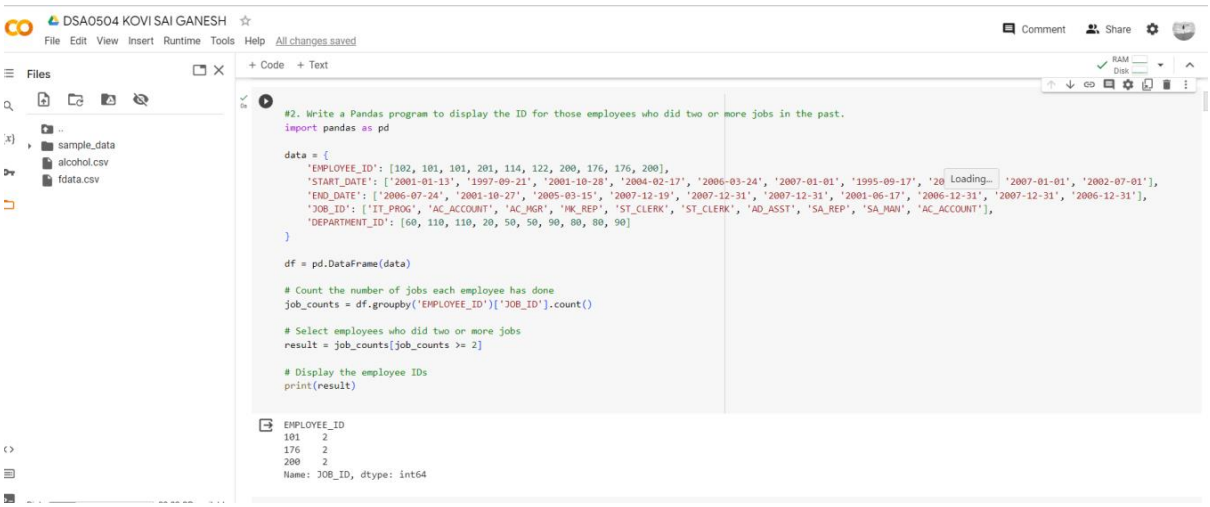
OUTPUT DOCUMENT – LAB EXPERIMENTS

NAME : Kovi Sai Ganesh
REGISTER NUMBER : 192124028
COURSE CODE: DSA0504
COURSE NAME: Query Processing for Data Science in Open Source Platform

1. Write a Pandas program to select distinct department id from employees file.

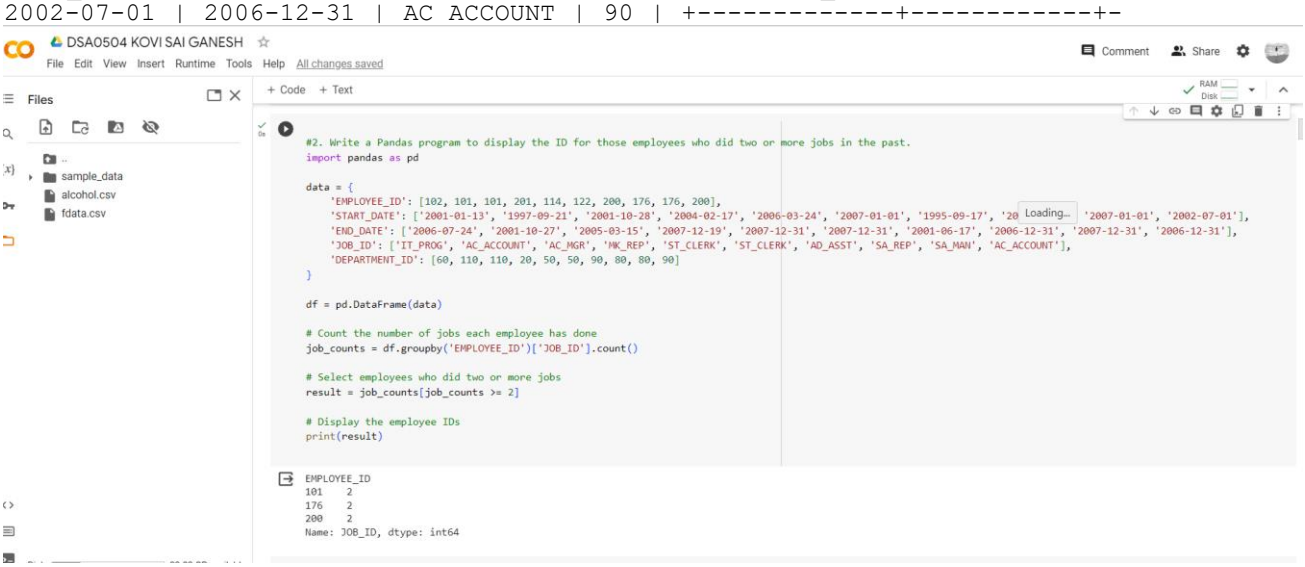
```
+---------------+----------------------+------------+-------------+   |
DEPARTMENT_ID | DEPARTMENT_NAME | MANAGER_ID | LOCATION_ID | +--------
-------+----------------------+------------+-------------+   |   10   |
Administration | 200 | 1700 | | 20 | Marketing | 201 | 1800 | | 30 |
Purchasing | 114 | 1700 | | 40 | Human Resources | 203 | 2400 | | 50 |
Shipping | 121 | 1500 | | 60 | IT | 103 | 1400 | | 70 | Public Relations
| 204 | 2700 | | 80 | Sales | 145 | 2500 | | 90 | Executive | 100 | 1700
| | 100 | Finance | 108 | 1700 | | 110 | Accounting | 205 | 1700 | |
120 | Treasury | 0 | 1700 | | 130 | Corporate Tax | 0 | 1700 | | 140 |
Control And Credit | 0 | 1700 | | 150 | Shareholder Services | 0 | 1700
| | 160 | Benefits | 0 | 1700 | | 170 | Manufacturing | 0 | 1700 | |
180 | Construction | 0 | 1700 | | 190 | Contracting | 0 | 1700 | | 200
| Operations | 0 | 1700 | | 210 | IT Support | 0 | 1700 | | 220 | NOC |
0 | 1700 | | 230 | IT Helpdesk | 0 | 1700 | | 240 | Government Sales |
0 | 1700 | | 250 | Retail Sales | 0 | 1700 | | 260 | Recruiting | 0 |
1700 | | 270 | Payroll | 0 | 1700 | +---------------+----------------
-----+------------+-------------
```



2. Write a Pandas program to display the ID for those employees who did two or more jobs in the past.

```
+-------------+------------+------------+------------+---------------+   |
EMPLOYEE_ID | START_DATE | END_DATE | JOB_ID | DEPARTMENT_ID | +---------
----+------------+------------+------------+---------------+ | 102 | 2001-
01-13 | 2006-07-24 | IT_PROG | 60 | | 101 | 1997-09-21 | 2001-10-27 |
AC_ACCOUNT | 110 | | 101 | 2001-10-28 | 2005-03-15 | AC_MGR | 110 | | 201
| 2004-02-17 | 2007-12-19 | MK_REP | 20 | | 114 | 2006-03-24 | 2007-12-31
| ST_CLERK | 50 | | 122 | 2007-01-01 | 2007-12-31 | ST_CLERK | 50 | | 200
| 1995-09-17 | 2001-06-17 | AD_ASST | 90 | | 176 | 2006-03-24 | 2006-12-31
```

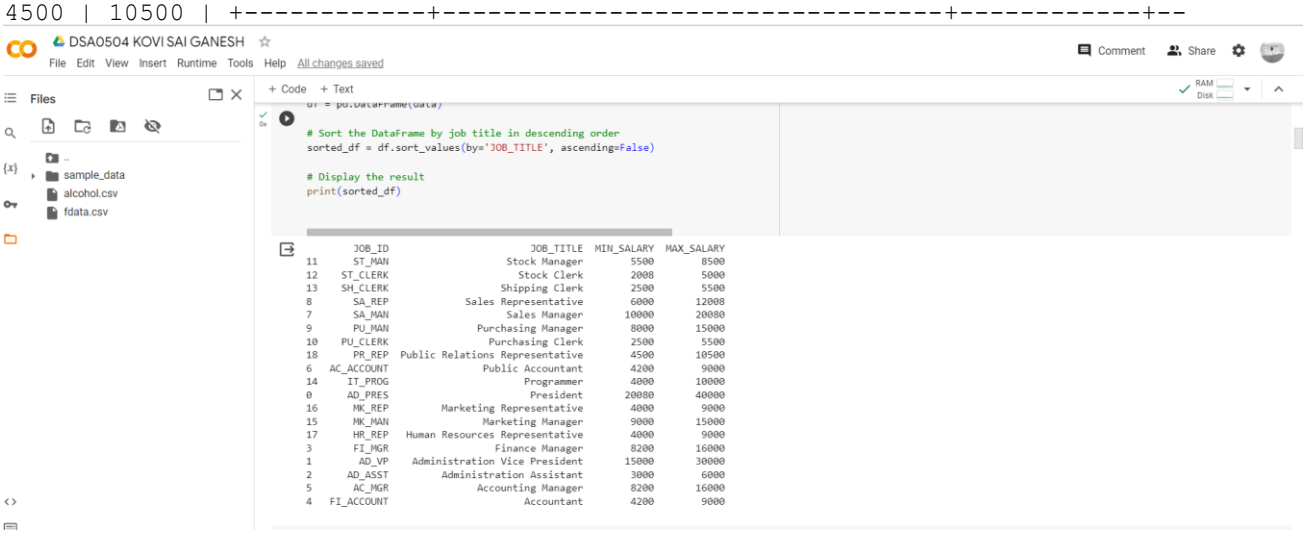
| SA_REP | 80 | | 176 | 2007-01-01 | 2007-12-31 | SA_MAN | 80 | | 200 | 2002-07-01 | 2006-12-31 | AC_ACCOUNT | 90 | +----------+----------+-



----------+----------+----------+

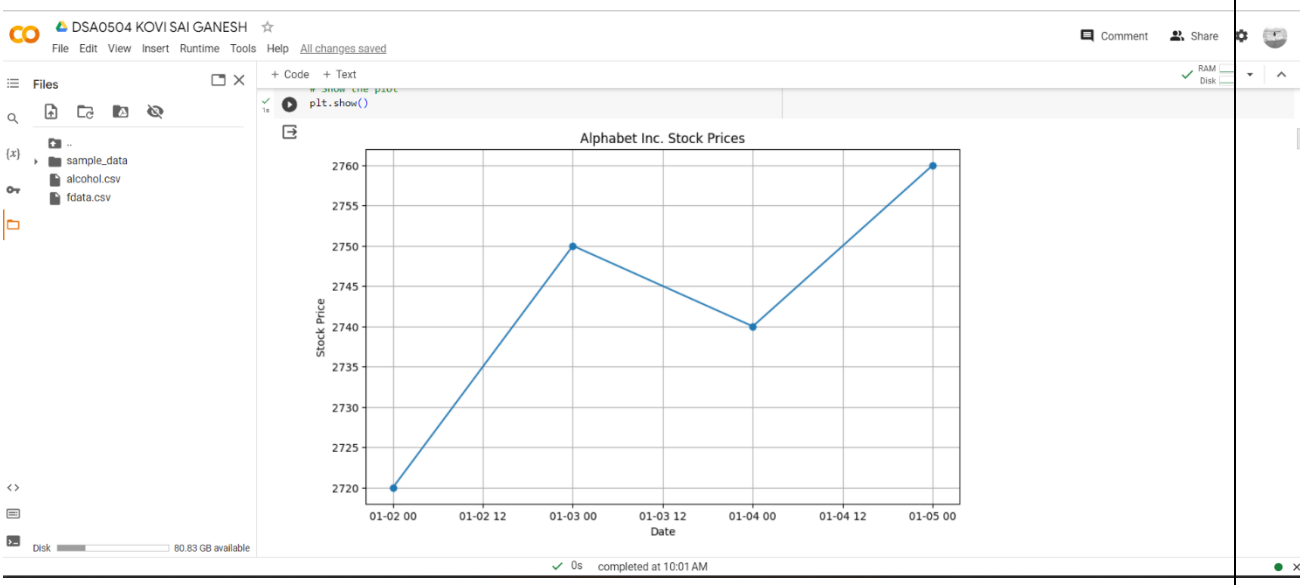
3. Write a Pandas program to display the details of jobs in descending sequence on job title.

