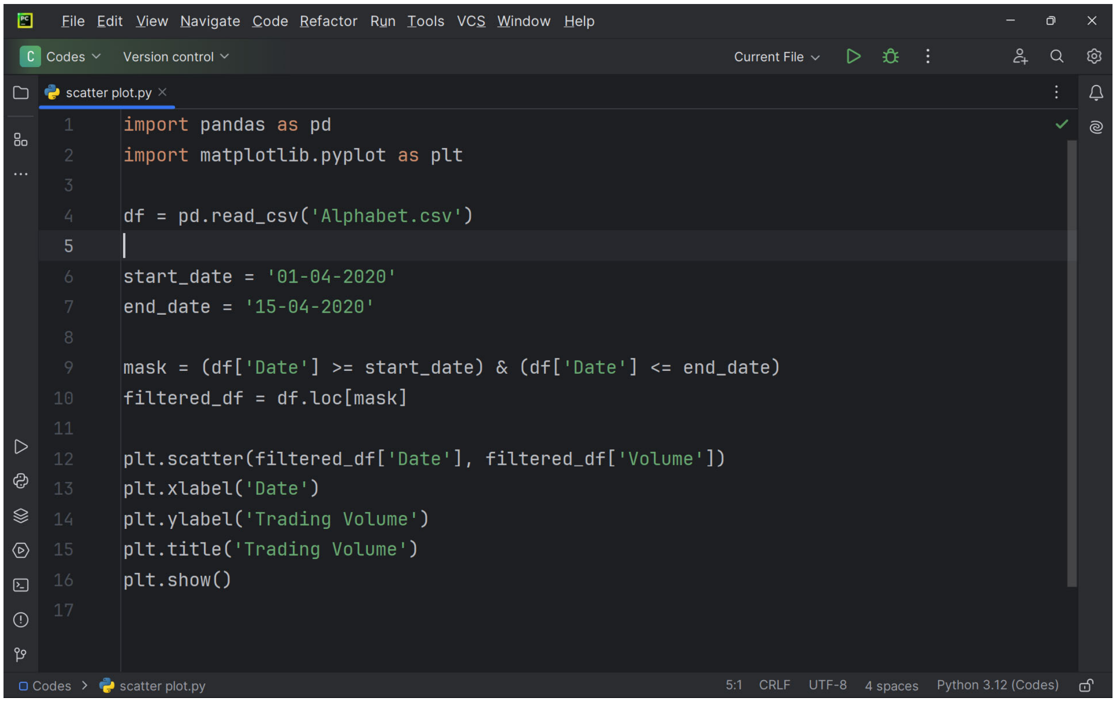
Experiment 6

Aim:

To develop a Pandas program to create a scatter plot of the trading volume of Alphabet Inc. between two specific dates

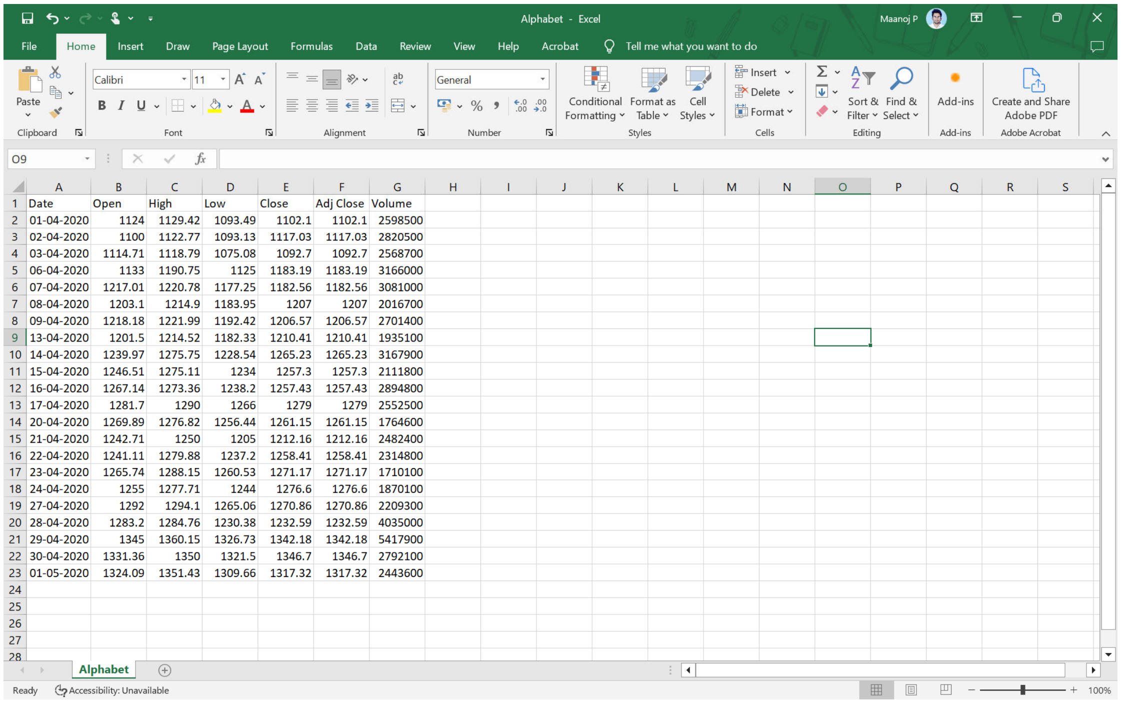
Code:



File Edit View Navigate Code Refactor Run Tools VCS Window Help
Codes
Version control
Current File
scatter plot.py
Codes > scatter plot.py
5:1 CRLF UTF-8 4 spaces Python 3.12 (Codes)

```python
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('Alphabet.csv')

start_date = '01-04-2020'
end_date = '15-04-2020'

mask = (df['Date'] >= start_date) & (df['Date'] <= end_date)
filtered_df = df.loc[mask]

plt.scatter(filtered_df['Date'], filtered_df['Volume'])
plt.xlabel('Date')
plt.ylabel('Trading Volume')
plt.title('Trading Volume')
plt.show()
```

Input:



