5/19 Python 測驗

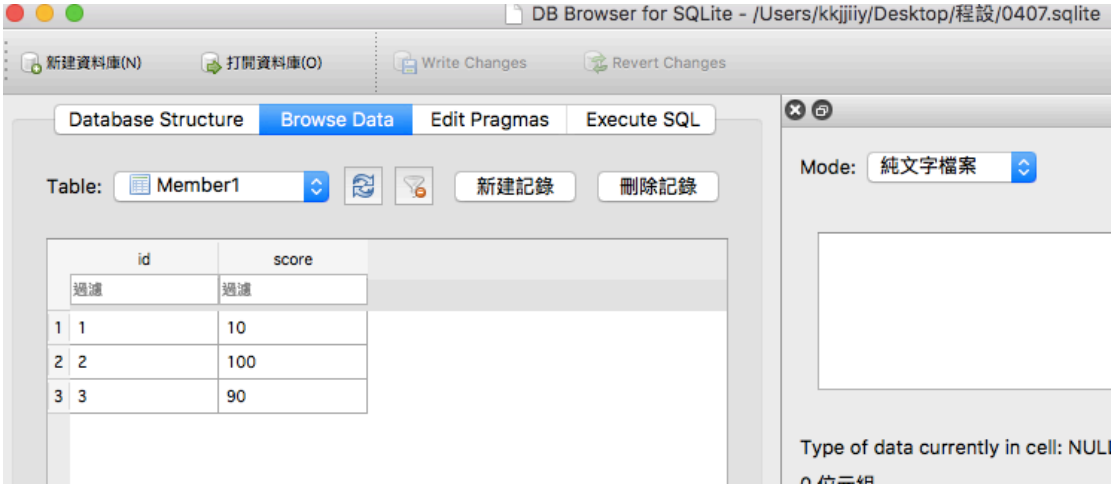
B10833012 湯敏摯

1. 請設計一組類別，此類別可以幫助使用者新增、修改以及刪除學生的成績， 並且透過屬性來記錄最佳成績。
   1. 請問這次成績如何? (使用者輸入數字，則程式碼會將這組自動存入資料庫)

      新增 id:3,成績:90

```
class Score:
    def __init__(self,i_d,score):
        self.iii = i_d
        self.sss = score
    def add(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "INSERT INTO Member1 Values (%d,%d)"%(self.iii,self.sss)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
i = int(input("請輸入id:"))
s = int(input("請輸入成績："))
s = Score(i,s)
s.add()
```

```
程設 — -bash — 80×24
~/Desktop/程設 — -bash    ~/Desktop/程設 — -bash
Last login: Tue May 19 18:10:57 on ttys000
tangminzhideAir:程設 kkjjiiy$ cd /Users/kkjjiiy/Desktop/程設
tangminzhideAir:程設 kkjjiiy$ python 0519.py
請輸入id:3
請輸入成績：90
tangminzhideAir:程設 kkjjiiy$
```

