QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect

```python
dist_df = pd.DataFrame({'DEPARTMENT_ID': dist_id})

# Display the result
print(dist_df)
```

```
    DEPARTMENT_ID
0              10
1              20
2              30
3              40
4              50
5              60
6              70
7              80
8              90
9             100
10            110
11            120
12            130
13            140
14            150
15            160
16            170
17            180
18            190
19            200
20            210
21            220
22            230
23            240
24            250
25            260
26            270
```

26°C Mostly cloudy

Search

ENG IN

08:54 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect

```python
#2. Write a Pandas program to display the ID for those employees who did two or more jobs in the past.
import pandas as pd

data = {
    'EMPLOYEE_ID': [102, 101, 101, 201, 114, 122, 200, 176, 176, 200],
    'START_DATE': ['2001-01-13', '1997-09-21', '2001-10-28', '2004-02-17', '2006-03-24', '2007-01-01', '1995-09-17', '2006-03-24', '2007-01-01', '2002-07-01'],
    'END_DATE': ['2006-07-24', '2001-10-27', '2005-03-15', '2007-12-19', '2007-12-31', '2007-12-31', '2001-06-17', '2006-12-31', '2007-12-31', '2006-12-31'],
    'JOB_ID': ['IT_PROG', 'AC_ACCOUNT', 'AC_MGR', 'MK_REP', 'ST_CLERK', 'ST_CLERK', 'AD_ASST', 'SA_REP', 'SA_MAN', 'AC_ACCOUNT'],
    'DEPARTMENT_ID': [60, 110, 110, 20, 50, 50, 90, 80, 80, 90]
}

df = pd.DataFrame(data)

# Count the number of jobs each employee has done
job_counts = df.groupby('EMPLOYEE_ID')['JOB_ID'].count()

# Select employees who did two or more jobs
result = job_counts[job_counts >= 2]

# Display the employee IDs
print(result)
```

```
EMPLOYEE_ID
101    2
176    2
200    2
Name: JOB_ID, dtype: int64
```

26°C Mostly cloudy

Search

ENG IN

08:54 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect

```
MAX_SALARY : [40000, 30000, 6000, 16000, 9000, 16000, 9000, 20080, 12008, 15000, 5500, 8500, 5000, 5500, 10000, 15000, 9000, 9000, 10500]
}

df = pd.DataFrame(data)

# Sort the DataFrame by job title in descending order
sorted_df = df.sort_values(by='JOB_TITLE', ascending=False)

# Display the result
print(sorted_df)
```

```
        JOB_ID                      JOB_TITLE  MIN_SALARY  MAX_SALARY
11      ST_MAN                  Stock Manager        5500        8500
12    ST_CLERK                    Stock Clerk        2008        5000
13    SH_CLERK                  Shipping Clerk        2500        5500
8       SA_REP             Sales Representative        6000       12008
7       SA_MAN                    Sales Manager       10000       20080
9       PU_MAN               Purchasing Manager        8000       15000
10    PU_CLERK                 Purchasing Clerk        2500        5500
18      PR_REP  Public Relations Representative        4500       10500
6   AC_ACCOUNT                Public Accountant        4200        9000
14     IT_PROG                       Programmer        4000       10000
0      AD_PRES                        President       20080       40000
16      MK_REP         Marketing Representative        4000        9000
15      MK_MAN                Marketing Manager        9000       15000
17      HR_REP   Human Resources Representative        4000        9000
3       FI_MGR                  Finance Manager        8200       16000
1        AD_VP    Administration Vice President       15000       30000
2      AD_ASST         Administration Assistant        3000        6000
5       AC_MGR               Accounting Manager        8200       16000
4   FI_ACCOUNT                       Accountant        4200        9000
```

26°C Mostly cloudy | Search | ENG IN | 08:54 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

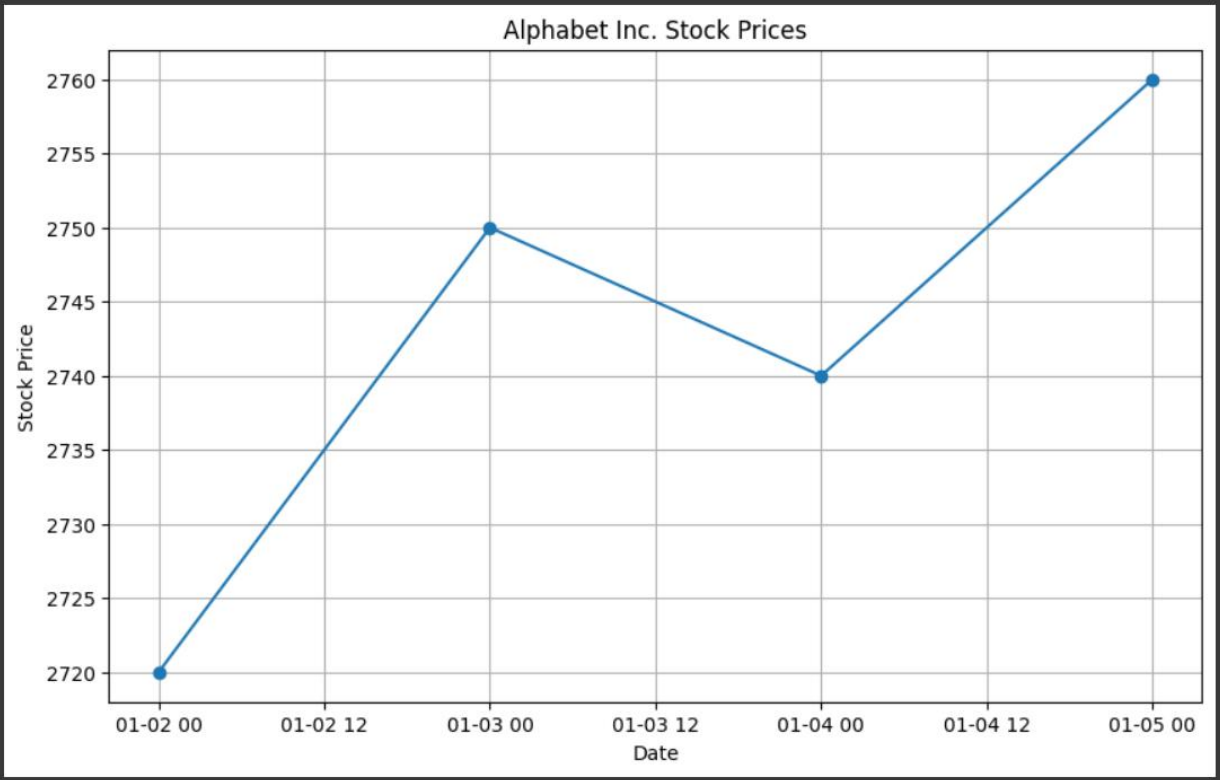
DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect



26°C Mostly cloudy

Search

ENG IN

08:55 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

DSA0504.ipynb

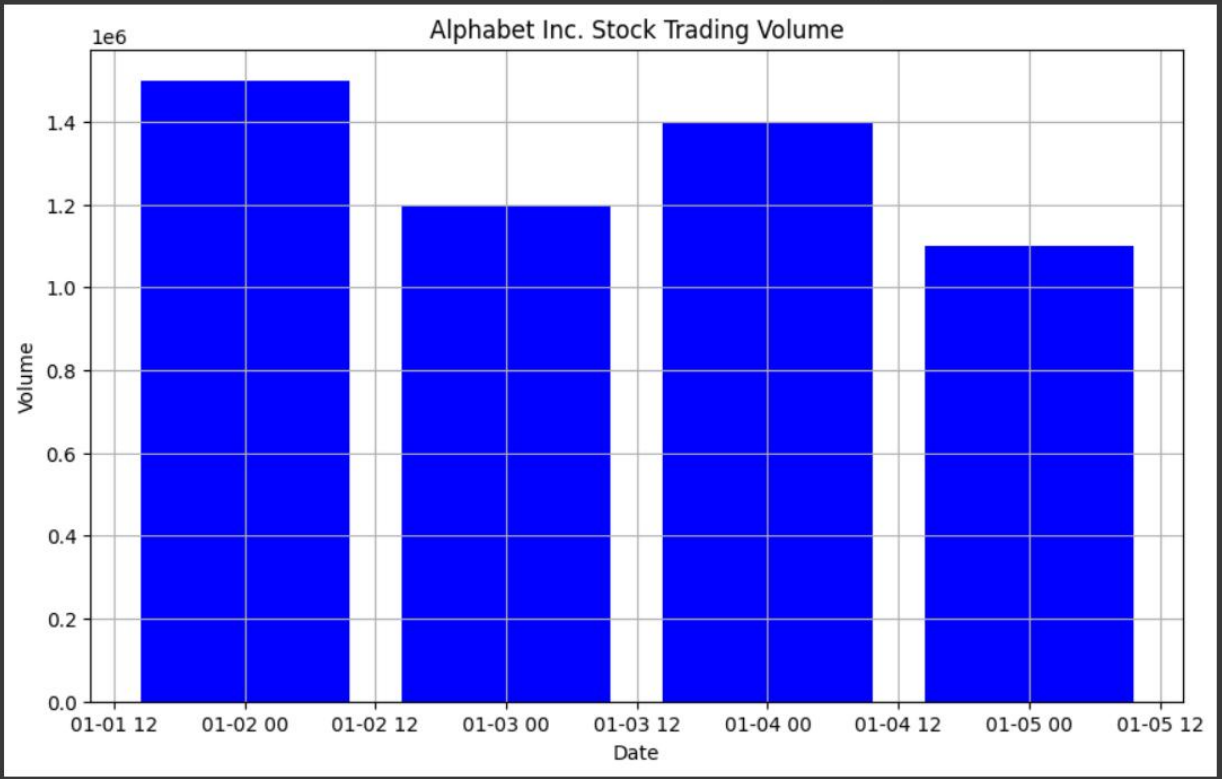
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect

```
plt.show()
```



26°C Mostly cloudy

Search

ENG IN

08:55 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

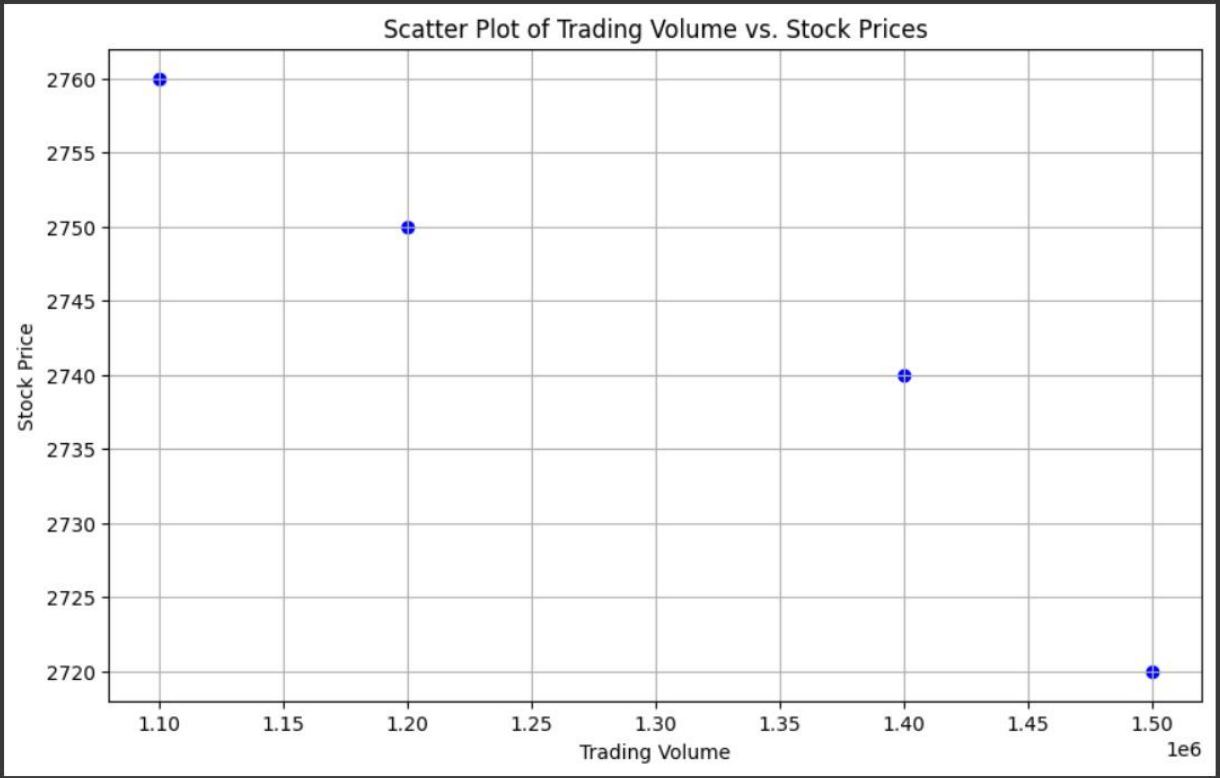
DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect



26°C Mostly cloudy

Search

ENG IN

08:55 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect

```python
#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)
import pandas as pd

# Sample sales_data table
data = {
    'Item': ['A', 'B', 'A', 'C', 'B', 'A', 'C', 'B', 'A', 'C'],
    'Sale': [100, 200, 150, 300, 250, 180, 320, 220, 170, 310]
}

# Create a DataFrame from the sample data
sales_data = pd.DataFrame(data)

# Create a pivot table to find the maximum and minimum sale values for each item
pivot_table = pd.pivot_table(sales_data, values='Sale', index='Item', aggfunc=[max, min])

# Rename the columns for clarity
pivot_table.columns = ['Max Sale', 'Min Sale']

# Display the pivot table
print(pivot_table)
```

```
      Max Sale  Min Sale
Item
A         180       100
B         250       200
C         320       300
```

[ ]