+----------+----------+----------+----------+ | JOB_ID | JOB_TITLE | MIN_SALARY | MAX_SALARY | +----------+----------+----------+----------+ | AD_PRES | President | 20080 | 40000 | | AD_VP | Administration Vice President | 15000 | 30000 | | AD_ASST | Administration Assistant | 3000 | 6000 | | FI_MGR | Finance Manager | 8200 | 16000 | | FI_ACCOUNT | Accountant | 4200 | 9000 | | AC_MGR | Accounting Manager | 8200 | 16000 | | AC_ACCOUNT | Public Accountant | 4200 | 9000 | | SA_MAN | Sales Manager | 10000 | 20080 | | SA_REP | Sales Representative | 6000 | 12008 | | PU_MAN | Purchasing Manager | 8000 | 15000 | | PU_CLERK | Purchasing Clerk | 2500 | 5500 | | ST_MAN | Stock Manager | 5500 | 8500 | | ST_CLERK | Stock Clerk | 2008 | 5000 | | SH_CLERK | Shipping Clerk | 2500 | 5500 | | IT_PROG | Programmer | 4000 | 10000 | | MK_MAN | Marketing Manager | 9000 | 15000 | | MK_REP | Marketing Representative | 4000 | 9000 | | HR_REP | Human Resources Representative | 4000 | 9000 | | PR_REP | Public Relations Representative | 4500 | 10500 | +----------+----------+----------+--

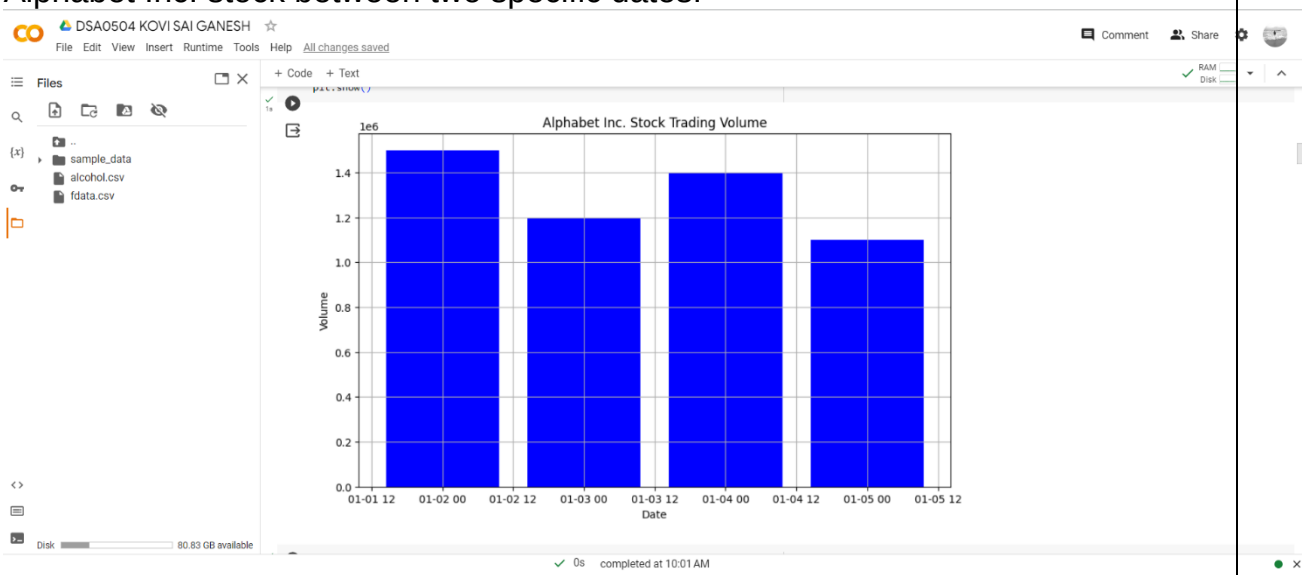


----------+

4. Write a Pandas program to create a line plot of the historical stock prices of Alphabet Inc. between two specific dates.

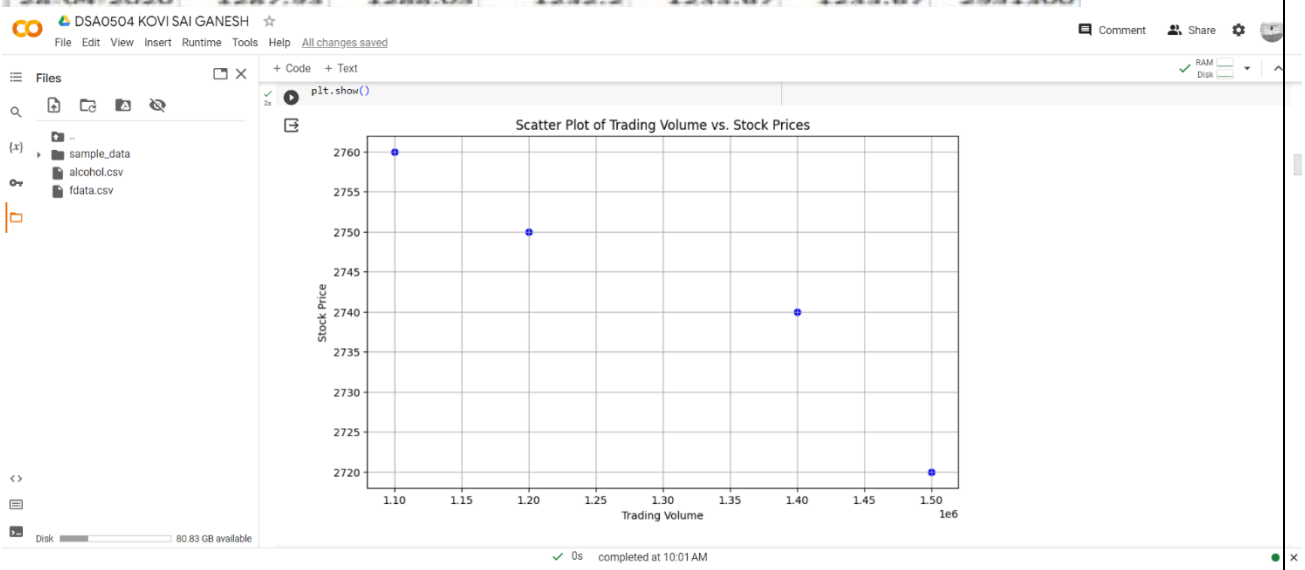


5. Write a Pandas program to create a bar plot of the trading volume of Alphabet Inc. stock between two specific dates.

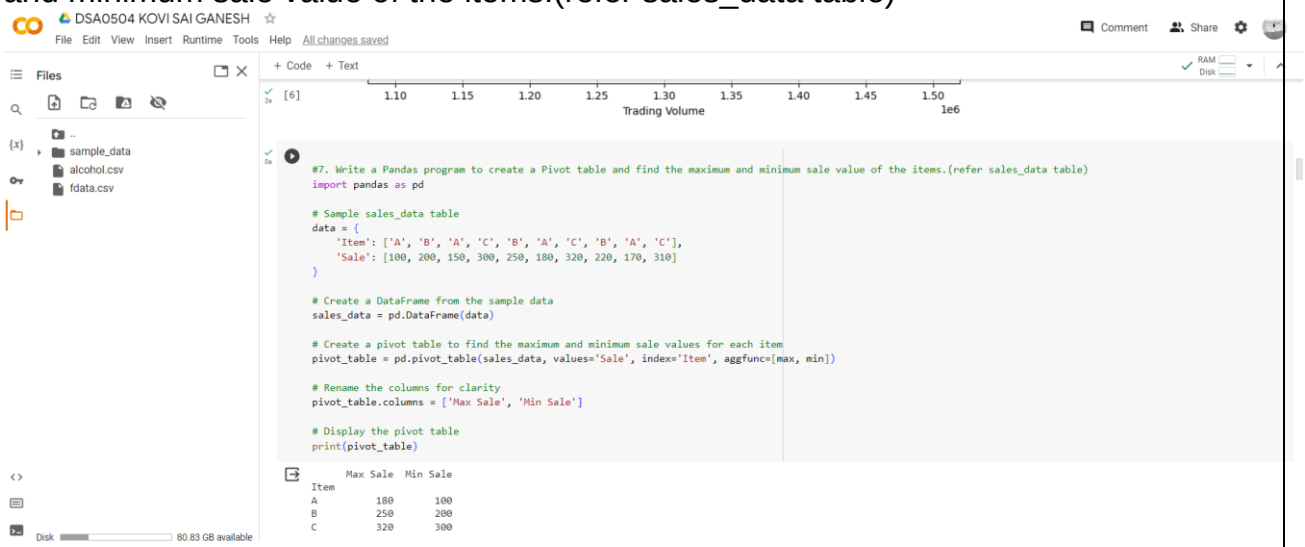


6. Write a Pandas program to create a scatter plot of the trading volume/stock prices of Alphabet Inc. stock between two specific dates.
alphabet_stock_data:

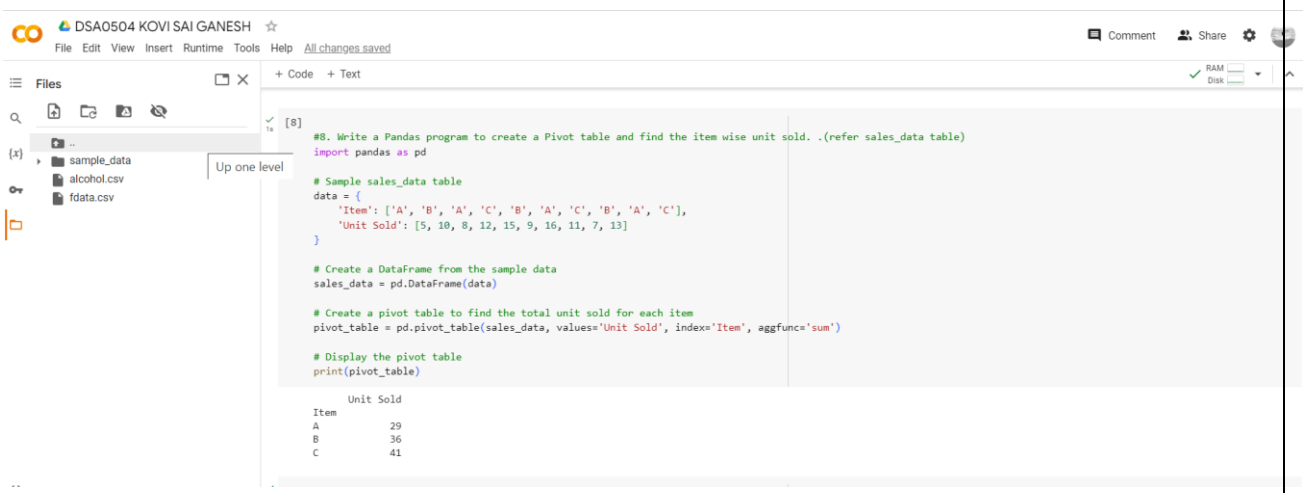
Date	Open	High	Low	Close	Adj Close	Volume
01-04-2020	1122	1129.69	1097.45	1105.62	1105.62	2343100
02-04-2020	1098.26	1126.86	1096.4	1120.84	1120.84	1964900
03-04-2020	1119.015	1123.54	1079.81	1097.88	1097.88	2313400
06-04-2020	1138	1194.66	1130.94	1186.92	1186.92	2664700
07-04-2020	1221	1225	1182.23	1186.51	1186.51	2387300
08-04-2020	1206.5	1219.07	1188.16	1210.28	1210.28	1975100
09-04-2020	1224.08	1225.57	1196.735	1211.45	1211.45	2175400
13-04-2020	1209.18	1220.51	1187.598	1217.56	1217.56	1739800
14-04-2020	1245.09	1282.07	1236.93	1269.23	1269.23	2470400
15-04-2020	1245.61	1280.46	1240.4	1262.47	1262.47	1671700
16-04-2020	1274.1	1279	1242.62	1263.47	1263.47	2518100
17-04-2020	1284.85	1294.43	1271.23	1283.25	1283.25	1949000
20-04-2020	1271	1281.6	1261.37	1266.61	1266.61	1695500
21-04-2020	1247	1254.27	1209.71	1216.34	1216.34	2153000
22-04-2020	1245.54	1285.613	1242	1263.21	1263.21	2093100
23-04-2020	1271.55	1293.31	1265.67	1276.31	1276.31	1566200
24-04-2020	1261.17	1280.4	1249.45	1279.31	1279.31	1640400
27-04-2020	1296	1296.15	1269	1275.88	1275.88	1600600
28-04-2020	1287.93	1288.05	1232.2	1233.67	1233.67	2951300