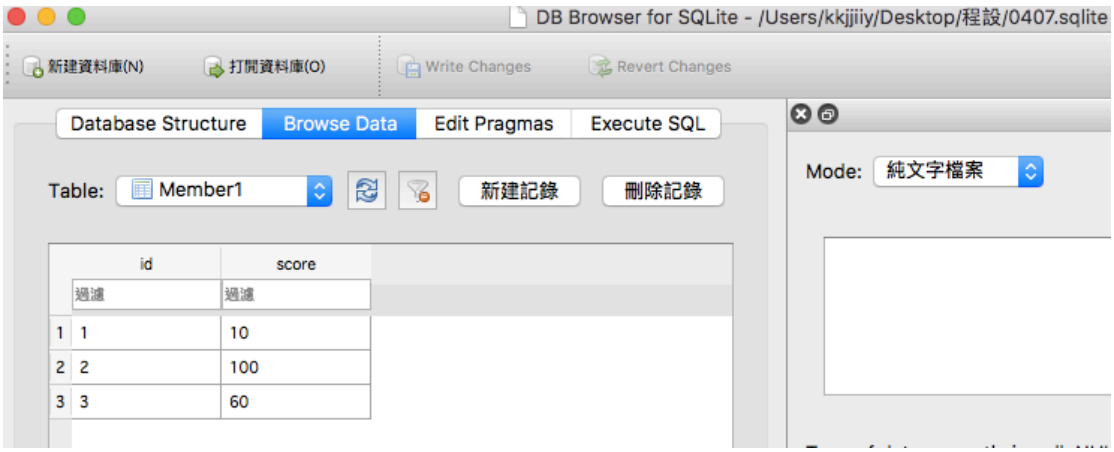


   2. 請輸入欲修改的學生 ID 以及該生的成績。(使用者輸入 ID 與 成績)

修改 id:3 的成績成 60

```python
class Score:
    def __init__(self,i_d,score):
        self.iii = i_d
        self.sss = score
    def add(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "INSERT INTO Member1 Values (%d,%d)"%(self.iii,self.sss)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def change(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "update Member1 set score='%d' where id='%d'"%(self.sss,self.iii)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
i = int(input("請輸入id:"))
s = int(input("請輸入欲修改成績："))
s = Score(i,s)
s.change()
```

```
程設 — -bash — 80×24
~/Desktop/程設 — -bash    ~/Desktop/程設 — -bash
Last login: Tue May 19 18:54:05 on ttys001
tangminzhideAir:程設 kkjjiiy$ cd /Users/kkjjiiy/Desktop/程設
tangminzhideAir:程設 kkjjiiy$ python 0519.py
請輸入id:3
請輸入欲修改成績：60
tangminzhideAir:程設 kkjjiiy$
```

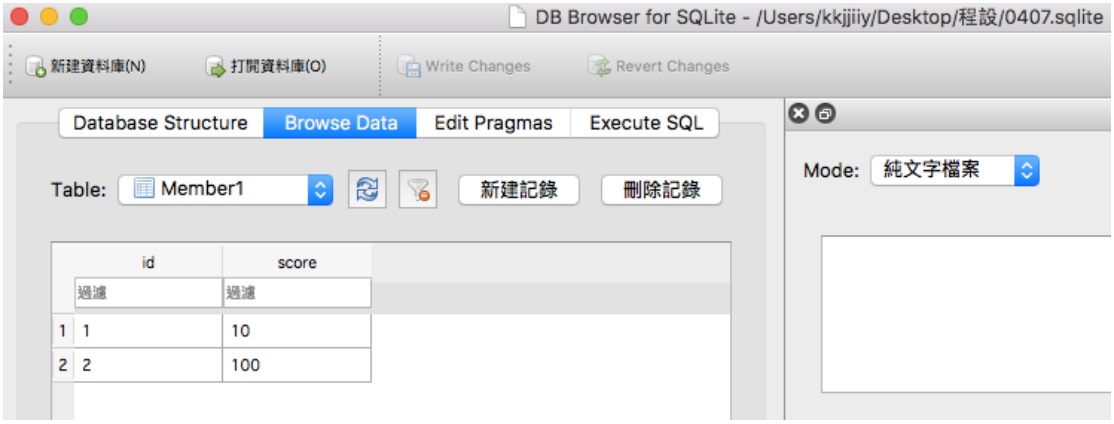


3. 請輸入欲刪除的學生 ID。(使用者輸入 ID)

刪除 id:3 的項目

```python
class Score:
    def __init__(self,i_d):
        self.iii = i_d
        # self.sss = score
    # def add(self):
    #     import sqlite3
    #     conn = sqlite3.connect('0407.sqlite')
    #     cursor = conn.cursor()
    #     sqlstr = "INSERT INTO Member1 Values (%d,%d)"%(self.iii,s
    #     cursor.execute(sqlstr)
    #     conn.commit()
    #     conn.close()
    # def change(self):
    #     import sqlite3
    #     conn = sqlite3.connect('0407.sqlite')
    #     cursor = conn.cursor()
    #     sqlstr = "update Member1 set score='%d' where id='%d'"%(s
    #     cursor.execute(sqlstr)
    #     conn.commit()
    #     conn.close()
    def delete(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "delete from Member1 where id='%d'"%(self.iii)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
i = int(input("請輸入欲刪除id:"))
# s = int(input("請輸入欲修改成績："))
s = Score(i)
s.delete()
```

```
程設 — -bash — 80×24
~/Desktop/程設 — -bash | ~/Desktop/程設 — -bash | ~/Desktop/程設 — -bash
Last login: Tue May 19 18:56:55 on ttys001
[tangminzhideAir:程設 kkjjiiy$ cd /Users/kkjjiiy/Desktop/程設
[tangminzhideAir:程設 kkjjiiy$ python 0519.py
請輸入欲刪除id:3
tangminzhideAir:程設 kkjjiiy$
```



