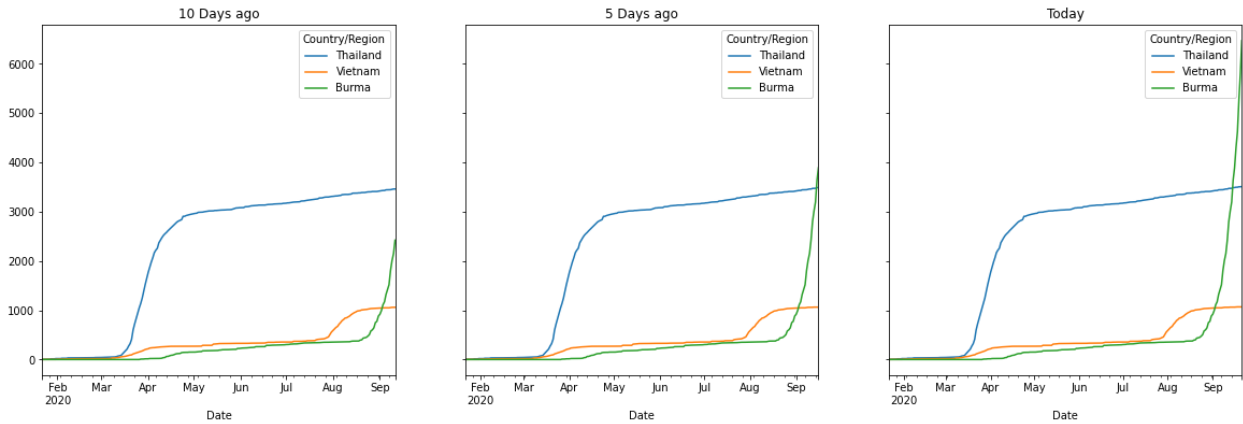


Assignment 7: Covid-19

สถานการณ์ Covid 19 ทั่วโลกเป็นสถานการณ์ที่เปลี่ยนแปลงตลอดเวลาและผู้คนทั่วโลกต่างให้ความสนใจ ซึ่ง ณ วันนี้สถานการณ์ในประเทศไทยเมื่อนำมาไม่น่าไว้วางใจ ตามกราฟด้านล่าง (ซึ่งพล็อตด้วยการเขียนโปรแกรมภาษา python) จะเห็นว่า สถานการณ์เปลี่ยนไปอย่างรวดเร็ว โดยจำนวน confirmed cases ตามเส้นสีเขียว พุ่งขึ้นสูงกว่าประเทศไทยและเวียดนามที่พล็อตอยู่ด้วยกัน



ข้างล่างนี้คือฟังก์ชันที่สามารถใช้เพื่อดึงข้อมูลจำนวน confirmed cases ของประเทศต่าง ๆ มีรายละเอียดดังนี้

```
import pandas as pd

def get_data():
    csv = 'https://raw.githubusercontent.com/CSSEGISandData/COVID-19/' + \
          'master/csse_covid_19_data/csse_covid_19_time_series/' + \
          'time_series_covid19_confirmed_global.csv'
    df=pd.read_csv(csv)
    df=df.groupby('Country/Region').sum()
    df=df.T.iloc[2:]
    return df

def get_confirmed_cases(df, country):
    return list(df[country])

def get_all_countries(df):
    return list(df.columns)
```

1. `get_data()` คืนฐานข้อมูล Covid-19 ของมหาวิทยาลัย John Hopkins ในรูปแบบ DataFrame (ไม่ต้องสนใจว่า DataFrame คืออะไร ให้คิดว่าเป็นที่เก็บข้อมูลแบบหนึ่ง) เราต้องใช้ข้อมูลแบบ DataFrame ที่คืนจาก `get_data()` นี้ส่งให้อีก 2 ฟังก์ชัน
2. `get_all_countries(df)` รับ DataFrame และคืนลิสต์ของชื่อประเทศทั้งหมดที่เก็บใน df
3. `get_confirmed_cases(df, country)` รับ DataFrame และชื่อประเทศ และคืนลิสต์ของจำนวน confirmed cases สะสม ในแต่ละวันของประเทศนั้น (ข้อมูลเริ่มตั้งแต่วันที่ 22 มกราคม 2563)