7. Write a Pandas program to create a Pivot table and find the maximum and minimum sale value of the items.(refer sales_data table)



8. Write a Pandas program to create a Pivot table and find the item wise unit sold. .(refer sales_data table)



9. Write a Pandas program to create a Pivot table and find the total sale amount region wise, manager wise, sales man wise. .(refer sales_data table)

Sales_data:

			Item		

OrderDate Region Manager SalesMan Units Unit_price Sale_amt

East	Martha		95		1,198.00

1-6-18 1,13,810.00

	nn Shelli		Home Theater	50	500.00

1-23-18 25,000.00

	nn Luis		Televisi		1,198.00

2-9-18 43,128.00

	Timothy		Cell Phone	27	225.00

2-26-18 6,075.00

West	Timothy

	Televisi				1,198.00

3-15-18 67,088.00

East	Martha	Alexander	Home Theater	60	500.00

4-1-18 30,000.00

		Steven	Televisi		1,198.00

4-18-18 89,850.00

	nn Luis		Televisi		1,198.00

5-5-18 1,07,820.00

West	Douglas	Michael	Televisi		1,198.00

5-22-18 38,336.00

East	Martha	Alexander	Home Theater	60	500.00

6-8-18 30,000.00

	nn Sigal		Televisi		1,198.00

6-25-18 1,07,820.00

East	Martha	Diana	Home Theater	29	500.00

7-12-18 14,500.00

East	Douglas

Karen	Home Theater	81	500.00

7-29-18 40,500.00

East	Martha		35		1,198.00

8-15-18 41,930.00

	Douglas	John	Desk	2	125.00

9-1-18 250.00

East	Martha	Alexander	Video Games	16	58.50

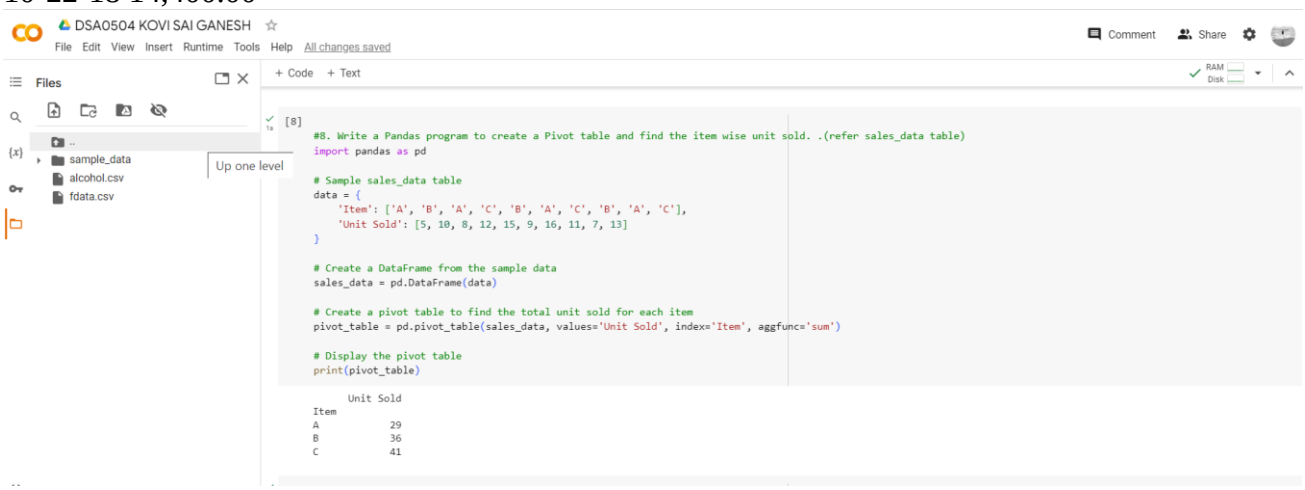
9-18-18 936.00

	nn Sigal		Home Theater	28	500.00

10-5-18 14,000.00

East	Martha	Alexander	Cell Phone	64	225.00

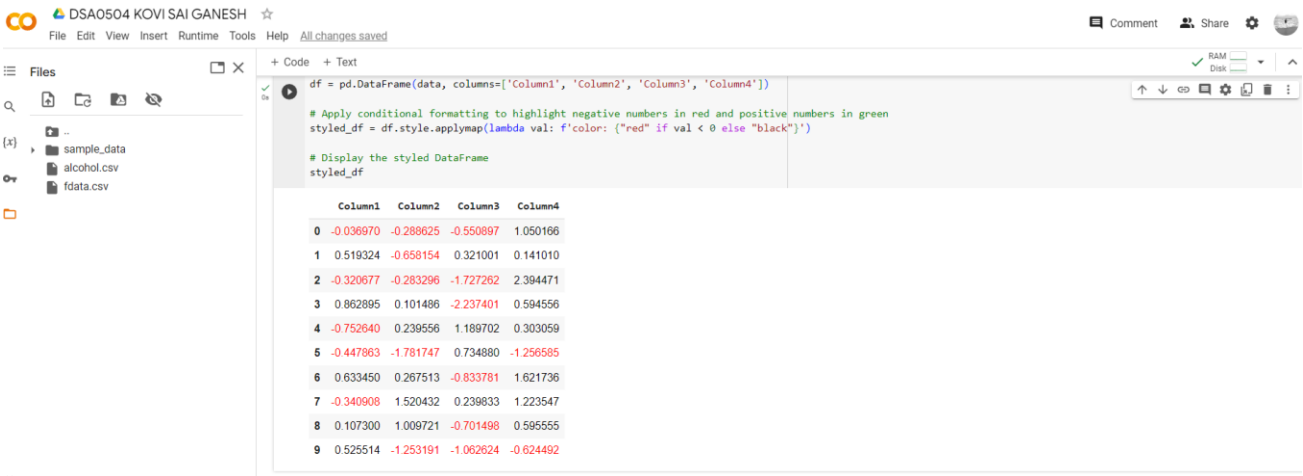
10-22-18 14,400.00



10.Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight the negative numbers red and positive numbers black.

Expected Output:

	A	B	C	D	E
0	1	1.32921	-0.770033	-0.31628	-0.99081
1	2	-1.07082	-1.43871	0.564417	0.295722
2	3	-1.6264	0.219565	0.678805	1.88927
3	4	0.961538	0.104011	-0.481165	0.850229
4	5	1.45342	1.05774	0.165562	0.515018
5	6	-1.33694	0.562861	1.39285	-0.063328
6	7	0.121668	1.2076	-0.00204021	1.6278
7	8	0.354493	1.03753	-0.385684	0.519818
8	9	1.68658	-1.32596	1.42898	-2.08935
9	10	-0.12982	0.631523	-0.586538	0.29072



	Column1	Column2	Column3	Column4
0	-0.036970	-0.288625	-0.550897	1.050166
1	0.519324	-0.658154	0.321001	0.141010
2	-0.320677	-0.283296	-1.727262	2.394471
3	0.862895	0.101486	-2.237401	0.594556
4	-0.752640	0.239556	1.189702	0.303059
5	-0.447863	-1.781747	0.734880	-1.256585
6	0.633450	0.267513	-0.833781	1.621736
7	-0.340908	1.520432	0.239833	1.223547
8	0.107300	1.009721	-0.701498	0.595555
9	0.525514	-1.253191	-1.062624	-0.624492

11.Create a dataframe of ten rows, four columns with random values. Convert some values to nan values. Write a Pandas program which will highlight the nan values.

	A	B	C	D	E
0	1	1.32921	nan	-0.31628	-0.99081
1	2	-1.07082	-1.43871	0.564417	0.295722
2	3	-1.6264	0.219565	0.678805	1.88927
3	4	0.961538	0.104011	nan	0.850229
4	5	nan	1.05774	0.165562	0.515018
5	6	-1.33694	0.562861	1.39285	-0.063328
6	7	0.121668	1.2076	-0.00204021	1.6278
7	8	0.354493	1.03753	-0.385684	0.519818
8	9	1.68658	-1.32596	1.42898	-2.08935
9	10	-0.12982	0.631523	-0.586538	nan

```
Original array:
     A         B         C         D         E
0  1.0  1.329212       NaN -0.316280 -0.990810
1  2.0 -1.070816 -1.438713  0.564417  0.295722
2  3.0 -1.626404  0.219565  0.678805  1.889273
3  4.0  0.961538  0.104011       NaN  0.850229
4  5.0       NaN  1.057737  0.165562  0.515018
5  6.0 -1.336936  0.562861  1.392855 -0.063328
6  7.0  0.121668  1.207603 -0.002040  1.627796
7  8.0  0.354493  1.037528 -0.385684  0.519818
8  9.0  1.686583 -1.325963  1.428984 -2.089354
9 10.0 -0.129820  0.631523 -0.586538       NaN

Negative numbers red and positive numbers black:
<ipython-input-12-0152cf84b42a>:21: FutureWarning: `null_color` is deprecated: use `color` instead
  df.style.highlight_null(null_color='red')
```

	A	B	C	D	E
0	1.000000	1.329212	nan	-0.316280	-0.990810
1	2.000000	-1.070816	-1.438713	0.564417	0.295722
2	3.000000	-1.626404	0.219565	0.678805	1.889273
3	4.000000	0.961538	0.104011	nan	0.850229
4	5.000000	nan	1.057737	0.165562	0.515018
5	6.000000	-1.336936	0.562861	1.392855	-0.063328
6	7.000000	0.121668	1.207603	-0.002040	1.627796
7	8.000000	0.354493	1.037528	-0.385684	0.519818
8	9.000000	1.686583	-1.325963	1.428984	-2.089354
9	10.000000	-0.129820	0.631523	-0.586538	nan

12.Create a dataframe of ten rows, four columns with random values. Write a Pandas program to set dataframe background Color black and font color yellow.

	A	B	C	D	E
0	1	1.32921	nan	-0.31628	-0.99081
1	2	-1.07082	-1.43871	0.564417	0.295722
2	3	-1.6264	0.219565	0.678805	1.88927
3	4	0.961538	0.104011	nan	0.850229
4	5	nan	1.05774	0.165562	0.515018
5	6	-1.33694	0.562861	1.39285	-0.063328
6	7	0.121668	1.2076	-0.00204021	1.6278
7	8	0.354493	1.03753	-0.385684	0.519818
8	9	1.68658	-1.32596	1.42898	-2.08935
9	10	-0.12982	0.631523	-0.586538	nan

```
Original array:
      A         B         C         D         E
0   1.0  1.329212       NaN -0.316280 -0.990810
1   2.0 -1.070816 -1.438713  0.564417  0.295722
2   3.0 -1.626404  0.219565  0.678805  1.889273
3   4.0  0.961538  0.104011       NaN  0.850229
4   5.0       NaN  1.057737  0.165562  0.515018
5   6.0 -1.336936  0.562861  1.392855 -0.063328
6   7.0  0.121668  1.207603 -0.002040  1.627796
7   8.0  0.354493  1.037528 -0.385684  0.519818
8   9.0  1.686583 -1.325963  1.428984 -2.089354
9  10.0 -0.129820  0.631523 -0.586538       NaN

Background:black - fontcolor:yelow
```

	A	B	C	D	E
0	1.000000	1.329212	nan	-0.316280	-0.990810
1	2.000000	-1.070816	-1.438713	0.564417	0.295722
2	3.000000	-1.626404	0.219565	0.678805	1.889273
3	4.000000	0.961538	0.104011	nan	0.850229
4	5.000000	nan	1.057737	0.165562	0.515018
5	6.000000	-1.336936	0.562861	1.392855	-0.063328
6	7.000000	0.121668	1.207603	-0.002040	1.627796
7	8.000000	0.354493	1.037528	-0.385684	0.519818
8	9.000000	1.686583	-1.325963	1.428984	-2.089354
9	10.000000	-0.129820	0.631523	-0.586538	nan