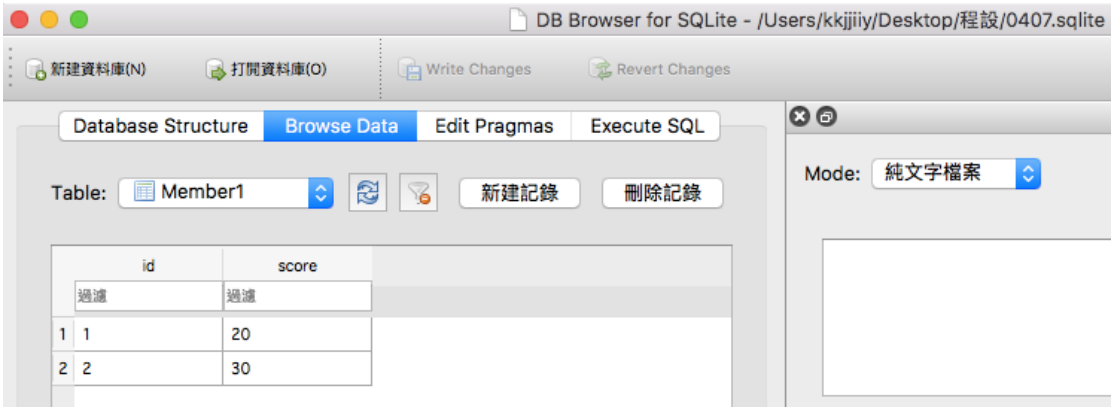
4. 當呼叫 Score.best()，則可查看目前最高分的分數。

```python
class Score:
    def __init__(self,i_d,score):
        self.iii = i_d
        self.sss = score
    def add(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "INSERT INTO Member1 Values (%d,%d)"%(self.iii,self.sss)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def change(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "update Member1 set score='%d' where id='%d'"%(self.sss,self.iii)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def delete(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "delete from Member1 where id='%d'"%(self.iii)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def best(self):
        global n_h
        print("最高分為%d"%(n_h))
    def operates(self):
        global j
        print("使用%d次SQL語法"%(j))
```

```python
j = 0
n_h = 0
while j < 1000:
    i = int(input("請輸入id:"))
    a = int(input("請輸入成績："))
    choice1 = input("是否新增資料？(y/n):")
    if choice1 == "y":
        j+=1
        Score(i,a).add()
    choice2 = input("是否修改資料？(y/n):")
    if choice2 == "y":
        j+=1
        c = int(input("請輸入要修改的成績："))
        Score(i,c).change()
    choice3 = input("是否刪除資料？(y/n):")
    if choice3 == "y":
        j+=1
        Score(i,a).delete()
    if n_h < a:
        n_h = a
    choice = input("是否繼續輸入？(y/n):")
    if choice == "y":
        continue
    else:
        Score(i,a).best()
        Score(i,a).operates()
        break
```

新增 id:1,成績 20、id:2,成績 30，可以得到最高成績為 30 的結果。

```
tangminzhideAir:程設 kkjjiiy$ python 0519.py
請輸入id:1
請輸入成績 : 20
是否新增資料？(y/n):y
是否修改資料？(y/n):n
是否刪除資料？(y/n):n
是否繼續輸入？(y/n):y
請輸入id:2
請輸入成績 : 30
是否新增資料？(y/n):y
是否修改資料？(y/n):n
是否刪除資料？(y/n):n
是否繼續輸入？(y/n):n
最高分為 30
使用2次SQL語法
tangminzhideAir:程設 kkjjiiy$
```



