

Lab Exercise - Python Libraries and Packages

Part 1 – Implement Basic Data Structures using Numpy, Pandas

1. Lists
2. Arrays
3. Identify their type using type()
4. Perform mathematical operations on these datasets created - multiplication, division, poweroff
5. Combine text with Numpy function to generate a textual output - "Addition of Two: array1 +array 2
6. Implement np.sin(), log(), log2(), np.exp())

The screenshot shows a Google Colab notebook titled 'Untitled0.ipynb'. The code in the first cell is as follows:

```
import numpy as np
import pandas as pd
# 1.Lists
list1 = [1, 2, 3, 4, 5]
list2 = [6, 7, 8, 9, 10]
# 2.Arrays using numpy
array1 = np.array(list1)
array2 = np.array(list2)
# 3.Identify their type using type()
print("Type of list1:", type(list1))
print("Type of array1:", type(array1))
# 4.Perform mathematical operations
# Multiplication
result_multiply = array1 * array2
print("Multiplication result:", result_multiply)
#Division
result_divide = array2 / array1
print("Division result:", result_divide)
# Power off
result_poweroff = array1 ** 2
print("Power off result:", result_poweroff)
# 5. Combine text with Numpy function
text_output = np.array2string(array1) + " + " + np.array2string(array2)
print("Textual output:", text_output)
# 6. Implement np.sin(), log(), log2(), np.exp()
sin_values = np.sin(array1)
log_values = np.log(array1)
log2_values = np.log2(array1)
exp_values = np.exp(array1)
print("sin values:", sin_values)
print("log values:", log_values)
print("log2 values:", log2_values)
print("exp values:", exp_values)
```

The output of the code is displayed in the second cell:

```
Type of list1: <class 'list'>
Type of array1: <class 'numpy.ndarray'>
Multiplication result: [ 6 14 24 36 50]
Division result: [ 0. 3.5 2.66666667 2.25 2. ]
Power off result: [ 1 4 9 16 25]
Textual output: [1 2 3 4 5] + [ 6 7 8 9 10]
sin values: [ 0.84147098  0.90929743  0.14112001 -0.7568025 -0.95892427]
log values: [ 0. 0.69314718 1.09861229 1.38629436 1.60943791]
log2 values: [ 0. 1. 1.5849625 2. 2.32192809]
exp values: [ 2.71828183  7.3890561 20.08553692 54.59815003 148.4131591 ]
```

This screenshot shows the same Google Colab notebook, but with the output of the code displayed in the second cell. The output is as follows:

```
Type of list1: <class 'list'>
Type of array1: <class 'numpy.ndarray'>
Multiplication result: [ 6 14 24 36 50]
Division result: [ 0. 3.5 2.66666667 2.25 2. ]
Power off result: [ 1 4 9 16 25]
Textual output: [1 2 3 4 5] + [ 6 7 8 9 10]
sin values: [ 0.84147098  0.90929743  0.14112001 -0.7568025 -0.95892427]
log values: [ 0. 0.69314718 1.09861229 1.38629436 1.60943791]
log2 values: [ 0. 1. 1.5849625 2. 2.32192809]
exp values: [ 2.71828183  7.3890561 20.08553692 54.59815003 148.4131591 ]
```

Part 2 – Visualization of Data using matplotlib, pyplots Packages

1. Generate a Data set for Health Care using the following: people = ['kiran', 'arun', 'vijay', 'varun'] age =[25, 30, 35, 40, 45] height =[145, 151, 165, 173] weight=[45, 55, 65, 75]
2. Using the generated dataset given above. Generate the following graphs and justify the relationships among the vectors.
3. Scatter Plot
4. Bar Chart
5. Histogram
6. Provide Graph Title, labels for X, Y axis with proper justification and explanation of the graph.

