Name : Mayur Girase
Div : E(E1), Roll No : 514

In [33]:
```
import numpy as nm
import pandas as pd
```

In [6]:
```
data=pd.read_csv("all_data.csv")
```

In [7]:
```
data.head()
```

Out[7]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	176558	USB-C Charging Cable	2	11.95	04/19/19 8:46	917 1st St, Dallas, TX 75001
1	NaN	NaN	NaN	NaN	NaN	NaN
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001

In [8]:
```
df1 = data[data.isna().any(axis=1)]
display(df1.head())

data = data.dropna(how='all')
data.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
1	NaN	NaN	NaN	NaN	NaN	NaN
356	NaN	NaN	NaN	NaN	NaN	NaN
735	NaN	NaN	NaN	NaN	NaN	NaN
1433	NaN	NaN	NaN	NaN	NaN	NaN
1553	NaN	NaN	NaN	NaN	NaN	NaN

Out[8]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	176558	USB-C Charging Cable	2	11.95	04/19/19 8:46	917 1st St, Dallas, TX 75001
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
5	176561	Wired Headphones	1	11.99	04/30/19 9:27	333 8th St, Los Angeles, CA 90001

In [9]:
```
data = data[data['Order Date'].str[0:2]!='Or']
```

In [11]:
```
data['Quantity Ordered']=pd.to_numeric(data['Quantity Ordered'])
data['Price Each']=pd.to_numeric(data['Price Each'])
```

In [12]:
```
data['Month'] = data['Order Date'].str[0:2]
data['Month'] = data['Month'].astype('int32')
data.head()
```

Out[12]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month
0	176558	USB-C Charging Cable	2	11.95	04/19/19 8:46	917 1st St, Dallas, TX 75001	4
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4
5	176561	Wired Headphones	1	11.99	04/30/19 9:27	333 8th St, Los Angeles, CA 90001	4

In [14]:
```
data['Month 2'] = pd.to_datetime(data['Order Date']).dt.month
data.head()
```

Out[14]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Month 2
0	176558	USB-C Charging Cable	2	11.95	04/19/19 8:46	917 1st St, Dallas, TX 75001	4	4
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4	4
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	4
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	4
5	176561	Wired Headphones	1	11.99	04/30/19 9:27	333 8th St, Los Angeles, CA 90001	4	4

In [15]:
```
def get_city(address):
    return address.split(",")[1].strip(" ")

def get_state(address):
    return address.split(",")[2].split(" ")[1]

data['City'] = data['Purchase Address'].apply(lambda x: f"{get_city(x)}  ({get
data.head()
```

Out[15]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Month 2	City
0	176558	USB-C Charging Cable	2	11.95	04/19/19 8:46	917 1st St, Dallas, TX 75001	4	4	Dallas (TX)
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4	4	Boston (MA)
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	4	Los Angeles (CA)
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	4	Los Angeles (CA)
5	176561	Wired Headphones	1	11.99	04/30/19 9:27	333 8th St, Los Angeles, CA 90001	4	4	Los Angeles (CA)

In [17]:
```
data['Sales'] = data['Quantity Ordered'].astype('int') * data['Price Each'].as
```

In [22]:
```
data.groupby(['Month']).sum()
```

Out[22]:

Month	Quantity Ordered	Price Each	Month 2	Sales
4	17739	2899439.68	63088	2918954.40
5	26	8851.62	125	8855.46

In [24]:
```
c_max=data.groupby(['City']).sum()
print(max(c_max))
```

```
Sales
```

In [26]:
```
df1 = data[data['Order ID'].duplicated(keep=False)]
df1['Grouped'] = df1.groupby('Order ID')['Product'].transform(lambda x: ','.jo
df2 = df1[['Order ID', 'Grouped']].drop_duplicates()
print(df1['Grouped'])
```

```
3                            Google Phone,Wired Headphones
4                            Google Phone,Wired Headphones
18                       Google Phone,USB-C Charging Cable
19                       Google Phone,USB-C Charging Cable
30       Bose SoundSport Headphones,Bose SoundSport Hea...
                               ...
15787                  USB-C Charging Cable,Wired Headphones
15818              Vareebadd Phone,Lightning Charging Cable
15819              Vareebadd Phone,Lightning Charging Cable
15874               Google Phone,Bose SoundSport Headphones
15875               Google Phone,Bose SoundSport Headphones
Name: Grouped, Length: 1269, dtype: object
```

```
C:\Users\student\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: Setting
WithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/s
table/user_guide/indexing.html#returning-a-view-versus-a-copy (https://panda
s.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-ver
sus-a-copy)
```

In [27]:
```
from itertools import combinations
from collections import Counter

count = Counter()

for row in df2['Grouped']:
    row_list = row.split(',')
    count.update(Counter(combinations(row_list, 2)))

for key,value in count.most_common(10):
    print(key, value)
```

```
('iPhone', 'Lightning Charging Cable') 94
('Google Phone', 'USB-C Charging Cable') 92
('Google Phone', 'Wired Headphones') 34
('iPhone', 'Wired Headphones') 33
('Vareebadd Phone', 'USB-C Charging Cable') 32
('iPhone', 'Apple Airpods Headphones') 29
('Google Phone', 'Bose SoundSport Headphones') 20
('Vareebadd Phone', 'Wired Headphones') 15
('USB-C Charging Cable', 'Wired Headphones') 11
('AA Batteries (4-pack)', 'Apple Airpods Headphones') 7
```

In [28]:
```
product_group = data.groupby('Product')
quantity_ordered = product_group.sum()['Quantity Ordered']
```

In [29]:
```
print(quantity_ordered)
```

```
Product
20in Monitor                  345
27in 4K Gaming Monitor        491
27in FHD Monitor              623
34in Ultrawide Monitor        583

AA Batteries (4-pack)        2446
AAA Batteries (4-pack)       2559
Apple Airpods Headphones     1303
Bose SoundSport Headphones   1110
Flatscreen TV                 398
Google Phone                  497
LG Dryer                       69
LG Washing Machine             56
Lightning Charging Cable     2027
Macbook Pro Laptop            400
ThinkPad Laptop               329
USB-C Charging Cable         1938
Vareebadd Phone               185
Wired Headphones             1823
iPhone                        593
Name: Quantity Ordered, dtype: int64
```

In [31]:
```
prices = data.groupby('Product').mean()['Price Each']
```

In [32]:
```
print(prices)
```

```
Product
20in Monitor                  109.99
27in 4K Gaming Monitor        389.99
27in FHD Monitor              149.99
34in Ultrawide Monitor        379.99

AA Batteries (4-pack)           3.84
AAA Batteries (4-pack)          2.99
Apple Airpods Headphones      150.00
Bose SoundSport Headphones     99.99
Flatscreen TV                 300.00
Google Phone                  600.00
LG Dryer                      600.00
LG Washing Machine            600.00
Lightning Charging Cable       14.95
Macbook Pro Laptop           1700.00
ThinkPad Laptop               999.99
USB-C Charging Cable           11.95
Vareebadd Phone               400.00
Wired Headphones               11.99
iPhone                        700.00
Name: Price Each, dtype: float64
```

In [ ]:
```

```