# data-analyst-internship-task8

November 14, 2025

[1]:
```
# 1. Load dataset (your file path)
import pandas as pd
file_path = r"D:\intern\Superstore.csv"
df = pd.read_csv(file_path, encoding='latin1')
print(df.head())
```

```
   Row ID        Order ID  Order Date   Ship Date       Ship Mode Customer ID  \
0       1  CA-2016-152156   11/8/2016  11/11/2016    Second Class    CG-12520
1       2  CA-2016-152156   11/8/2016  11/11/2016    Second Class    CG-12520
2       3  CA-2016-138688   6/12/2016   6/16/2016    Second Class    DV-13045
3       4  US-2015-108966  10/11/2015  10/18/2015  Standard Class    SO-20335
4       5  US-2015-108966  10/11/2015  10/18/2015  Standard Class    SO-20335

     Customer Name    Segment        Country             City  …  \
0      Claire Gute   Consumer  United States        Henderson  …
1      Claire Gute   Consumer  United States        Henderson  …
2  Darrin Van Huff  Corporate  United States      Los Angeles  …
3   Sean O'Donnell   Consumer  United States  Fort Lauderdale  …
4   Sean O'Donnell   Consumer  United States  Fort Lauderdale  …

  Postal Code  Region       Product ID         Category Sub-Category  \
0       42420   South  FUR-BO-10001798        Furniture    Bookcases
1       42420   South  FUR-CH-10000454        Furniture       Chairs
2       90036    West  OFF-LA-10000240  Office Supplies       Labels
3       33311   South  FUR-TA-10000577        Furniture       Tables
4       33311   South  OFF-ST-10000760  Office Supplies      Storage

                                        Product Name     Sales  Quantity  \
0                 Bush Somerset Collection Bookcase  261.9600         2
1  Hon Deluxe Fabric Upholstered Stacking Chairs,…  731.9400         3
2  Self-Adhesive Address Labels for Typewriters b…   14.6200         2
3     Bretford CR4500 Series Slim Rectangular Table  957.5775         5
4                   Eldon Fold 'N Roll Cart System   22.3680         2

   Discount    Profit
0      0.00   41.9136
1      0.00  219.5820
2      0.00    6.8714
```

```
3      0.45 -383.0310
4      0.20    2.5164

[5 rows x 21 columns]
```

[2]:
```
# 2. Convert Order Date to datetime
df["Order Date"] = pd.to_datetime(df["Order Date"], errors="coerce")
```

[3]:
```
#removing rows where data could not convert
df=df.dropna(subset=["Order Date"])
```

[4]:
```
# 3. Remove missing or wrong values in Sales / Profit
df=df.dropna(subset=["Sales"])
df=df[df["Sales"] >=0] #remove negative sales
df.dropna(subset=["Profit"])
```

[4]:

| | Row ID | Order ID | Order Date | Ship Date | Ship Mode |
|---|---|---|---|---|---|
| 0 | 1 | CA-2016-152156 | 2016-11-08 | 11/11/2016 | Second Class |
| 1 | 2 | CA-2016-152156 | 2016-11-08 | 11/11/2016 | Second Class |
| 2 | 3 | CA-2016-138688 | 2016-06-12 | 6/16/2016 | Second Class |
| 3 | 4 | US-2015-108966 | 2015-10-11 | 10/18/2015 | Standard Class |
| 4 | 5 | US-2015-108966 | 2015-10-11 | 10/18/2015 | Standard Class |
| … | … | … | … | … | … |
| 9989 | 9990 | CA-2014-110422 | 2014-01-21 | 1/23/2014 | Second Class |
| 9990 | 9991 | CA-2017-121258 | 2017-02-26 | 3/3/2017 | Standard Class |
| 9991 | 9992 | CA-2017-121258 | 2017-02-26 | 3/3/2017 | Standard Class |
| 9992 | 9993 | CA-2017-121258 | 2017-02-26 | 3/3/2017 | Standard Class |
| 9993 | 9994 | CA-2017-119914 | 2017-05-04 | 5/9/2017 | Second Class |