The screenshot shows a Google Colaboratory notebook interface. The browser tabs at the top include 'My Profile | Networking Acad...', 'I01c01_introduction_to_colab...', and 'Untitled0.ipynb - Colaboratory'. The address bar shows the URL: `colab.research.google.com/drive/1B3m0vxZABW0m1RAuF_sb8aos0Mv5LFD#scrollTo=OzyoQgWd5wvm`. The notebook title is 'Untitled0.ipynb'. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', 'Help', and 'All changes saved'. On the left, there is a 'Table of contents' sidebar with a search bar and icons for sections, code, and files. The main code area contains the following Python code:

```
import matplotlib.pyplot as plt

# 1. Generate Data set for Health Care
people = ['Kiran', 'Arun', 'Vijay', 'Varun']
age = [25, 30, 35, 40]
height = [145, 151, 165, 173]
weight = [45, 55, 65, 75]

# 2. Scatter Plot
plt.figure(figsize=(8, 6))
plt.scatter(age, height, color='blue', label='Height')
plt.scatter(age, weight, color='red', label='Weight')
plt.title('Scatter Plot of Age vs Height/Weight')
plt.xlabel('Age')
plt.ylabel('Height/Weight')
plt.legend()
plt.grid(True)
plt.show()

# 3. Bar Chart
plt.figure(figsize=(8, 6))
bar_width = 0.35
bar_positions1 = [a - bar_width/2 for a in age]
bar_positions2 = [a + bar_width/2 for a in age]
plt.bar(bar_positions1, height, width=bar_width, color='blue', label='Height')
plt.bar(bar_positions2, weight, width=bar_width, color='red', label='Weight')
plt.title('Bar Chart of Age vs Height/Weight')
plt.xlabel('Age')
plt.ylabel('Height/Weight')
plt.legend()
plt.grid(axis='y')
plt.show()
```

On the right side, the 'Release notes' panel is open, showing updates for 2022-07-22 and 2022-07-01. The 2022-07-22 notes include updates to scipy, pytorch, jax, opencv-python, spaCy, and dlib, as well as fixes for Open In tab doc link rendering and default tab orientation. The 2022-07-01 notes include updates to drives and spacy, and improvements to LSP handling and FAQ entries. At the bottom of the notebook, a status bar indicates '1s completed at 7:55PM'.

My Profile | Networking Academy | x | I01c01_introduction_to_colab... x | Untitled0.ipynb - Colaboratory x +

colab.research.google.com/drive/1Bm0vzZABW0m1RAuF_sb8aos0Mv5LdF#scrollTo=OzyoQgWd5wmm

Untitled0.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Table of contents

+ Code + Text

```

plt.figure(figsize=(8, 6))
plt.scatter(age, height, color='blue', label='Height')
plt.scatter(age, weight, color='red', label='Weight')
plt.title('Scatter Plot of Age vs Height/Weight')
plt.xlabel('Age')
plt.ylabel('Height/Weight')
plt.legend()
plt.grid(True)
plt.show()

# 3. Bar Chart
plt.figure(figsize=(8, 6))
bar_width = 0.35
bar_positions1 = [a - bar_width/2 for a in age]
bar_positions2 = [a + bar_width/2 for a in age]
plt.bar(bar_positions1, height, width=bar_width, color='blue', label='Height')
plt.bar(bar_positions2, weight, width=bar_width, color='red', label='Weight')
plt.title('Bar Chart of Age vs Height/Weight')
plt.xlabel('Age')
plt.ylabel('Height/Weight')
plt.legend()
plt.grid(axis='y')
plt.show()

# 4. Histogram
plt.figure(figsize=(8, 6))
plt.hist(age, bins=5, color='green', edgecolor='black')
plt.title('Histogram of Age')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.grid(axis='y')
plt.show()