26°C Mostly cloudy

Search

ENG IN

08:55 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

DSA0504.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

Connect

```
A          180       100
B          250       200
C          320       300
```

```
#8. Write a Pandas program to create a Pivot table and find the item wise unit sold. .(refer sales_data table)
import pandas as pd

# Sample sales_data table
data = {
    'Item': ['A', 'B', 'A', 'C', 'B', 'A', 'C', 'B', 'A', 'C'],
    'Unit Sold': [5, 10, 8, 12, 15, 9, 16, 11, 7, 13]
}

# Create a DataFrame from the sample data
sales_data = pd.DataFrame(data)

# Create a pivot table to find the total unit sold for each item
pivot_table = pd.pivot_table(sales_data, values='Unit Sold', index='Item', aggfunc='sum')

# Display the pivot table
print(pivot_table)
```

```
      Unit Sold
Item
A            29
B            36
C            41
```

```
#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)
```

26°C Mostly cloudy

Search

ENG IN

08:55 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

+ Code + Text All changes saved Connect

```python
#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)
import pandas as pd

# Create the Sales_data DataFrame
data = {
    'OrderDate': ['1-6-18', '1-23-18', '2-9-18', '2-26-18', '3-15-18', '4-1-18', '4-18-18', '5-5-18', '5-22-18', '6-8-18', '6-25-18', '7-12-18', '7-29-18', '8-15-18', '9-1-18', '9-18-
    'Region': ['East', 'Central', 'Central', 'Central', 'West', 'East', 'Central', 'Central', 'West', 'East', 'Central', 'East', 'East', 'East', 'Central', 'East', 'Central', 'East'],
    'Manager': ['Martha', 'Hermann', 'Hermann', 'Timothy', 'Timothy', 'Martha', 'Martha', 'Hermann', 'Douglas', 'Martha', 'Hermann', 'Martha', 'Douglas', 'Martha', 'Douglas', 'Martha'
    'SalesMan': ['Alexander', 'Shelli', 'Luis', 'David', 'Stephen', 'Alexander', 'Steven', 'Luis', 'Michael', 'Alexander', 'Sigal', 'Diana', 'Karen', 'Alexander', 'John', 'Alexander',
    'Item': ['Television', 'Home Theater', 'Television', 'Cell Phone', 'Television', 'Home Theater', 'Television', 'Television', 'Television', 'Home Theater', 'Television', 'Home Thea
    'Units': [95, 50, 36, 27, 56, 60, 75, 90, 32, 60, 90, 29, 81, 35, 2, 16, 28, 64],
    'Unit_price': [1198.00, 500.00, 1198.00, 225.00, 1198.00, 500.00, 1198.00, 1198.00, 1198.00, 500.00, 1198.00, 500.00, 500.00, 1198.00, 125.00, 58.50, 500.00, 225.00],
    'Sale_amt': [113810.00, 25000.00, 43128.00, 6075.00, 67088.00, 30000.00, 89850.00, 107820.00, 38336.00, 30000.00, 107820.00, 14500.00, 40500.00, 41930.00, 250.00, 936.00, 14000.00
}

sales_data = pd.DataFrame(data)

# Create a pivot table to find total sale amount region-wise, manager-wise, and salesman-wise
pivot_table = pd.pivot_table(sales_data, values='Sale_amt', index=['Region', 'Manager', 'SalesMan'], aggfunc='sum')

print(pivot_table)
```

```
                          Sale_amt
Region  Manager SalesMan
Central Douglas John         250.0
        Hermann Luis       150948.0
                Shelli      25000.0
                Sigal      121820.0
        Martha  Steven      89850.0
        Timothy David        6075.0
East    Douglas Karen       40500.0
        Martha  Alexander  231076.0
                Diana       14500.0
West    Douglas Michael     38336.0
        Timothy Stephen     67088.0
```

26°C Mostly cloudy Search ENG IN 08:55 01-12-2023

QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

+ Code + Text All changes saved Connect

```
[ ]
#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.
import pandas as pd
import numpy as np

# Create a DataFrame with random values
data = np.random.randn(10, 4)

df = pd.DataFrame(data, columns=['Column1', 'Column2', 'Column3', 'Column4'])

# Apply conditional formatting to highlight negative numbers in red and positive numbers in green
styled_df = df.style.applymap(lambda val: f'color: {"red" if val < 0 else "black"}')

# Display the styled DataFrame
styled_df
```

	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

26°C Mostly cloudy Search ENG IN 08:55 01-12-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=3TIjUtZXM53F

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
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-1-29d4b3fe49dc>:17: 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

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:03 AM

28°C Mostly cloudy

Search

ENG IN 10:05 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=uq8JPbR5NkR3

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help Saving...

Comment Share

+ Code + Text

RAM Disk

```
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

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:06 AM

28°C High winds soon

Search

ENG IN 10:06 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=KqfWDL4bNwZ5

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help Saving...

Comment Share

+ Code + Text

RAM Disk

```
print(df.isna())
```

```
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
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:07 AM

28°C High winds soon

Search

ENG IN

10:07 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=ABq9_zEfN4vW

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk

```
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
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:08 AM

28°C Mostly cloudy

Search

ENG IN

10:08 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=LJ8F1ZpQOUYn

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
print(result)
```

```
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
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:09 AM

Top events Event brief

Search

ENG IN 10:09 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=hxgU0ulrOidr

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
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

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

    address
S2  street2
S5  street2

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

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

    address
S6  street4
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:10 AM

28°C Mostly cloudy

Search

ENG IN 10:10 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=ygiWbPTkOuU6

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk

```
print("\nMean, min, and max value of age for each school with customized column names: ")
grouped_single = student_data.groupby('school_code').agg(Age_Mean = ('age','mean'),
print(grouped_single)
```

```
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
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4

0s completed at 10:11 AM

28°C Mostly cloudy

Search

ENG IN 10:11 28-11-2023

DSA0504 - Query Processing fo | LAB_QUESTIONS - Google Doc | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=efquhsXYO3eE

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

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

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

    address
S2  street2
S5  street2

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

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

    address
S6  street4
```

Release notes

Please follow our blog to see more information about new features, tips and tricks, and featured notebooks such as Analyzing a Bank Failure with Colab.

2023-11-08

- Launched Secrets, for safe storage of private keys on Colab (tweet)
- Fixed issue where TensorBoard would not load (#3990)
- Python package upgrades
  - lightgbm 4.0.0 -> 4.1.0
  - bigframes 0.10.0 -> 0.12.0
  - bokeh 3.2.2 -> 3.3.0
  - duckdb 0.8.1 -> 0.9.1
  - numba 0.56.4 -> 0.58.1
  - tweepy 4.13.0 -> 4.14.0
  - jax 0.4.16 -> 0.4.20
  - jaxlib 0.4.16 -> 0.4.20

2023-10-23

- Updated the **Open notebook** dialog for better usability and support for smaller screen sizes
- Added smart paste support for data from Google Sheets for R notebooks
- Enabled showing release notes in a tab
- Launched AI coding features for Pro/Pro+ users in Australia AU Canada CA India IN and Japan JP (tweet)
- Python package upgrades
  - earthengine-api 0.1.357 -> 0.1.375
  - flax 0.7.2 -> 0.7.4
  - geemap 0.27.4 -> 0.28.2
  - jax 0.4.14 -> 0.4.16

28°C Mostly cloudy

Search

ENG IN

10:12 28-11-2023

DSA0504 - Query Processing fo | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=Valnq4Z5TZq3

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

Files

..
sample_data
Al.csv
Alcohol.CSV

Disk 80.83 GB available

+ Code + Text

RAM
Disk

