In [9]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [73]:

```
df=pd.read_csv("C:/Users/DELL/Desktop/P6-SuperStoreUS-2015.csv",parse_dates=["Order Date","Ship_Date"])
df.head()
```

```
C:\Users\DELL\AppData\Local\Temp\ipykernel_33356\2878548922.py:1: UserWarning: Parsing dates in DD/MM/YYYY format when dayf
irst=False (the default) was specified. This may lead to inconsistently parsed dates! Specify a format to ensure consistent
parsing.
  df=pd.read_csv("C:/Users/DELL/Desktop/P6-SuperStoreUS-2015.csv",parse_dates=["Order Date","Ship_Date"])
C:\Users\DELL\AppData\Local\Temp\ipykernel_33356\2878548922.py:1: UserWarning: Parsing dates in DD/MM/YYYY format when dayf
irst=False (the default) was specified. This may lead to inconsistently parsed dates! Specify a format to ensure consistent
parsing.
  df=pd.read_csv("C:/Users/DELL/Desktop/P6-SuperStoreUS-2015.csv",parse_dates=["Order Date","Ship_Date"])
```

Out[73]:

	Row_ID	Order_Priority	Discount	Unit_Price	Shipping_Cost	Customer ID	Customer Name	Ship Mode	Customer Segment	Product_Category	...	Region	Province	
0	20847	High	0.01	2.84	0.93	3	Bonnie Potter	Express Air	Corporate	Office Supplies	...	West	Washington	Ana
1	20228	Not Specified	0.02	500.98	26.00	5	Ronnie Proctor	Delivery Truck	Home Office	Furniture	...	West	California	G
2	21776	Critical	0.06	9.48	7.29	11	Marcus Dunlap	Regular Air	Home Office	Furniture	...	East	New Jersey	R
3	24844	Medium	0.09	78.69	19.99	14	Gwendolyn F Tyson	Regular Air	Small Business	Furniture	...	Central	Minnesota	
4	24846	Medium	0.08	3.28	2.31	14	Gwendolyn F Tyson	Regular Air	Small Business	Office Supplies	...	Central	Minnesota	

5 rows × 25 columns

In [15]:

```
df.columns
```

Out[15]:

```
Index(['Row_ID', 'Order_Priority', 'Discount', 'Unit_Price', 'Shipping_Cost',
       'Customer ID', 'Customer Name', 'Ship Mode', 'Customer Segment',
       'Product_Category', 'Product_Sub_Category', 'Product_Container',
       'Product Name', 'Product_Base Margin', 'Country', 'Region', ' Province',
       'City', 'Postal Code', 'Order Date', 'Ship_Date', 'Profit', 'Quantity',
       'Sales', 'Order_ID'],
      dtype='object')
```

In [16]:

```
#Total_profits
df["Profit"].sum()
```

Out[16]:

```
224077.6118353
```

In [17]:

```
#Region wise profit
a=df[["Region","Profit"]].groupby("Region").sum()
a
```

Out[17]:

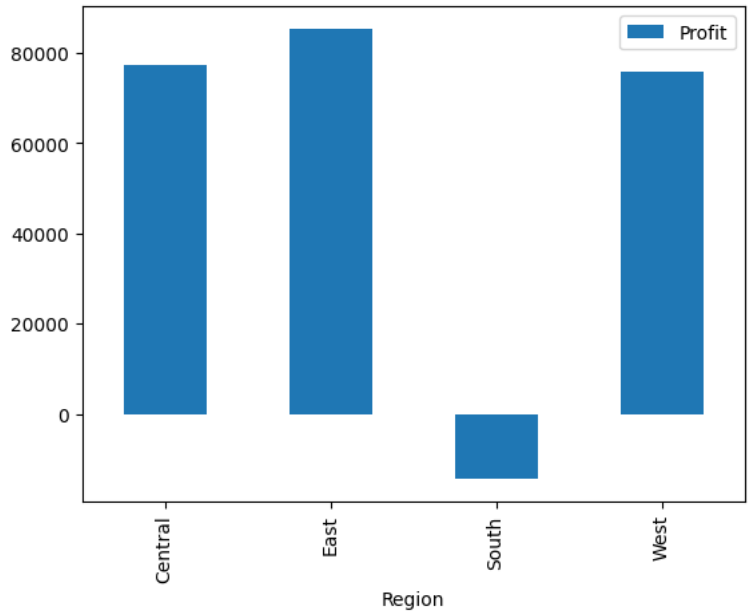
	Profit
Region	
Central	77365.472669
East	85291.403446
South	-14424.054379
West	75844.790100