Date	Open	High	Low	Close	Adj Close	Volume
01-04-2020	1124	1129.42	1093.49	1102.1	1102.1	2598500
02-04-2020	1100	1122.77	1093.13	1117.03	1117.03	2820500
03-04-2020	1114.71	1118.79	1075.08	1092.7	1092.7	2568700
06-04-2020	1133	1190.75	1125	1183.19	1183.19	3166000
07-04-2020	1217.01	1220.78	1177.25	1182.56	1182.56	3081000
08-04-2020	1203.1	1214.9	1183.95	1207	1207	2016700
09-04-2020	1218.18	1221.99	1192.42	1206.57	1206.57	2701400
13-04-2020	1201.5	1214.52	1182.33	1210.41	1210.41	1935100
14-04-2020	1239.97	1275.75	1228.54	1265.23	1265.23	3167900
15-04-2020	1246.51	1275.11	1234	1257.3	1257.3	2111800
16-04-2020	1267.14	1273.36	1238.2	1257.43	1257.43	2894800
17-04-2020	1281.7	1290	1266	1279	1279	2552500
20-04-2020	1269.89	1276.82	1256.44	1261.15	1261.15	1764600
21-04-2020	1242.71	1250	1205	1212.16	1212.16	2482400
22-04-2020	1241.11	1279.88	1237.2	1258.41	1258.41	2314800
23-04-2020	1265.74	1288.15	1260.53	1271.17	1271.17	1710100
24-04-2020	1255	1277.71	1244	1276.6	1276.6	1870100
27-04-2020	1292	1294.1	1265.06	1270.86	1270.86	2209300
28-04-2020	1283.2	1284.76	1230.38	1232.59	1232.59	4035000
29-04-2020	1345	1360.15	1326.73	1342.18	1342.18	5417900
30-04-2020	1331.36	1350	1321.5	1346.7	1346.7	2792100
01-05-2020	1324.09	1351.43	1309.66	1317.32	1317.32	2443600

Output:

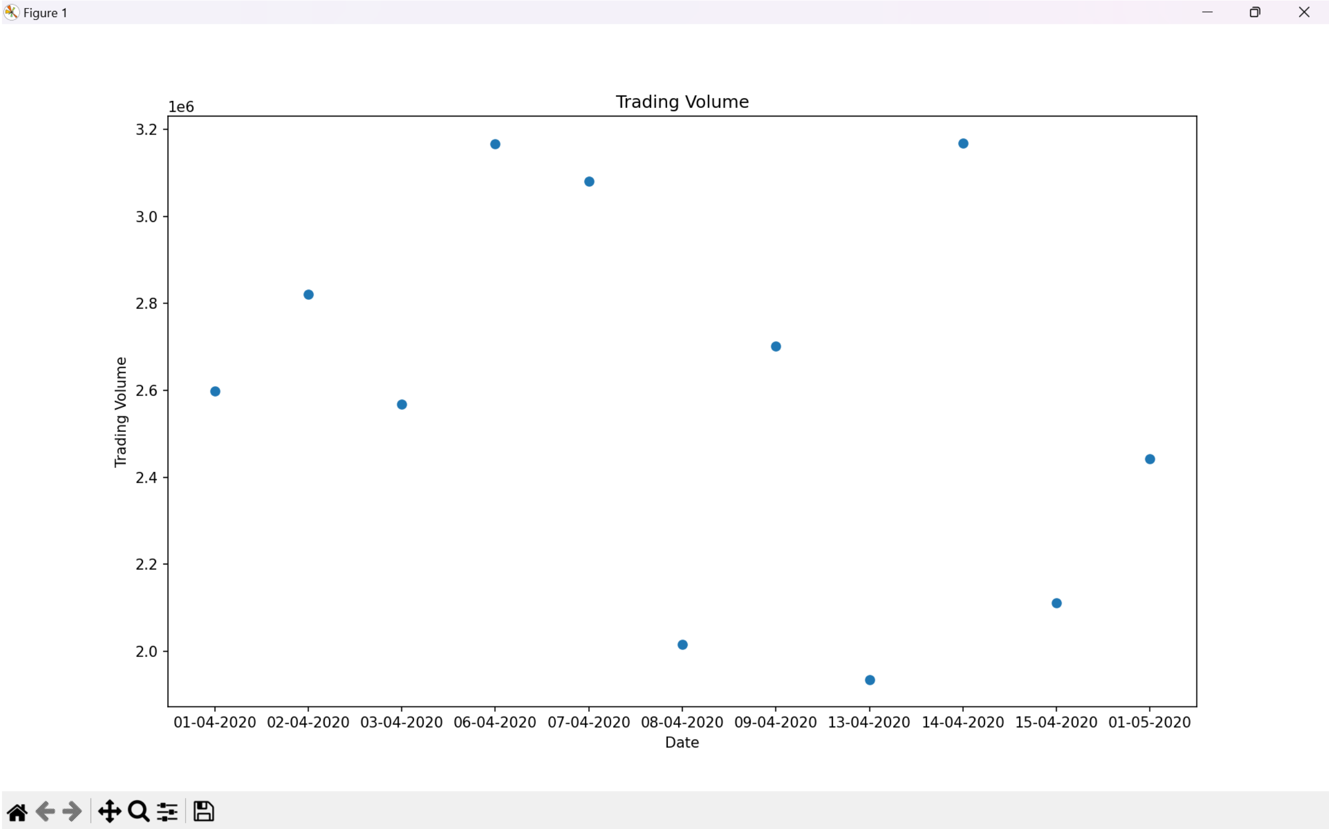


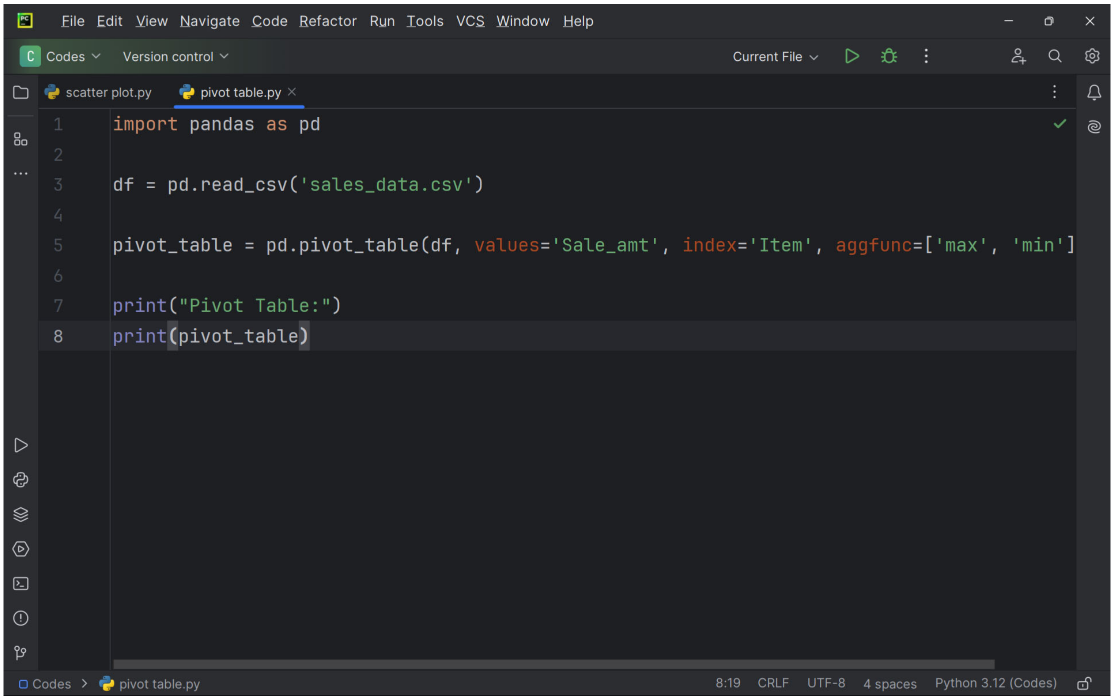
Figure 1
Trading Volume
1e6
3.2
3.0
2.8
2.6
2.4
2.2
2.0
Trading Volume
01-04-2020
02-04-2020
03-04-2020
06-04-2020
07-04-2020
08-04-2020
09-04-2020
13-04-2020
14-04-2020
15-04-2020
01-05-2020
Date