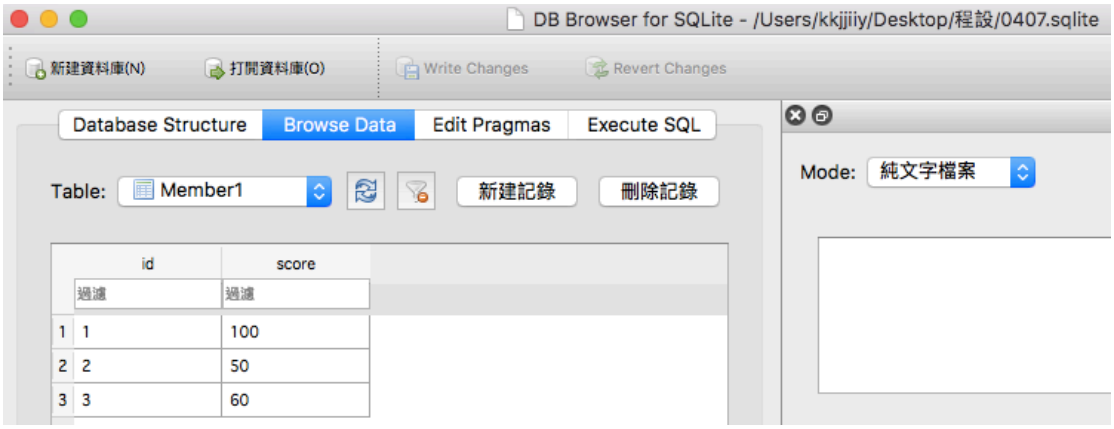
5. 當呼叫 Score.operates()，則可查看目前共使用多少次 SQL 語法(使用新增、修改與刪除的總次數)。

```python
class Score:
    def __init__(self,i_d,score):
        self.iii = i_d
        self.sss = score
    def add(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "INSERT INTO Member1 Values (%d,%d)"%(self.iii,self.sss)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def change(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "update Member1 set score='%d' where id='%d'"%(self.sss,self.iii)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def delete(self):
        import sqlite3
        conn = sqlite3.connect('0407.sqlite')
        cursor = conn.cursor()
        sqlstr = "delete from Member1 where id='%d'"%(self.iii)
        cursor.execute(sqlstr)
        conn.commit()
        conn.close()
    def best(self):
        global n_h
        print("最高分為%d"%(n_h))
    def operates(self):
        global j
        print("使用%d次SQL語法"%(j))
```

```python
j = 0
n_h = 0
while j < 1000:
    i = int(input("請輸入id:"))
    a = int(input("請輸入成績："))
    choice1 = input("是否新增資料？(y/n):")
    if choice1 == "y":
        j+=1
        Score(i,a).add()
    choice2 = input("是否修改資料？(y/n):")
    if choice2 == "y":
        j+=1
        c = int(input("請輸入要修改的成績："))
        Score(i,c).change()
    choice3 = input("是否刪除資料？(y/n):")
    if choice3 == "y":
        j+=1
        Score(i,a).delete()
    if n_h < a:
        n_h = a
    choice = input("是否繼續輸入？(y/n):")
    if choice == "y":
        continue
    else:
        Score(i,a).best()
        Score(i,a).operates()
        break
```

新增 id:1,成績 100，id:2,成績 50，id:3,成績 70，修改 id:3,成績 70 為成績 60，總共使用語法 4 次

```
tangminzhideAir:程設 kkjjiiy$ python 0519.py
請輸入id:1
請輸入成績 : 100
是否新增資料？(y/n):y
是否修改資料？(y/n):n
是否刪除資料？(y/n):n
是否繼續輸入？(y/n):y
請輸入id:2
請輸入成績 : 50
是否新增資料？(y/n):y
是否修改資料？(y/n):n
是否刪除資料？(y/n):n
是否繼續輸入？(y/n):y
請輸入id:3
請輸入成績 : 70
是否新增資料？(y/n):y
是否修改資料？(y/n):y
請輸入要修改的成績 : 60
是否刪除資料？(y/n):n
是否繼續輸入？(y/n):n
最高分為100
使用4次SQL語法
tangminzhideAir:程設 kkjjiiy$
```

