



Intermediate Pandas 2

Tags	Pandas	Python
Class		
Finished Yet?	<input checked="" type="checkbox"/>	
Knowledge	The Ninth Sprint: Essential Python for Data Science	

Lesson 31: Describe

-เรามาระดูว่า Data Frame ได้ด้วย `.describe()`

```
[55] > 0.1s
#Summarize dataframe
penguins.describe(include = 'all')

Table Raw Visualize Statistics

```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
count	344	344	342.0	342.0	342.0	342.0	333
unique	3	3	nan	nan	nan	nan	2
top	Adelie	Biscoe	nan	nan	nan	nan	MALE
freq	152	168	nan	nan	nan	nan	168
mean	nan	nan	43.9219298245614	17.151169590643278	200.91520467836258	4201.754385964912	nan
std	nan	nan	5.4595837139265315	1.9747931568167816	14.061713679356888	801.9545356980956	nan
min	nan	nan	32.1	13.1	172.0	2700.0	nan
25%	nan	nan	39.225	15.6	190.0	3550.0	nan
50%	nan	nan	44.45	17.3	197.0	4050.0	nan
75%	nan	nan	48.5	18.7	213.0	4750.0	nan
max	nan	nan	59.6	21.5	231.0	6380.0	nan

-ถ้าอยากรู้แค่ค่าที่เราต้องการ เช่น ค่าเฉลี่ยของ column ชื่อ `bill_length_mm` เราสามารถเลือกเฉพาะ column ที่ต้องการ ตามด้วย Function ค่าสถิติที่ต้องการ (ในกรณีนี้คือ `.mean()`) ได้

Lesson 32: Group By

- เราสามารถจัดกลุ่มของข้อมูลได้ด้วย `.groupby("ชื่อ column ที่เราอยากระบุ")` ตามด้วย `column ที่เราอยากระบุ` คำนวนเชิงสถิติ และtabular ด้วย `Function เชิงสถิติ`

```
[59] ⏷ ▶ 0.1s
      #Group by + sum/mean
      penguins.groupby("species")['bill_length_mm'].mean()

      ⏷ Table Raw Visualize Statistics

      bill_length_mm
      Adelie    38.79139072847682
      Chinstrap 48.83382352941176
      Gentoo    47.50487804878049
```

Lesson 33: Group By Aggregation

- เราสามารถหาค่าสกัดมากกว่า 1 ค่าได้ด้วย .agg()

```
[61] ▶ 0.1s
#Group by aggregation
penguins.groupby("species")['bill_length_mm'].agg(['min', 'mean', 'median', 'std', 'max'])

Table Raw Visualize Statistics
```

	min	mean	median	std	max
Adelie	32.1	38.79139072847682	38.8	2.6634048483686197	46.0
Chinstrap	40.9	48.83382352941176	49.55	3.339255895935887	58.0
Gentoo	40.9	47.50487804878049	47.3	3.081857372114286	59.6

3 rows x 6 columns Jump to top Jump to bottom

- เรากำลังจัดกลุ่มได้มากกว่า 1 column

Attached data

- Notebook files
- Upload files (24.0 kB) /data/notebook_files/

Name	Date	Size
.private		
environment.yml	12 Feb 2023 07:01	110.0 B
penguins.csv	12 Feb 2023 07:03	13.2 kB
result.csv	13 Feb 2023 06:49	243.0 B

```
[64] ▶ 0.1s
#Group by aggregation
result = penguins.groupby(["island", "species"])['bill_length_mm'].agg(['min', 'mean', 'max']).reset_index()

result.to_csv('result.csv')
```

Add code cell

- Save file เป็น .csv ให้นำไปใช้ต่อได้

Attached data

- Notebook files
- Upload files (24.0 kB) /data/notebook_files/

result.csv					
	island	species	min	mean	max
0	Biscoe	Adelie	34.5	38.975	45.6
1	Biscoe	Gentoo	40.9	47.50487804878049	59.6
2	Dream	Adelie	32.1	38.50178571428571	44.1
3	Dream	Chinstrap	40.9	48.83382352941176	58.0
4	Torgersen	Adelie	33.5	38.958980392156865	46.0