```

1s completed at 7:55 PM

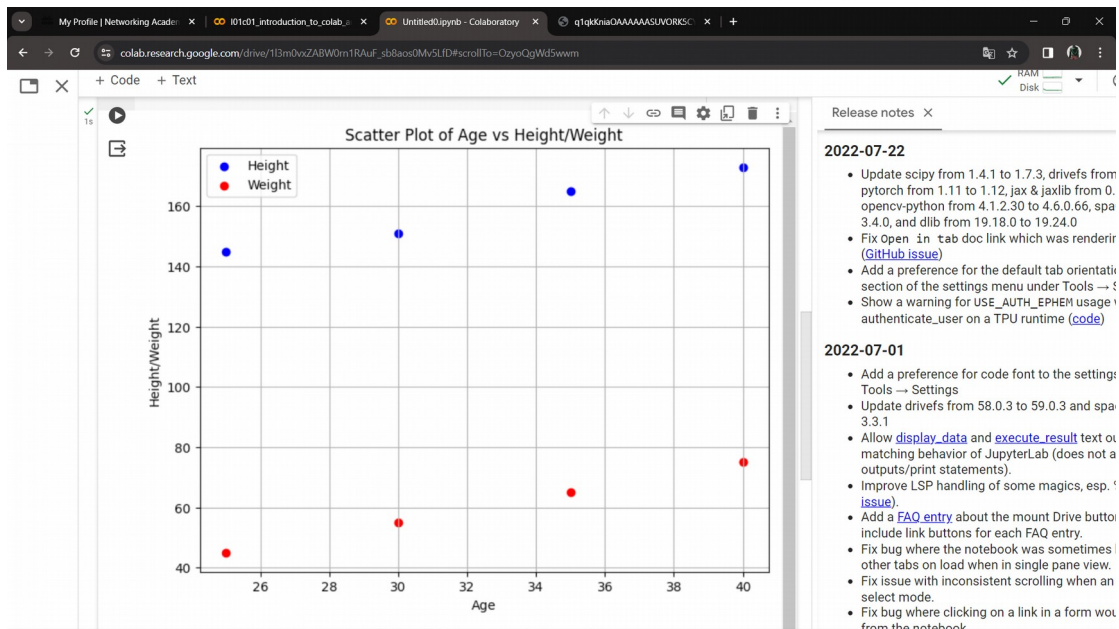
Release notes