```python
print(w_a_con.head())
print('\nShape of the dataframe: ',w_a_con.shape)
print('\nNumber of rows: ',w_a_con.shape[0])
print('\nNumber of column: ',w_a_con.shape[1])
print("\nExtract Column Names:")
print(w_a_con.columns)
```

```
World alcohol consumption sample data:
                      Year    WHO region Country Beverage Types  \
1986        Western Pacific      Viet Nam                 Wine
1986               Americas       Uruguay                Other
1985                 Africa  Cte d'Ivoire                 Wine
1986               Americas      Colombia                 Beer
1987  Saint Kitts and Nevis          Beer                 1.98

      Display Value
1986           0.00
1986           0.50
1985           1.62
1986           4.27
1987            NaN

Shape of the dataframe:  (5, 4)

Number of rows:  5

Number of column:  4

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

28°C Mostly cloudy

Search

ENG IN

10:38 28-11-2023

DSA0504 - Query Processing fo | solutions - Google Docs | Untitled0.ipynb - Colaboratory

colab.research.google.com/drive/16sXSTSt6zwy9W5O7s0T5Mqf8tuldlXX2#scrollTo=8XSpUkqkV-yF

Gmail YouTube Maps

Untitled0.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

Files

..
sample_data
Al.csv
Alcohol.CSV

+ Code + Text

RAM
Disk

```python
import pandas as pd
df = pd.DataFrame({
    'name_code': ['c0001','1000c','b00c2', 'b2c02', 'c2222'],
    'date_of_birth ': ['12/05/2002','16/02/1999','25/09/1998','12/02/2022','15/09/1997'],
    'age': [18.5, 21.2, 22.5, 22, 23]
})
print("Original DataFrame:")
print(df)
print("\nIndex of a substring in a specified column of a dataframe:")
df['Index'] = list(map(lambda x: x.find('c', 0, 5), df['name_code']))
print(df)
```

```
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
```

Disk 80.83 GB available

28°C Mostly cloudy

Search

ENG IN

10:42 28-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW

Gmail YouTube Maps

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```python
import pandas as pd
df = pd.DataFrame({
    'company_code': ['Abcd','EFGF', 'zefsalf', 'sdfslew', 'zekfsdf'],
    'date_of_sale': ['12/05/2002','16/02/1999','25/09/1998','12/02/2022','15/09/1997'],
    'sale_amount': [12348.5, 233331.2, 22.5, 2566552.0, 23.0]
})
print("Original DataFrame:")
print(df)
print("\nSwapp cases in comapny_code:")
df['swapped_company_code'] = list(map(lambda x: x.swapcase(), df['company_code']))
print(df)
```

```
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

Swapp cases in comapny_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
```

1s completed at 8:27 AM

25°C Mostly cloudy

Search

ENG IN 08:27 30-11-2023

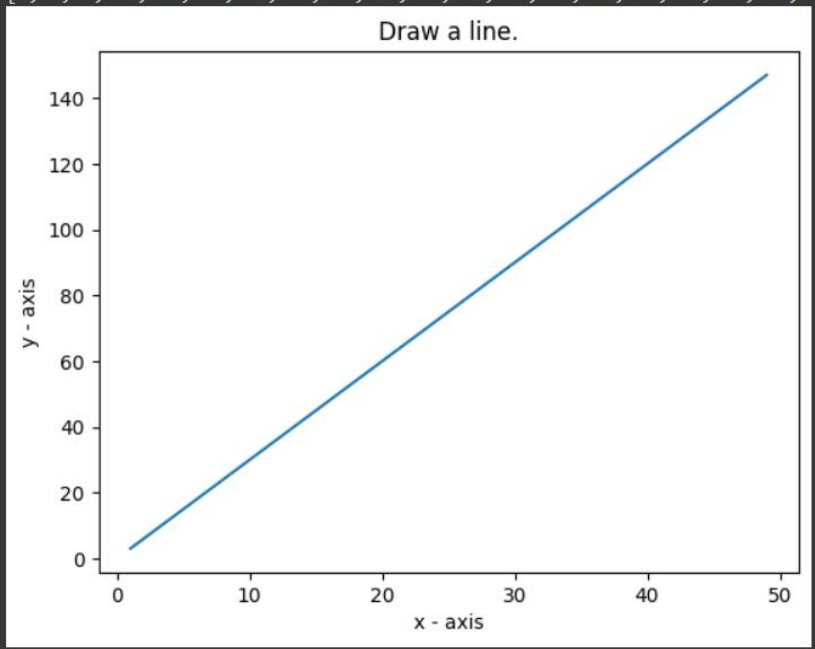
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=7mAsb41XKMLs

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

```
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, 123, 126, 129, 132,
```



0s completed at 8:27 AM

25°C Mostly cloudy

Search

ENG IN

08:27 30-11-2023

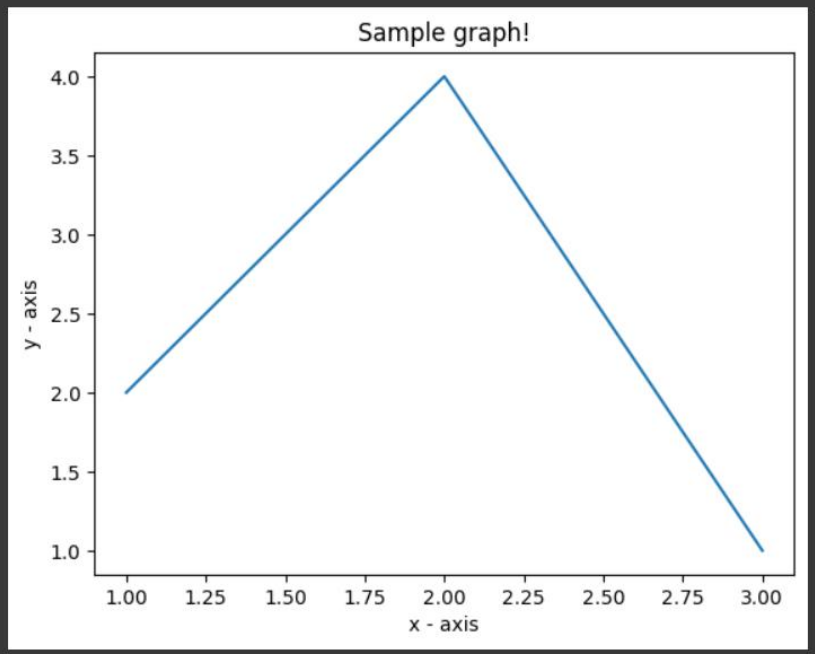
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=4A4WF5zhKVQu

Untitled1.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text



0s completed at 8:28 AM

25°C Mostly cloudy

Search

ENG IN

08:28 30-11-2023

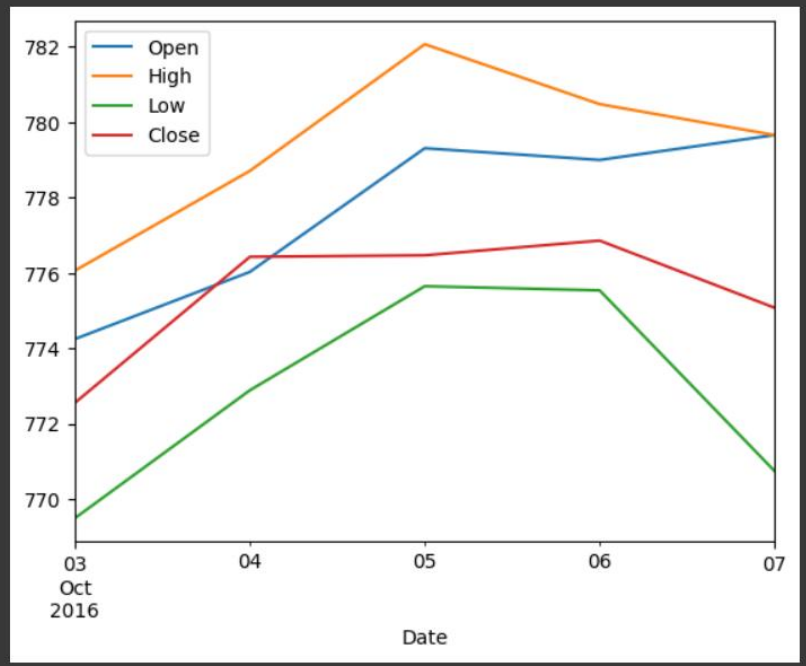
QUERY PROCESSING FOR DATA | DSA0504.ipynb - Colaboratory

colab.research.google.com/drive/1tPDD-9jCZBs9rwYMSVYlAQ1E2IKNril4

Gmail YouTube Maps

+ Code + Text All changes saved Connect