In [18]:

```
a.plot.bar()
```

Out[18]:

```
<AxesSubplot: xlabel='Region'>
```



In [19]:

```
#unquie values of ship mode
df["Ship Mode"].unique()
```

Out[19]:

```
array(['Express Air', 'Delivery Truck', 'Regular Air'], dtype=object)
```

In [20]:

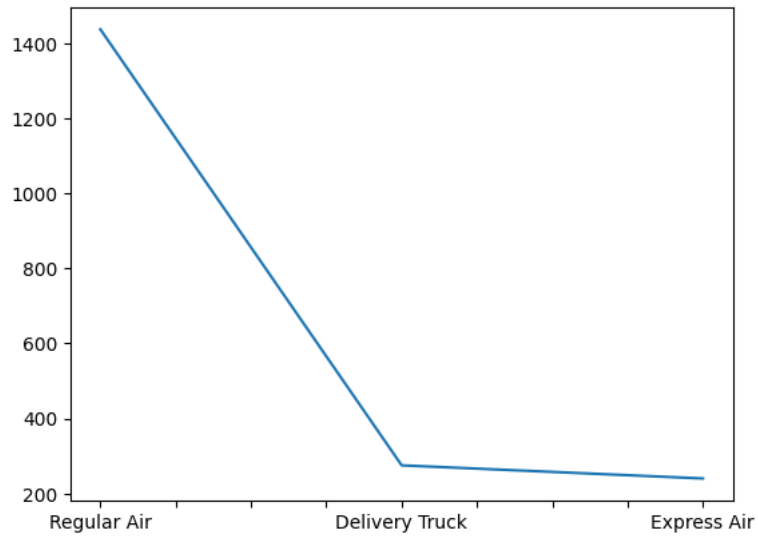
```
#counts
a=df["Ship Mode"].value_counts()
```

In [21]:

```
a.plot()
```

Out[21]:

```
<AxesSubplot: >
```



In [22]:

```
#start date
df["Order Date"].min()
```

Out[22]:

```
Timestamp('2015-01-01 00:00:00')
```

In [23]:

```
#End date
df["Order Date"].max()
```

Out[23]:

```
Timestamp('2015-12-06 00:00:00')
```

In [24]:

```
#total number of days
df["Order Date"].max()-df["Order Date"].min()
```

Out[24]:

```
Timedelta('339 days 00:00:00')
```

In [ ]:

```

```

In [25]:

```
#I want to had month column
df["month"]=df["Order Date"].dt.month
df.head()
```

Out[25]:

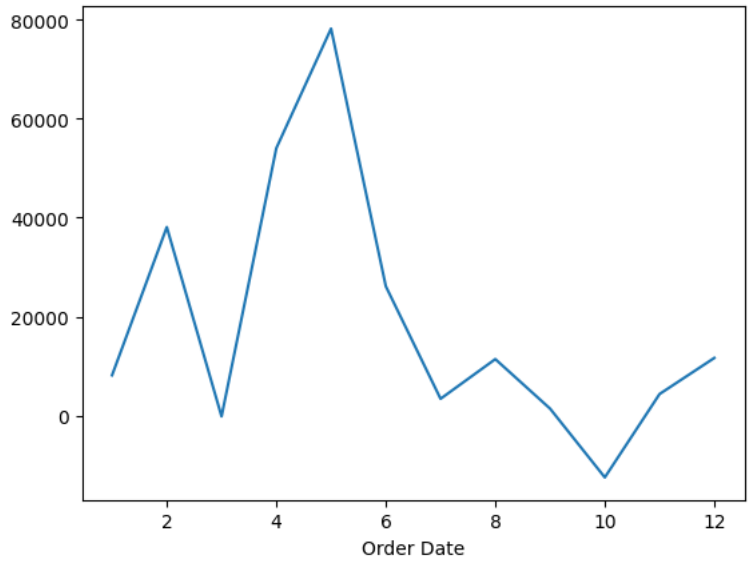
Customer ID	Customer Name	Ship Mode	Customer Segment	Product_Category	...	Province	City	Postal Code	Order Date	Ship_Date	Profit	Quantity	Sales	Order_ID	m
3	Bonnie Potter	Express Air	Corporate	Office Supplies	...	Washington	Anacortes	98221	2015-07-01	08-01-2015	4.5600	4	13.01	88522	
5	Ronnie Proctor	Delivery Truck	Home Office	Furniture	...	California	San Gabriel	91776	2015-06-13	15-06-2015	4390.3665	12	6362.85	90193	
11	Marcus Dunlap	Regular Air	Home Office	Furniture	...	New Jersey	Roselle	7203	2015-02-15	17-02-2015	-53.8096	22	211.15	90192	
14	Gwendolyn F Tyson	Regular Air	Small Business	Furniture	...	Minnesota	Prior Lake	55372	2015-12-05	14-05-2015	803.4705	16	1164.45	86838	
14	Gwendolyn F Tyson	Regular Air	Small Business	Office Supplies	...	Minnesota	Prior Lake	55372	2015-12-05	13-05-2015	-24.0300	7	22.23	86838	