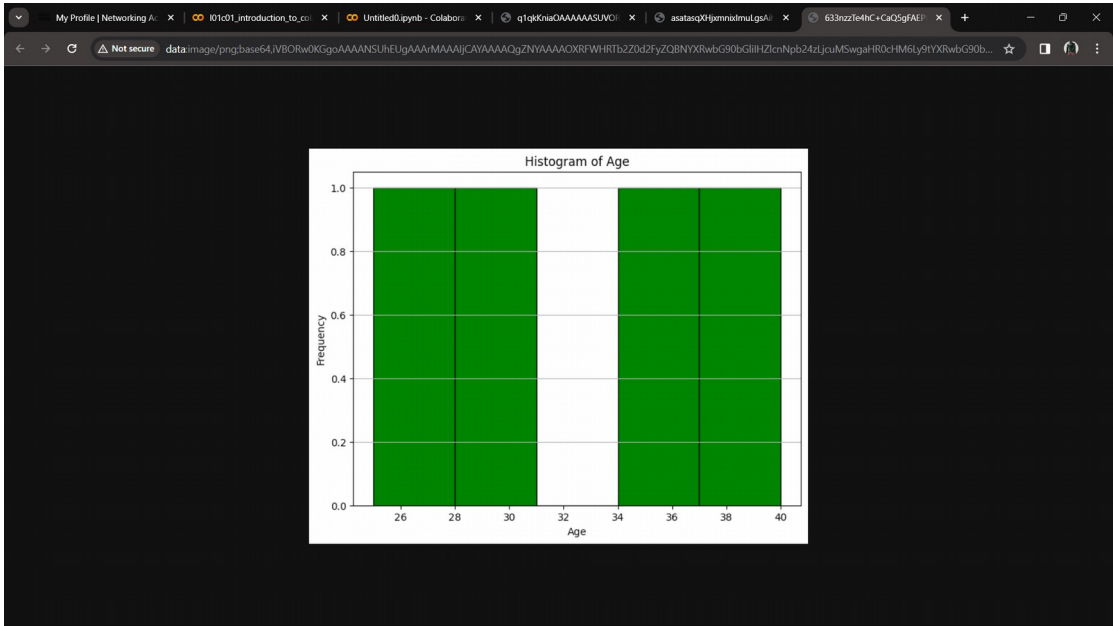
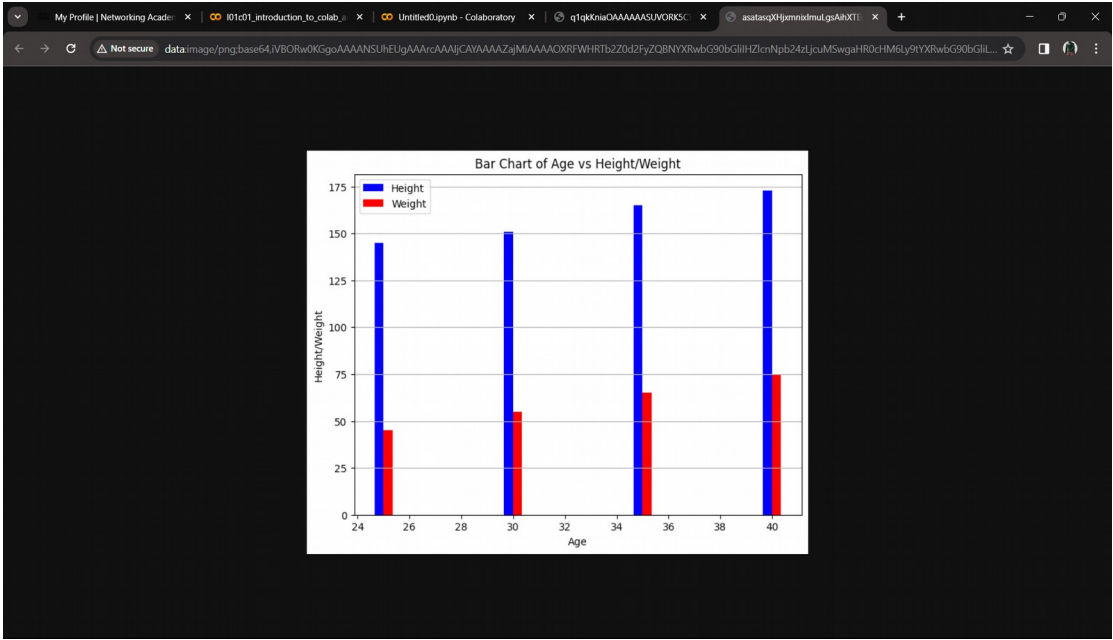
2022-07-22