Experiment 7

Aim:

To develop a Pandas program to create a pivot table and find the maximum and minimum sale value of the items

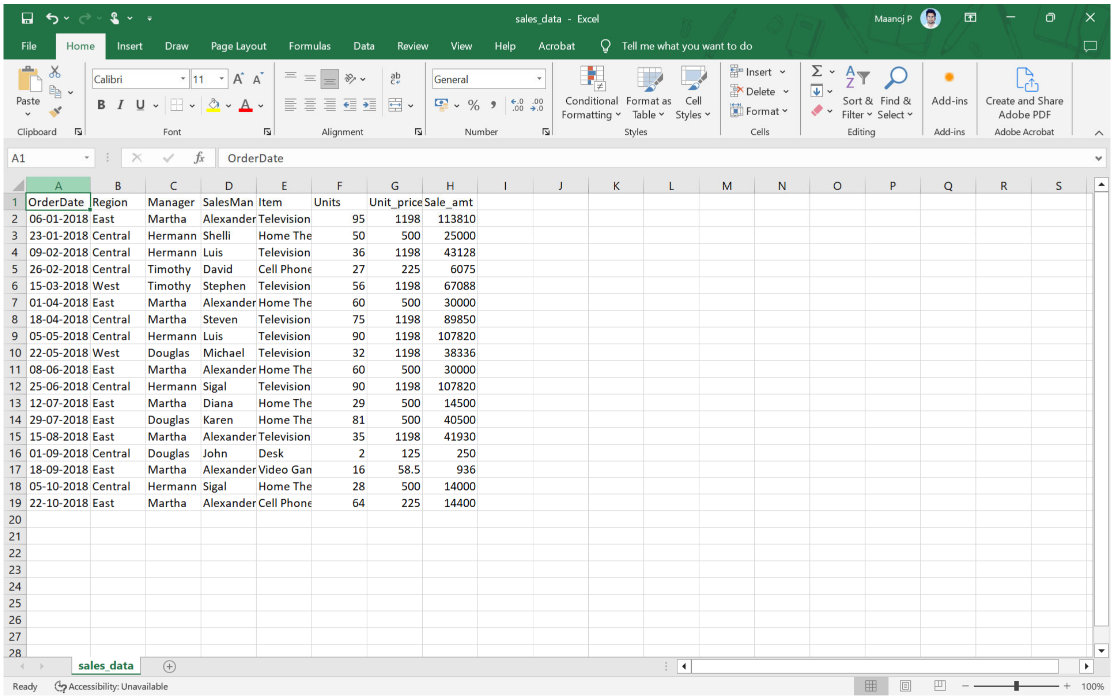
Code:



File Edit View Navigate Code Refactor Run Tools VCS Window Help
Codes
Version control
Current File
scatter plot.py
pivot table.py
Codes > pivot table.py
8:19 CRLF UTF-8 4 spaces Python 3.12 (Codes)

```python
import pandas as pd

df = pd.read_csv('sales_data.csv')

pivot_table = pd.pivot_table(df, values='Sale_amt', index='Item', aggfunc=['max', 'min']

print("Pivot Table:")
print(pivot_table)
```

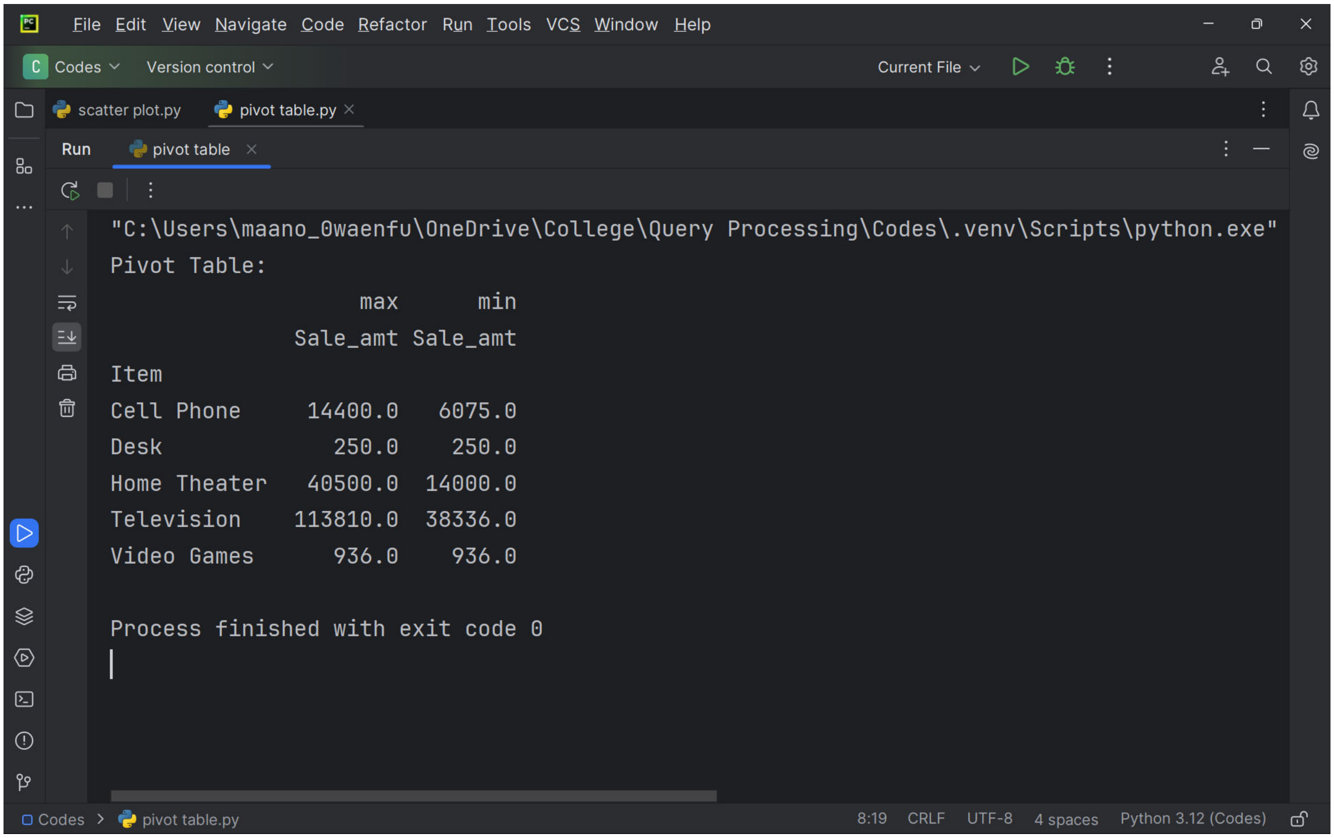
Input:



sales_data - Excel
Maanoj P
File Home Insert Draw Page Layout Formulas Data Review View Help Acrobat Tell me what you want to do
Clipboard Font Alignment Number Styles Cells Editing Add-ins Adobe Acrobat
A1
OrderDate
sales_data
Ready
Accessibility: Unavailable
100%

	A	B	C	D	E	F	G	H
1	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
2	06-01-2018	East	Martha	Alexander	Television	95	1198	113810
3	23-01-2018	Central	Hermann	Shelli	Home The	50	500	25000
4	09-02-2018	Central	Hermann	Luis	Television	36	1198	43128
5	26-02-2018	Central	Timothy	David	Cell Phone	27	225	6075
6	15-03-2018	West	Timothy	Stephen	Television	56	1198	67088
7	01-04-2018	East	Martha	Alexander	Home The	60	500	30000
8	18-04-2018	Central	Martha	Steven	Television	75	1198	89850
9	05-05-2018	Central	Hermann	Luis	Television	90	1198	107820
10	22-05-2018	West	Douglas	Michael	Television	32	1198	38336
11	08-06-2018	East	Martha	Alexander	Home The	60	500	30000
12	25-06-2018	Central	Hermann	Sigal	Television	90	1198	107820
13	12-07-2018	East	Martha	Diana	Home The	29	500	14500
14	29-07-2018	East	Douglas	Karen	Home The	81	500	40500
15	15-08-2018	East	Martha	Alexander	Television	35	1198	41930
16	01-09-2018	Central	Douglas	John	Desk	2	125	250
17	18-09-2018	East	Martha	Alexander	Video Gan	16	58.5	936
18	05-10-2018	Central	Hermann	Sigal	Home The	28	500	14000
19	22-10-2018	East	Martha	Alexander	Cell Phone	64	225	14400

Output:



File Edit View Navigate Code Refactor Run Tools VCS Window Help
Codes
Version control
Current File
scatter plot.py
pivot table.py
Run
pivot table
Codes > pivot table.py
8:19 CRLF UTF-8 4 spaces Python 3.12 (Codes)

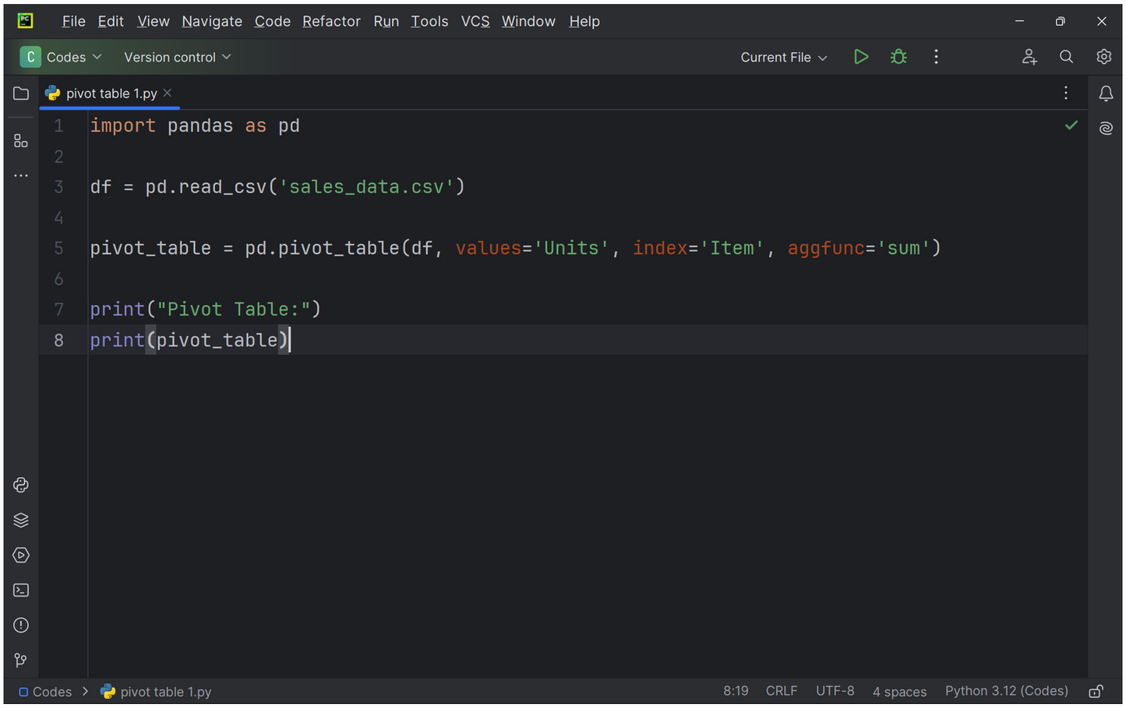
```
"C:\Users\maano_0waenfu\OneDrive\College\Query Processing\Codes\.venv\Scripts\python.exe"
Pivot Table:
                   max      min
              Sale_amt Sale_amt
Item
Cell Phone     14400.0   6075.0
Desk             250.0    250.0
Home Theater   40500.0  14000.0
Television    113810.0  38336.0
Video Games      936.0    936.0

Process finished with exit code 0
```

Experiment 8

Aim:

To develop a Pandas program to create a pivot table and find the item wise unit sold