13.Write a Pandas program to detect missing values of a given DataFrame. Display True or False.

	ord_no	purch_amt	ord_date	customer_id	salesman_id
0	70001.0	150.50	2012-10-05	3002	5002.0
1	NaN	270.65	2012-09-10	3001	5003.0
2	70002.0	65.26	NaN	3001	5001.0
3	70004.0	110.50	2012-08-17	3003	NaN
4	NaN	948.50	2012-09-10	3002	5002.0
5	70005.0	2400.60	2012-07-27	3001	5001.0
6	NaN	5760.00	2012-09-10	3001	5001.0
7	70010.0	1983.43	2012-10-10	3004	NaN
8	70003.0	2480.40	2012-10-10	3003	5003.0
9	70012.0	250.45	2012-06-27	3002	5002.0
10	NaN	75.29	2012-08-17	3001	5003.0
11	70013.0	3045.60	2012-04-25	3001	NaN

```
Original Orders DataFrame:
     ord_no  purch_amt    ord_date  customer_id  salesman_id
0   70001.0     150.50  2012-10-05         3002       5002.0
1       NaN     270.65  2012-09-10         3001       5003.0
2   70002.0      65.26         NaN         3001       5001.0
3   70004.0     110.50  2012-08-17         3003          NaN
4       NaN     948.50  2012-09-10         3002       5002.0
5   70005.0    2400.60  2012-07-27         3001       5001.0
6       NaN    5760.00  2012-09-10         3001       5001.0
7   70010.0    1983.43  2012-10-10         3004          NaN
8   70003.0    2480.40  2012-10-10         3003       5003.0
9   70012.0     250.45  2012-06-27         3002       5002.0
10      NaN      75.29  2012-08-17         3001       5003.0
11  70013.0    3045.60  2012-04-25         3001          NaN

Missing values of the said dataframe:
    ord_no  purch_amt  ord_date  customer_id  salesman_id
0    False      False     False        False        False
1     True      False     False        False        False
2    False      False      True        False        False
3    False      False     False        False         True
4     True      False     False        False        False
5    False      False     False        False        False
6     True      False     False        False        False
7    False      False     False        False         True
8    False      False     False        False        False
9    False      False     False        False        False
10    True      False     False        False        False
11   False      False     False        False         True
```

14. Write a Pandas program to find and replace the missing values in a given DataFrame which do not have any valuable information.

	ord_no	purch_amt	ord_date	customer_id	salesman_id
0	70001	150.5	?	3002	5002
1	NaN	270.65	2012-09-10	3001	5003
2	70002	65.26	NaN	3001	?
3	70004	110.5	2012-08-17	3003	5001
4	NaN	948.5	2012-09-10	3002	NaN
5	70005	2400.6	2012-07-27	3001	5002
6	--	5760	2012-09-10	3001	5001
7	70010	?	2012-10-10	3004	?
8	70003	12.43	2012-10-10	--	5003
9	70012	2480.4	2012-06-27	3002	5002
10	NaN	250.45	2012-08-17	3001	5003
11	70013	3045.6	2012-04-25	3001	--

```
Original Orders DataFrame:
   ord_no purch_amt    ord_date customer_id salesman_id
0   70001     150.5           ?        3002        5002
1     NaN    270.65  2012-09-10        3001        5003
2   70002     65.26         NaN        3001           ?
3   70004     110.5  2012-08-17        3003        5001
4     NaN     948.5  2012-09-10        3002         NaN
5   70005    2400.6  2012-07-27        3001        5002
6      --      5760  2012-09-10        3001        5001
7   70010         ?  2012-10-10        3004           ?
8   70003     12.43  2012-10-10          --        5003
9   70012    2480.4  2012-06-27        3002        5002
10    NaN    250.45  2012-08-17        3001        5003
11  70013    3045.6  2012-04-25        3001          --

Replace the missing values with NaN:
     ord_no  purch_amt    ord_date  customer_id  salesman_id
0   70001.0     150.50         NaN       3002.0       5002.0
1       NaN     270.65  2012-09-10       3001.0       5003.0
2   70002.0      65.26         NaN       3001.0          NaN
3   70004.0     110.50  2012-08-17       3003.0       5001.0
4       NaN     948.50  2012-09-10       3002.0          NaN
5   70005.0    2400.60  2012-07-27       3001.0       5002.0
6       NaN    5760.00  2012-09-10       3001.0       5001.0
7   70010.0        NaN  2012-10-10       3004.0          NaN
8   70003.0      12.43  2012-10-10          NaN       5003.0
9   70012.0    2480.40  2012-06-27       3002.0       5002.0
10      NaN     250.45  2012-08-17       3001.0       5003.0
11  70013.0    3045.60  2012-04-25       3001.0          NaN
```

15.Write a Pandas program to keep the rows with at least 2 NaN values in a given DataFrame.

	ord_no	purch_amt	ord_date	customer_id
0	NaN	NaN	NaN	NaN
1	NaN	270.65	2012-09-10	3001.0
2	70002.0	65.26	NaN	3001.0
3	NaN	NaN	NaN	NaN
4	NaN	948.50	2012-09-10	3002.0
5	70005.0	2400.60	2012-07-27	3001.0
6	NaN	5760.00	2012-09-10	3001.0
7	70010.0	1983.43	2012-10-10	3004.0
8	70003.0	2480.40	2012-10-10	3003.0
9	70012.0	250.45	2012-06-27	3002.0
10	NaN	75.29	2012-08-17	3001.0
11	NaN	NaN	NaN	NaN

```
Original Orders DataFrame:
     ord_no  purch_amt    ord_date  customer_id
0       NaN        NaN         NaN          NaN
1       NaN     270.65  2012-09-10       3001.0
2   70002.0      65.26         NaN       3001.0
3       NaN        NaN         NaN          NaN
4       NaN     948.50  2012-09-10       3002.0
5   70005.0    2400.60  2012-07-27       3001.0
6       NaN    5760.00  2012-09-10       3001.0
7   70010.0    1983.43  2012-10-10       3004.0
8   70003.0    2480.40  2012-10-10       3003.0
9   70012.0     250.45  2012-06-27       3002.0
10      NaN      75.29  2012-08-17       3001.0
11      NaN        NaN         NaN          NaN

Keep the rows with at least 2 NaN values of the said DataFrame:
     ord_no  purch_amt    ord_date  customer_id
1       NaN     270.65  2012-09-10       3001.0
2   70002.0      65.26         NaN       3001.0
4       NaN     948.50  2012-09-10       3002.0
5   70005.0    2400.60  2012-07-27       3001.0
6       NaN    5760.00  2012-09-10       3001.0
7   70010.0    1983.43  2012-10-10       3004.0
8   70003.0    2480.40  2012-10-10       3003.0
9   70012.0     250.45  2012-06-27       3002.0
10      NaN      75.29  2012-08-17       3001.0
```

16.Write a Pandas program to split the following dataframe into groups based on school code. Also check the type of GroupBy object.

	school	class	name	date_Of_Birth	age	height	weight	address
S1	s001	V	Alberto Franco	15/05/2002	12	173	35	street1
S2	s002	V	Gino Mcneill	17/05/2002	12	192	32	street2
S3	s003	VI	Ryan Parkes	16/02/1999	13	186	33	street3
S4	s001	VI	Eesha Hinton	25/09/1998	13	167	30	street1
S5	s002	V	Gino Mcneill	11/05/2002	14	151	31	street2
S6	s004	VI	David Parkes	15/09/1997	12	159	32	street4

```
   school_code class           name date_of_birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35
S2        s002     V    Gino Mcneill    17/05/2002   12     192      32
S3        s003    VI     Ryan Parkes    16/02/1999   13     186      33
S4        s001    VI    Eesha Hinton    25/09/1998   13     167      30
S5        s002     V    Gino Mcneill    11/05/2002   14     151      31
S6        s004    VI    David Parkes    15/09/1997   12     159      32

    address
S1  street1
S2  street2
S3  street3
S4  street1
S5  street2
S6  street4

Split the data on school_code wise:

Group:
s001
   school_code class           name date_of_birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35
S4        s001    VI    Eesha Hinton    25/09/1998   13     167      30

    address
S1  street1
S4  street1

Type of the object:
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>

Group:
s002
   school_code class          name date_of_birth  age  height  weight  address
S2        s002     V  Gino Mcneill    17/05/2002   12     192      32  street2
S5        s002     V  Gino Mcneill    11/05/2002   14     151      31  street2

Type of the object:
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>
```

✓ 0s completed at 10:01 AM

17.Write a Pandas program to split the following dataframe by school code and get mean, min, and max value of age for each school.

	school	class	name	date_Of_Birth	age	height	weight	address
S1	s001	V	Alberto Franco	15/05/2002	12	173	35	street1
S2	s002	V	Gino Mcneill	17/05/2002	12	192	32	street2
S3	s003	VI	Ryan Parkes	16/02/1999	13	186	33	street3
S4	s001	VI	Eesha Hinton	25/09/1998	13	167	30	street1

```
Original DataFrame:
   school_code class           name date_of_birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35
S2        s002     V    Gino Mcneill    17/05/2002   12     192      32
S3        s003    VI     Ryan Parkes    16/02/1999   13     186      33
S4        s001    VI    Eesha Hinton    25/09/1998   13     167      30
S5        s002     V    Gino Mcneill    11/05/2002   14     151      31
S6        s004    VI    David Parkes    15/09/1997   12     159      32

    address
S1  street1
S2  street2
S3  street3
S4  street1
S5  street2
S6  street4

Mean, min, and max value of age for each school with customized column names:
             Age_Mean  Age_Max  Age_Min
school_code
s001             12.5       13       12
s002             13.0       14       12
s003             13.0       13       13
s004             12.0       12       12
```

18.Write a Pandas program to split the following given dataframe into groups based on school code and class.

	school	class	name	date_Of_Birth	age	height	weight	address
S1	s001	V	Alberto Franco	15/05/2002	12	173	35	street1
S2	s002	V	Gino Mcneill	17/05/2002	12	192	32	street2
S3	s003	VI	Ryan Parkes	16/02/1999	13	186	33	street3
S4	s001	VI	Eesha Hinton	25/09/1998	13	167	30	street1
S5	s002	V	Gino Mcneill	11/05/2002	14	151	31	street2
S6	s004	VI	David Parkes	15/09/1997	12	159	32	street4

```
Original DataFrame:
   school_code class           name date_of_birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35
S2        s002     V    Gino Mcneill    17/05/2002   12     192      32
S3        s003    VI     Ryan Parkes    16/02/1999   13     186      33
S4        s001    VI    Eesha Hinton    25/09/1998   13     167      30
S5        s002     V    Gino Mcneill    11/05/2002   14     151      31
S6        s004    VI    David Parkes    15/09/1997   12     159      32

    address
S1  street1
S2  street2
S3  street3
S4  street1
S5  street2
S6  street4

Split the said data on school_code wise:

Group:
s001
   school_code class           name date_of_birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35
S4        s001    VI    Eesha Hinton    25/09/1998   13     167      30

    address
S1  street1
S4  street1

Type of the object:
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>

Group:
s002
   school_code class          name date_of_birth  age  height  weight  address
S2        s002     V  Gino Mcneill    17/05/2002   12     192      32  street2
S5        s002     V  Gino Mcneill    11/05/2002   14     151      31  street2

Type of the object:
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>

Group:
s003
   school_code class         name date_of_birth  age  height  weight  address
S3        s003    VI  Ryan Parkes    16/02/1999   13     186      33  street3

Type of the object:
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>

Group:
s004
   school_code class          name date_of_birth  age  height  weight  address
S6        s004    VI  David Parkes    15/09/1997   12     159      32  street4

Type of the object:
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>
```

✓ 0s completed at 10:01 AM

19.Write a Pandas program to display the dimensions or shape of the World alcohol consumption dataset. Also extract the column names from the dataset.