- Update scipy from 1.4.1 to 1.7.3, drives from 59.0.3 to 60.0.2, pytorch from 1.11 to 1.12, jax & jaxlib from 0.3.8 to 0.3.14, opencv-python from 4.1.2.30 to 4.6.0.66, spaCy from 3.3.1 to 3.4.0, and dlib from 19.18.0 to 19.24.0
- Fix Open in tab doc link which was rendering incorrectly ([GitHub issue](#))
- Add a preference for the default tab orientation to the Site section of the settings menu under Tools → Settings
- Show a warning for USE_AUTH_EPHEM usage when running authenticate_user on a TPU runtime ([code](#))

2022-07-01

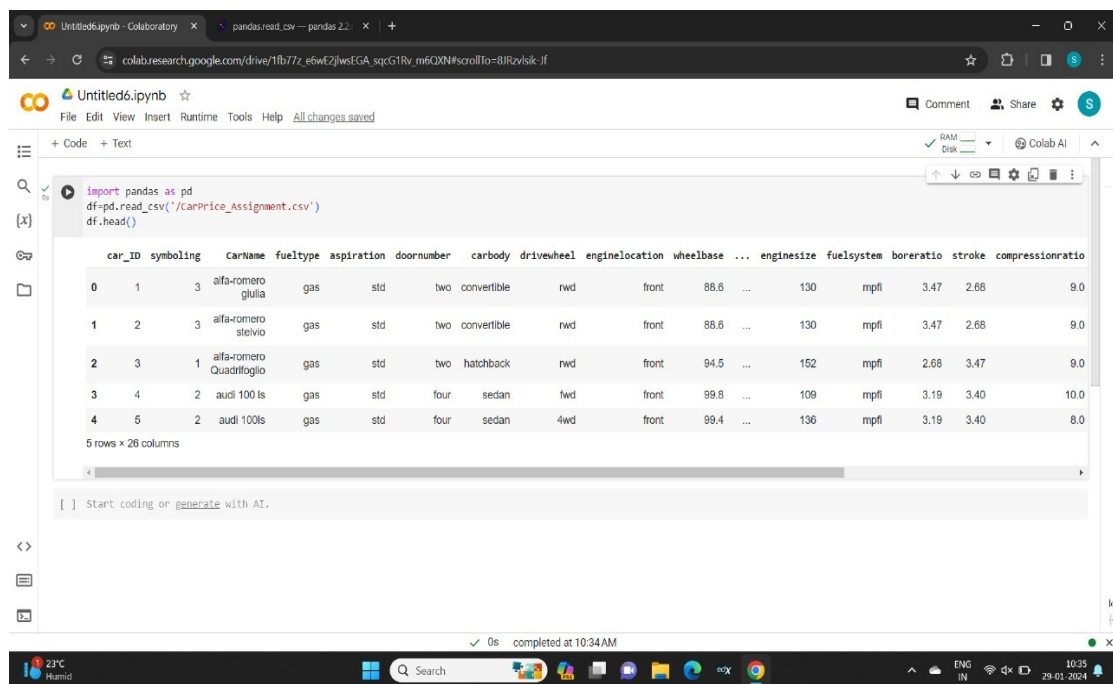
- Add a preference for code font to the settings menu under Tools → Settings
- Update drives from 58.0.3 to 59.0.3 and spacys from 2.2.4 to 3.3.1
- Allow `display_data` and `execute_result` text outputs to wrap, matching behavior of JupyterLab (does not affect stream outputs/print statements).
- Improve LSP handling of some magics, esp. `%writefile` ([GitHub issue](#)).
- Add a [FAQ entry](#) about the mount Drive button behavior and include link buttons for each FAQ entry.
- Fix bug where the notebook was sometimes hidden behind other tabs on load when in single pane view.
- Fix issue with inconsistent scrolling when an editor is in multi-select mode.
- Fix bug where clicking on a link in a form would navigate away from the notebook
- Show a confirmation dialog before performing Replace all from the Find and replace pane.





Part 3 – Access Data from Various Data Sources using builtin Function of Numpy, Pandas

1. Generate your own dataset using MS Excel and Notepad to prepare the dataset. Save it in Google Drive and access it in Google Colab
2. Upload the text file using tab separated value(.tsv) and access the data from the file.
3. Upload the text file using comma separated value(.csv) and access the data from file.
4. Access the excel file using .xlsx
5. Access the text from the URL



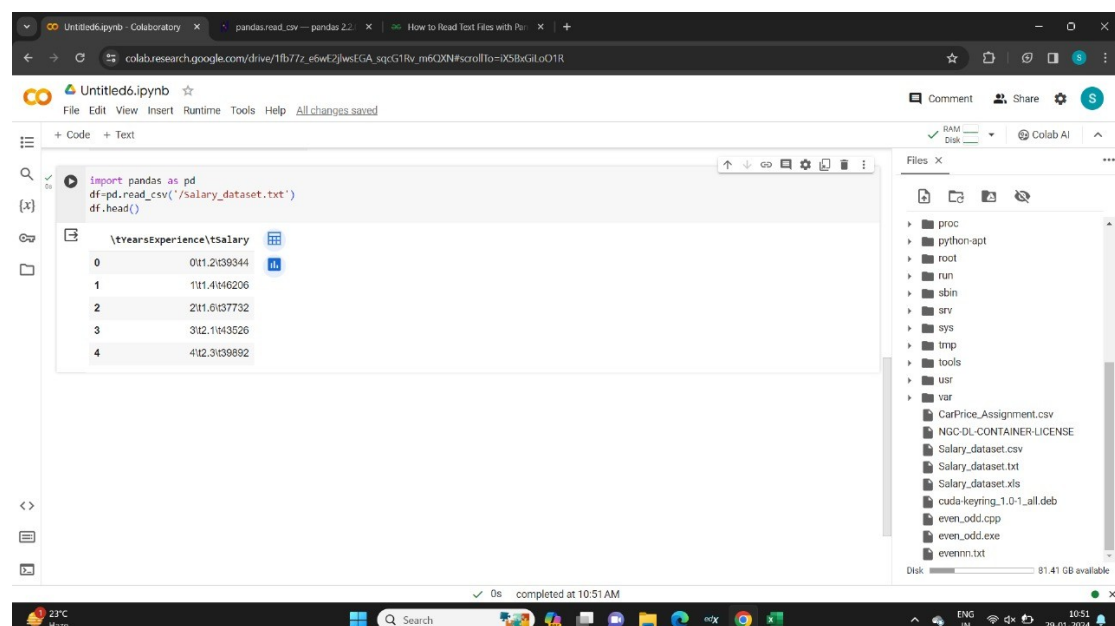
The screenshot shows a Google Colab environment with a Jupyter Notebook. The code cell contains the following Python code:

```
import pandas as pd
df=pd.read_csv('/CarPrice_Assignment.csv')
df.head()
```