In [83]:

```
z=df.groupby(df["Order Date"].dt.month)["Profit"].sum()
z.plot()
```

Out[83]:

```
<AxesSubplot: xlabel='Order Date'>
```



In [27]:

```
#profit per month
x=df.groupby("month")["Profit"].sum()
```

In [28]:

```
z=df.groupby(df["Order Date"].dt.month_name())["Profit"].sum()
z
```

Out[28]:

```
Order Date
April         54034.594984
August        11397.569240
December      11638.594872
February      38086.675304
January        8138.978548
July           3381.085700
June          26129.807917
March          -174.547233
May           78211.808754
November       4344.775500
October      -12508.388680
September      1396.656930
Name: Profit, dtype: float64
```

In [29]:

```
#profits of weekdays on basis of regions
day_per_profits=df.groupby([df["Order Date"].dt.weekday,"Region"])["Profit"].sum()
day_per_profits
```

Out[29]:

```
Order Date  Region
0           Central    13727.207860
            East        1476.992164
            South       1780.680200
            West        5160.203680
1           Central     9542.148560
            East       20987.587369
            South       6811.773101
            West       12525.970580
2           Central     4419.936352
            East       13448.053959
            South       3386.647400
            West       -3891.305386
3           Central     4930.283635
            East       11330.480700
            South      -2059.309540
            West        6028.850228
4           Central    20729.218162
            East        4206.001788
            South      -4768.246220
            West       -1665.007438
5           Central    15190.151990
            East       31939.604049
            South     -18507.420030
            West       33111.440840
6           Central     8826.526110
            East        1902.683417
            South      -1068.179290
            West       24574.637596
Name: Profit, dtype: float64
```

In [30]:

```
a=df["Order Date"].dt.weekday.unique()
a
```

Out[30]:

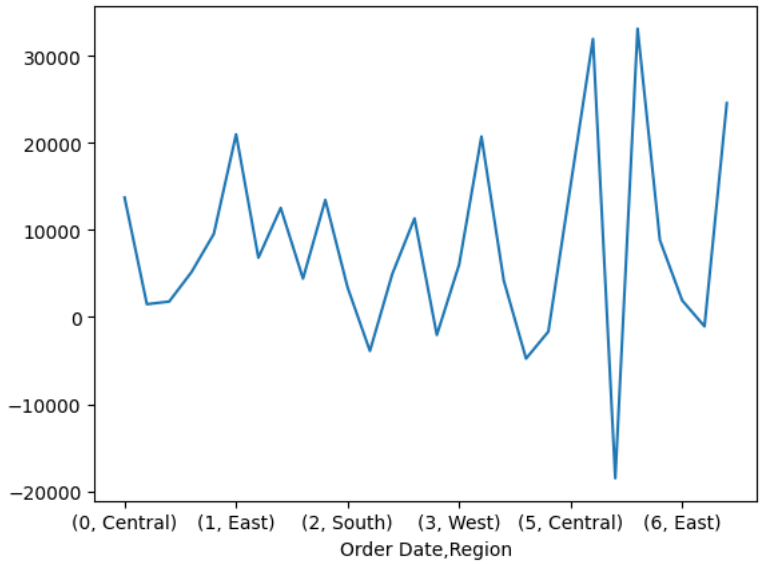
```
array([2, 5, 6, 1, 3, 4, 0], dtype=int64)
```