```
[ ] import matplotlib.pyplot as plt
    import pandas as pd
    df = pd.read_csv('fdata.csv', sep=',', parse_dates=True,
    index_col=0)
    df.plot()
    plt.show()
```



26°C Mostly cloudy Search ENG IN 08:56 01-12-2023

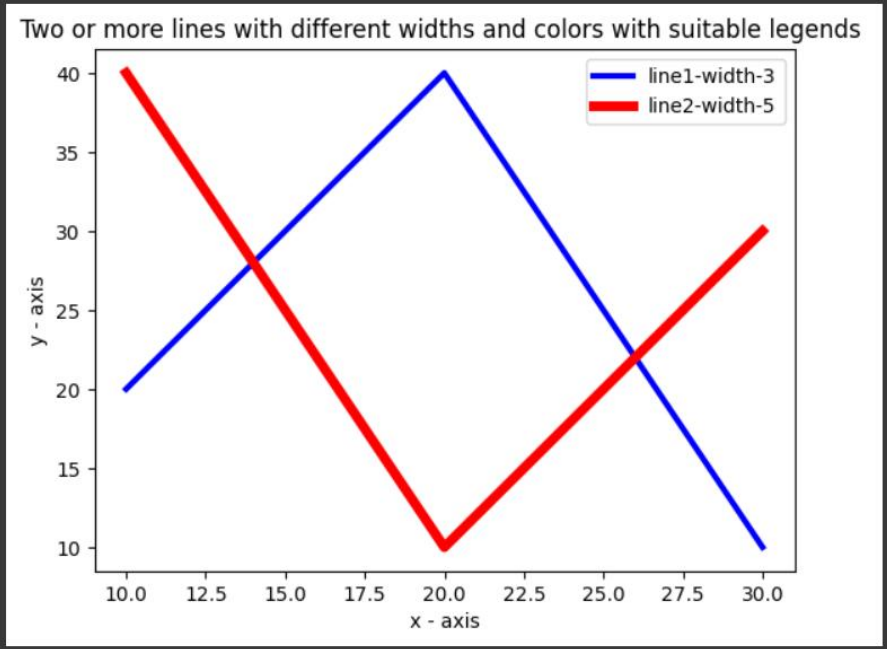
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=MA8jdNzIKc9L

Untitled1.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text



0s completed at 8:28 AM

25°C Mostly cloudy

Search

ENG IN

08:28 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=DrfuT9oVKmV1

Gmail YouTube Maps

Untitled1.ipynb

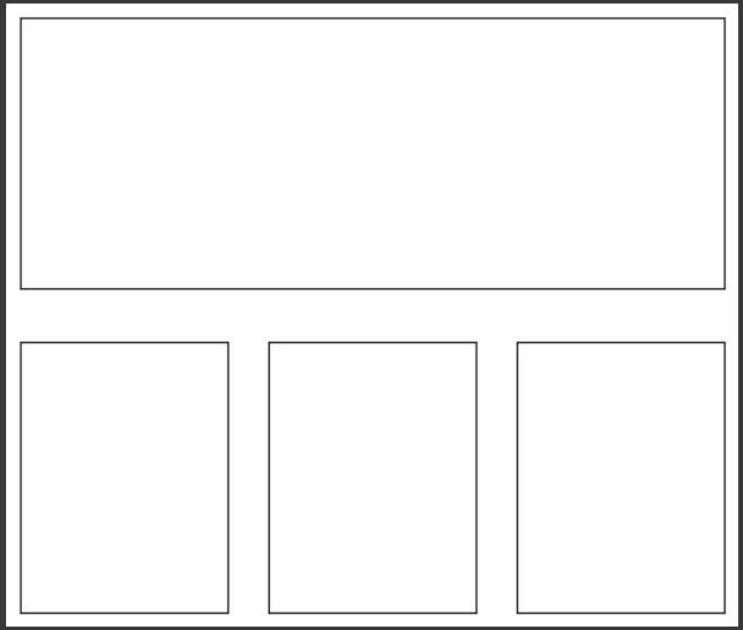
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
plt.show()
```



