique values.*



Week 1

8. Checking for Missing Values

▶ `df.isna().sum()`

Used to know where cleaning is required.

9. Filtering Rows with Missing Postal Code

▶ `filtered_df = df[df['Postal Code'].isnull()]
filtered_df`

Used to investigate missing data in Postal Code column.

10. Filtering Rows for a Specific City

▶ `filtered2_df = df[df['City'] == "Burlington"]
filtered2_df`

Useful for focused analysis and validation.

11. Converting Date Columns to Proper Date Format

▶ `df['Order Date'] = pd.to_datetime(df['Order Date'], format='%d/%m/%Y')
df['Ship Date'] = pd.to_datetime(df['Ship Date'], format='%d/%m/%Y')`

Essential for performing date-based analysis (like delivery time or sales trend).



Week 1

12. Checking Data Again After Conversion

▶ `df.info()`

✓ *Verification step after transformation.*

13. Cleaning and Formatting Postal Codes

▶ `df['Postal Code'] = df['Postal Code'].astype(str)`
`df['Postal Code'] = df['Postal Code'].replace('nan', '05405')`
`df['Postal Code'] = df['Postal Code'].str.replace(r'\.0$', '', regex=True)`

✓ *Ensures consistency and correctness of postal code data.*

14. Verifying Data After Cleaning

▶ `df.info()`

✓ *Final validation before export.*

15. Saving the Cleaned Data

▶ `df.to_excel("cleaned_data.xlsx", index=False)`

✓ *Final step: saving our cleaned data and ready-to-analyze dataset.*



References:

Data References: https://drive.google.com/drive/folders/1kVSkliWO0o6dl-o10qkkbjkFK7LI_Fyk

Codes References:

<https://colab.research.google.com/drive/1vkAX88uHliTGWz9YWxpPdQNOYBbfwFYe?usp=sharing#scrolledTo=cEnm86rnIC7o>