In [31]:

```
day_per_profits.plot()
```

Out[31]:

```
<AxesSubplot: xlabel='Order Date,Region'>
```



In [32]:

```
df.head()
```

Out[32]:

	Row_ID	Order_Priority	Discount	Unit_Price	Shipping_Cost	Customer ID	Customer Name	Ship Mode	Customer Segment	Product_Category	...	Province	City	P
0	20847	High	0.01	2.84	0.93	3	Bonnie Potter	Express Air	Corporate	Office Supplies	...	Washington	Anacortes	9
1	20228	Not Specified	0.02	500.98	26.00	5	Ronnie Proctor	Delivery Truck	Home Office	Furniture	...	California	San Gabriel	9
2	21776	Critical	0.06	9.48	7.29	11	Marcus Dunlap	Regular Air	Home Office	Furniture	...	New Jersey	Roselle	
3	24844	Medium	0.09	78.69	19.99	14	Gwendolyn F Tyson	Regular Air	Small Business	Furniture	...	Minnesota	Prior Lake	5
4	24846	Medium	0.08	3.28	2.31	14	Gwendolyn F Tyson	Regular Air	Small Business	Office Supplies	...	Minnesota	Prior Lake	5

5 rows × 26 columns

In [33]:

```
#unique values of customer segment column
df["Customer Segment"].unique()
```

Out[33]:

```
array(['Corporate', 'Home Office', 'Small Business', 'Consumer'],
      dtype=object)
```

In [34]:

```
a=df["Order Date"].dt.weekday.unique()
a
```

Out[34]:

```
array([2, 5, 6, 1, 3, 4, 0], dtype=int64)
```

In [35]:

```
#total profit of weekdays
b=df.groupby(df["Order Date"].dt.weekday)["Profit"].sum()
b
```

Out[35]:

```
Order Date
0    22145.083903
1    49867.479610
2    17363.332325
3    20230.305023
4    18501.966292
5    61733.776849
6    34235.667833
Name: Profit, dtype: float64
```

In [36]:

```
c=np.array(b.index)
c
```

Out[36]:

```
array([0, 1, 2, 3, 4, 5, 6], dtype=int64)
```

In [37]:

```
d=np.array(["Mon","Tue","Wed","Thur","Fri","Sat","Sun"])
d
```

Out[37]:

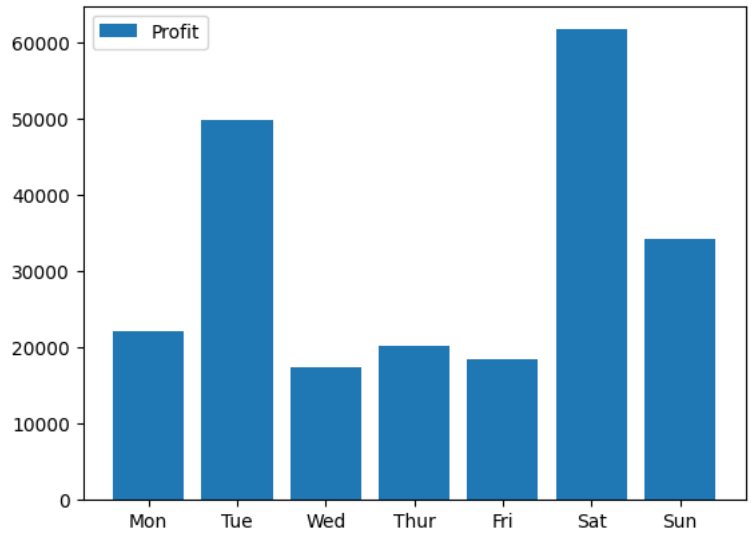
```
array(['Mon', 'Tue', 'Wed', 'Thur', 'Fri', 'Sat', 'Sun'], dtype='<U4')
```