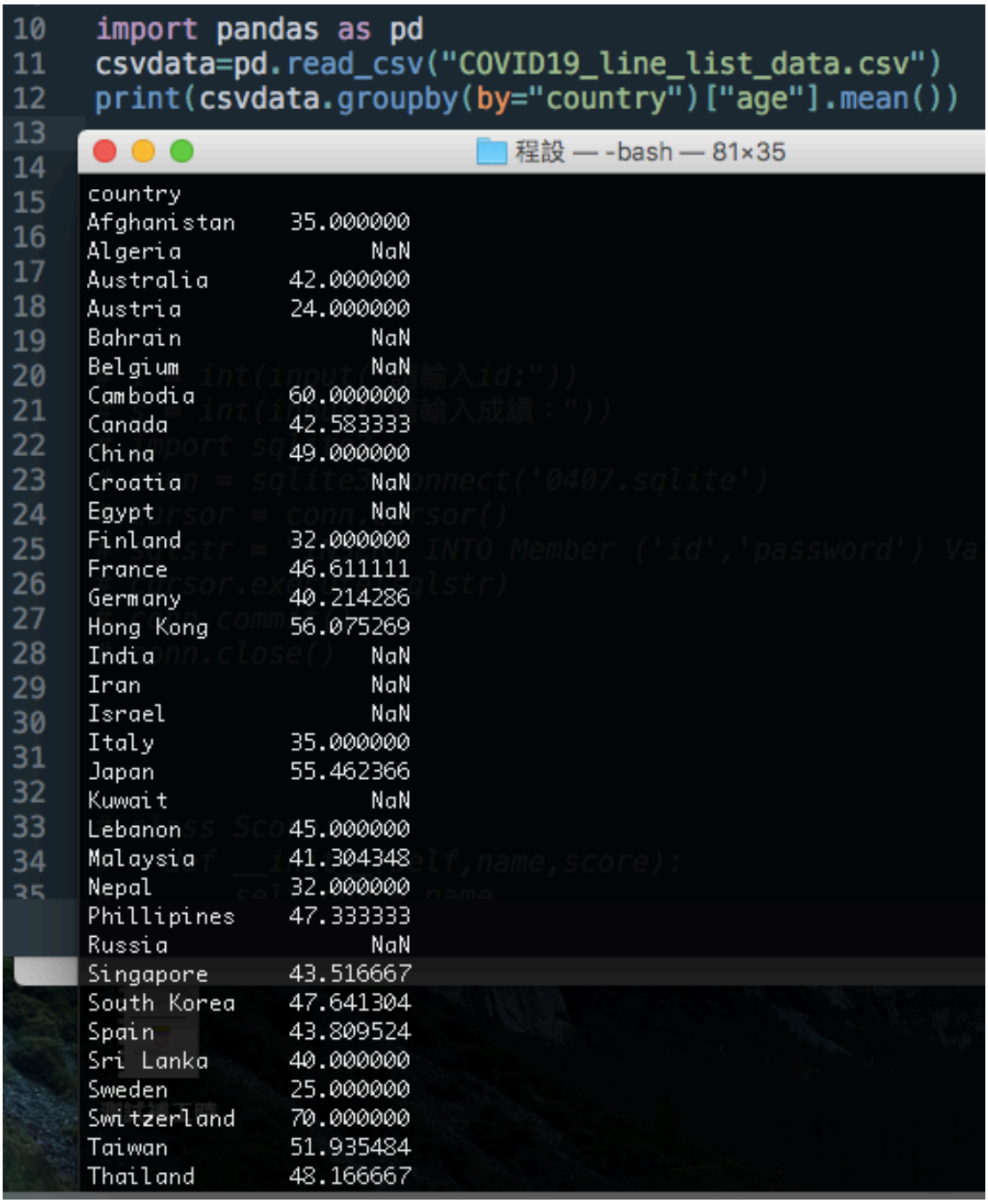


2. 請使用 COVID19_line_list_data.csv，

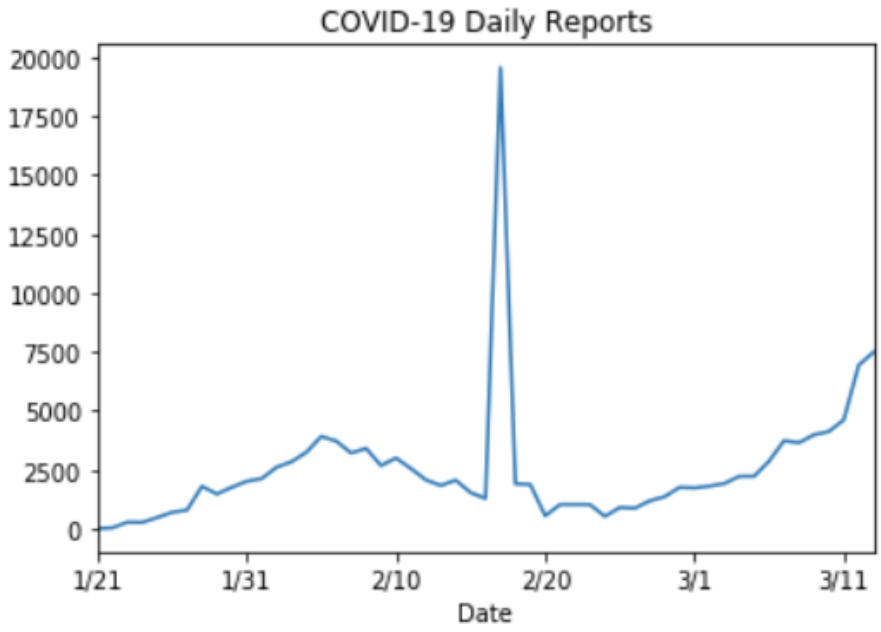
A. 列出 location 為 Tianjin 且年紀大於 35 的案例。

```
12  import pandas as pd
13  csvdata=pd.read_csv("COVID19_line_list_data.csv")
14  print(csvdata[(csvdata['location']=="Tianjin") & (csvdata['age'] > 35)])
15
16
17      print(plot.show())
18  NameError: name 'plot' is not defined
19  tangminzhideAir:程設 kkjjiiy$ python 0519.py
20  None
21  tangminzhideAir:程設 kkjjiiy$ python 0519.py
22      id  case_in_country reporting date  Unnamed: 3  \
23  3    4              NaN      1/21/2020         NaN
24  4    5              NaN      1/21/2020         NaN
25  80   81             NaN      1/23/2020         NaN
26  95   96             NaN      1/24/2020         NaN
27  117  118            NaN      1/21/2020         NaN
28  118  119            NaN      1/21/2020         NaN
29  119  120            NaN      1/22/2020         NaN
30  120  121            NaN      1/22/2020         NaN
31  121  122            NaN      1/23/2020         NaN
32  122  123            NaN      1/24/2020         NaN
33  125  126            NaN      1/25/2020         NaN
34  150  151            NaN      1/25/2020         NaN
35  151  152            NaN      1/25/2020         NaN
36  163  164            NaN      1/26/2020         NaN
37  173  174            NaN      1/26/2020         NaN
38  174  175            NaN      1/26/2020         NaN
39  175  176            NaN      1/26/2020         NaN
```