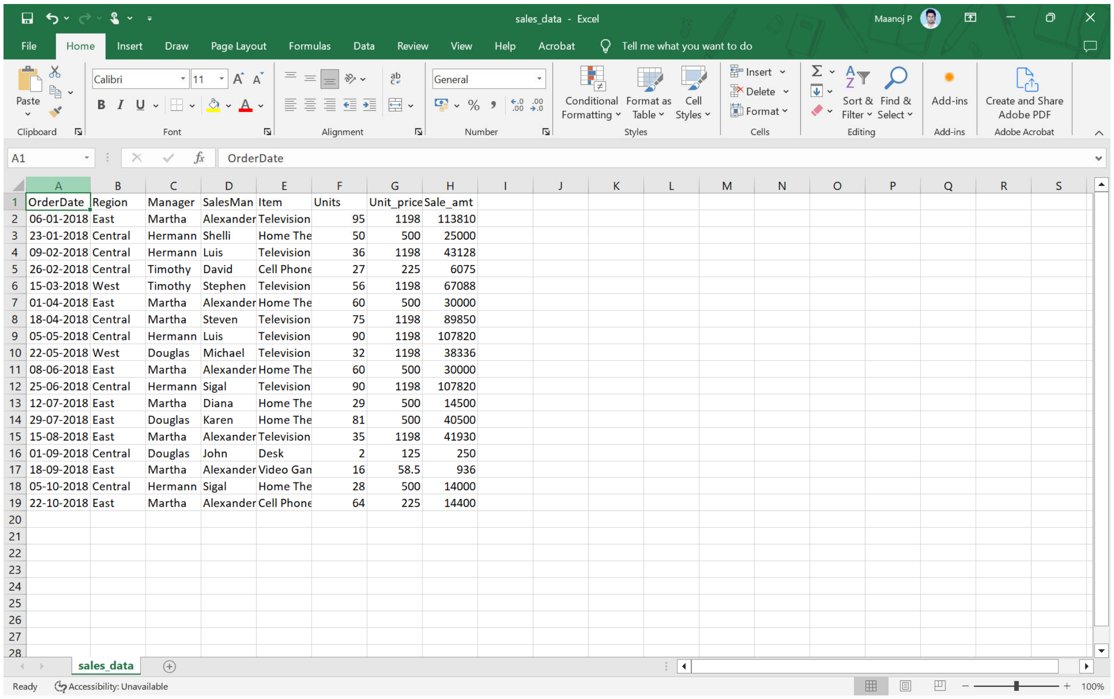
Code:



File Edit View Navigate Code Refactor Run Tools VCS Window Help
Codes
Version control
Current File
pivot table 1.py
Codes > pivot table 1.py
8:19 CRLF UTF-8 4 spaces Python 3.12 (Codes)

```python
import pandas as pd

df = pd.read_csv('sales_data.csv')

pivot_table = pd.pivot_table(df, values='Units', index='Item', aggfunc='sum')

print("Pivot Table:")
print(pivot_table)
```

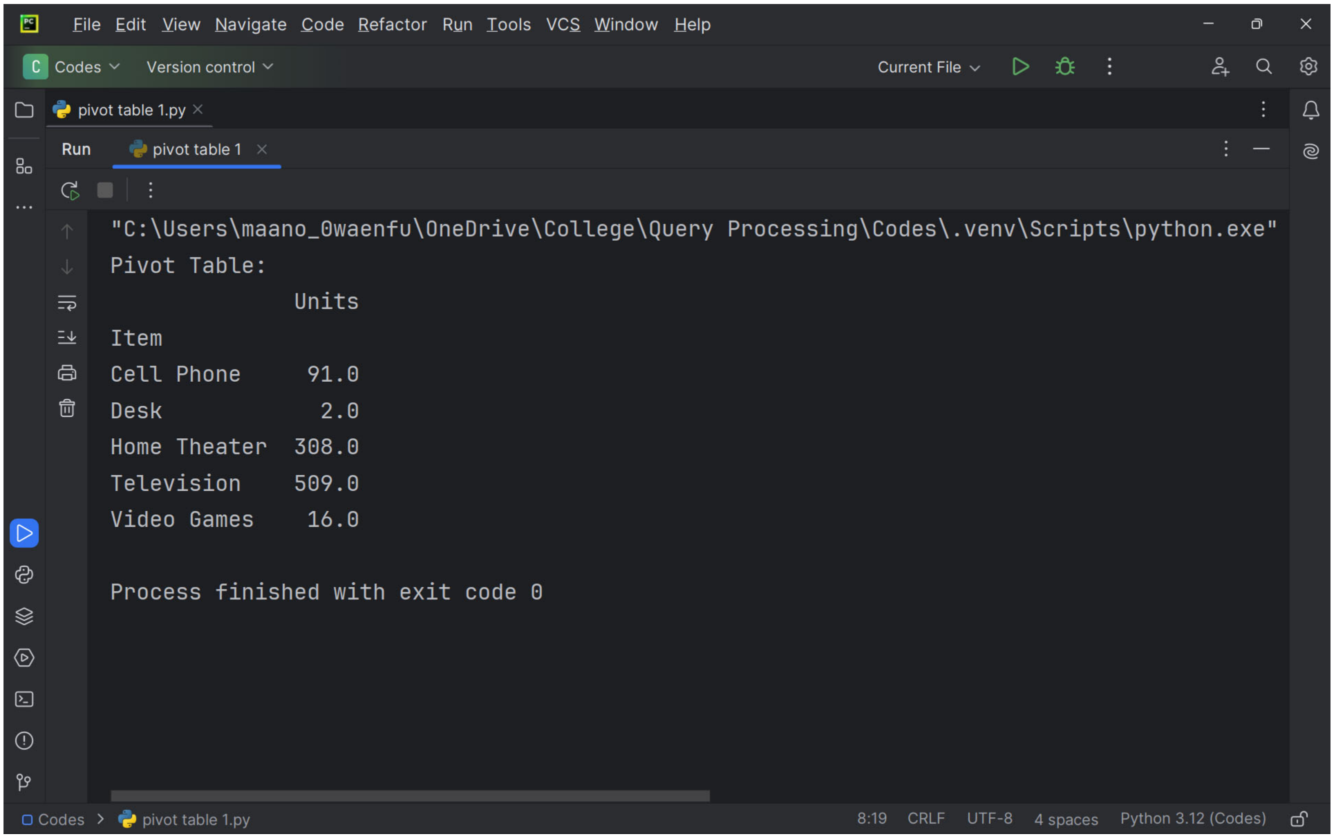
Input:



sales_data - Excel
Maanoj P
File Home Insert Draw Page Layout Formulas Data Review View Help Acrobat Tell me what you want to do
Paste
Calibri 11
General
Conditional Formatting
Format as Table
Cell Styles
Insert
Delete
Format
Sort & Filter
Find & Select
Add-ins
Create and Share Adobe PDF
Clipboard Font Alignment Number Styles Cells Editing Add-ins Adobe Acrobat
A1
OrderDate
sales_data
Ready
Accessibility: Unavailable
100%

OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
06-01-2018	East	Martha	Alexander	Television	95	1198	113810
23-01-2018	Central	Hermann	Shelli	Home The	50	500	25000
09-02-2018	Central	Hermann	Luis	Television	36	1198	43128
26-02-2018	Central	Timothy	David	Cell Phone	27	225	6075
15-03-2018	West	Timothy	Stephen	Television	56	1198	67088
01-04-2018	East	Martha	Alexander	Home The	60	500	30000
18-04-2018	Central	Martha	Steven	Television	75	1198	89850
05-05-2018	Central	Hermann	Luis	Television	90	1198	107820
22-05-2018	West	Douglas	Michael	Television	32	1198	38336
08-06-2018	East	Martha	Alexander	Home The	60	500	30000
25-06-2018	Central	Hermann	Sigal	Television	90	1198	107820
12-07-2018	East	Martha	Diana	Home The	29	500	14500
29-07-2018	East	Douglas	Karen	Home The	81	500	40500
15-08-2018	East	Martha	Alexander	Television	35	1198	41930
01-09-2018	Central	Douglas	John	Desk	2	125	250
18-09-2018	East	Martha	Alexander	Video Gan	16	58.5	936
05-10-2018	Central	Hermann	Sigal	Home The	28	500	14000
22-10-2018	East	Martha	Alexander	Cell Phone	64	225	14400

Output:



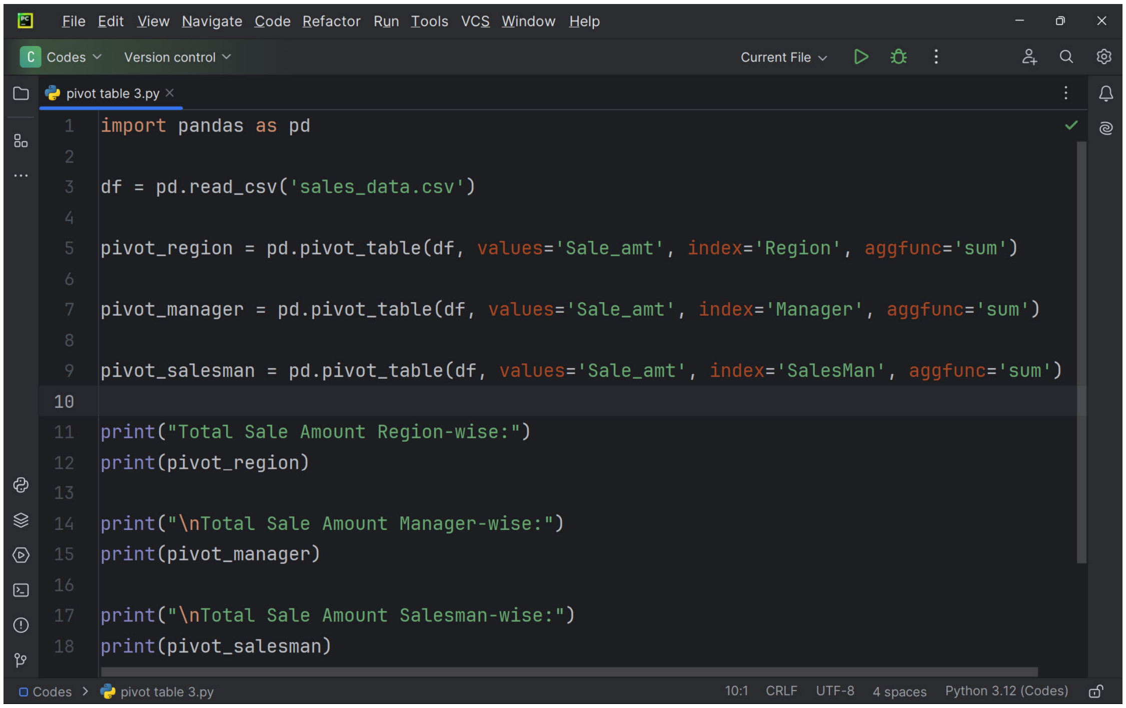
```
"C:\Users\maano_0waenfu\OneDrive\College\Query Processing\Codes\.venv\Scripts\python.exe"
Pivot Table:
              Units
Item
Cell Phone     91.0
Desk            2.0
Home Theater  308.0
Television    509.0
Video Games    16.0

Process finished with exit code 0
```

Experiment 9

Aim:

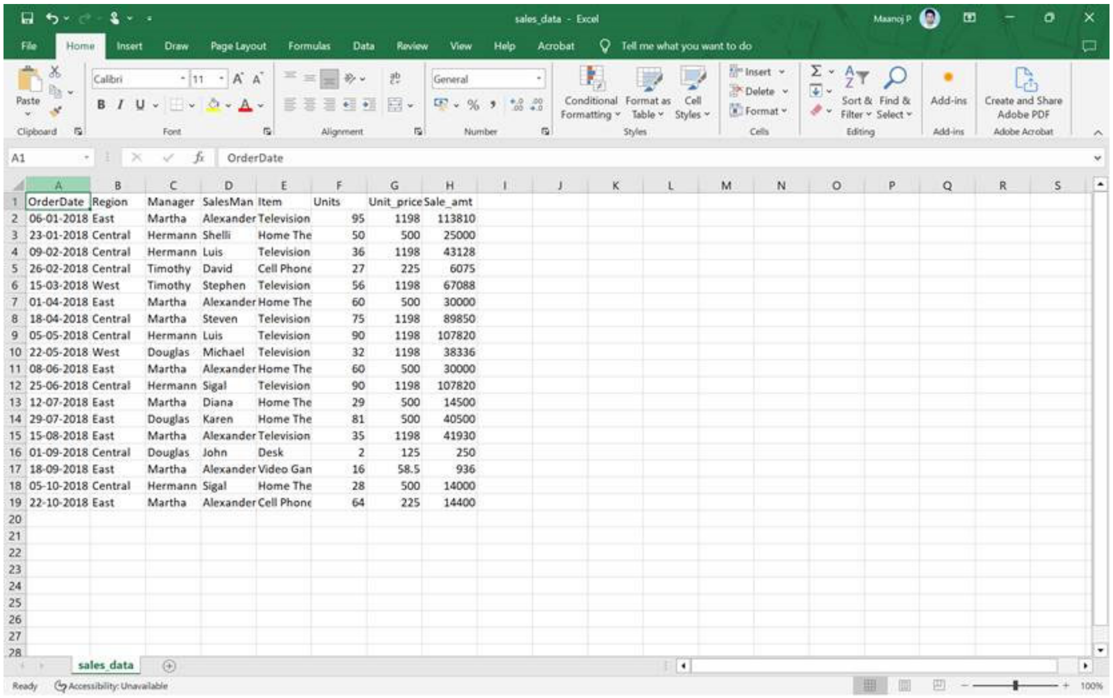
To develop a Pandas program to create a pivot table and find the total sale amount region wise, manager wise, sales man wise

