Installing Necessary Python Packages

In [ ]:
```
import kaggle
```

In [3]:
```
!kaggle datasets download ankitbansal06/retail-orders -f orders.csv
```

```
Dataset URL: https://www.kaggle.com/datasets/ankitbansal06/retail-orders
License(s): CC0-1.0
Downloading orders.csv.zip to E:\Projects\data_analysis\Retail_orders_analysis

  0%|          | 0.00/200k [00:00<?, ?B/s]
100%|##########| 200k/200k [00:00<00:00, 276kB/s]
100%|##########| 200k/200k [00:00<00:00, 276kB/s]
```

In [3]:
```
!kaggle datasets download ankitbansal06/retail-orders -f orders.csv
```

```
Dataset URL: https://www.kaggle.com/datasets/ankitbansal06/retail-orders
License(s): CC0-1.0
Downloading orders.csv.zip to E:\Projects\data_analysis\Retail_orders_analysis

  0%|          | 0.00/200k [00:00<?, ?B/s]
100%|##########| 200k/200k [00:00<00:00, 276kB/s]
100%|##########| 200k/200k [00:00<00:00, 276kB/s]
```

Extracting The zip file

In [4]:
```
import zipfile
zip_ref = zipfile.ZipFile('orders.csv.zip')
zip_ref.extractall() # extract file to dir
zip_ref.close() # close file
```

Reading the orderscsv datasets with the pandas pacakge

In [5]:
```
import pandas as pd
df = pd.read_csv('orders.csv',na_values=['Not Available','unknown'])
df['Ship Mode'].unique()
```

Out[5]:
```
array(['Second Class', 'Standard Class', nan, 'First Class', 'Same Day'],
      dtype=object)
```

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

Out[8]:

	Order Id	Order Date	Ship Mode	Segment	Country	City	State	Postal Code	Region	Cate
0	1	2023-03-01	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furr
1	2	2023-08-15	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furr
2	3	2023-01-10	Second Class	Corporate	United States	Los Angeles	California	90036	West	C Sup
3	4	2022-06-18	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furr
4	5	2022-07-13	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	C Sup

In [11]:
```
df.columns= df.columns.str.lower()
df.columns=df.columns.str.replace(" ","_")
```

In [17]:
```
df["discount"] =df["list_price"]*df["discount_percent"]*.01
df["sale_price"] =df["list_price"]-df["discount"]
df["profit"] =df["sale_price"]-df["cost_price"]
df.head(10)
```

Out[17]:

	order_id	order_date	ship_mode	segment	country	city	state	postal_code
0	1	2023-03-01	Second Class	Consumer	United States	Henderson	Kentucky	42420
1	2	2023-08-15	Second Class	Consumer	United States	Henderson	Kentucky	42420
2	3	2023-01-10	Second Class	Corporate	United States	Los Angeles	California	90036
3	4	2022-06-18	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311
4	5	2022-07-13	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311
5	6	2022-03-13	NaN	Consumer	United States	Los Angeles	California	90032
6	7	2022-12-28	Standard Class	Consumer	United States	Los Angeles	California	90032
7	8	2022-01-25	Standard Class	Consumer	United States	Los Angeles	California	90032
8	9	2023-03-23	NaN	Consumer	United States	Los Angeles	California	90032
9	10	2023-05-16	Standard Class	Consumer	United States	Los Angeles	California	90032

In [19]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 19 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   order_id          9994 non-null   int64  
 1   order_date        9994 non-null   object 
 2   ship_mode         9988 non-null   object 
 3   segment           9994 non-null   object 
 4   country           9994 non-null   object 
 5   city              9994 non-null   object 
 6   state             9994 non-null   object 
 7   postal_code       9994 non-null   int64  
 8   region            9994 non-null   object 
 9   category          9994 non-null   object 
 10  sub_category      9994 non-null   object 
 11  product_id        9994 non-null   object 
 12  cost_price        9994 non-null   int64  
 13  list_price        9994 non-null   int64  
 14  quantity          9994 non-null   int64  
 15  discount_percent  9994 non-null   int64  
 16  discount          9994 non-null   float64
 17  sale_price        9994 non-null   float64
 18  profit            9994 non-null   float64
dtypes: float64(3), int64(6), object(10)
memory usage: 1.4+ MB
```

In [21]:
```
df["order_date"]= pd.to_datetime(df["order_date"],format="%Y-%m-%d")
```

In [23]:
```
df.drop(columns=["list_price","discount_percent","cost_price"],inplace=True)
```

In [24]:
```
df.head(10)
```

Out[24]:

	order_id	order_date	ship_mode	segment	country	city	state	postal_code
0	1	2023-03-01	Second Class	Consumer	United States	Henderson	Kentucky	42420
1	2	2023-08-15	Second Class	Consumer	United States	Henderson	Kentucky	42420
2	3	2023-01-10	Second Class	Corporate	United States	Los Angeles	California	90036
3	4	2022-06-18	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311
4	5	2022-07-13	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311
5	6	2022-03-13	NaN	Consumer	United States	Los Angeles	California	90032
6	7	2022-12-28	Standard Class	Consumer	United States	Los Angeles	California	90032
7	8	2022-01-25	Standard Class	Consumer	United States	Los Angeles	California	90032
8	9	2023-03-23	NaN	Consumer	United States	Los Angeles	California	90032
9	10	2023-05-16	Standard Class	Consumer	United States	Los Angeles	California	90032

In [27]:
```
import sqlalchemy as sal
engine =sal.create_engine('mssql://LAPTOP-S67TFTS4/retail?driver=ODBC+DRIVER+17+FOR
                          )
conn=engine.connect()
```

In [29]:
```
df.to_sql("orders",index=False, if_exists='replace', con=conn)
```

Out[29]:
```
38
```