In [38]:

```
plt.xticks(c,d)
plt.bar(d,b,label="Profit")
plt.legend()
```

Out[38]:

```
<matplotlib.legend.Legend at 0x27a7334f5d0>
```



In [39]:

```
#month list
c=np.array(df["Order Date"].dt.month.unique())
c
```

Out[39]:

```
array([ 7,  6,  2, 12,  8,  5,  1,  3, 10,  4,  9, 11], dtype=int64)
```

In [40]:

```
#month name
c=np.array(df["Order Date"].dt.month_name().unique())
c
```

Out[40]:

```
array(['July', 'June', 'February', 'December', 'August', 'May', 'January',
       'March', 'October', 'April', 'September', 'November'], dtype=object)
```

In [41]:

```
df.head()
```

Out[41]:

	Row_ID	Order_Priority	Discount	Unit_Price	Shipping_Cost	Customer ID	Customer Name	Ship Mode	Customer Segment	Product_Category	...	Province	City	P
0	20847	High	0.01	2.84	0.93	3	Bonnie Potter	Express Air	Corporate	Office Supplies	...	Washington	Anacortes	9
1	20228	Not Specified	0.02	500.98	26.00	5	Ronnie Proctor	Delivery Truck	Home Office	Furniture	...	California	San Gabriel	9
2	21776	Critical	0.06	9.48	7.29	11	Marcus Dunlap	Regular Air	Home Office	Furniture	...	New Jersey	Roselle	
3	24844	Medium	0.09	78.69	19.99	14	Gwendolyn F Tyson	Regular Air	Small Business	Furniture	...	Minnesota	Prior Lake	5
4	24846	Medium	0.08	3.28	2.31	14	Gwendolyn F Tyson	Regular Air	Small Business	Office Supplies	...	Minnesota	Prior Lake	5

5 rows × 26 columns

In [42]:

```
#month profits with groupby Customer Segment

Month_profits=df.groupby([df["Order Date"].dt.month_name(),"Customer Segment"])["Profit"].sum()
Month_profits
```

Out[42]:

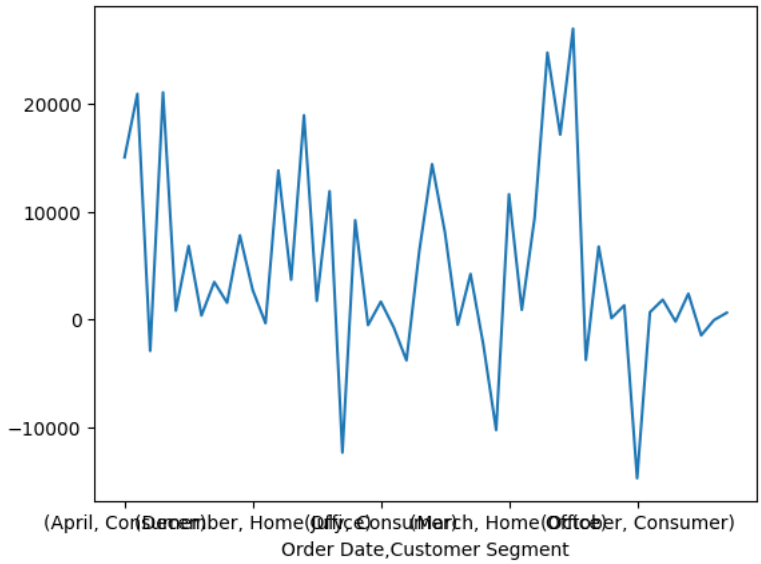
```
Order Date  Customer Segment
April       Consumer            15015.005752
            Corporate           20909.717488
            Home Office         -2935.614836
            Small Business      21045.486580
August      Consumer              796.898340
            Corporate            6803.548300
            Home Office           344.591900
            Small Business       3452.530700
December    Consumer             1526.351422
            Corporate            7778.914550
            Home Office          2705.856300
            Small Business       -372.527400
February    Consumer            13804.379546
            Corporate            3659.269877
            Home Office         18922.900841
            Small Business       1700.125040
January     Consumer            11877.078720
            Corporate          -12373.822365
            Home Office          9183.897966
            Small Business       -548.175773
July        Consumer             1624.276840
            Corporate            -744.662300
            Home Office         -3821.836820
            Small Business       6323.307980
June        Consumer            14382.828550
            Corporate            8068.996775
            Home Office          -522.251720
            Small Business       4200.234312
March       Consumer            -2367.276140
            Corporate          -10278.690826
            Home Office         11598.552600
            Small Business        872.867133
May         Consumer             9401.014163
            Corporate           24730.024423
            Home Office         17126.092708
            Small Business      26954.677460
November    Consumer            -3780.965100
            Corporate            6740.407700
            Home Office           100.028000
            Small Business       1285.304900
October     Consumer           -14753.668080
            Corporate            645.859700
            Home Office          1810.404800
            Small Business       -210.985100
September   Consumer             2369.066260
            Corporate           -1495.372320
            Home Office           -79.100870
            Small Business        602.063860
Name: Profit, dtype: float64
```