Code:



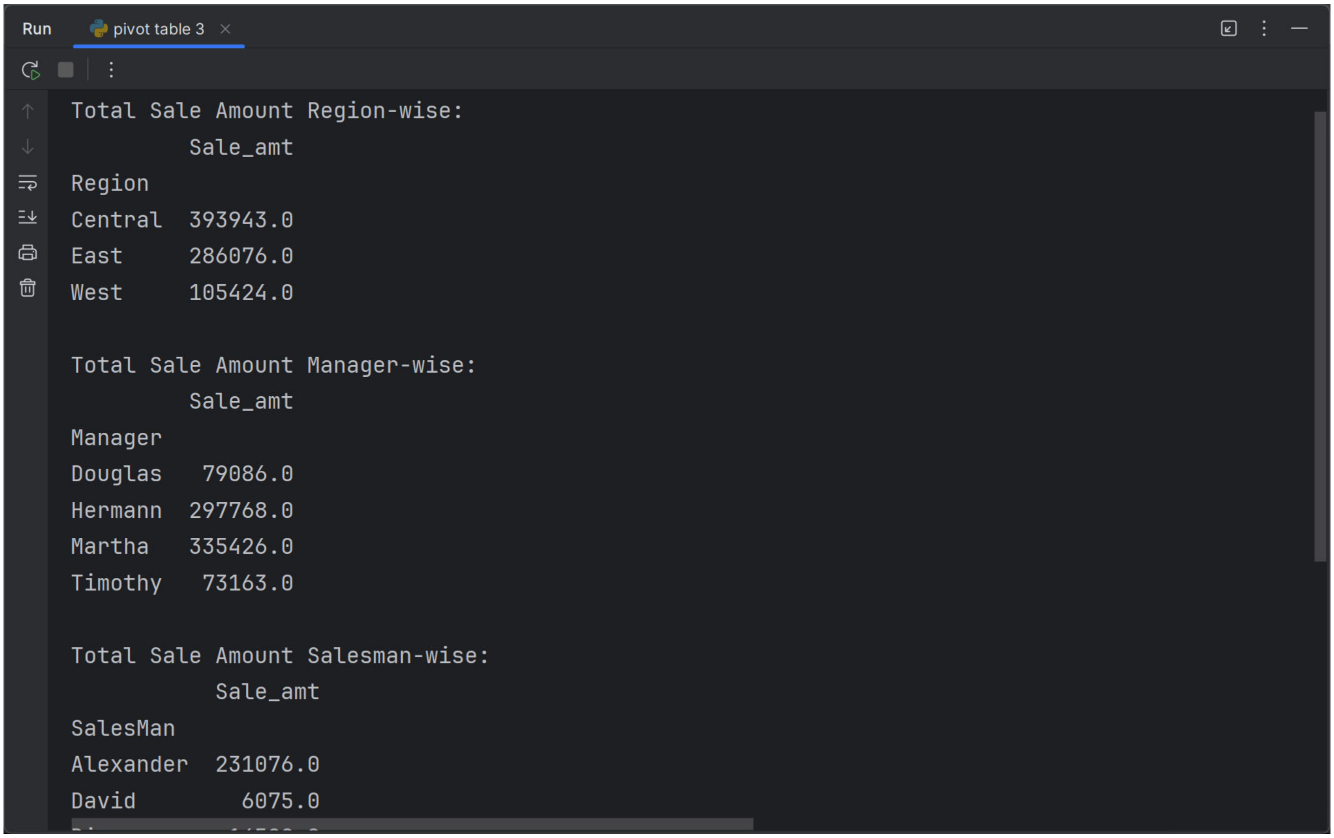
```python
import pandas as pd

df = pd.read_csv('sales_data.csv')

pivot_region = pd.pivot_table(df, values='Sale_amt', index='Region', aggfunc='sum')

pivot_manager = pd.pivot_table(df, values='Sale_amt', index='Manager', aggfunc='sum')

pivot_salesman = pd.pivot_table(df, values='Sale_amt', index='SalesMan', aggfunc='sum')

print("Total Sale Amount Region-wise:")
print(pivot_region)

print("\nTotal Sale Amount Manager-wise:")
print(pivot_manager)

print("\nTotal Sale Amount Salesman-wise:")
print(pivot_salesman)
```

Input:



OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
06-01-2018	East	Martha	Alexander	Television	95	1198	113810
23-01-2018	Central	Hermann	Shelli	Home The	50	500	25000
09-02-2018	Central	Hermann	Luis	Television	36	1198	43128
26-02-2018	Central	Timothy	David	Cell Phone	27	225	6075
15-03-2018	West	Timothy	Stephen	Television	56	1198	67088
01-04-2018	East	Martha	Alexander	Home The	60	500	30000
18-04-2018	Central	Martha	Steven	Television	75	1198	89850
05-05-2018	Central	Hermann	Luis	Television	90	1198	107820
22-05-2018	West	Douglas	Michael	Television	32	1198	38336
08-06-2018	East	Martha	Alexander	Home The	60	500	30000
25-06-2018	Central	Hermann	Sigal	Television	90	1198	107820
12-07-2018	East	Martha	Diana	Home The	29	500	14500
29-07-2018	East	Douglas	Karen	Home The	81	500	40500
15-08-2018	East	Martha	Alexander	Television	35	1198	41930
01-09-2018	Central	Douglas	John	Desk	2	125	250
18-09-2018	East	Martha	Alexander	Video Gan	16	58.5	936
05-10-2018	Central	Hermann	Sigal	Home The	28	500	14000
22-10-2018	East	Martha	Alexander	Cell Phone	64	225	14400

Output:



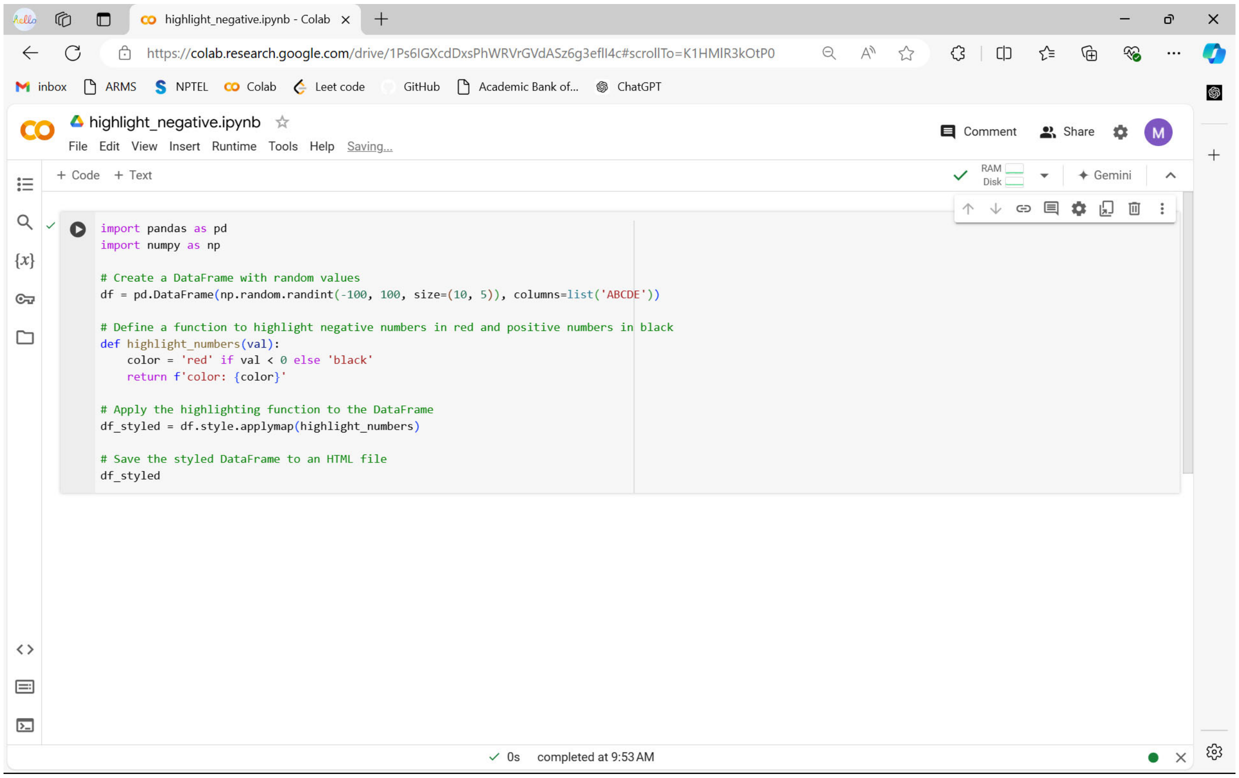
```
Total Sale Amount Region-wise:
         Sale_amt
Region
Central  393943.0
East     286076.0
West     105424.0

Total Sale Amount Manager-wise:
         Sale_amt
Manager
Douglas   79086.0
Hermann  297768.0
Martha   335426.0
Timothy   73163.0

Total Sale Amount Salesman-wise:
           Sale_amt
SalesMan
Alexander  231076.0
David        6075.0
```

Experiment 10

Aim:

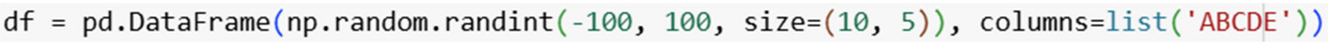
To develop a Pandas program to highlight the negative numbers red and positive numbers black.

Code:



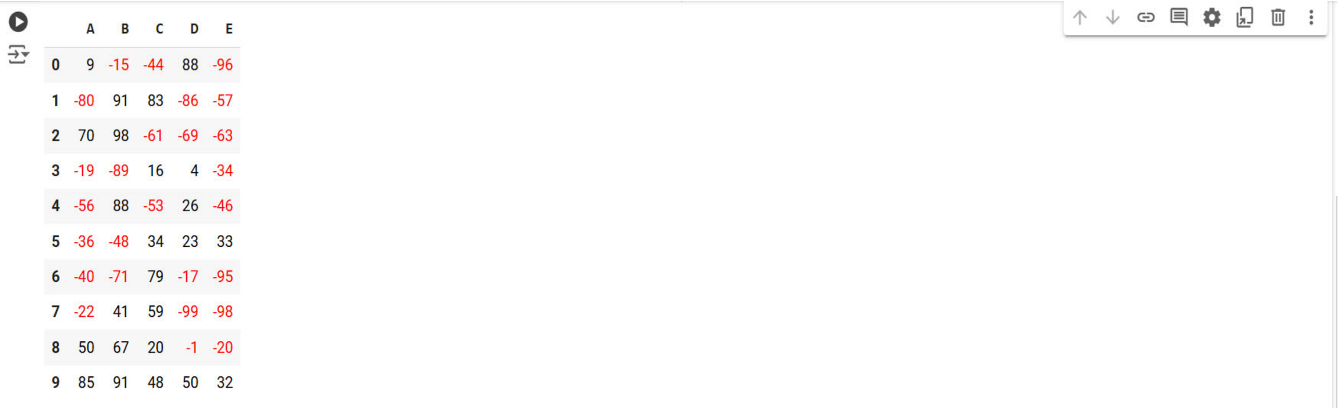
```
import pandas as pd
import numpy as np

# Create a DataFrame with random values
df = pd.DataFrame(np.random.randint(-100, 100, size=(10, 5)), columns=list('ABCDE'))

# Define a function to highlight negative numbers in red and positive numbers in black
def highlight_numbers(val):
    color = 'red' if val < 0 else 'black'
    return f'color: {color}'

# Apply the highlighting function to the DataFrame
df_styled = df.style.applymap(highlight_numbers)

# Save the styled DataFrame to an HTML file
df_styled
```

Input:



```
df = pd.DataFrame(np.random.randint(-100, 100, size=(10, 5)), columns=list('ABCDE'))
```

Output:



```
     A    B    C    D    E
0    9  -15  -44   88  -96
1  -80   91   83  -86  -57
2   70   98  -61  -69  -63
3  -19  -89   16    4  -34
4  -56   88  -53   26  -46
5  -36  -48   34   23   33
6  -40  -71   79  -17  -95
7  -22   41   59  -99  -98
8   50   67   20   -1  -20
9   85   91   48   50   32
```