	Year	WHO region	Country	Beverage Types	Display Value
0	1986	Western Pacific	Viet Nam	Wine	0.00
1	1986	Americas	Uruguay	Other	0.50
2	1985	Africa	Cte d'Ivoire	Wine	1.62
3	1986	Americas	Colombia	Beer	4.27
4	1987	Americas	Saint Kitts and Nevis	Beer	1.98

```
World alcohol consumption sample data:
    Year        H0 region               Country Beverage Types  Display Value
0   1986  Western Pacific              Viet Nam           Wine           0.00
1   1986         Americas               Uruguay          Other           0.50
2   1985           Africa          Cte d'Ivoire           Wine           1.62
3   1986         Americas              Colombia           Beer           4.27
4   1987         Americas  Saint Kitts and Nevi           Beer           1.98

Shape of the dataframe:  (5, 5)

Number of rows:  5

Number of column:  5

Extract Column Names:
Index([' Year', 'H0 region', 'Country', 'Beverage Types', 'Display Value'], dtype='object')
```

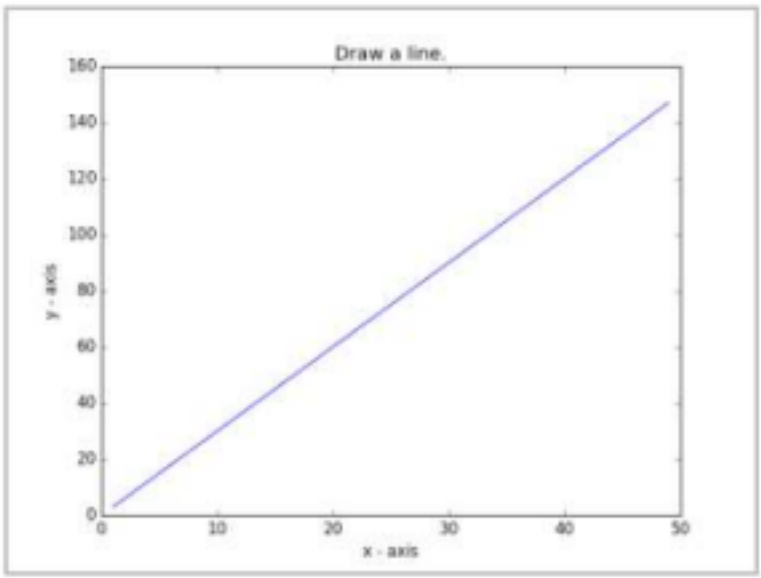
20.Write a Pandas program to find the index of a given substring of a DataFrame column.

```
Original DataFrame:
  name_code date_of_birth    age
0     c0001     12/05/2002  18.5
1     1000c     16/02/1999  21.2
2     b00c2     25/09/1998  22.5
3     b2c02     12/02/2022  22.0
4     c2222     15/09/1997  23.0

Index of a substring in a specified column of a dataframe:
  name_code date_of_birth    age  Index
0     c0001     12/05/2002  18.5      0
1     1000c     16/02/1999  21.2      4
2     b00c2     25/09/1998  22.5      3
3     b2c02     12/02/2022  22.0      2
4     c2222     15/09/1997  23.0      0
```

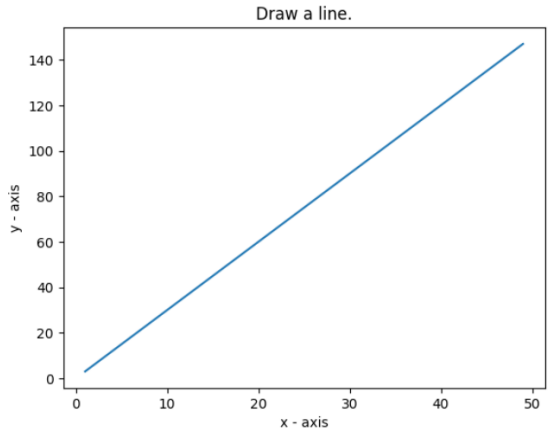
21.Write a Pandas program to swap the cases of a specified character column in a given DataFrame.

```
Original DataFrame:
  company_code date_of_sale  sale_amount
0         Abcd   12/05/2002      12348.5
1         EFGF   16/02/1999     233331.2
2      zefsalf   25/09/1998         22.5
3      sdfslew   12/02/2022    2566552.0
4      zekfsdf   15/09/1997         23.0

Swap cases in company_code:
  company_code date_of_sale  sale_amount swapped_company_code
0         Abcd   12/05/2002      12348.5                 aBCD
1         EFGF   16/02/1999     233331.2                 efgf
2      zefsalf   25/09/1998         22.5              ZEFSALF
3      sdfslew   12/02/2022    2566552.0              SDFSLEW
4      zekfsdf   15/09/1997         23.0              ZEKFSDF
```

22.Write a Python program to draw a line with suitable label in the x axis, y axis and a title.



```
Values of X:
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
Values of Y (thrice of X):
[3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96, 99, 102, 105, 108, 111, 114, 117, 120, 12
```



23.Write a Python program to draw a line using given axis values taken from a text file, with suitable label in the x axis, y axis and a title.

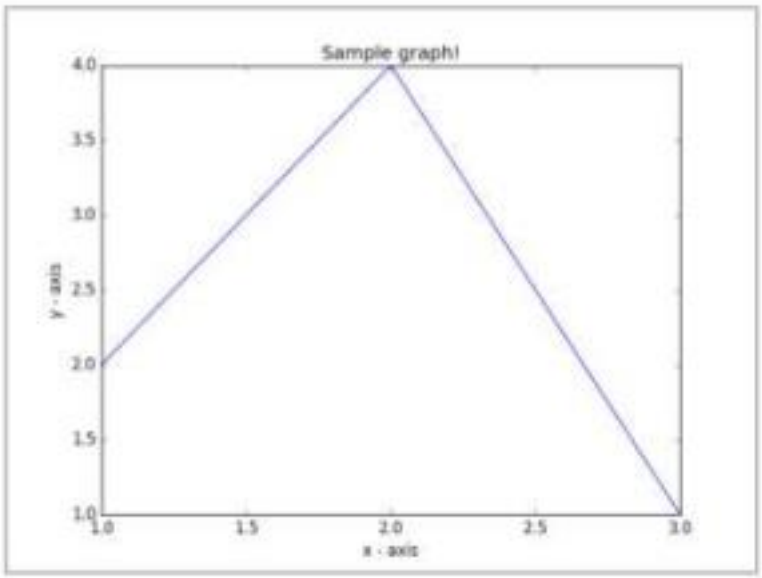
Test Data:

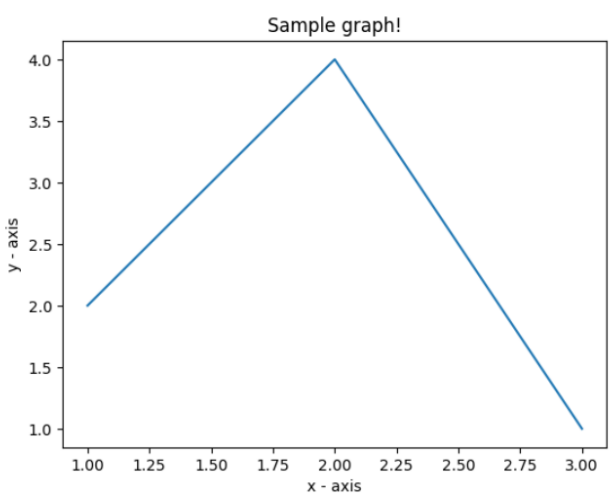
test.txt

1 2

2 4

3 1





24.Write a Python program to draw line charts of the financial data of Alphabet Inc. between October 3, 2016 to October 7, 2016.

Sample Financial data (fdata.csv):

Date,Open,High,Low,Close

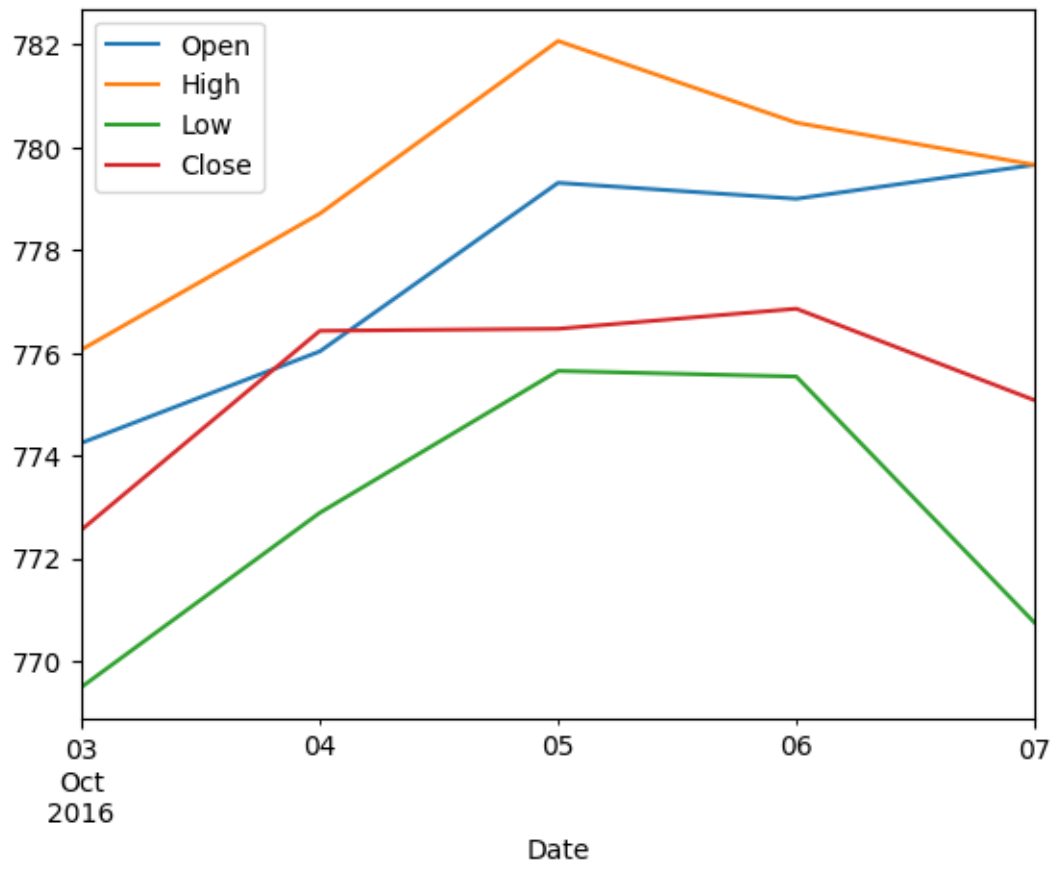
10-03-16,774.25,776.065002,769.5,772.559998

10-04-16,776.030029,778.710022,772.890015,776.429993

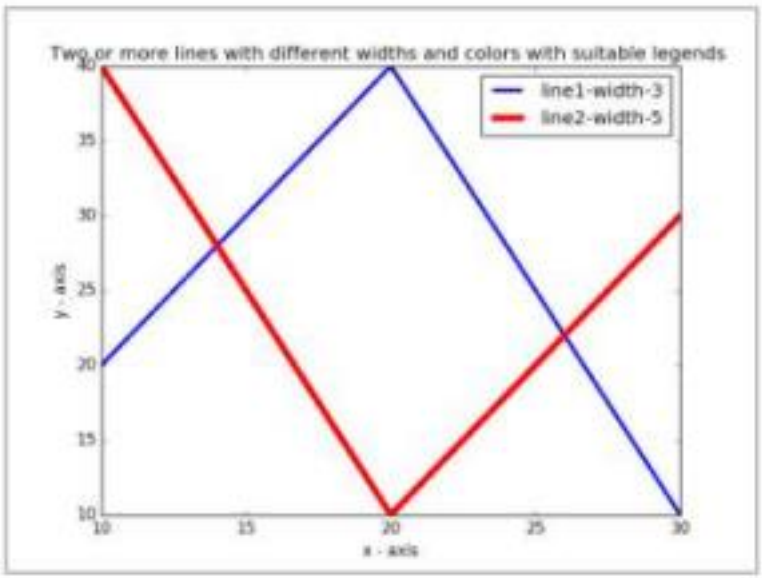
10-05-16,779.309998,782.070007,775.650024,776.469971