In [43]:

```
Month_profits.plot.line()
```

Out[43]:

```
<AxesSubplot: xlabel='Order Date,Customer Segment'>
```



In [80]:

```
#difference between Order date and Ship date
df["Difference"]=(df["Order Date"]-df["Ship_Date"]).dt.days
df["Difference"]
```

Out[80]:

```
0       -31
1        -2
2        -2
3       205
4       206
       ...
1947    -30
1948    -30
1949     -2
1950      0
1951    -92
Name: Difference, Length: 1952, dtype: int64
```

In [90]:

```
#average time for shipping
aveage_days_shipping=df.groupby("Ship Mode")["Difference"].mean()
aveage_days_shipping
```

Out[90]:

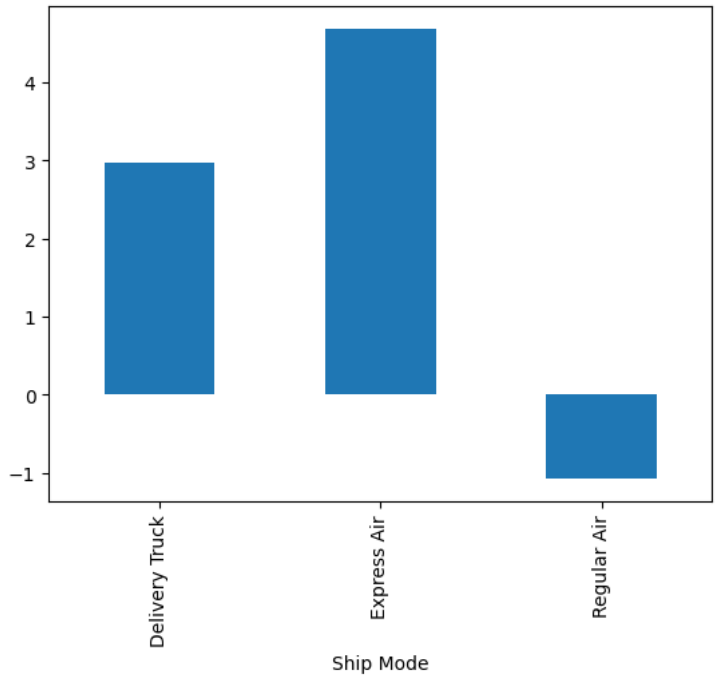
```
Ship Mode
Delivery Truck    2.970909
Express Air       4.683333
Regular Air      -1.071677
Name: Difference, dtype: float64
```

In [88]:

```
aveage_days_shipping.plot.bar()
```

Out[88]:

```
<AxesSubplot: xlabel='Ship Mode'>
```



In [92]:

```
#total discount grouped by Order_Priority
df.groupby(["Order_Priority"])["Discount"].sum()
```

Out[92]:

```
Order_Priority
Critical         18.75
Critical          0.06
High             20.00
Low              20.39
Medium           18.23
Not Specified    18.17
Name: Discount, dtype: float64
```

In [116]:

```
#count of order priority for particular city

f=df.groupby(df["Order_Priority"])["City"].value_counts()
f
```

Out[116]:

```
Order_Priority  City
Critical        Los Angeles       7
                New York City     7
                Seattle           6
                Boston            4
                Bowling Green     4
                                 ..
Not Specified   Whitehall         1
                Wilmington        1
                Winter Park       1
                Winter Springs    1
                Woburn            1
Name: City, Length: 1263, dtype: int64
```