TRANSFORMING AND MERGING DATA

In [11]:

```
import pandas as pd
import os
```

In [13]:

```
# merge all 12 files
df=pd.read_csv("./Sales_Data/Sales_April_2019.csv")
files= [file for file in os.listdir("./Sales_Data")]
all_months_data=pd.DataFrame()
for file in files:
    df=pd.read_csv("./Sales_Data/"+file)
    all_months_data=pd.concat([ all_months_data,df])
all_months_data.head()
all_months_data_clean = all_months_data.dropna(how='all').reset_index(drop=True)

all_months_data_clean.to_csv("all_data.csv",index= False)
```

read all updated data as new df

In [15]:

```
all_data=pd.read_csv("all_data.csv")
```

Q1 Find the month with highest sales and the sales figure in that month?

In [17]:

```
# starting by making a new months column

#all_data['month']= all_data['Order Date'].str[0:2]
#all_data['month']=all_data['month'].astype(int32)

#removing OR in months group
all_data['month']= all_data['Order Date'].str[0:2]
all_data=all_data[all_data['month'] != 'Or']
all_data['month']=all_data['month'].astype('int32')
```

In [23]:

```
# introducing sales column and coverting other colums acccording to our need
all_data['Quantity Ordered']=all_data['Quantity Ordered'].astype('float')
all_data['Price Each']=all_data['Price Each'].astype('float')
all_data['Sales']= all_data['Quantity Ordered']*all_data['Price Each']
```

In [25]:

```
max_df=all_data.groupby('month')['Sales'].sum()
result_1=max_df.sort_values(ascending= False)
# Therefore best sales were in 12th month which  is december and sales revenue w

0
```

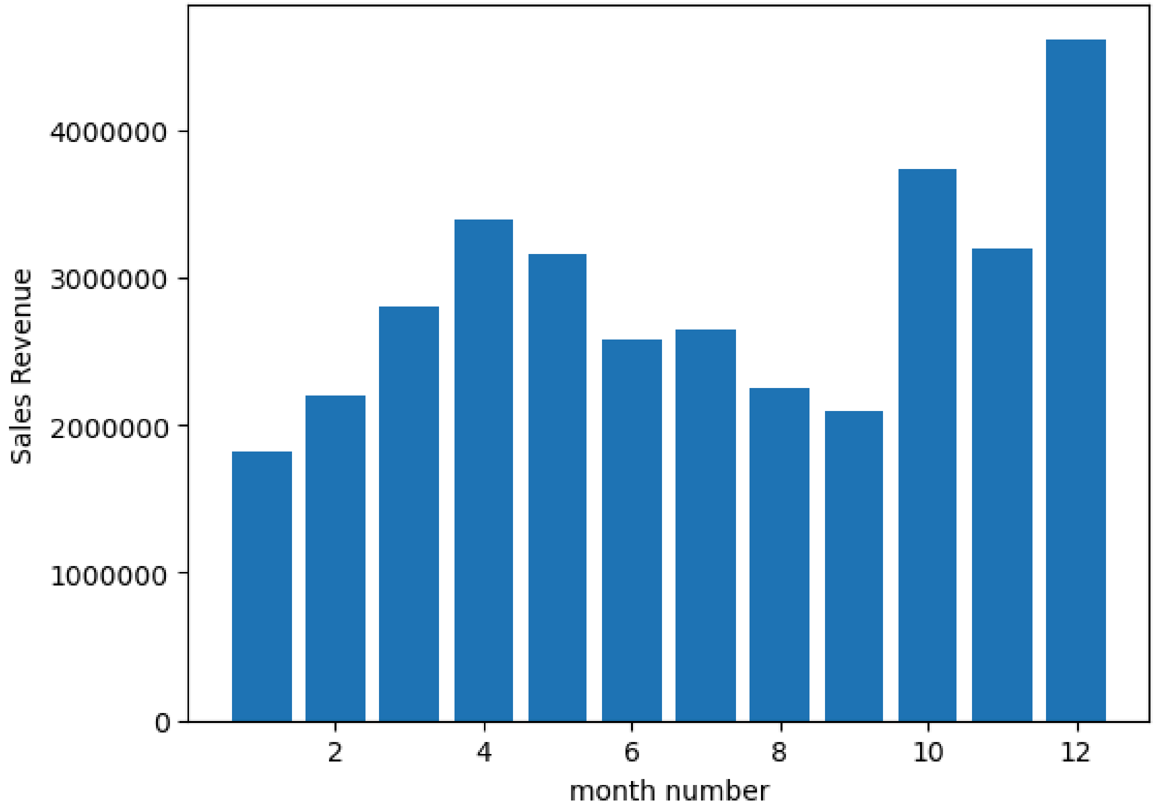
Out[25]:

```
0
```

In [27]:

```
#plot for the same observation
import matplotlib.pyplot as plt
month= range(1,13)
```

```
plt.bar(month,max_df)
plt.ticklabel_format(style='plain', axis='y')
plt.ylabel("Sales Revenue")
plt.xlabel("month number")
plt.show()
```



find the city with the highest sales?