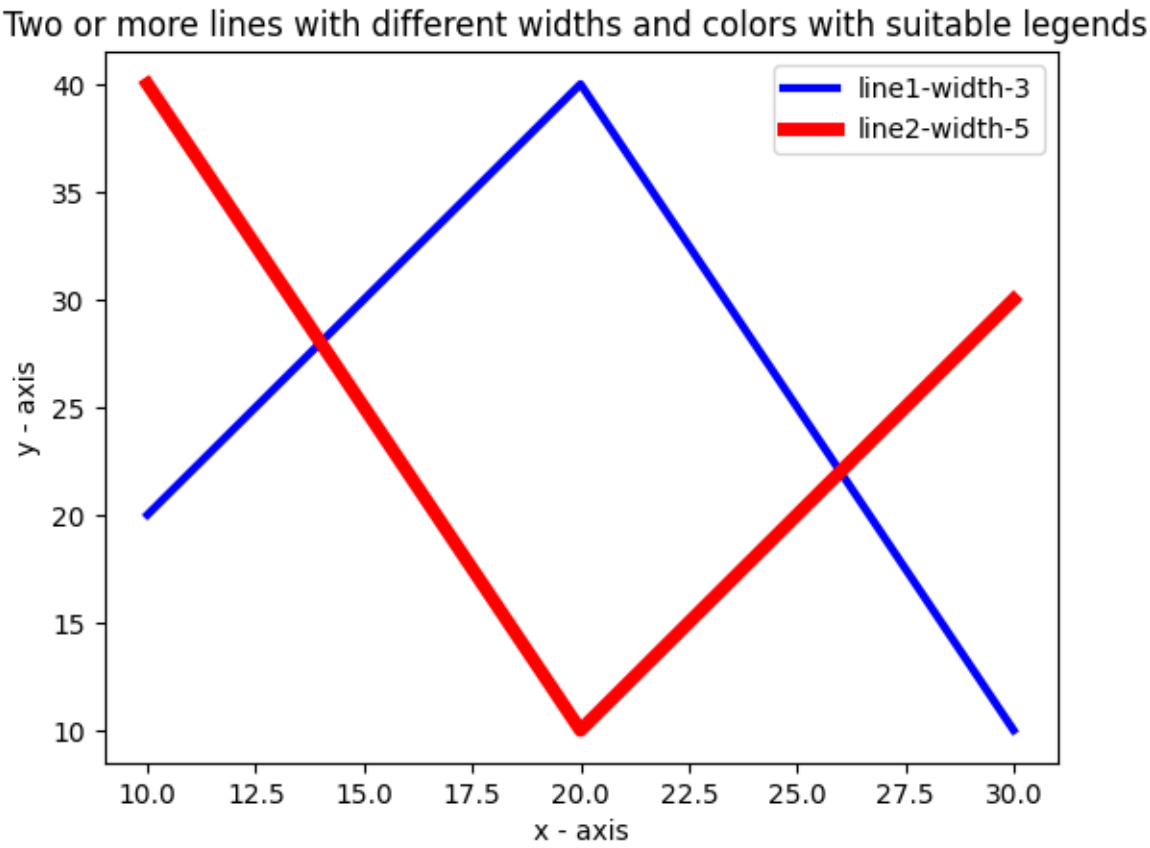
10-06-16,779,780.47998,775.539978,776.859985

10-07-16,779.659973,779.659973,770.75,775.080017

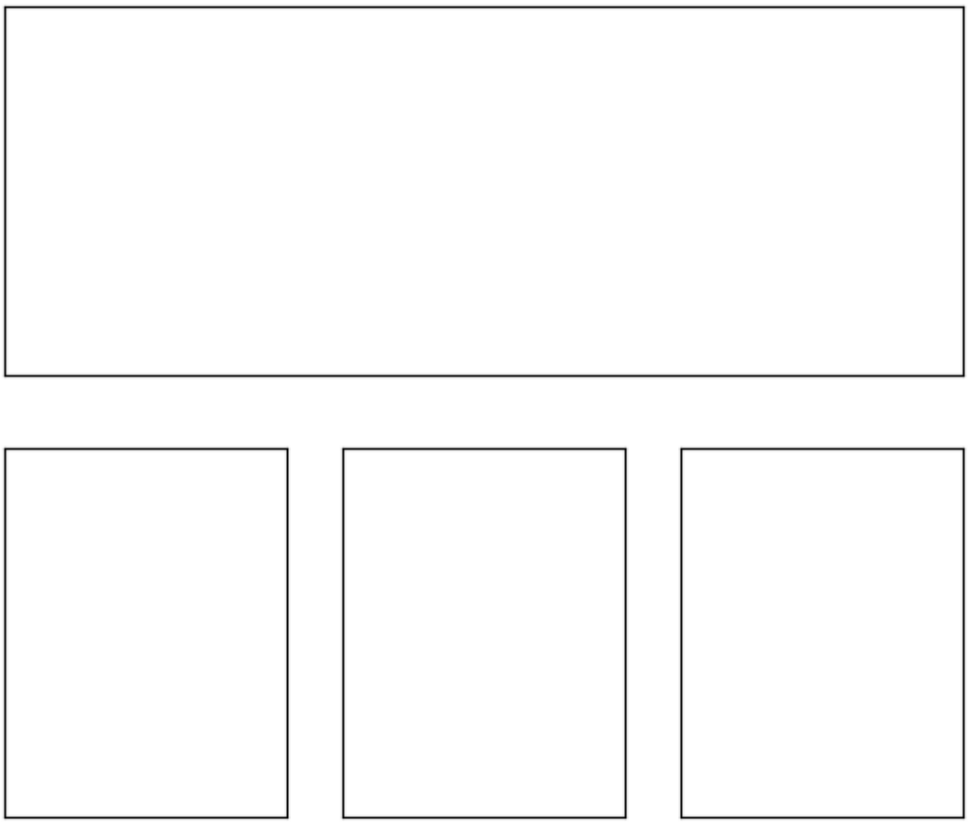


25.Write a Python program to plot two or more lines with legends, different widths and colors.



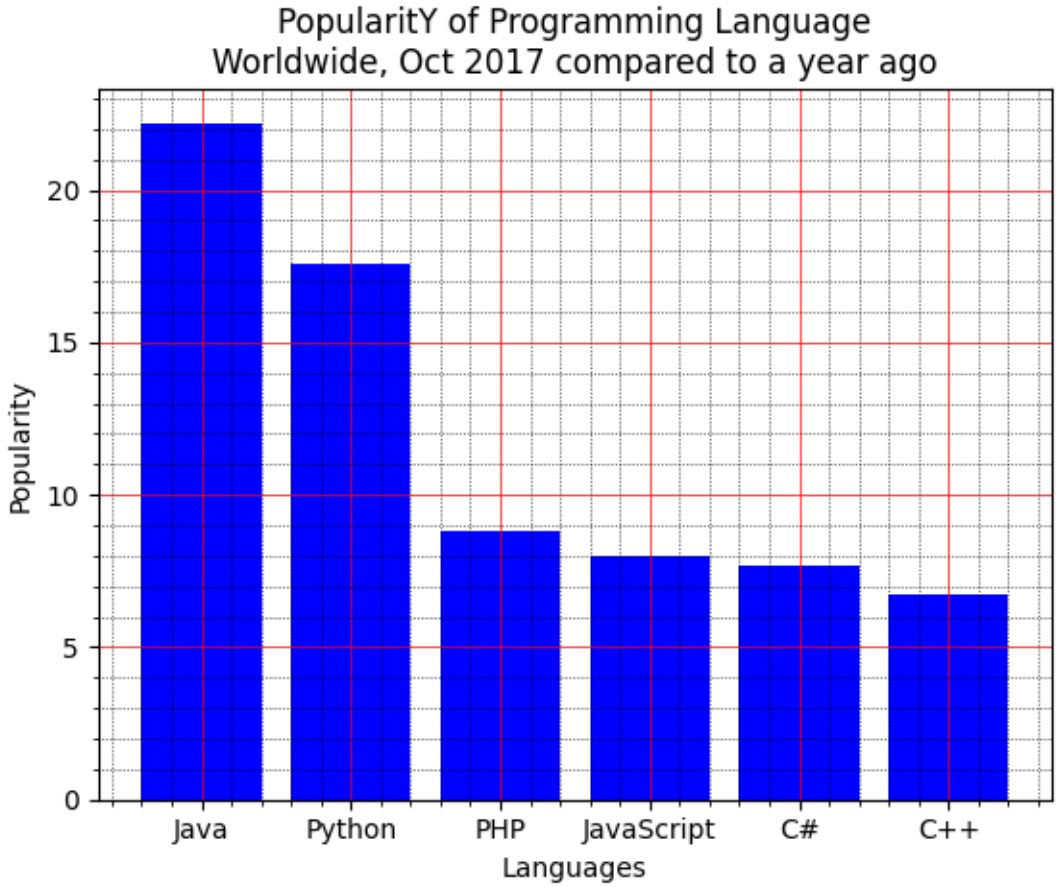


26.Write a Python program to create multiple plots.

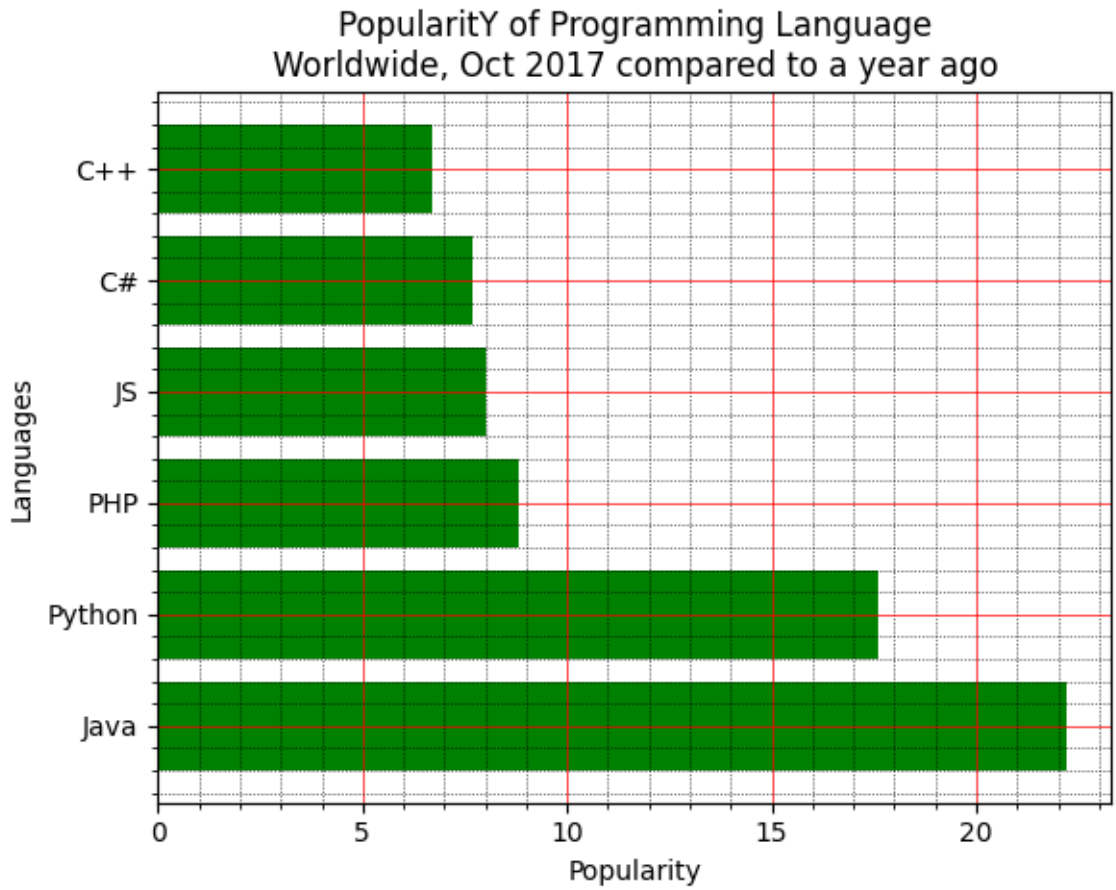


27.Write a Python programming to display a bar chart of the popularity of

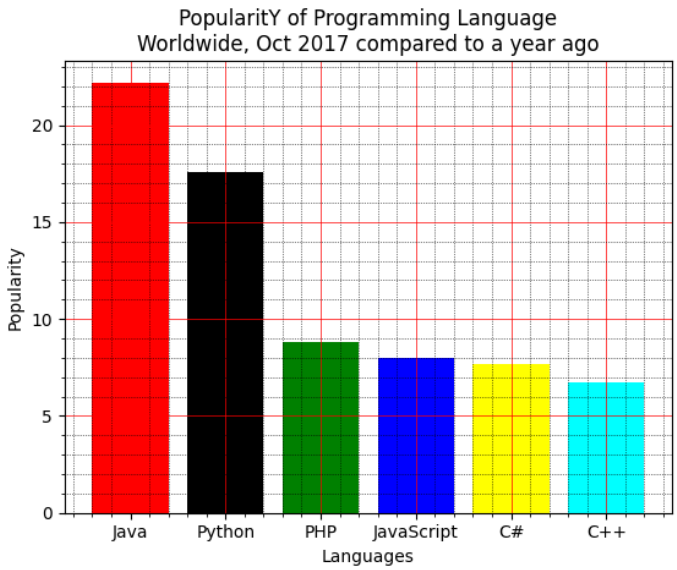
programming Languages.
Sample data:
Programming languages: Java, Python, PHP, JavaScript, C#, C++
Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7