| | Customer ID | Customer Name | Segment | Country | City |
|---|---|---|---|---|---|
| 0 | CG-12520 | Claire Gute | Consumer | United States | Henderson |
| 1 | CG-12520 | Claire Gute | Consumer | United States | Henderson |
| 2 | DV-13045 | Darrin Van Huff | Corporate | United States | Los Angeles |
| 3 | SO-20335 | Sean O'Donnell | Consumer | United States | Fort Lauderdale |
| 4 | SO-20335 | Sean O'Donnell | Consumer | United States | Fort Lauderdale |
| … | … | … | … | … | … |
| 9989 | TB-21400 | Tom Boeckenhauer | Consumer | United States | Miami |
| 9990 | DB-13060 | Dave Brooks | Consumer | United States | Costa Mesa |
| 9991 | DB-13060 | Dave Brooks | Consumer | United States | Costa Mesa |
| 9992 | DB-13060 | Dave Brooks | Consumer | United States | Costa Mesa |
| 9993 | CC-12220 | Chris Cortes | Consumer | United States | Westminster |

| | … | Postal Code | Region | Product ID | Category | Sub-Category |
|---|---|---|---|---|---|---|
| 0 | … | 42420 | South | FUR-BO-10001798 | Furniture | Bookcases |
| 1 | … | 42420 | South | FUR-CH-10000454 | Furniture | Chairs |
| 2 | … | 90036 | West | OFF-LA-10000240 | Office Supplies | Labels |
| 3 | … | 33311 | South | FUR-TA-10000577 | Furniture | Tables |

```
4     …       33311   South  OFF-ST-10000760  Office Supplies      Storage
…   …         …     …              …               …          …
9989  …       33180   South  FUR-FU-10001889        Furniture  Furnishings
9990  …       92627    West  FUR-FU-10000747        Furniture  Furnishings
9991  …       92627    West  TEC-PH-10003645       Technology       Phones
9992  …       92627    West  OFF-PA-10004041  Office Supplies        Paper
9993  …       92683    West  OFF-AP-10002684  Office Supplies   Appliances

                                              Product Name     Sales  Quantity  \
0                        Bush Somerset Collection Bookcase  261.9600         2
1     Hon Deluxe Fabric Upholstered Stacking Chairs,…  731.9400         3
2     Self-Adhesive Address Labels for Typewriters b…   14.6200         2
3         Bretford CR4500 Series Slim Rectangular Table  957.5775         5
4                           Eldon Fold 'N Roll Cart System   22.3680         2
…                                                      …       …       …
9989                              Ultra Door Pull Handle   25.2480         3
9990  Tenex B1-RE Series Chair Mats for Low Pile Car…   91.9600         2
9991                               Aastra 57i VoIP phone  258.5760         2
9992  It's Hot Message Books with Stickers, 2 3/4" x 5"   29.6000         4
9993  Acco 7-Outlet Masterpiece Power Center, Wihtou…  243.1600         2

      Discount    Profit
0         0.00   41.9136
1         0.00  219.5820
2         0.00    6.8714
3         0.45 -383.0310
4         0.20    2.5164
…          …       …
9989      0.20    4.1028
9990      0.00   15.6332
9991      0.20   19.3932
9992      0.00   13.3200
9993      0.00   72.9480

[9994 rows x 21 columns]
```

[5]:
```python
# 4. Create Month-Year string (for display)
df["MonthYear"] = df["Order Date"].dt.strftime("%b %Y")
```

[6]:
```python
# 5. Create YearMonth numeric for proper sorting
df["YearMonth"] = df["Order Date"].dt.year * 100 + df["Order Date"].dt.month
```

[7]:
```python
# 6. Optional extra fields
df["Year"] = df["Order Date"].dt.year
df["Months"] = df["Order Date"].dt.month_name()
df["Profit_Margin"] = (df["Profit"] / df["Sales"]).round(4)
```

[8]:
```
# 7. Sort by YearMonth
df = df.sort_values("YearMonth")
```

[9]:
```
# 8. Export cleaned file
df.to_csv(r"D:\intern\Superstore_clean.csv", index=False)
print("Cleaning complete! Saved as D:\\intern\\Superstore_clean.csv")
```

```
Cleaning complete! Saved as D:\intern\Superstore_clean.csv
```