The output of the code is a preview of the first five rows of the CSV file. The data is as follows:

	car_ID	symboling	CarName	fueltype	aspiration	doornumber	carbody	drivewheel	engineLocation	wheelbase	...	engineSize	fuelsystem	boreRatio	stroke	compressionratio
0	1	3	alfa-romero glulia	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0
1	2	3	alfa-romero stelvio	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0
2	3	1	alfa-romero Quadrifoglio	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0
3	4	2	audi 100 ls	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0
4	5	2	audi 100ls	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0

The notebook interface shows the code cell with a green checkmark, indicating successful execution. The output is displayed below the code cell. The bottom status bar shows the notebook was completed at 10:34 AM.



The screenshot shows a Google Colab environment with a Jupyter Notebook. The code cell contains the following Python code:

```
import pandas as pd
df=pd.read_csv('/Salary_dataset.txt')
df.head()
```

The output of the code is a preview of the first five rows of the TSV file. The data is as follows:

	yearsExperience	salary
0	0.11.2	39344
1	1.11.4	46206
2	2.11.6	37732
3	3.12.1	43526
4	4.12.3	39892

The notebook interface shows the code cell with a green checkmark, indicating successful execution. The output is displayed below the code cell. The bottom status bar shows the notebook was completed at 10:51 AM.

Untitled6.ipynb - Colaboratory

colab.research.google.com/drive/1fb77z_e5wt2jwstGA_sqrG1Rv_m6DXNpsrollto=8JRzVsk-if

Untitled6.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
import pandas as pd
df=pd.read_csv('/carPrice_Assignment.csv')
df.head()
```

car_id	symboling	CarName	fueltype	aspiration	doornumber	carbody	drivewheel	engineLocation	wheelbase	...	engineSize	fuelsystem	boreratio	stroke	compressionratio		
0	1	3	alfa-romero	glulia	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0
1	2	3	alfa-romero	stelvio	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0
2	3	1	alfa-romero	Quadrifoglio	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0
3	4	2	audi	100 ls	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0
4	5	2	audi	100ls	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0

5 rows x 26 columns

[] Start coding or generate with AI.

completed at 10:34AM

Assignment 2

colab.research.google.com/drive/1LD06kaWX-x_kkU0TYSYow-01AINWdms#scrollto=nAKGDMCaxdX

Untitled7.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
import pandas as pd
X=pd.read_excel("/content/salary_dataset.xls")
X.head()
```

Unnamed: 0	YearsExperience	Salary	
0	0	1.2	39344
1	1	1.4	46206
2	2	1.6	37732
3	3	2.1	43526
4	4	2.3	39892

Double-click (or enter) to edit

completed at 3:25PM

Assignment 2

AIML_A2.pdf

2203AS1235 AIML Assignment

Untitled7.ipynb - Colaboratory

Breadth First Search in Python

colab.research.google.com/drive/1LDO6kaWX-x_kkU0TYsYsw-o1AINWdms#scrollTo=nAKGDMCaxLx

Untitled7.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

[x] sample_data Salary_dataset.xls

URL

```
import pandas as pd
X=pd.read_excel("/content/Salary_dataset.xls")
X.head()
```

Unnamed: 0	YearsExperience	Salary
0	0	1.2 39344
1	1	1.4 46206
2	2	1.6 37732
3	3	2.1 43526
4	4	2.3 39892

Double-click (or enter) to edit

Disk 61.40 GB available

completed at 3:25 PM

33°C
Haze

Search

15:30
30-01-2024