เช่น ขณะที่เขียนโจทย์นี้ (คือวันที่ 23 กันยายน 2563) ถ้าเรียกใช้ชุดคำสั่งข้างล่างนี้

```
df = get_data()
print(len(get_all_countries(df)), 'countries')
c = 'Burma'
ncc = get_confirmed_cases(df, c)
print(ncc)
```

จะได้ผลดังนี้

```
188 countries
Burma
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 8.0, 8.0, 10.0, 14.0, 15.0,
 15.0, 20.0, 20.0, 21.0, 21.0, 22.0, 22.0, 22.0, 23.0, 27.0,
 38.0, 41.0, 62.0, 63.0, 74.0, 85.0, 88.0, 98.0, 111.0, 119.0,
 121.0, 123.0, 139.0, 144.0, 146.0, 146.0, 146.0, 150.0, 150.0, 151.0,
 151.0, 151.0, 155.0, 161.0, 161.0, 161.0, 176.0, 177.0, 178.0, 180.0,
 180.0, 180.0, 181.0, 181.0, 182.0, 182.0, 184.0, 191.0, 193.0, 199.0,
 199.0, 199.0, 201.0, 201.0, 203.0, 206.0, 206.0, 206.0, 207.0, 224.0,
 224.0, 228.0, 232.0, 233.0, 236.0, 236.0, 240.0, 242.0, 244.0, 246.0,
 248.0, 260.0, 261.0, 261.0, 261.0, 262.0, 262.0, 262.0, 286.0, 286.0,
 287.0, 290.0, 291.0, 292.0, 293.0, 293.0, 293.0, 296.0, 299.0, 299.0,
 299.0, 303.0, 304.0, 306.0, 313.0, 313.0, 316.0, 316.0, 317.0, 321.0,
 326.0, 330.0, 331.0, 336.0, 337.0, 337.0, 339.0, 339.0, 340.0, 341.0,
 341.0, 341.0, 343.0, 343.0, 346.0, 348.0, 350.0, 350.0, 351.0, 351.0,
 353.0, 353.0, 353.0, 353.0, 355.0, 356.0, 357.0, 357.0, 359.0, 359.0,
 360.0, 360.0, 360.0, 361.0, 369.0, 374.0, 374.0, 375.0, 376.0, 376.0,
 394.0, 399.0, 435.0, 441.0, 450.0, 474.0, 504.0, 580.0, 602.0, 643.0,
 749.0, 775.0, 882.0, 919.0, 995.0, 1111.0, 1171.0, 1319.0, 1419.0, 1518.0,
 1807.0, 2009.0, 2150.0, 2422.0, 2796.0, 3015.0, 3195.0, 3636.0, 3894.0, 4299.0,
 4621.0, 5263.0, 5805.0, 6471.0]
```

สิ่งที่ต้องการหา คือ มีประเทศใดบ้างที่มีสถานการณ์จำนวนการเพิ่มของ confirmed cases ใกล้เคียงกับประเทศที่เราสนใจ

- กำหนดให้ t คือจำนวน confirmed cases เริ่มต้น และ n คือจำนวนวัน สถานการณ์ที่วุ่นวายนี้คือ ค่าความชันของกราฟจำนวน confirmed cases เมื่อเริ่มมีผู้ติดเชื้อมากกว่า t คน โดยใช้ข้อมูลอีกจำนวน n วัน (ถัดจากวันที่เริ่มมีผู้ติดเชื้อมากกว่า t คน) ในการคำนวณค่าความชัน
- เช่น จากตัวอย่างของประเทศเมียนมาข้างบนนี้ ถ้าให้ $t = 1000$, $n = 7$ จะได้ข้อมูลพื้นที่สี่เหลี่ยมข้างบนนี้ มีความชัน = $\frac{2150 - 1111}{7} = 148.43$ (ขอคำนวณความชันแบบง่าย ๆ)
- คำถามคือ มีประเทศใดบ้างที่มีสถานการณ์ใกล้เคียงแบบนี้ คือมีค่าความชันเมื่อเริ่มมีจำนวน confirmed cases เกิน t คน (โดยใช้ข้อมูลอีก n วันถัดไป) ใกล้เคียงกับของประเทศเมียนมาที่สุดบ้าง
- ถ้าได้ลองค้นดูจะพบว่า ข้อมูลของประเทศนอร์เวย์ ข้างล่างนี้ มีความชันใกล้เคียงกับของประเทศเมียนมามากที่สุด คือมีค่าความชันเป็น $\frac{2118 - 1090}{7} = 146.86$ (อันนี้คือข้อมูลขณะเขียนโจทย์ อาจเปลี่ยนแปลงได้)

```
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
 0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 1.0, 6.0, 15.0, 19.0,
 25.0, 32.0, 56.0, 87.0, 108.0, 147.0, 176.0, 205.0, 400.0, 598.0,
 702.0, 996.0, 1090.0, 1221.0, 1333.0, 1463.0, 1550.0, 1746.0, 1914.0, 2118.0,
 2385.0, 2621.0, 2863.0, 3084.0, 3369.0, 3755.0, 4015.0, 4284.0, 4445.0, 4641.0, ... ]
```