0s completed at 8:29 AM

25°C Mostly cloudy

Search

ENG IN

08:29 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=eV7t-ElgKrzL

Gmail YouTube Maps

Untitled1.ipynb

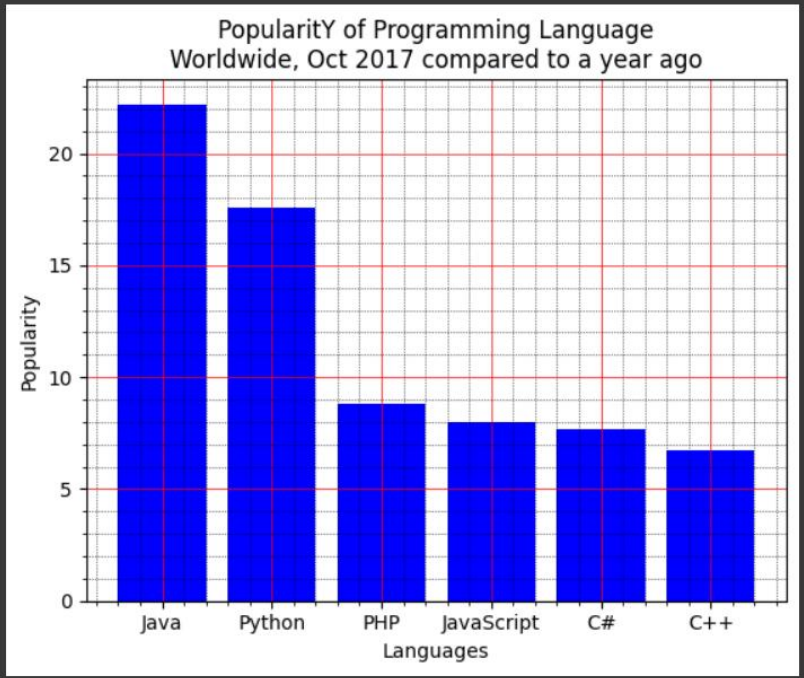
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
# customize the minor grid
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.show()
```



0s completed at 8:29 AM

25°C Mostly cloudy

Search

ENG IN

08:29 30-11-2023

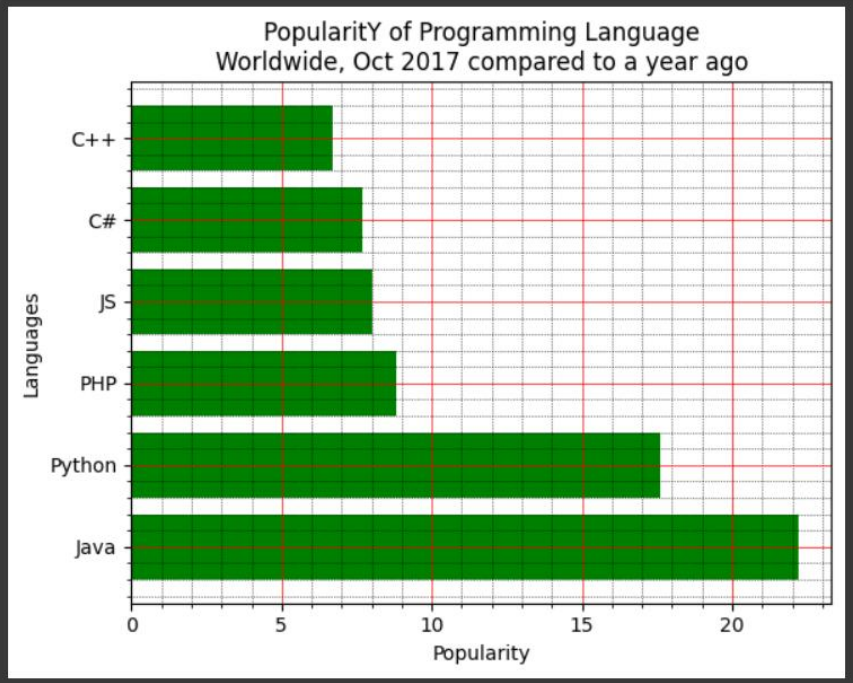
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=26KE4TpzK1Rx

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text



1s completed at 8:30 AM

25°C Mostly cloudy

Search

ENG IN

08:30 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=RV3T9XfkK_oD

Gmail YouTube Maps

Untitled1.ipynb

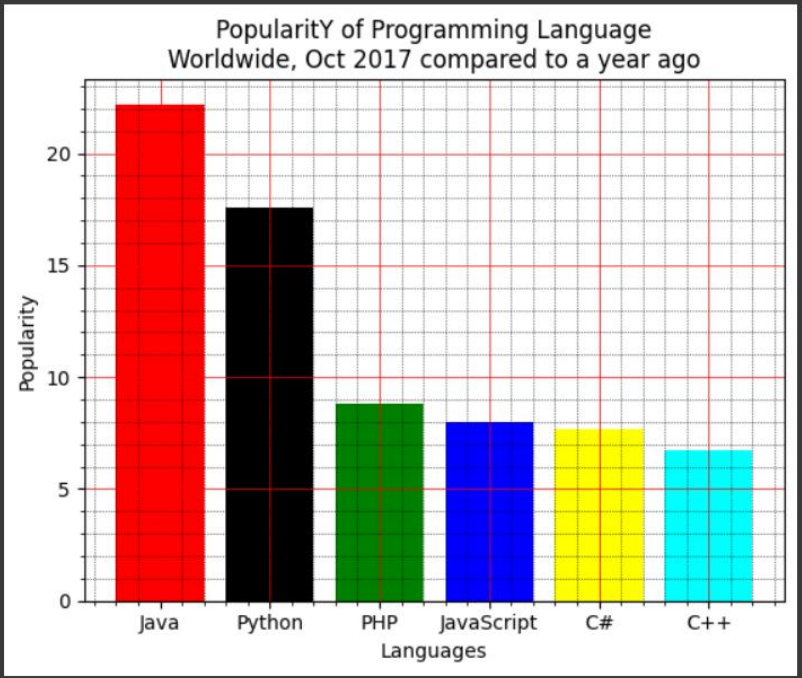
File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk

```
plt.show()
```



1s completed at 8:30 AM

25°C Mostly cloudy

Search

ENG IN

08:30 30-11-2023

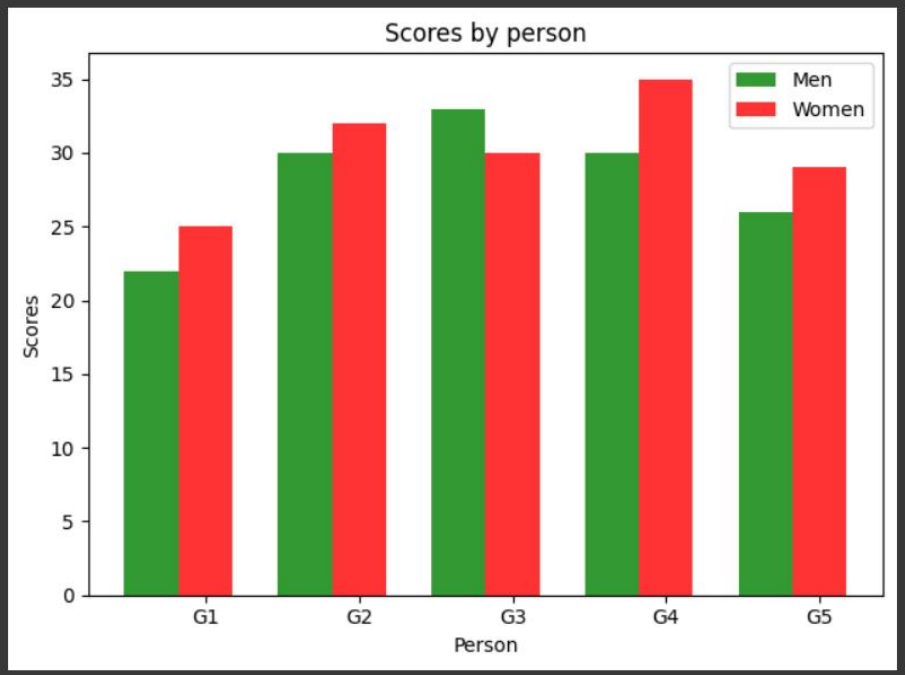
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=h-9gvewVLFS-

Untitled1.ipynb

File Edit View Insert Runtime Tools Help Saving...