Lesson 34: If Your Code is Long

- ในกรณีที่ Code ของเรายาวเกินไป เราสามารถพิมพ์ \ เพื่อขับบรรทัดใหม่ได้

```
▶ 0.1s
#If your code is long
penguins.groupby(["island", "species"])['bill_length_mm']\
    .agg(['min', 'mean', 'max'])\
    .reset_index()

Table Raw Visualize Statistics
```

	island	species	min	mean	max
0	Biscoe	Adelie	34.5	38.975	45.6
1	Biscoe	Gentoo	40.9	47.50487804878049	59.6
2	Dream	Adelie	32.1	38.50178571428571	44.1
3	Dream	Chinstrap	40.9	48.83382352941176	58.0
4	Torgersen	Adelie	33.5	38.950980392156865	46.0

Lesson 35: Map

-Map เป็นเทคนิคที่มีประโยชน์มากในการใช้งาน Pandas โดยจะรับค่าเป็น Dictionary {'ค่าเก่า' : 'ค่าใหม่'}

```

▶ 0.1s
#map values MALE: m, FEMALE: f
#penguins['sex'].head()
💡
penguins['sex'].map( {'MALE': 'm', 'FEMALE': 'f'} ).head(10).fillna('other')

```

Table Raw Visualize Statistics

	sex
0	m
1	f
2	f
3	other
4	f
5	m
6	f
7	m
8	other
9	other

- เราสามารถฝึกค่าไว้กับตัวแปรเพื่อสร้าง column ใหม่ที่มีค่าเท่ากับค่าที่เรา Map ไว้ได้

```

▶ 0.1s
#map values MALE: m, FEMALE: f
#penguins['sex'].head()

penguins['sex_new'] = penguins['sex'].map( {'MALE': 'm', 'FEMALE': 'f'} ).fillna('other')
penguins.head()

```

Table Raw Visualize Statistics

island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex	sex_new
Torgersen	39.1	18.7	181.0	3750.0	MALE	m
Torgersen	39.5	17.4	186.0	3800.0	FEMALE	f
Torgersen	40.3	18.0	195.0	3250.0	FEMALE	f
Torgersen	nan	nan	nan	nan	nan	other
Torgersen	36.7	19.3	193.0	3450.0	FEMALE	f

Lesson 36: Numpy

-Numpy = Numerical Python เป็น library ที่ใช้ในการคำนวณ computation ได้อย่างรวดเร็ว

```
[70] #Pandas style
      penguins['bill_length_mm'].mean()

      43.9219298245614

[71] ▶ 0.1s #Numpy style
      import numpy as np
      np.mean(penguins['bill_length_mm'])

      43.9219298245614
```

-Numpy มี Function ทางสถิติให้ใช้เยอะมาก

```
[73] ▶ 0.1s #Other functions of numpy
      print(np.sum(penguins['bill_length_mm']))
      print(np.std(penguins['bill_length_mm']))

      15021.3
      5.4515960231618195
```

Lesson 37: Where

-np.where = if-else

```
[74]
```

```
#Numpy where  
score = pd.Series( [80, 55, 62, 95, 20] )  
print(score)
```

```
0    80  
1    55  
2    62  
3    95  
4    20  
dtype: int64
```

```
[75]
```

```
np.where(score >= 80, "Passed", "Failed")
```

Table Raw Visualize Statistics

	Ab
0	Passed
1	Failed
2	Failed
3	Passed
4	Failed

- เราสามารถประยุกต์ใช้ np.where ในการสร้าง column ของ Data Frame ได้

```
[81] df = penguins.query("species == 'Adelie'")[['species', 'island', 'bill_length_mm']].dropna()

[82] df['new_column'] = np.where(df['bill_length_mm'] > 40, True, False) #Boolean

[83] ▶ 0.1s
df.head(10)
```

Table Raw Visualize Statistics

	species	island	bill_length_mm	new_column
0	Adelie	Torgersen	39.1	False
1	Adelie	Torgersen	39.5	False
2	Adelie	Torgersen	40.3	True
4	Adelie	Torgersen	36.7	False
5	Adelie	Torgersen	39.3	False

Lesson 38: Merge

-เราสามารถ JOIN ตาราง 2 อันได้ด้วย pd.merge() (เหมือน JOIN ใน SQL) เช่น:

```
result = pd.merge(left, right, on = 'key')
```

Merge Dataframes

```
▶ 0.1s
left = {
    'key': [1, 2, 3, 4],
    'name': ['toy', 'joe', 'jane', 'anna'],
    'age': [25, 28, 30, 22]
}

right = {
    'key': [1, 2, 3, 4],
    'city': ['Bangkok', 'London', 'Seoul', 'Tokyo'],
    'zip': [1001, 2504, 2094, 9802]
}

df_left = pd.DataFrame(left)
df_right = pd.DataFrame(right)
```

```
▶ 0.5s
import pandas as pd
pd.merge(df_left, df_right, on='key')
```

Table Visualize

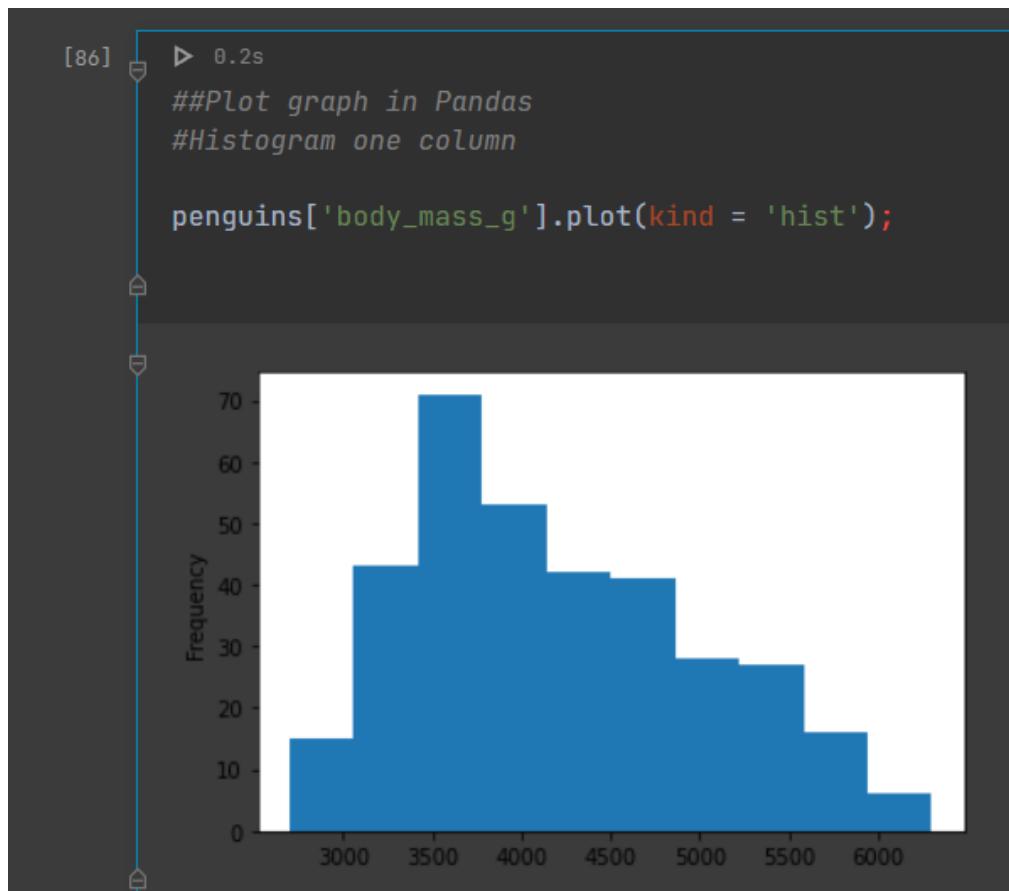
	key	name	age	city	zip
0	1	toy	25	Bangkok	1001
1	2	joe	28	London	2504
2	3	jane	30	Seoul	2094
3	4	anna	22	Tokyo	9802

4 rows x 5 columns

Lesson 39: Intro to Pandas Plots

-Pandas สามารถสร้าง Plot เบื้องต้นได้

-Histogram 1 column:

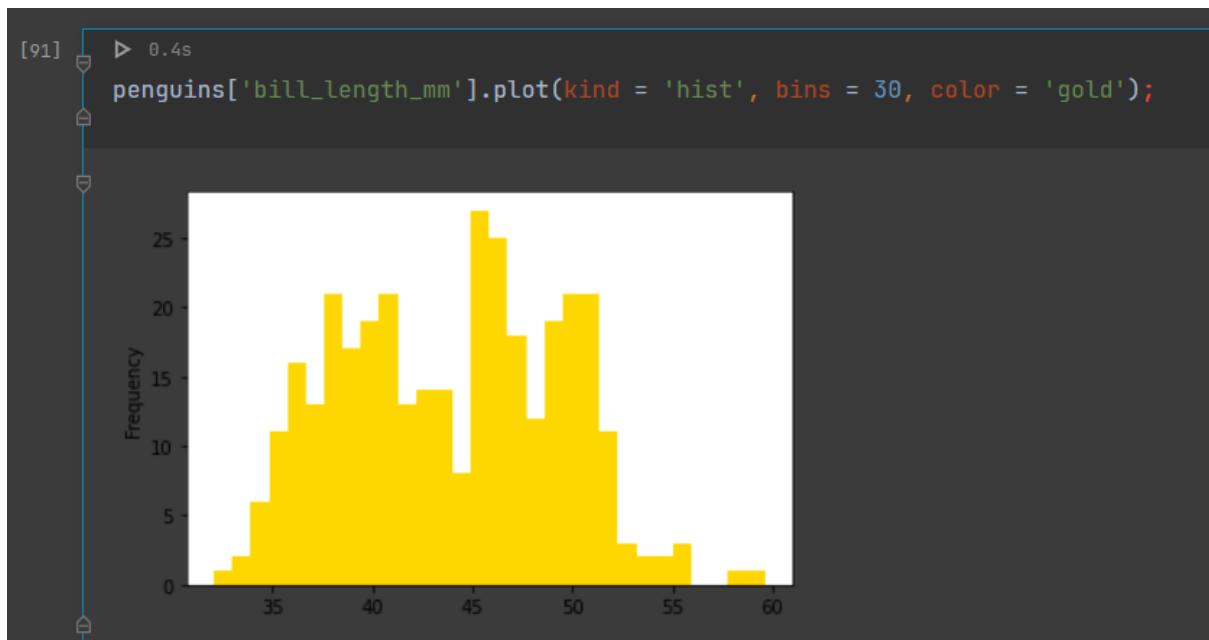


-Histogram 2 columns:

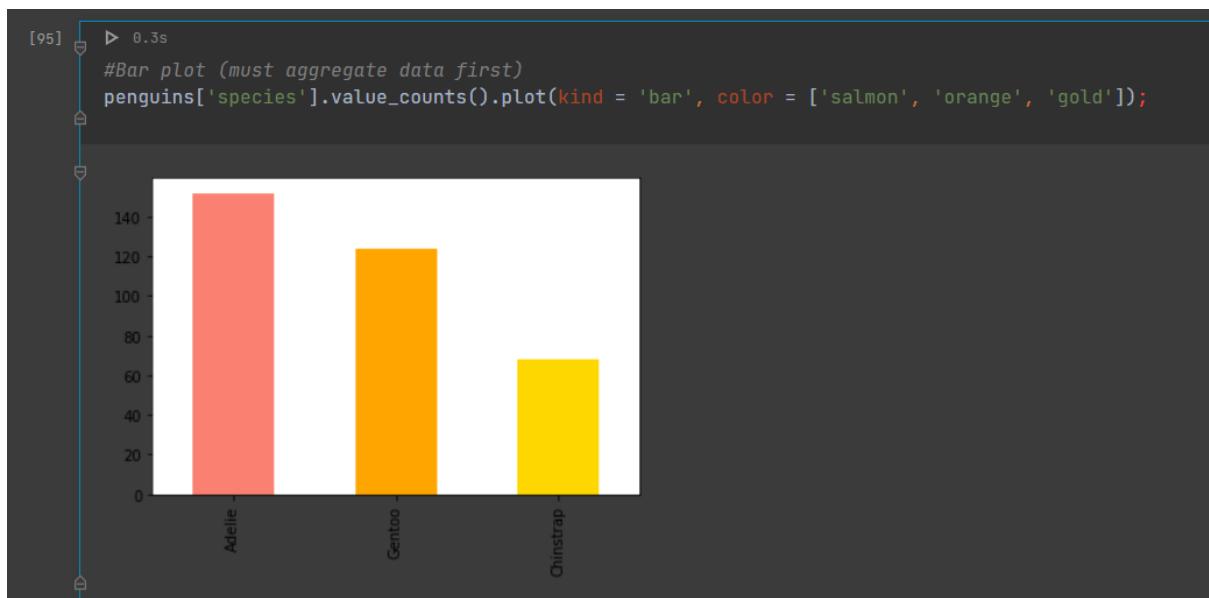


*ในบางครั้งอาจต้องทำ Normalization ข้อมูลก่อน เพื่อให้ใช้งานด้วยกันได้

-เราสามารถเปลี่ยนสีของกราฟที่เราสร้างได้



-การสร้าง Bar Graph จำเป็นต้องสรุปผลข้อมูลก่อน ถึงจะสามารถสร้างกราฟได้



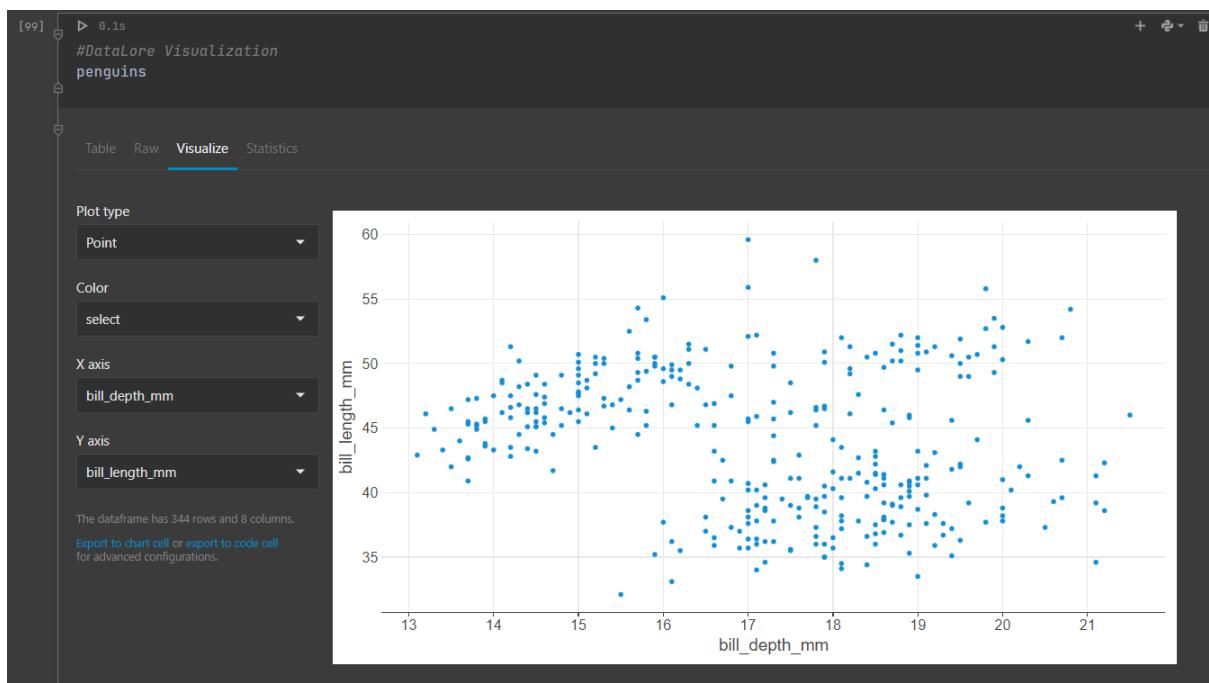
-เราสามารถสร้าง Scatter Plot เพื่อดูความสัมพันธ์ของข้อมูลได้