งานของคุณ

จากโปรแกรมต้นฉบับข้างล่างนี้ จงเขียนชุดคำสั่งให้กับฟังก์ชัน 3 ฟังก์ชันในบริเวณสี่เหลี่ยม (20 คะแนน ไม่มีโบนัส)

```
# Prog-07: COVID-19
# Fill in your ID & Name
# ...
# Declare that you do this by yourself
import pandas as pd

def get_data():
    return pd.read_pickle('covid19_df.pickle')
    csv = 'https://raw.githubusercontent.com/CSSEGISandData/COVID-19/' + \
          'master/csse_covid_19_data/csse_covid_19_time_series/' + \
          'time_series_covid19_confirmed_global.csv'
    df=pd.read_csv(csv)
    df=df.groupby('Country/Region').sum()
    df=df.T.iloc[2:]
    return df

def get_confirmed_cases(df, country):
    return list(df[country])

def get_all_countries(df):
    return list(df.columns)

#-----

def steepest_countries(df, t, n):

def most_similar_countries(df, country, t, n):

def top_k_similar_countries(df, country, t, n, k):

#-----
def main():
    df=get_data()
    print(steepest_countries(df, 1000000, 10))
    print(steepest_countries(df, 10000, 10))
    print(steepest_countries(df, 100, 10))
    print(most_similar_countries(df, 'Burma', 1000, 7))
    print(top_k_similar_countries(df, 'Cyprus', 1055, 5, 4))
#-----
main()
```

[download](#) เพิ่ม covid19_df.pickle ไว้ใน folder เดียวกับโปรแกรม
หากต้องการอ่านข้อมูลปัจจุบันจากแหล่งข้อมูลอินเทอร์เน็ต ให้ comment บรรทัดนี้ โดยเติม # ไว้หน้าบรรทัดนี้

download code นี้ได้

ไม่เพิ่ม ลบ หรือ เปลี่ยนแปลง
บริเวณคำสั่ง สีแดง เด็ดขาด

เขียนชุดคำสั่งในฟังก์ชันเท่านั้น
จะเขียนฟังก์ชันอื่นเพิ่มเติมในบริเวณนี้ก็ได้อีก

ไม่ใช่ตัวแปรใด ๆ
ที่อยู่นอกฟังก์ชัน

- `steepest_countries(df, t, n)`
 - รับ df เป็น DataFrame, t และ n คือจำนวนเต็ม
 - คืนลิสต์ของชื่อประเทศ ที่มีค่าความชันมากที่สุดจากวันที่เริ่มมีจำนวนผู้ติดเชื้อเกิน t คน ไปอีก n วัน
- `most_similar_countries(df, country, t, n)`
 - รับ df เป็น DataFrame, country เป็นชื่อประเทศ, t และ n คือจำนวนเต็ม
 - คืนลิสต์ของชื่อประเทศ ที่มีค่าความชันใกล้เคียงของประเทศ country มากที่สุด
 - ค่าความชันคำนวณจากวันที่เริ่มมีจำนวนผู้ติดเชื้อเกิน t คน ไปอีก n วัน
- `top_k_similar_countries(df, country, t, n, k)`
 - รับ df เป็น DataFrame, country เป็นชื่อประเทศ, t และ n คือจำนวนเต็ม
 - คืนลิสต์ของชื่อประเทศ ที่มีค่าความชันใกล้เคียงของประเทศ country มากที่สุด k ประเทศแรก
 - ค่าความชันคำนวณจากวันที่เริ่มมีจำนวนผู้ติดเชื้อเกิน t คน ไปอีก n วัน