```
plt.show()
```



0s completed at 8:31 AM

25°C Mostly cloudy

Search

ENG IN

08:31 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=5JWdy1GRLJaz

Gmail YouTube Maps

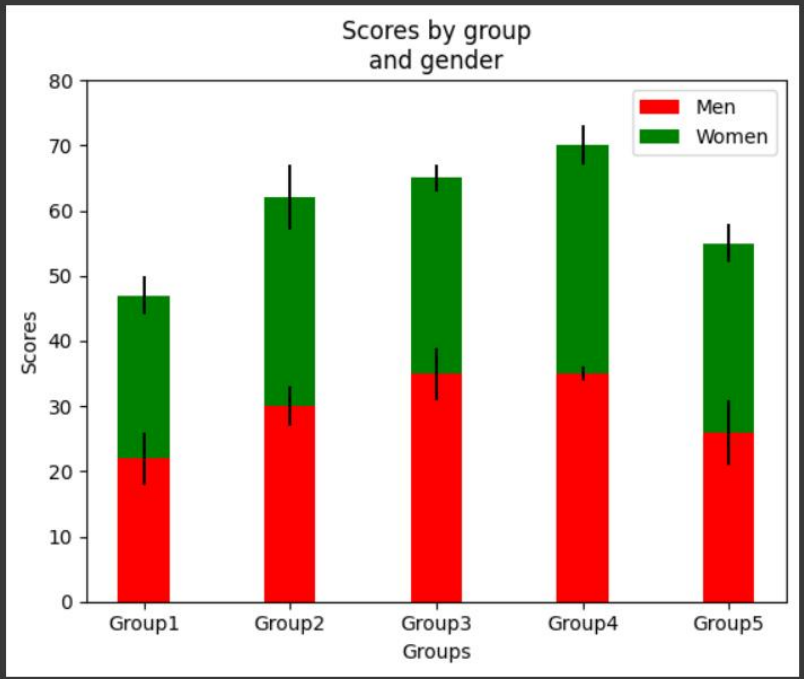
Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk



[ ]

0s completed at 8:31 AM

25°C Mostly cloudy

Search

ENG IN

08:31 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=j6BiADFxLZDy

Gmail YouTube Maps

Untitled1.ipynb

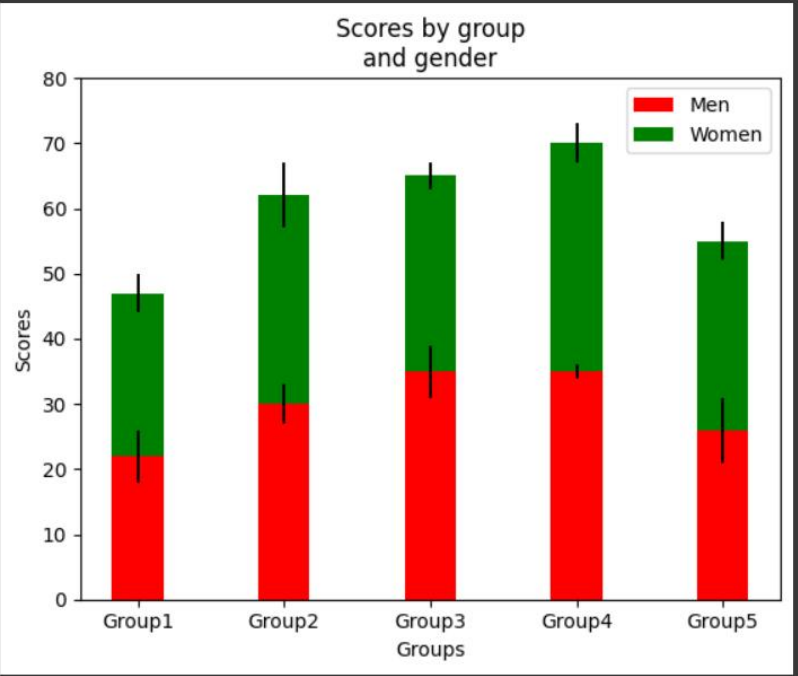
File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

[10]
```
plt.show()
```



0s completed at 8:34 AM

25°C Mostly cloudy

Search

ENG IN

08:40 30-11-2023

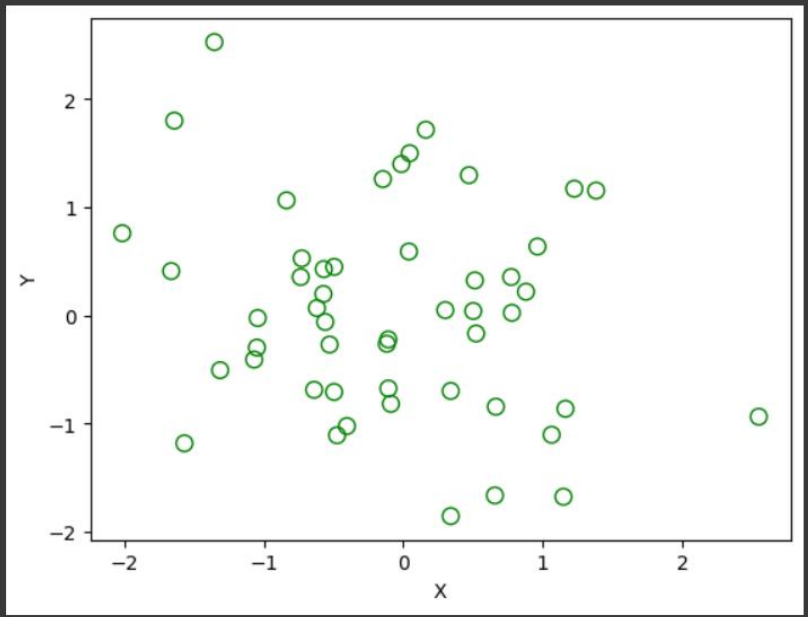
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=V9MOcBwELUDE

Untitled1.ipynb

File Edit View Insert Runtime Tools Help Saving...

```
plt.ylabel("Y")
plt.show()
```



0s completed at 8:32 AM

25°C Mostly cloudy

ENG IN 08:32 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=BmLoXcvtLYwo

Gmail YouTube Maps

Untitled1.ipynb

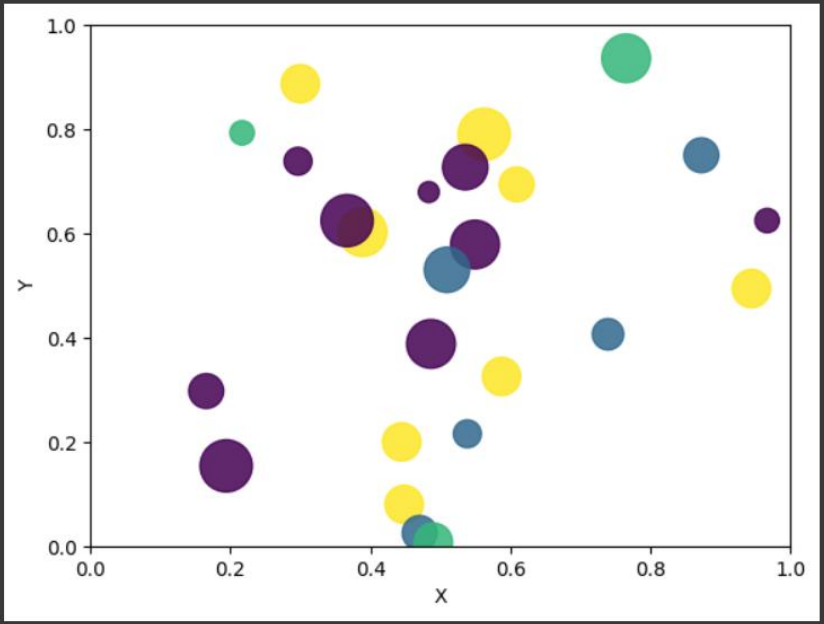
File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk

```
plt.xlabel("X")
plt.ylabel("Y")
plt.show()
```



0s completed at 8:32 AM

25°C Mostly cloudy

Search

ENG IN

08:32 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=blXKuh7pLd5X

Gmail YouTube Maps

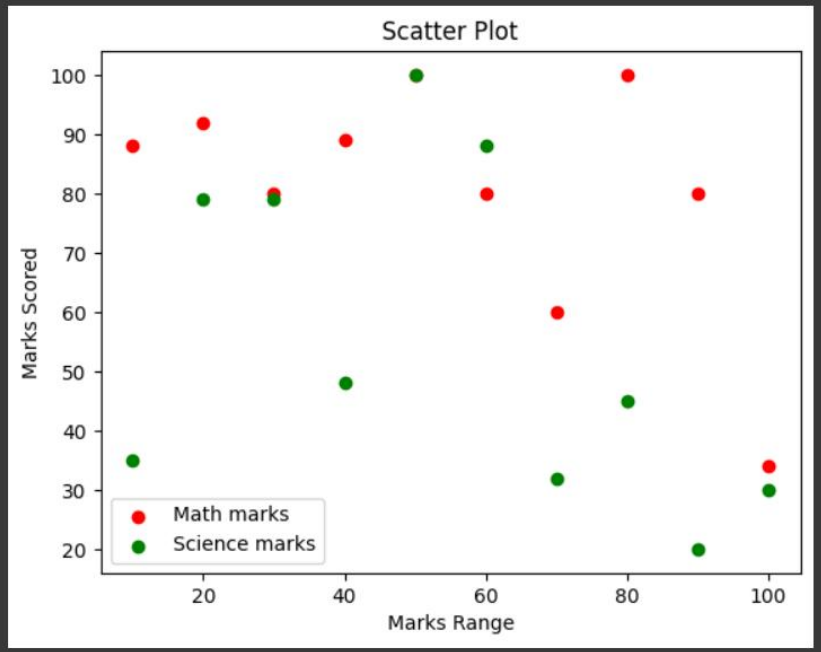
Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk



[ ]

0s completed at 8:32 AM

25°C Mostly cloudy

Search

ENG IN

08:32 30-11-2023

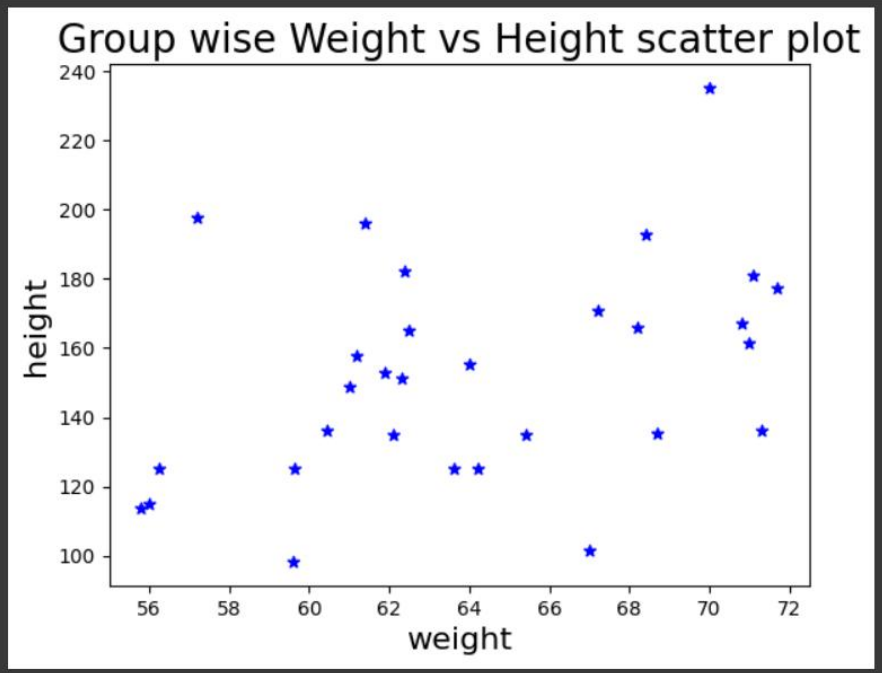
solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=gN8HmR1YLh-B

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text



[ ]

0s completed at 8:33 AM

25°C Mostly cloudy

Search

ENG IN

08:33 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=OFgzyNzkLn_E

Gmail YouTube Maps

Untitled1.ipynb

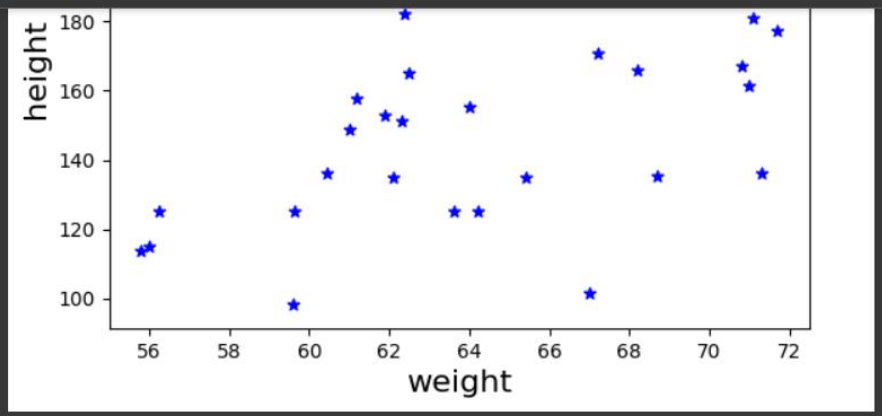
File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk

[15]




```python
import pandas as pd
df = pd.DataFrame({'X':[78,85,96,80,86], 'Y':[84,94,89,83,86],'Z':[86,97,96,72,83]});
print(df)
```

```
    X  Y  Z
0  78 84 86
1  85 94 97
2  96 89 96
3  80 83 72
4  86 86 83
```

[ ]

0s completed at 8:33 AM

25°C Mostly cloudy

Search

ENG IN

08:33 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=lEVFXifwLsqK

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

[16]

```
3  80  83  72
4  86  86  83
```

```python
import pandas as pd
import numpy as np

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
```

[ ]

0s completed at 8:33 AM

25°C Mostly cloudy

Search

ENG IN 08:34 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=9VZRyze3LzDz

Gmail YouTube Maps

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```python
import pandas as pd
import numpy as np

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("First three rows of the data frame:")
print(df.iloc[:3])
```

```
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
```

[ ]

0s completed at 8:34 AM

25°C Mostly cloudy

Search

ENG IN

08:34 30-11-2023

solutions - Google Docs | Untitled1.ipynb - Colaboratory

colab.research.google.com/drive/17segfgVkgr4C_6sngy6AEwaG9WlPdFNW#scrollTo=j6BiADFxLZDy

Untitled1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

```python
import pandas as pd
import numpy as np

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("Select specific columns:")
print(df[['name', 'score']])
```

```
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
```

0s completed at 8:34 AM

25°C Mostly cloudy

Search

ENG IN 08:34 30-11-2023