In [29]:

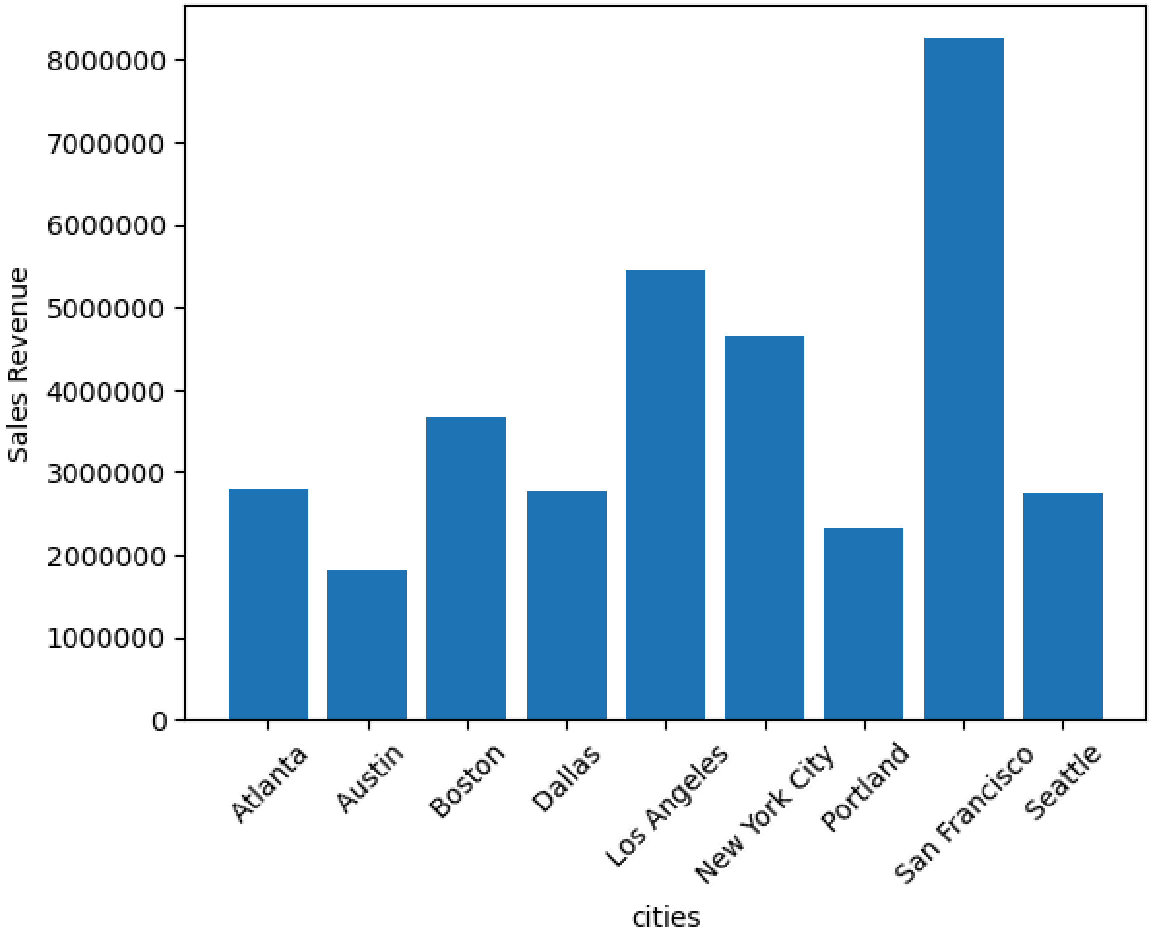
```
# we will make a new column city by seperating and filtering out the cities from
all_data['City'] = all_data['Purchase Address'].str.split(",").str[1]
all_data['City']
max_city=all_data.groupby('City')['Sales'].sum()
max_city.sort_values(ascending= False)
#San Francisco    8262203.91
```