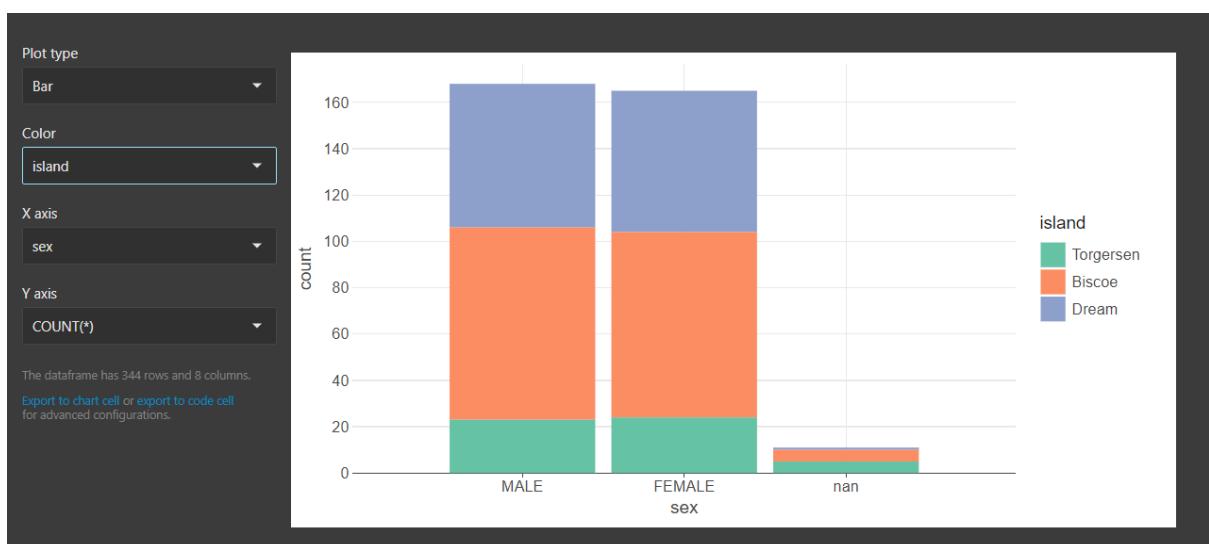
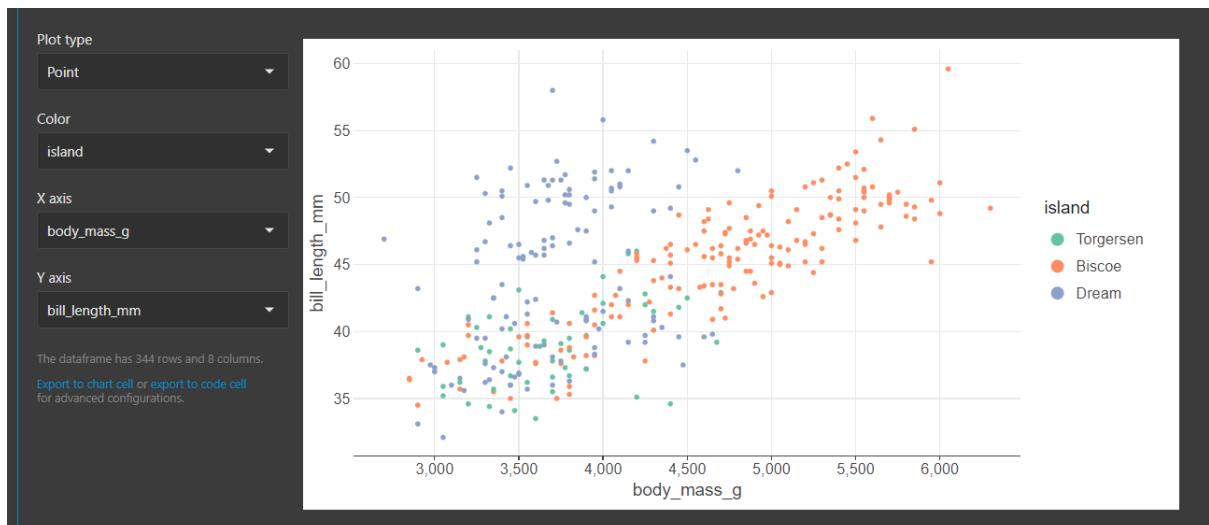


*จริง ๆ Python จะมี Library ที่ใช้ในการสร้าง Plot เช่น Matplotlib หรือ Seaborn อยู่ และ Pandas ก็สามารถสร้าง Plot ได้หลากหลายแบบในตัวเองเช่นกัน

Lesson 40: DataLore Visualization

-เราสามารถสร้าง Chart ใน DataLore ได้โดยไม่ต้องเขียนโค้ด เลือกรูปแบบของ Chart, สี, ตัวแปรแกน x และตัวแปรแกน y ได้ตามใจชอบ





My DataLore Notebook (Intermediate Pandas 1 + 2):

<https://datalore.jetbrains.com/notebook/bK6ww1mlzO8jOk6ASOvLAG/AQRq08OBVYqgm325Fu1tGx/>

[Pandas Final Project]

CSV File:

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/34963761-315e-4e28-9652-e176bfef0db7/Untitled.csv>

Final Project Link:

<https://datalore.jetbrains.com/notebook/bK6ww1mlzO8jOk6ASOvLAG/nZFrcYYqeRy1MoXH8xByRk/>