Pandas

import pandas as pd

* df[columns][rows]
└ dataframe

- นำเข้า Data : pd.read_csv('....path...')
  (excel)
- link google drive: from google.colab drive
  drive.mount('...path...')
  pd.read_csv('ใช้ path ใน google drive ได้เลย')

- .head(x) ดูหัว-ท้าย ตาราง x แถวแรก
  .tail(x)
- .shape => ดูขนาด กี่แถว กี่หลัก?
- .describe() => ข้อมูลเบื้องต้น

- .columns => เรียกดูคอลัมน์ทั้งหมด
- df[[col1, col2, ... coln]] จัดลำดับตามที่เราเลือก
  ↑ ชื่อชุดข้อมูล
  [x : y] "เอาแถวที่ x-y"
  มี [ ] คลุมอีกรอบ จะแสดงเป็นตาราง

ดูผลของแบบ Array
ดูผลของแบบ table

	.loc[] - By label	.iloc[] - By Index (ใช้ลำดับ ตัวเลข)
Select Single Row	df.loc['r2']	df.iloc[1]
" Single Column	df.loc[:,"_"]	df.iloc[:,0]
" Multiple Row	df.loc[['r2','r3']]	df.iloc[[1,2]]
" Multiple columns	df.loc[:,['_','_']]	df.iloc[:,[0,1]]
" Rows Range	df.loc['r1':'r4']	df.iloc[0:4]
" Columns Range	df.loc[:,'E':'_']	df.iloc[:,1:4]
" Alternate Row	df.loc['r1':'r4':1]	df.iloc[0:4:1]
" Alternate Column	df.loc[:,'_':'_':1]	df.iloc[:,1:4:1]
Using Condition	df.loc[df['_'] >= ...]	df.iloc[list(df['_'] >= ...)]
Using lambda function	df.loc[lambda x: x[3]]	df.iloc[lambda x: x[3]]

if-elif-else

if เงื่อนไขที่จะตรวจสอบ ใช้ ==, !=, <, >, <=, >=, in, is_null() :
    ถ้าเป็นจริง ทำงานที่ 1
elif เงื่อนไขที่จะตรวจสอบต่อไปถ้าเงื่อนไขข้างบน ไม่เป็นจริง :
    ถ้าเป็นจริง ทำงานที่ a1
else : *ตรวจสอบเงื่อนไขข้างบนแล้วไม่เป็นจริง
    ทำงานที่ x

Ex

```
import numpy as np

score = np.random.uniform(60,100)
print("score=", np.round(score,2))

if score > 90: print('A')
elif score > 80: print('B')
else: print('C')
```

▶ Loop : for

```
for ตัวแปรที่เก็บแต่ละค่าจากลิสต์ in ลิสต์ที่ต้องการจะวนลูป :
    do something
    do another thing
this is outside the loop
```

Ex
```
for x in food:
    print("I like", x)
```
↳ I like sushi

▶ Loop : for, while

```
while true_condition :
    do something
```

Ex
```
max_count = 5
count = 0
while (count < max_count):
    print(count)
    count += 1
```

▶ Function

```
def ชื่อฟังก์ชัน (input1, input2, .....) :
    do something
    do another thing
    return output
```

- set(df['col_name'])

"ดูค่าใน col_name โดยที่ไม่ซ้ำกัน"

Ex

	col_name	set(['col_name'])
1	2	1
2	3	2
1	9	3
1	1	9
2		
3		
9	1	
1	9	

▶ Missing value checking

- df.isnull() ※ True = missing value
- " .all() เอาค่าความจริงใน column มา AND กัน (เช็คว่าทั้งคอลัมน์เป็นค่า Null หรือไม่?)
- " .any() เอาค่าความจริงใน column มา OR กัน (เช็คว่าในคอลัมน์มี Null?)
- " .any().any() เช็คว่าทั้ง df มีค่า null หรือไม่?

* .dropna() ลบ record ที่มี Missing value
↳ subset = ['col_name'] "ดูเฉพาะค่าที่อยู่ใน column __ (ลบเฉพาะที่อยู่ใน col_name)"

* update ข้อมูล inplace : .dropna(inplace = True) ← ลบไปแล้วเลย
False ลบแล้วข้อมูลยังอยู่

* แทน missing ด้วยค่าที่เหมาะสม : .fillna('_', inplace = True)
↑ ค่าแทนที่ข้อมูล

ต่อตาราง

- ต่อในแนวแกน Y (เพิ่มจำนวนข้อมูล)

```
pd.concat([df1, df2])
```

- ต่อในแนวแกน X (เพิ่มจำนวน/รายละเอียดข้อมูล)
  Column

```
df1['__'] = df1['FK'].map(df2.set_index('PK')['ให้แสดงชื่ออะไร'])
```

ชื่อใหม่ (ไม่ซ้ำกัน)

Group by

```
df.groupby('col_name')
```

ใช้ร่วมกับ (ไม่ต้องเขียน)

- .sort_values(by= __ , ascending = True น้อย -> มาก / False มาก -> น้อย)   ← เรียงโดยใช้ column ใดเป็นเกณฑ์
- .max() .min() mean()
- .head()
- .reset_index => ทำให้อยู่ในรูปตาราง?
  จัดเรียง Index ใหม่ Ex. หลังจากตัดข้อมูลแล้ว อยากเรียง Index ใหม่