Out[29]:

```
City
San Francisco    8262203.91
Los Angeles      5452570.80
New York City    4664317.43
Boston           3661642.01
Atlanta          2795498.58
Dallas           2767975.40
Seattle          2747755.48
Portland         2320490.61
Austin           1819581.75
Name: Sales, dtype: float64
```

In [33]:

```
import matplotlib.pyplot as plt
cities = list(max_city.index)
plt.xticks(rotation=45)
plt.bar(cities,max_city)
plt.ticklabel_format(style='plain', axis='y')
plt.ylabel("Sales Revenue")
plt.xlabel("cities")
plt.show()
```



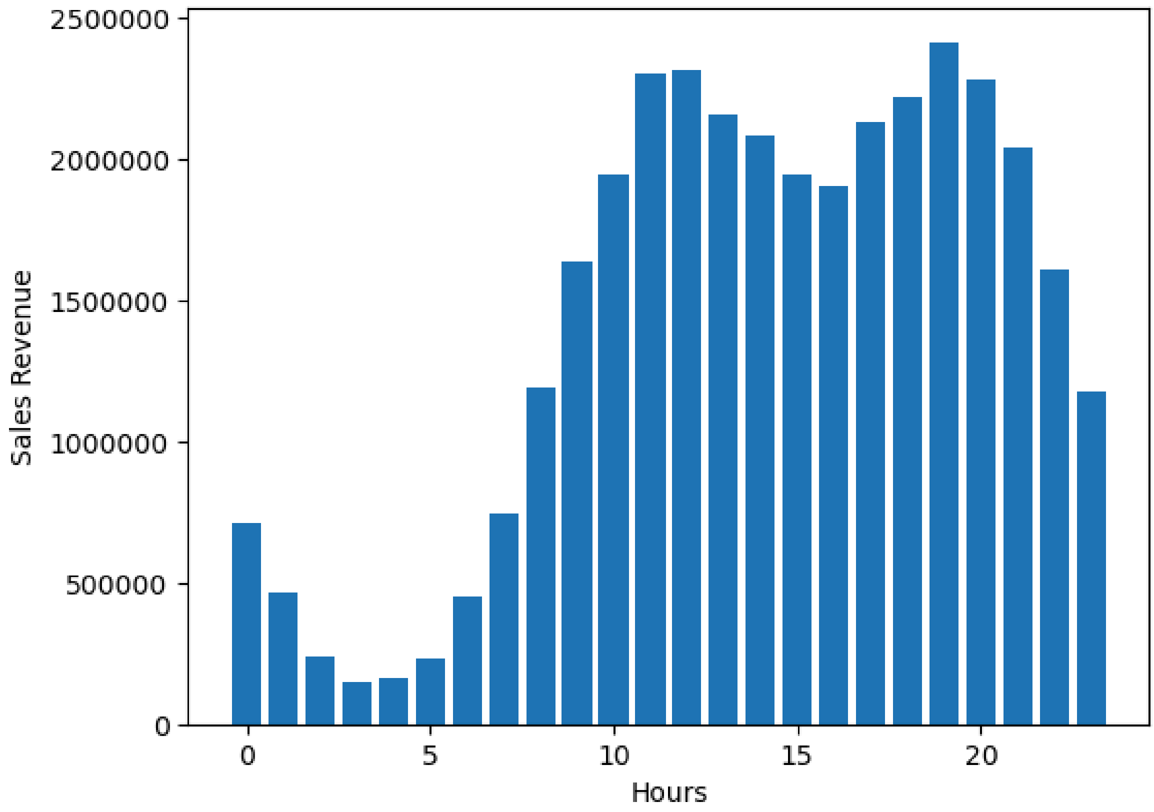
What time should be choose for advertising to maximise sales?

In [46]:
```
all_data['Order Date']=pd.to_datetime(all_data['Order Date'])
```

In [62]:
```
all_data['Hour']= all_data['Order Date'].dt.hour
all_data['minute']= all_data['Order Date'].dt.minute
all_data['Hour'].head()
```

In [72]:
```
hour_sales= all_data.groupby('Hour')['Sales'].sum()
```

In [78]:
```
hours = [hour for hour, df in all_data.groupby('Hour')]
plt.bar(hours,hour_sales)
plt.ticklabel_format(style='plain', axis='y')
plt.ylabel("Sales Revenue")
plt.xlabel("Hours")
plt.show()
# according to thr graph we can see the highest sales are usually in the 19th ho
# Also the insight shows that sales in the night hours which are 0:00hrs to 6:00
```



Which products are often sold in pairs?

In [159…

```
all_data_grouped=all_data.groupby('Order ID')['Product'].apply(list)
all_data_grouped = all_data.groupby('Order ID')['Product'].apply(list).reset_ind
all_data_grouped.rename(columns={'Product': 'Grouped Products'}, inplace=True)
```

In [167…

```
from itertools import combinations
from collections import Counter

combo_counter = Counter()

for products in all_data_grouped['Grouped Products']:
    if len(products) > 1:
        combo_counter.update(combinations(products, 2))

combo_df = pd.DataFrame(combo_counter.most_common(10), columns=['Product Pair',
combo_df
# the most often sold pair is (iPhone, Lightning Charging Cable)
```

Out[167…

	Product Pair	Count
0	(iPhone, Lightning Charging Cable)	1005
1	(Google Phone, USB-C Charging Cable)	987
2	(iPhone, Wired Headphones)	447
3	(Google Phone, Wired Headphones)	414
4	(Vareebadd Phone, USB-C Charging Cable)	361
5	(iPhone, Apple Airpods Headphones)	360
6	(Google Phone, Bose SoundSport Headphones)	220
7	(USB-C Charging Cable, Wired Headphones)	160
8	(Vareebadd Phone, Wired Headphones)	143
9	(Lightning Charging Cable, Wired Headphones)	92

What product sold the most and why?

In [170…

```
all_data
```

Out[170…

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	month	Sales	
0	176558	USB-C Charging Cable	2.0	11.95	2019-04-19 08:46:00	917 1st St, Dallas, TX 75001	4	23.90	
1	176559	Bose SoundSport Headphones	1.0	99.99	2019-04-07 22:30:00	682 Chestnut St, Boston, MA 02215	4	99.99	B
2	176560	Google Phone	1.0	600.00	2019-04-12 14:38:00	669 Spruce St, Los Angeles, CA 90001	4	600.00	Ar
3	176560	Wired Headphones	1.0	11.99	2019-04-12 14:38:00	669 Spruce St, Los Angeles, CA 90001	4	11.99	Ar
4	176561	Wired Headphones	1.0	11.99	2019-04-30 09:27:00	333 8th St, Los Angeles, CA 90001	4	11.99	Ar
...	...	...	...	...	...	...	...	...	
186300	259353	AAA Batteries (4-pack)	3.0	2.99	2019-09-17 20:56:00	840 Highland St, Los Angeles, CA 90001	9	8.97	Ar
186301	259354	iPhone	1.0	700.00	2019-09-01 16:00:00	216 Dogwood St, San Francisco, CA 94016	9	700.00	Fra
186302	259355	iPhone	1.0	700.00	2019-09-23 07:39:00	220 12th St, San Francisco, CA 94016	9	700.00	Fra
186303	259356	34in Ultrawide Monitor	1.0	379.99	2019-09-19 17:30:00	511 Forest St, San Francisco, CA 94016	9	379.99	Fra
186304	259357	USB-C Charging Cable	1.0	11.95	2019-09-30 00:18:00	250 Meadow St, San Francisco, CA 94016	9	11.95	Fra

185950 rows × 11 columns

In [178…

```
highest_selling=all_data.groupby('Product')['Quantity Ordered'].sum()
highest_selling.sort_values(ascending=False)
```

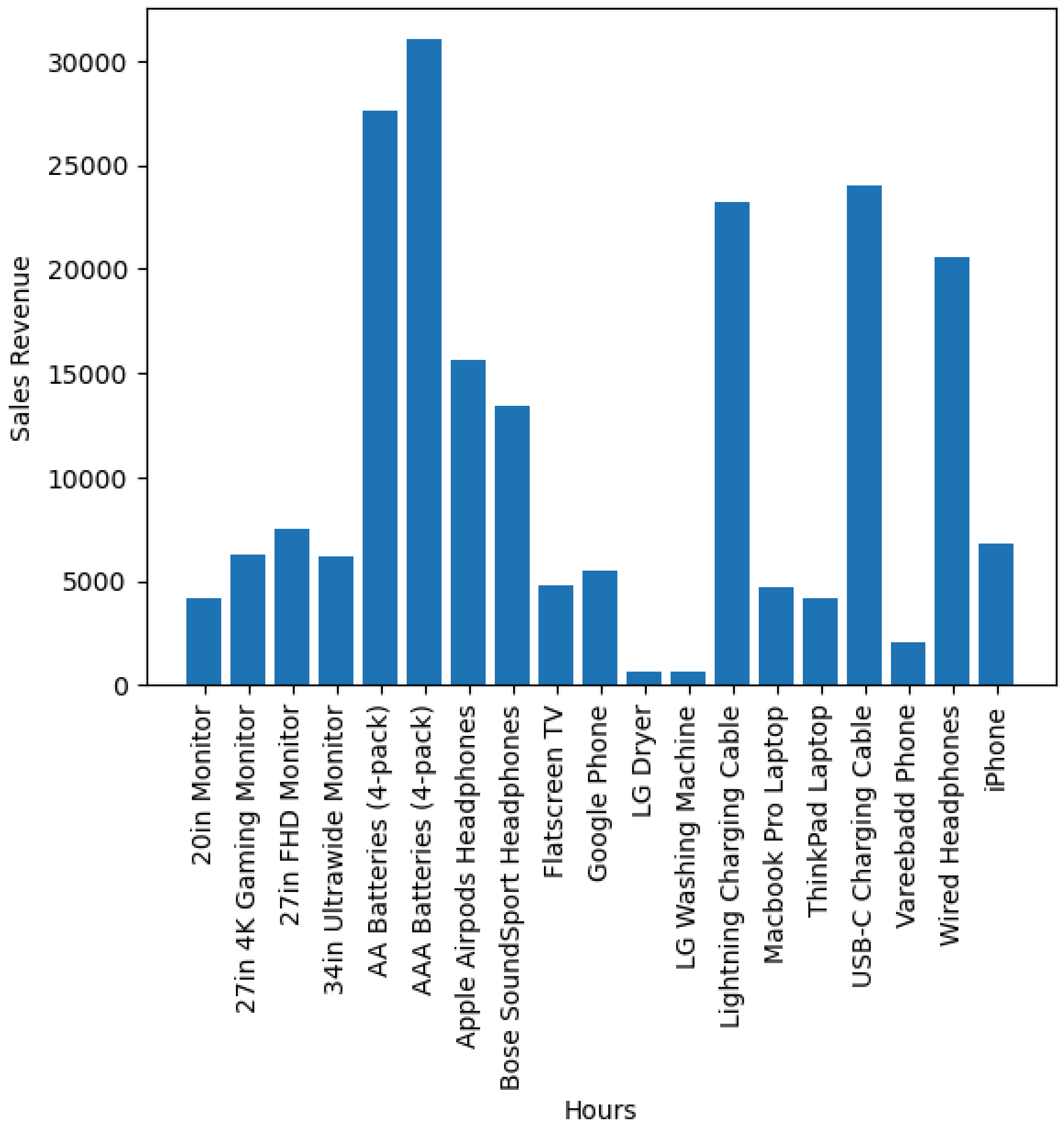
Out[178…

```
Product
AAA Batteries (4-pack)        31017.0
AA Batteries (4-pack)         27635.0
USB-C Charging Cable          23975.0
Lightning Charging Cable      23217.0
Wired Headphones              20557.0
Apple Airpods Headphones      15661.0
Bose SoundSport Headphones    13457.0
27in FHD Monitor               7550.0
iPhone                         6849.0
27in 4K Gaming Monitor         6244.0
34in Ultrawide Monitor         6199.0
Google Phone                   5532.0
Flatscreen TV                  4819.0
Macbook Pro Laptop             4728.0
ThinkPad Laptop                4130.0
20in Monitor                   4129.0
Vareebadd Phone                2068.0
LG Washing Machine              666.0
LG Dryer                        646.0
Name: Quantity Ordered, dtype: float64
```

In [189…

```
Products = [products for products, df in all_data.groupby('Product')]
plt.bar(Products,highest_selling)
plt.xticks(rotation=90)
plt.ticklabel_format(style='plain', axis='y')
plt.ylabel("Sales Revenue")
plt.xlabel("Hours")
plt.show()

#Answer- the highest selling product is AAA Batteries(4-pack) and it is so becau
# relatively low compared to other products
```


Sales Revenue
30000
25000
20000
15000
10000
5000
0
20in Monitor
27in 4K Gaming Monitor
27in FHD Monitor
34in Ultrawide Monitor
AA Batteries (4-pack)
AAA Batteries (4-pack)
Apple Airpods Headphones
Bose SoundSport Headphones
Flatscreen TV
Google Phone
LG Dryer
LG Washing Machine
Lightning Charging Cable
Macbook Pro Laptop
ThinkPad Laptop
USB-C Charging Cable
Vareebadd Phone
Wired Headphones
iPhone
Hours

```
In [ ]:
```

```
In [ ]:
```