Import Liberaries

In [1]:
```
import pandas as pd
```

Load Data

In [2]:
```
df = pd.read_csv("superstore_sales.csv", encoding="latin1")
```

In [3]:
```
df.head(10)
```

Out[3]:

	order_date	customer_name	country	region	category	sales	profit
0	01-01-2011	Toby Braunhardt	Algeria	Africa	Office Supplies	408	106.140
1	01-01-2011	Joseph Holt	Australia	Oceania	Office Supplies	120	36.036
2	01-01-2011	Annie Thurman	Hungary	EMEA	Office Supplies	66	29.640
3	01-01-2011	Eugene Moren	Sweden	North	Office Supplies	45	-26.055
4	01-01-2011	Joseph Holt	Australia	Oceania	Furniture	114	37.770
5	01-01-2011	Joseph Holt	Australia	Oceania	Office Supplies	55	15.342
6	02-01-2011	Magdelene Morse	Canada	Canada	Technology	314	3.120
7	03-01-2011	Kean Nguyen	Australia	Oceania	Office Supplies	276	110.412
8	03-01-2011	Ken Lonsdale	New Zealand	Oceania	Technology	912	-319.464
9	03-01-2011	Lindsay Williams	Iraq	EMEA	Furniture	667	253.320

EDA - Exploratory Data Analysis

In [4]:
```
df.columns
```

Out[4]:
```
Index(['order_date', 'customer_name', 'country', 'region', 'category', 'sales',
       'profit'],
      dtype='object')
```

In [5]:
```
df.size
```

Out[5]:
```
359030
```

In [6]:
```
df.index
```

Out[6]:
```
RangeIndex(start=0, stop=51290, step=1)
```

Information about the data

In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51290 entries, 0 to 51289
Data columns (total 7 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   order_date     51290 non-null  object 
 1   customer_name  51290 non-null  object 
 2   country        51290 non-null  object 
 3   region         51290 non-null  object 
 4   category       51290 non-null  object 
 5   sales          51290 non-null  object 
 6   profit         51290 non-null  float64
dtypes: float64(1), object(6)
memory usage: 2.7+ MB
```

In [8]:
```
df.describe()
```

Out[8]:

	profit
count	51290.000000
mean	28.641740
std	174.424113
min	-6599.978000
25%	0.000000
50%	9.240000
75%	36.810000
max	8399.976000

In [9]:
```
df.describe(include = object)
```

Out[9]:

	order_date	customer_name	country	region	category	sales
count	51290	51290	51290	51290	51290	51290
unique	1430	795	147	13	3	2246
top	18-06-2014	Muhammed Yedwab	United States	Central	Office Supplies	13
freq	135	108	9994	11117	31273	589

In [10]:
```
df.describe(include = "number")
```

Out[10]:

	profit
count	51290.000000
mean	28.641740
std	174.424113
min	-6599.978000
25%	0.000000
50%	9.240000
75%	36.810000
max	8399.976000

Data cleaning

Duplicate value

In [12]:
```
df.duplicated().sum()
```

Out[12]:
```
np.int64(11)
```

Drop Duplicate Row

In [13]:
```
df.drop_duplicates(inplace=True)
```

In [14]:
```
df.duplicated().sum()
```

Out[14]:
```
np.int64(0)
```

Check Data-type

In [15]:
```
df.dtypes
```

Out[15]:
```
order_date        object
customer_name     object
country           object
region            object
category          object
sales             object
profit           float64
dtype: object
```

Covert Data-type (fixing data type)

In [19]:
```
df['sales'] = pd.to_numeric(df['sales'], errors='coerce')
```

In [20]:
```
df.dtypes
```

Out[20]:
```
order_date        object
customer_name     object
country           object
region            object
category          object
sales            float64
profit           float64
dtype: object
```

In [22]:
```
df['order_date'] = pd.to_datetime(df['order_date'], errors='coerce')
```

In [23]:
```
df.dtypes
```

Out[23]:
```
order_date       datetime64[ns]
customer_name            object
country                  object
region                   object
category                 object
sales                   float64
profit                  float64
dtype: object
```

Convert Order Date to “Month-Year” format

In [26]:
```
# Columns to convert
cols_to_format = ['customer_name', 'country', 'region', 'category']

# Convert to title case
for col in cols_to_format:
    df[col] = df[col].astype(str).str.strip().str.title()

# Convert order_date to datetime first
df['order_date'] = pd.to_datetime(df['order_date'], errors='coerce')

# Convert to DD/MM/YYYY format
df['order_date'] = df['order_date'].dt.strftime('%d/%m/%Y')

# Create month_year column (format: Jan-2011)
df['month_year'] = pd.to_datetime(df['order_date'], format='%d/%m/%Y').dt.strftime('%b-%Y')
```

In [27]:
```
df.head()
```

Out[27]:

	order_date	customer_name	country	region	category	sales	profit	month_year
0	01/01/2011	Toby Braunhardt	Algeria	Africa	Office Supplies	408.0	106.140	Jan-2011
1	01/01/2011	Joseph Holt	Australia	Oceania	Office Supplies	120.0	36.036	Jan-2011
2	01/01/2011	Annie Thurman	Hungary	Emea	Office Supplies	66.0	29.640	Jan-2011
3	01/01/2011	Eugene Moren	Sweden	North	Office Supplies	45.0	-26.055	Jan-2011
4	01/01/2011	Joseph Holt	Australia	Oceania	Furniture	114.0	37.770	Jan-2011

Clean data save

In [29]:
```
df.to_csv("superstore_sale_clean.csv", index=False)
```

In [ ]: