Weekly Project 17 :

Data Visualisation on Sales Dataset :

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df = pd.read_csv("sales file.csv")
```

In [3]:
```
df
```

Out[3]:

	order_id	product	quantity	price	date	address	month
0	236670	Wired Headphones	2	11.99	08/31/19 22:21	359 Spruce St, Seattle, WA 98101	August
1	236671	Bose SoundSport Headphones	1	99.99	08/15/19 15:11	492 Ridge St, Dallas, TX 75001	August
2	236672	iPhone	1	700.0	08/06/19 14:40	149 7th St, Portland, OR 97035	August
3	236673	AA Batteries (4-pack)	2	3.84	08/29/19 20:59	631 2nd St, Los Angeles, CA 90001	August
4	236674	AA Batteries (4-pack)	2	3.84	08/15/19 19:53	736 14th St, New York City, NY 10001	August
...	...	...	...	...	...	...	...
186300	162004	Apple Airpods Headphones	1	150	02/12/19 22:02	227 Church St, San Francisco, CA 94016	February
186301	162005	AAA Batteries (4-pack)	2	2.99	02/04/19 20:44	417 Jefferson St, Los Angeles, CA 90001	February
186302	162006	USB-C Charging Cable	1	11.95	02/24/19 06:31	498 8th St, Atlanta, GA 30301	February
186303	162007	USB-C Charging Cable	1	11.95	02/24/19 19:09	715 7th St, Dallas, TX 75001	February
186304	162008	27in FHD Monitor	1	149.99	02/26/19 17:15	677 West St, Los Angeles, CA 90001	February

186305 rows × 7 columns

1. Deal with Null values:

In [4]:
```
df.isnull().sum()
```

Out[4]:
```
order_id    0
product     0
quantity    0
price       0
date        0
address     0
month       0
dtype: int64
```

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 186305 entries, 0 to 186304
Data columns (total 7 columns):
 #   Column    Non-Null Count   Dtype 
---  ------    --------------   ----- 
 0   order_id  186305 non-null  object
 1   product   186305 non-null  object
 2   quantity  186305 non-null  object
 3   price     186305 non-null  object
 4   date      186305 non-null  object
 5   address   186305 non-null  object
 6   month     186305 non-null  object
dtypes: object(7)
memory usage: 9.9+ MB
```

2.number of unique orders placed so far :

In [6]:
```
no_of_unique_orders = df["order_id"].nunique()
print("Number of unique orders placed so far","-",no_of_unique_orders)
```

```
Number of unique orders placed so far - 178438
```

3. Number of unique products we're having as per the Dataset:

In [7]:
```
no_of_unique_products = df["product"].nunique()
print("Number of unique products as per the dataset","-",no_of_unique_products)
```

```
Number of unique products as per the dataset - 20
```

4.Name the most expensive product on the line :

In [8]:
```
len(df["order_id"])
```

Out[8]:
```
186305
```

In [9]:
```
df["price"].mode().values[0]
```

Out[9]:
```
'11.95'
```

In [10]:
```
k = len(df['order_id'])
for i in range(k):
    if 'Price' in df['price'][i]:
        df['price'][i] = 0
df['price'].unique()
```

Out[10]:
```
array(['11.99', '99.99', '700.0', '3.84', '379.99', '109.99', '1700.0',
       '600.0', '149.99', '14.95', '150.0', '2.99', '11.95', '389.99',
       '999.99', '300.0', '400.0', 0, '600', '150', '1700', '300', '400',
       '700'], dtype=object)
```

In [11]:
```
l = []
n = len(df["order_id"])
for i in range(n):
    l.append(int(float(df['price'][i])))
a = max(l)
df['price'] = l
for j in range(n):
    if(df['price'][j]==a):
        b = df['product'][j]
print("The most expensive product on the list is",b,"and its price is",a,".")
```

```
The most expensive product on the list is Macbook Pro Laptop and its price is 1700 .
```

5. From which address does the most number of orders are placed?

In [12]:
```
n = len(df['order_id'])
for i in range(n):
    if(df['address'][i]=="Purchase Address"):
        df=df.drop(i)
most_orders_address = df["address"].value_counts().idxmax()
print("The address with the most orders is:",most_orders_address)
```

```
The address with the most orders is: 193 Forest St, San Francisco, CA 94016
```

6.Plot number of orders in each month in line graph

In [13]:
```
monthly_orders = df.groupby('month')['order_id'].count().reset_index()
fig,axs = plt.subplots(figsize = (15,4))
sns.lineplot(x = 'month', y='order_id', data = monthly_orders)
fig.autofmt_xdate()
plt.xlabel('MONTH')
plt.ylabel('Number of Orders')
plt.title("Monthly orders")
plt.show()
```