28.Write a Python programming to display a horizontal bar chart of the popularity of programming Languages.
Sample data:
Programming languages: Java, Python, PHP, JavaScript, C#, C++
Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7



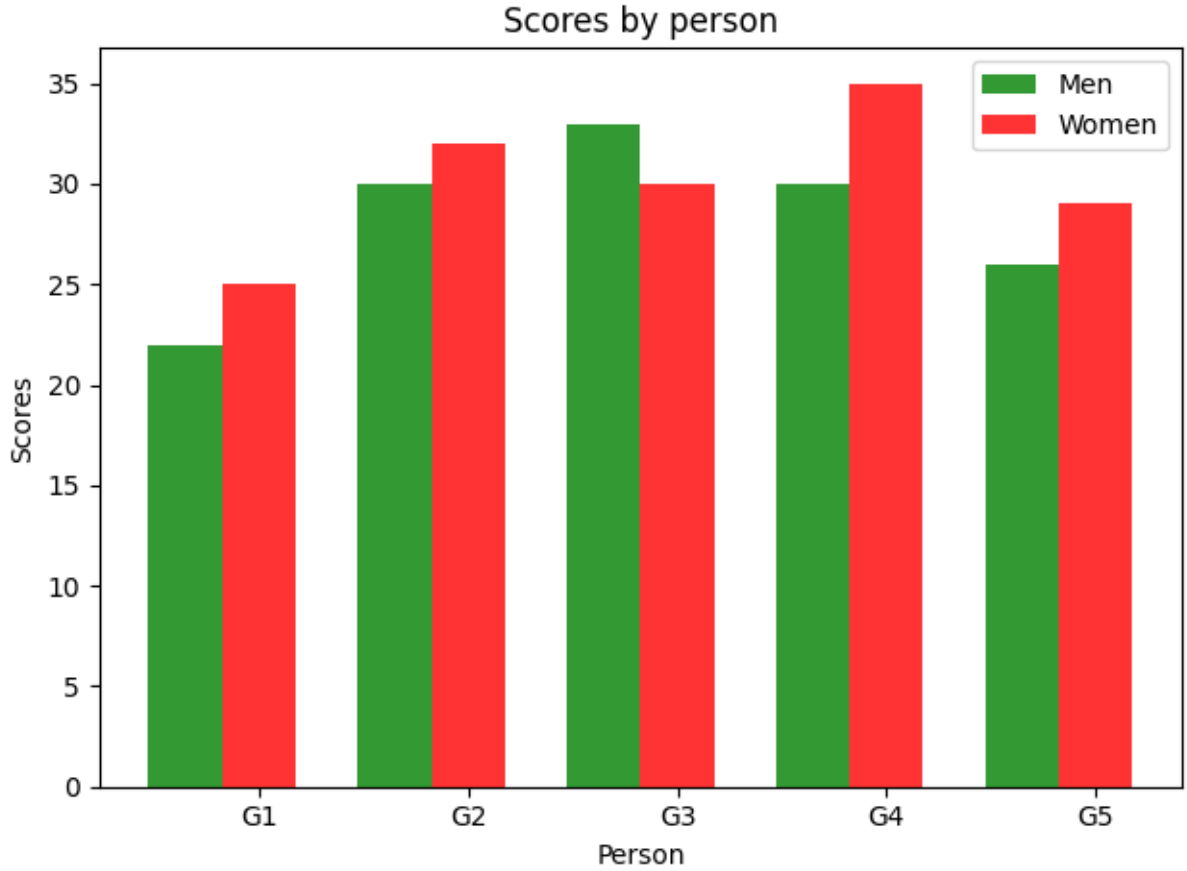
29.Write a Python programming to display a bar chart of the popularity of programming Languages. Use different color for each bar. Sample data:
Programming languages: Java, Python, PHP, JavaScript, C#, C++
Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7



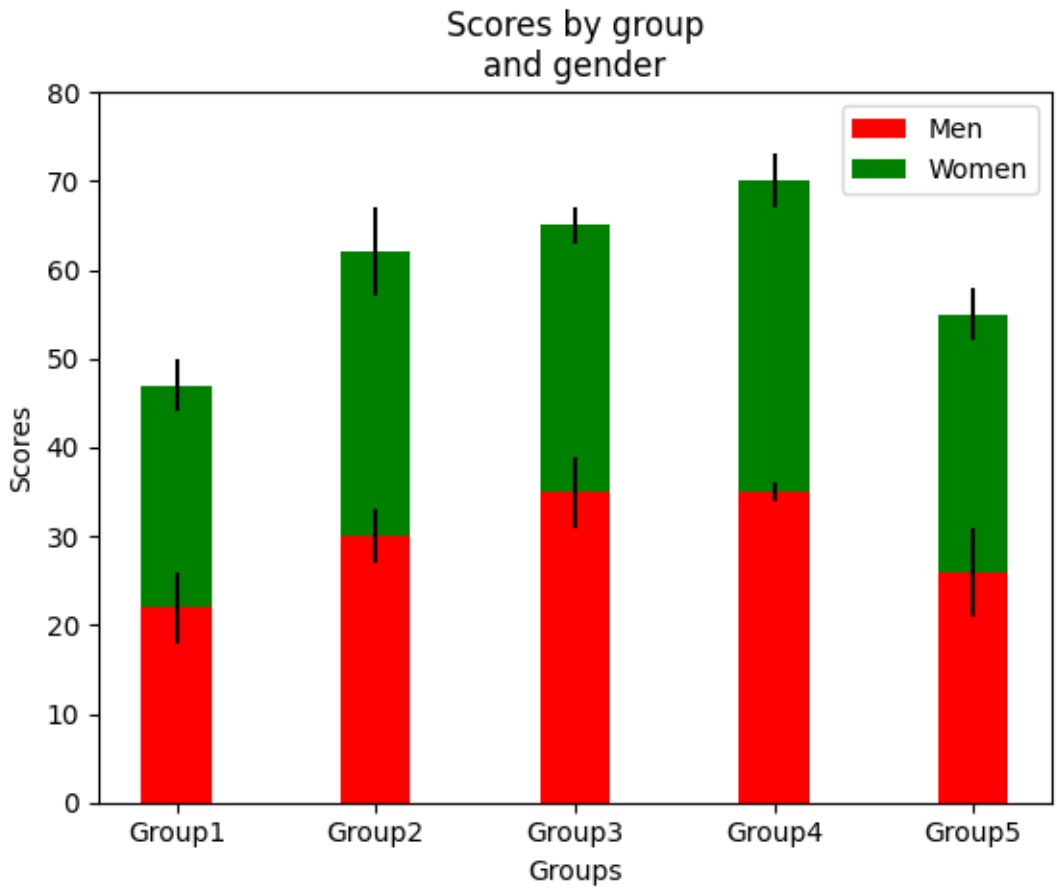
30.Write a Python program to create bar plot of scores by group and gender.

Use multiple X values on the same chart for men and women.

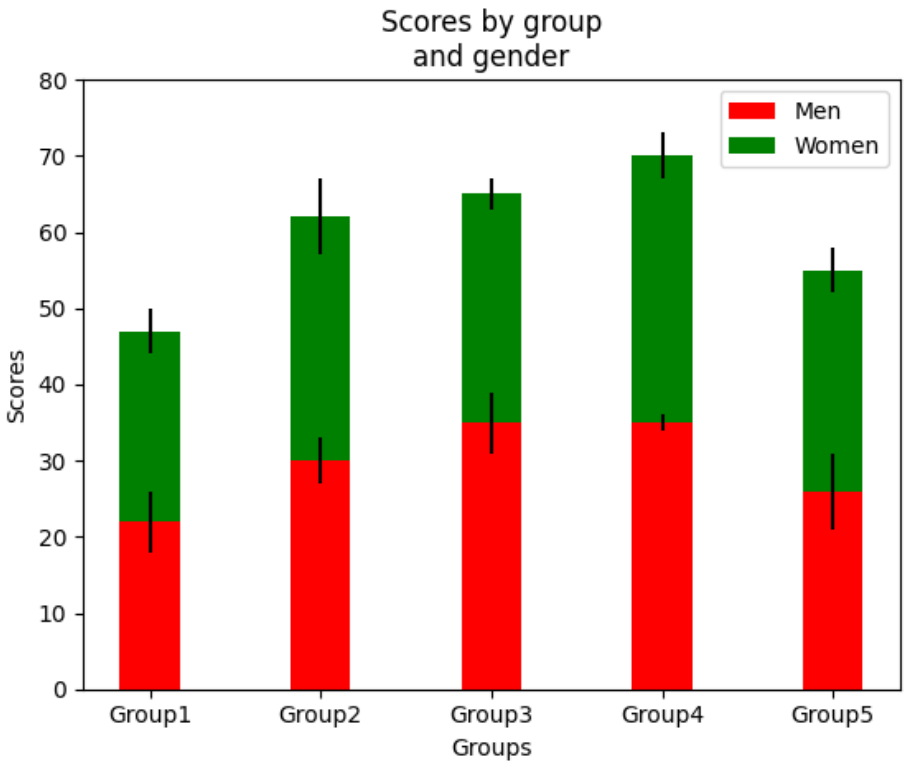
Sample Data:
Means (men) = (22, 30, 35, 35, 26)
Means (women) = (25, 32, 30, 35, 29)



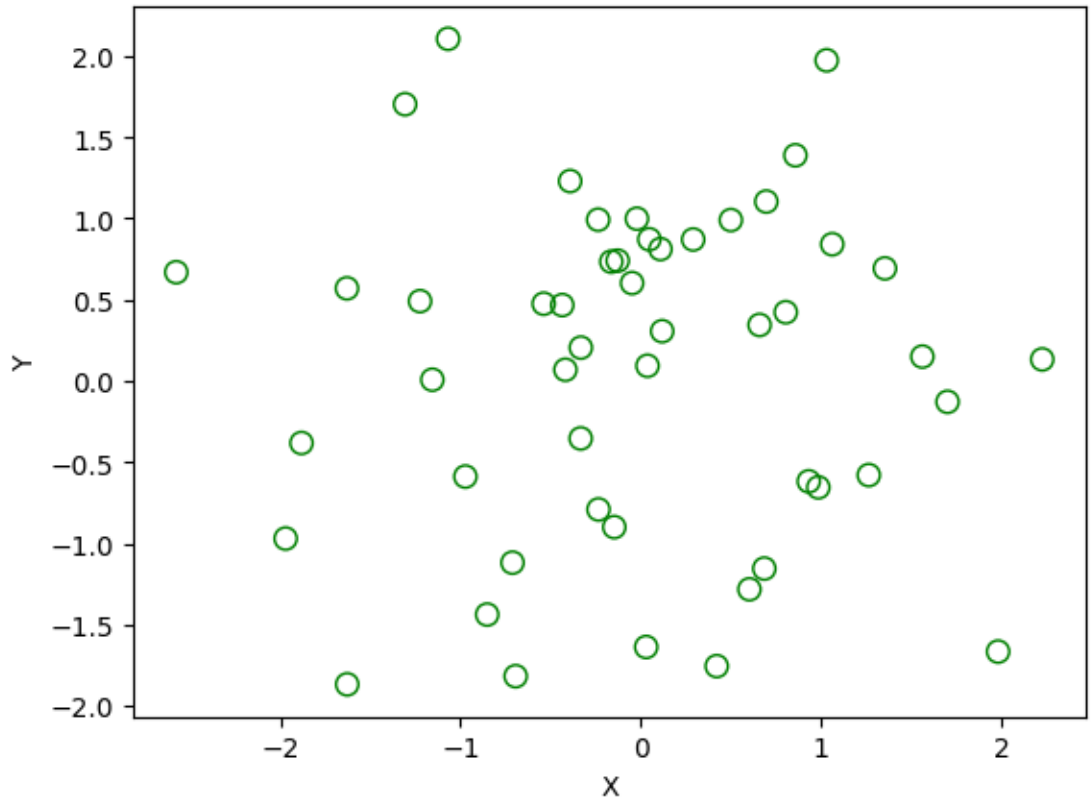
31.Write a Python program to create a stacked bar plot with error bars.
Note: Use bottom to stack the women?s bars on top of the men?s bars.
Sample Data:
Means (men) = (22, 30, 35, 35, 26)
Means (women) = (25, 32, 30, 35, 29)
Men Standard deviation = (4, 3, 4, 1, 5)
Women Standard deviation = (3, 5, 2, 3, 3)