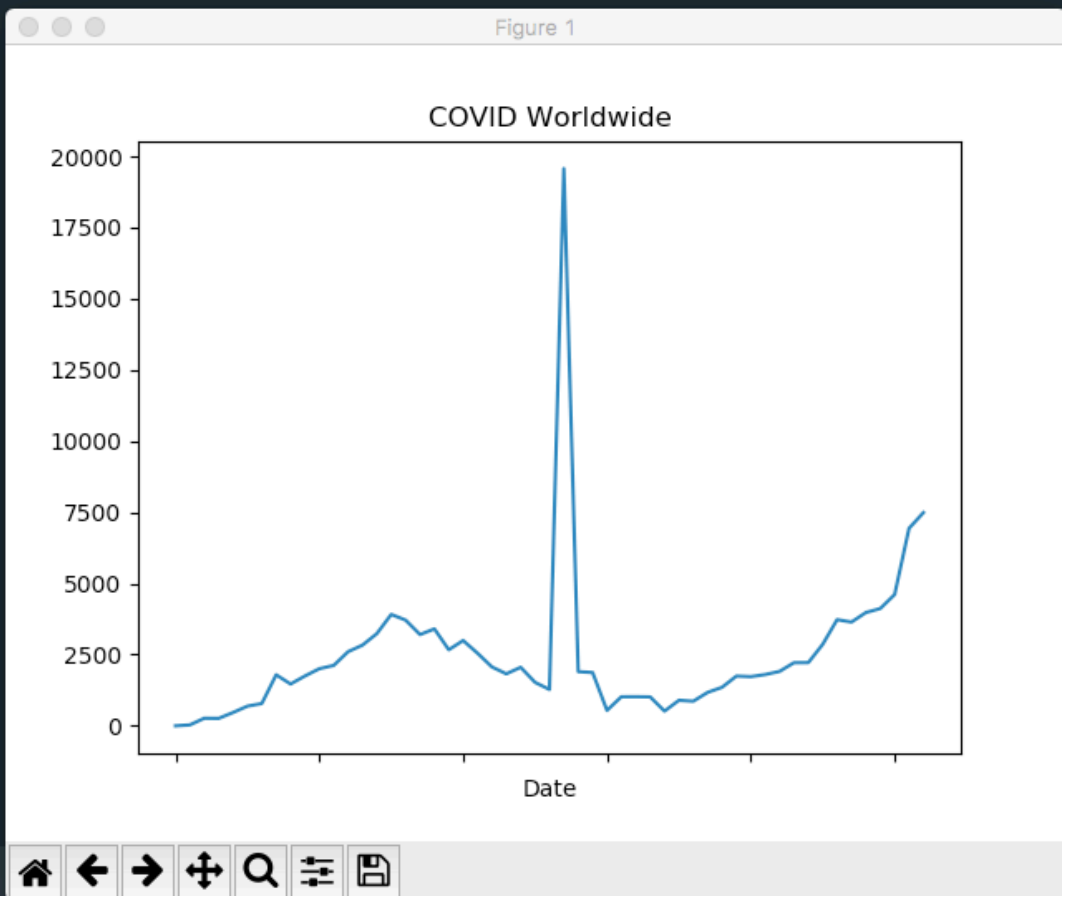
B. 算出各國病患的平均年紀。


10 import pandas as pd
11 csvdata=pd.read_csv("COVID19_line_list_data.csv")
12 print(csvdata.groupby(by="country")["age"].mean())
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
程設 — -bash — 81×35
country
Afghanistan 35.000000
Algeria NaN
Australia 42.000000
Austria 24.000000
Bahrain NaN
Belgium NaN
Cambodia 60.000000
Canada 42.583333
China 49.000000
Croatia NaN
Egypt NaN
Finland 32.000000
France 46.611111
Germany 40.214286
Hong Kong 56.075269
India NaN
Iran NaN
Israel NaN
Italy 35.000000
Japan 55.462366
Kuwait NaN
Lebanon 45.000000
Malaysia 41.304348
Nepal 32.000000
Phillipines 47.333333
Russia NaN
Singapore 43.516667
South Korea 47.641304
Spain 43.809524
Sri Lanka 40.000000
Sweden 25.000000
Switzerland 70.000000
Taiwan 51.935484
Thailand 48.166667

3. 請使用 cases_add_Taiwan.csv，透過 Python 繪製 COVID-19 世界案例變化的折線圖。



```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
tdata = pd.read_csv("cases_add_Taiwan.csv",encoding='big5')
tdata[['Date','Worldwide']].plot(kind = 'line',x = 'Date',y = 'Worldw
print(plt.show())
```



4. 請使用 cases_add_Taiwan.csv，透過 Python 運用折線圖比較三個國家的 COVID-19 疫情爆發的狀況 (三條線必須使用不同的顏色)。