ในกรณีที่มีค่าความชันที่เท่ากันหลายประเทศ คือ

- ชันมากที่สุดเท่ากัน ในกรณีของ `steepest_countries`,
- ชันใกล้เคียงที่สุดเท่ากัน ในกรณีของ `most_similar_countries`
- ชันใกล้เคียงมากที่สุด k ประเทศ ประเทศที่ k เท่ากัน ในกรณีของ `top_k_similar_countries`

ให้นำทุกประเทศที่ความชันเท่ากันที่อยู่ในเกณฑ์ เก็บในลิสต์ผลลัพธ์ด้วย เช่น

`top_k_similar_countries(df, 'Cyprus', 1055, 5, 3)` ได้มา 3 ประเทศคือ

`['Jordan', 'Lesotho', 'Liberia']`

แต่ถ้าขยายเป็น `top_k_similar_countries(df, 'Cyprus', 1055, 5, 4)` ได้มา 7 ประเทศ คือ

`['Jordan', 'Lesotho', 'Liberia', 'Andorra', 'Georgia', 'Jamaica', 'Togo']`

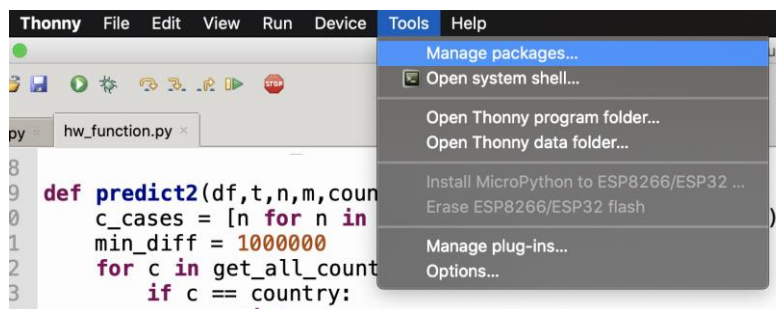
นั่นเป็นเพราะ 4 ประเทศหลังมีความชันเท่ากัน ก็เลยต้องรวมไว้ในผลให้หมด

ปฏิบัติตัวอย่างเคร่งครัด

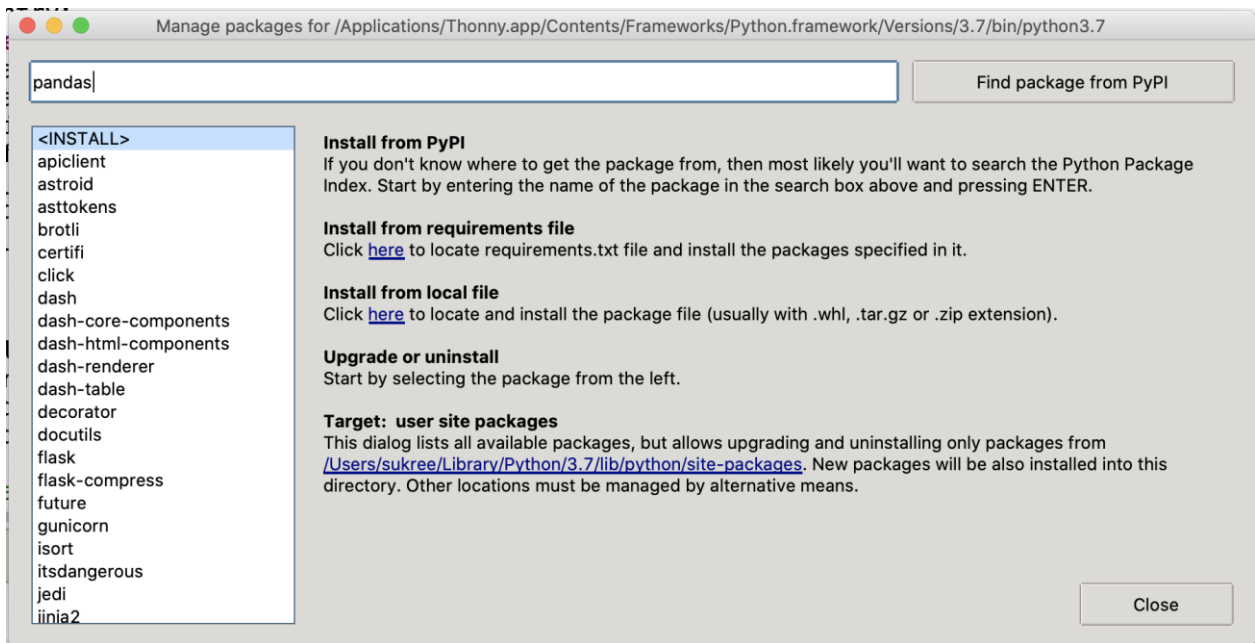
- ห้ามเขียนคำสั่งในฟังก์ชันที่ไปใช้ตัวแปรที่ยอยู่นอกฟังก์ชัน (หรือที่เรียกว่าตัวแปร global)
- อนุญาตให้เพิ่มคำสั่งในบริเวณพื้นที่สีขาวเท่านั้น จะเขียนฟังก์ชันเพิ่มเติมก็ได้
- ตั้งแต่ชื่อแฟ้ม ให้ถูกต้องตามที่เขียนใน CourseVille
- เพิ่มเติมข้อมูลใน `comment` ต้นโปรแกรมให้ตรงตามความจริง
- ก่อนส่งโปรแกรม ควรส่งงานโปรแกรมที่ส่งอีกครั้ง ว่าทำงานได้ตามที่ต้องการ