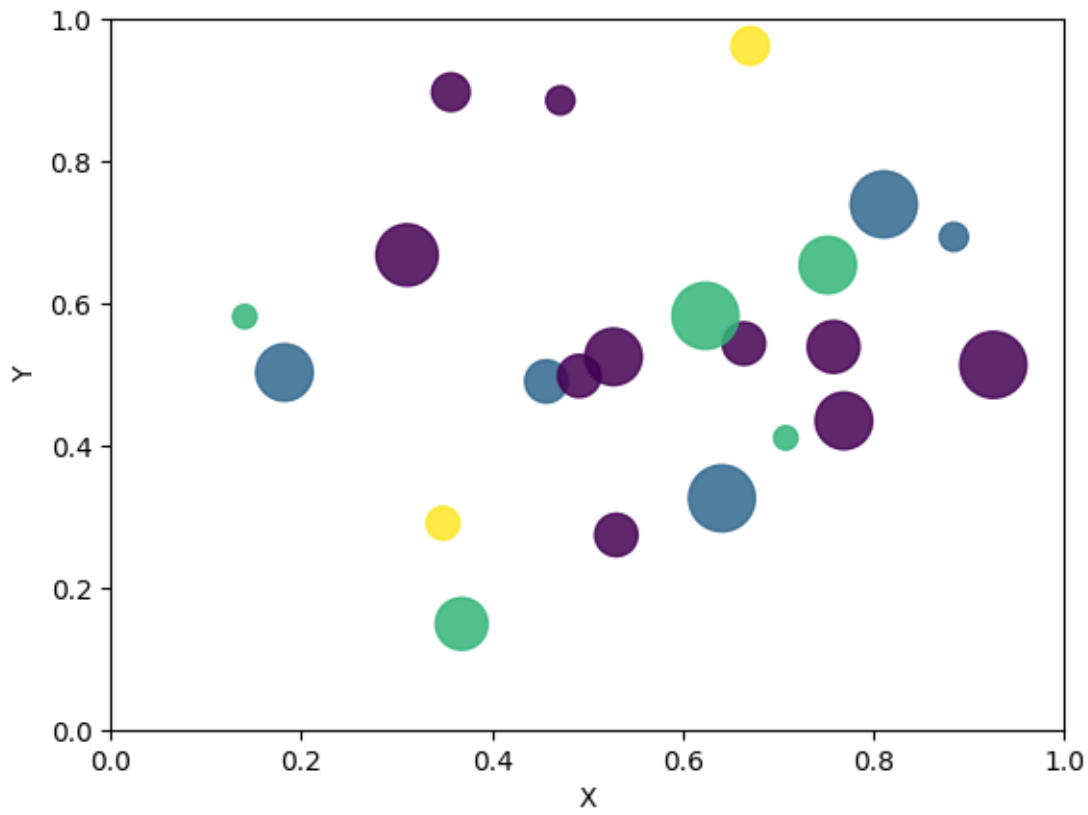
32.Write a Python program to draw a scatter graph taking a random distribution in X and Y and plotted against each other.



33.Write a Python program to draw a scatter plot with empty circles taking a random distribution in X and Y and plotted against each other.

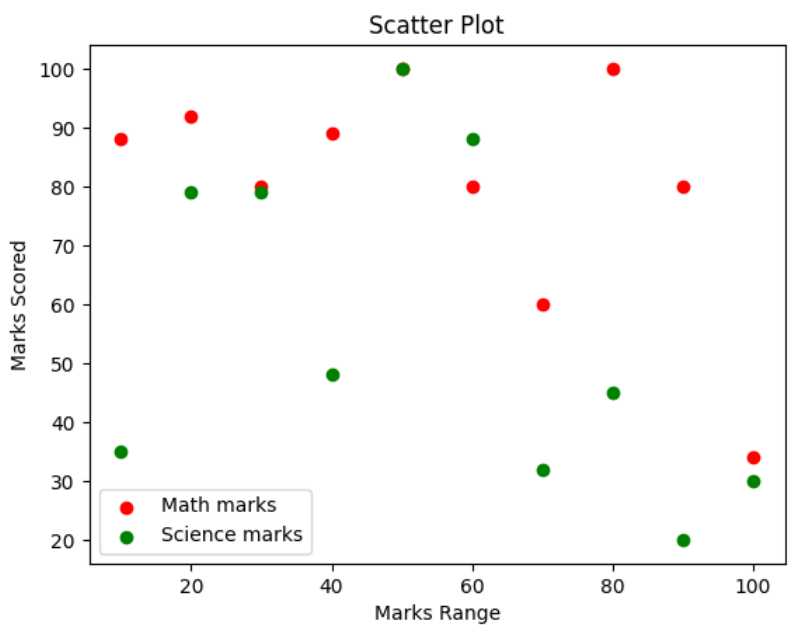


34.Write a Python program to draw a scatter plot using random distributions to generate balls of different sizes.

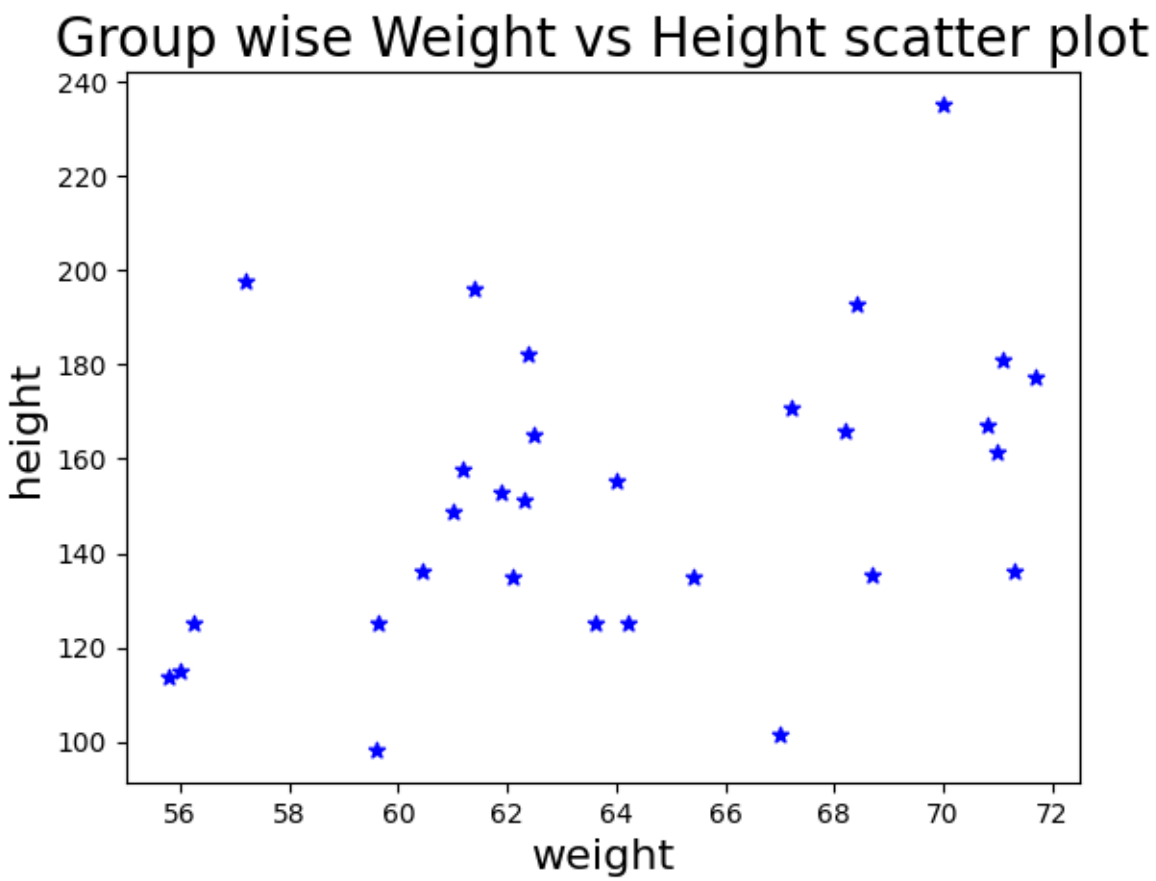


35.Write a Python program to draw a scatter plot comparing two subject marks of Mathematics and Science. Use marks of 10 students. Sample data:

Test Data:
math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]
science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]
marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

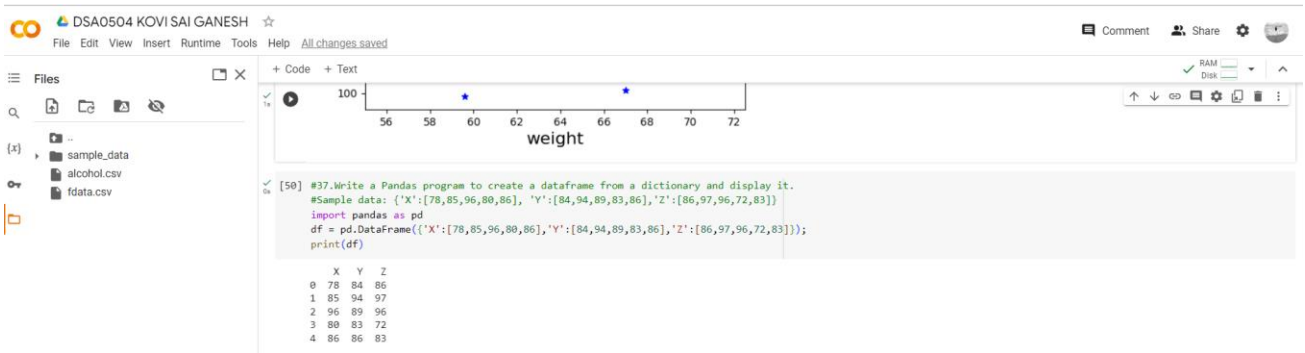


36.Write a Python program to draw a scatter plot for three different groups comparing weights and heights.



37.Write a Pandas program to create a dataframe from a dictionary and display it.

Sample data: {'X':[78,85,96,80,86], 'Y':[84,94,89,83,86],'Z':[86,97,96,72,83]}



38.Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.

Sample Python dictionary data and list labels:

exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

```
df = pd.DataFrame(exam_data, index=labels)
print(df)
```

```
         name  score  attempts qualify
a  Anastasia   12.5         1     yes
b       Dima    9.0         3      no
c  Katherine   16.5         2     yes
d      James    NaN         3      no
e      Emily    9.0         2      no
f    Michael   20.0         3     yes
g    Matthew   14.5         1     yes
h      Laura    NaN         1      no
i      Kevin    8.0         2      no
j      Jonas   19.0         1     yes
```

39.Write a Pandas program to get the first 3 rows of a given DataFrame.
Sample Python dictionary data and list labels:
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

```
First three rows of the data frame:
         name  score  attempts qualify
a  Anastasia   12.5         1     yes
b       Dima    9.0         3      no
c  Katherine   16.5         2     yes
```

40. Write a Pandas program to select the 'name' and 'score' columns from the following DataFrame.
Sample Python dictionary data and list labels:
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

```
Select specific columns:
        name  score
a  Anastasia   12.5
b       Dima    9.0
c  Katherine   16.5
d      James    NaN
e      Emily    9.0
f    Michael   20.0
g    Matthew   14.5
h      Laura    NaN
i      Kevin    8.0
j      Jonas   19.0
```