In [3]:
```
# import libraries
# !pip install kaggle
import kaggle
!kaggle datasets download ankitbansal06/retail-orders -f orders.csv
```

```
  0%|          | 0.00/200k [00:00<?, ?B/s]
100%|##########| 200k/200k [00:00<00:00, 570kB/s]
100%|##########| 200k/200k [00:00<00:00, 568kB/s]
Dataset URL: https://www.kaggle.com/datasets/ankitbansal06/retail-orders
License(s): CC0-1.0
Downloading orders.csv.zip to C:\Users\najmi\Downloads\Data Analytics with Python Training\SQL Python Kaggl
e Project
```

In [6]:
```
# extract file from zip file
import zipfile
zip_ref = zipfile.ZipFile('orders.csv.zip')
zip_ref.extractall() # extract file to dir
zip_ref.close() # close the file
```

In [8]:
```
# read data from the file and handle null values
import pandas as pd
df = pd.read_csv('orders.csv', na_values=['Not Available', 'unknown'])
df['Ship Mode'].unique()
```

Out[8]:
```
array(['Second Class', 'Standard Class', nan, 'First Class', 'Same Day'],
      dtype=object)
```

In [13]:
```
# rename columns name
df.columns = df.columns.str.lower().str.replace(' ', '_')
df.columns
```

Out[13]:
```
Index(['order_id', 'order_date', 'ship_mode', 'segment', 'country', 'city',
       'state', 'postal_code', 'region', 'category', 'sub_category',
       'product_id', 'cost_price', 'list_price', 'quantity',
       'discount_percent'],
      dtype='object')
```

In [14]:
```
# derive new columns discount, sale price and profit
df.head()
```

Out[14]:

	order_id	order_date	ship_mode	segment	country	city	state	postal_code	region	category	sub_category	pro
0	1	2023-03-01	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	F 1(
1	2	2023-08-15	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	F 1(
2	3	2023-01-10	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	1(
3	4	2022-06-18	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	1(
4	5	2022-07-13	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	1(

In [20]:
```
# df['discount'] = df['list_price'] * df['discount_percent'] * .01
# df['sale_price'] = df['list_price'] - df['discount']
df['profitQ'] = df['sale_price'] - df['cost_price']
df
```

Out[20]:

	order_id	order_date	ship_mode	segment	country	city	state	postal_code	region	category	sub_categor
0	1	2023-03-01	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcas
1	2	2023-08-15	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chai
2	3	2023-01-10	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labe
3	4	2022-06-18	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tabl
4	5	2022-07-13	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storag
...	...	...	...	...	...	...	...	...	...	...	
9989	9990	2023-02-18	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furnishing
9990	9991	2023-03-17	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furnishing
9991	9992	2022-08-07	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Technology	Phon
9992	9993	2022-11-19	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Office Supplies	Pap
9993	9994	2022-07-17	Second Class	Consumer	United States	Westminster	California	92683	West	Office Supplies	Applianc

9994 rows × 19 columns

In [21]:
```
df.dtypes
```

Out[21]:
```
order_id              int64
order_date           object
ship_mode            object
segment              object
country              object
city                 object
state                object
postal_code           int64
region               object
category             object
sub_category         object
product_id           object
cost_price            int64
list_price            int64
quantity              int64
discount_percent      int64
discount            float64
sale_price          float64
profit              float64
dtype: object
```

In [23]:
```
# convert order date from object data type to datetime
df['order_date'] = pd.to_datetime(df['order_date'], format = "%Y-%m-%d")
```

In [28]:
```
# drop cost price, list price and discount percent columns
# df.drop(columns=['list_price', 'cost_price', 'discount_percent'], inplace = True)
df.head()
```

Out[28]:

	order_id	order_date	ship_mode	segment	country	city	state	postal_code	region	category	sub_category	pro
0	1	2023-03-01	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases	F 1(
1	2	2023-08-15	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs	F 1(
2	3	2023-01-10	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels	1(
3	4	2022-06-18	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables	1(
4	5	2022-07-13	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage	1(

In [29]:
```
# load data into sql server using replace option
import sqlalchemy as sal
engine = sal.create_engine('mssql://Najmi-XPS\SQLEXPRESS/master?driver=ODBC+DRIVER+17+FOR+SQL+SERVER')
conn = engine.connect()
```

In [44]:
```
# laod data into sql server using append option
df.to_sql('df_orders', con=conn, index=False, if_exists='append')
```

Out[44]:
```
-1
```

In [43]:
```
df.columns
```

Out[43]:
```
Index(['order_id', 'order_date', 'ship_mode', 'segment', 'country', 'city',
       'state', 'postal_code', 'region', 'category', 'sub_category',
       'product_id', 'quantity', 'discount', 'sale_price', 'profit'],
      dtype='object')
```

In [ ]:
```

```