ข้อแนะนำ

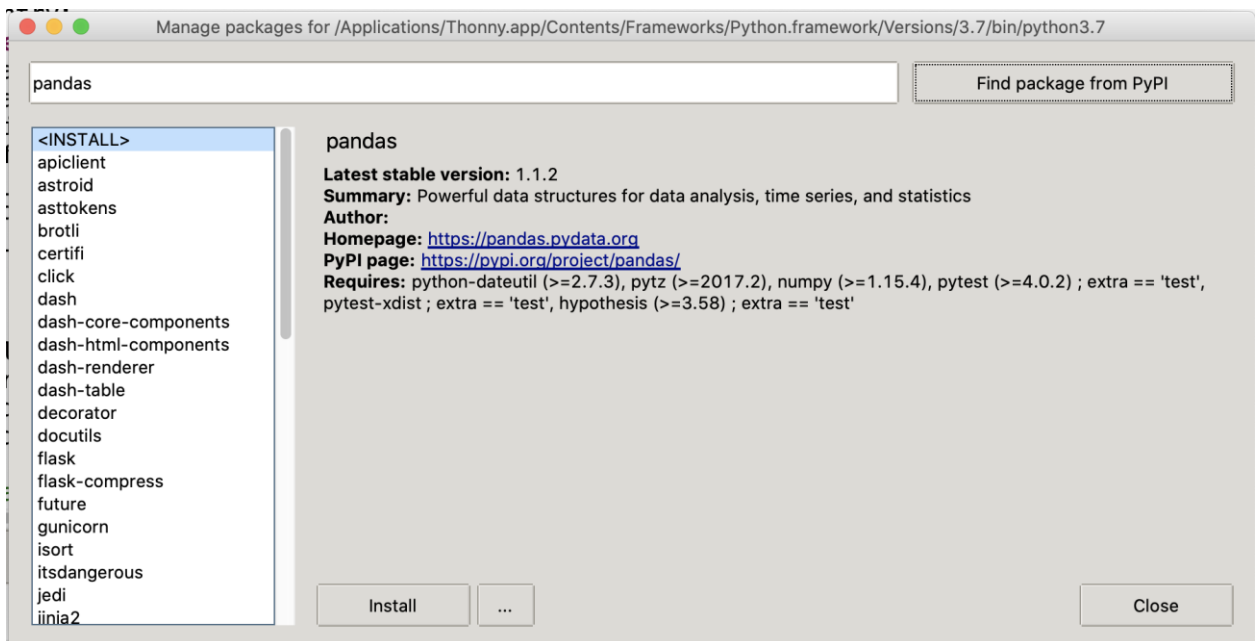
ก่อนจะทำโจทย์ได้ ต้องติดตั้ง package ที่ชื่อ `pandas` (`pandas` เก็บชุดคำสั่งวิเคราะห์ข้อมูลที่ได้รับความนิยมสูงมาก) ตามรูป (หรือเขียนโปรแกรมใน Google Colab ก็ได้ มี `pandas` ให้ใช้ได้เลย)



เลือกเมนู Manage packages



พิมพ์ pandas ลงใน dialog ที่ขึ้นมา จากนั้นคลิกที่ Find package from PyPI



จากนั้นในคลิก Install ก็จะสามารถเรียกใช้ library pandas ได้ตามต้องการ