

Exercises

By Nachai Paramesthanakorn

Calculate data from text

Input:

อ่านข้อมูลจากไฟล์ "Score.txt" ที่อยู่ในโฟลเดอร์ "D:\Read_file"

- บรรทัดแรกเป็นชื่อ header
- บรรทัดที่ 2 ถึงบรรทัดสุดท้าย แต่ละบรรทัด เก็บข้อมูลชื่อ ตามด้วยช่องว่างหนึ่งช่อง ตามด้วยคะแนน

Process:

- ให้ใช้การต่อ path จาก library os เท่านั้น
- คำนวณหาคะแนนเฉลี่ย
- คำนวณหาว่าใครได้คะแนนสูงสุด
- คำนวณหาว่าใครได้คะแนนต่ำสุด
- คำนวณหาว่า มีกี่คนที่ได้คะแนนต่ำกว่า 14.00

Output:

- แสดงผลคะแนนเฉลี่ย (ทศนิยม 3 ตำแหน่ง)
- แสดงชื่อและคะแนนสูงสุด
- แสดงชื่อและคะแนนต่ำสุด
- แสดงลิสต์จำนวนคน และชื่อ ที่ได้คะแนนต่ำกว่า 14.00

ตัวอย่าง

Input	Output
input from text file	max socre:19.99 name:E min socre:10.01 name:X average:14.308 name score < 14:['A', 'B', 'C', 'D', 'G', 'J', 'K', 'T', 'U', 'X', 'Y'] total:11

Check answer

Input:

อ่านข้อมูลจากไฟล์ "Anwsers.txt" ที่อยู่ในโฟลเดอร์ "D:\Read_file"

- บรรทัดที่ 1 เป็นเฉลยคำตอบ A, B, C, D
- บรรทัดที่ 2 ถึงบรรทัดสุดท้าย แต่ละบรรทัด เก็บข้อมูลเลขประจำตัว ตามด้วยช่องว่างหนึ่งช่อง ตามด้วยชุดคำตอบ A, B, C, D (กรณีเป็นช่องว่าง หรือ ตัวอักษรอื่นที่ไม่ใช่ A, B, C, D คือ ไม่ตอบข้อนั้น) เรียงลำดับจากข้อ 1 ถึงข้อสุดท้าย

Process:

- เขียนไฟล์ใหม่ ชื่อ "result.txt" เก็บไว้ในโฟลเดอร์ "D:\Read_file"
- แต่ละบรรทัดของไฟล์ประกอบด้วยเลขประจำตัวตามด้วยคะแนนที่ได้ คั่นด้วย comma

Output:

บันทึกคะแนนของแต่ละคนลงไฟล์ "result.txt"

ตัวอย่าง

Input	Output
input from text file	write to file "result.txt"
	0001, 25
	0002, 14
	0003, 7
	0004, 24
	0005, 21
	0006, 19
	0007, 15
	0008, 14
	0009, 21

Distance between 2 points

Input:

กำหนด points เป็นลิสต์ของพิกัดข้อมูลจุด 2 จุด (x1, y1, x2, y2) ตามลำดับ

Process:

- คำนวณหาระยะห่างระหว่างจุด (x1, y1) กับ (x2, y2) โดยใช้สูตร ระยะห่างระหว่างจุด = รากที่สองของ $(x1-x2)^2 + (y1-y2)^2$
- คำนวณหาจุดกึ่งกลางระหว่างจุด (x1, y1) กับ (x2, y2) โดยใช้สูตร จุดกึ่งกลาง $x = (x1 + x2)/2$, $y = (y1 + y2)/2$

Output:

distance (0,0) กับ (1,1) = 1.414, half = (0.5, 0.5)

distance (0,0) กับ (-2,2) = 2.828, half = (-1.0, 1.0)

distance (-1,0) กับ (3,1) = 4.123, half = (1.0, 0.5)

distance (-1,-2) กับ (2,5) = 7.616, half = (0.5, 1.5)

distance (0,-2) กับ (1,2) = 4.123, half = (0.5, 0.0)

```
points = [ (0, 0), (3, 1), (0, -1), (-2, 1), (2, -2), (-1, -2), (-2, -3), (3, 4),]
```

Distance between 2 points ต่อ

```
import math

def distance_between_2_points(x1, y1, x2, y2):
    # คำนวณระยะห่างระหว่างจุด (x1, y1) กับ (x2, y2) และส่งค่ากลับ
    pass

def half_between_2_points(x1, y1, x2, y2):
    # คำนวณจุดกึ่งกลางระหว่างจุด (x1, y1) กับ (x2, y2) และส่งค่า (x,y) กลับ
    pass

points = [
    (0, 0, 1, 1),    # x1=0 ,x2=0, y1=1, y2=1
    (0, 0, -2, 2),   # x1=0 ,x2=0, y1=-2, y2=2
    (-1, 0, 3, 1),    # x1=-1 ,x2=0, y1=3, y2=1
    (-1, -2, 2, 5),   # x1=-1 ,x2=-2, y1=2, y2=4
    (0, -2, 1, 2)     # x1=0 ,x2=-2, y1=1, y2=2
]

# Code here
```

Distance between line and points

Input: กำหนด

- เส้นตรง $L = 3x + 4y + 8 = 0$ โดย $A=3$, $B=4$, $C=8$ ตามสมการเส้นตรง $Ax+By+C = 0$
- points เป็นลิสต์ของพิกัดจุด (x, y) ตามลำดับ

Process:

- คำนวณหาระยะห่างระหว่างจุด (x, y) กับ เส้นตรง L โดยใช้สูตร ระยะห่างระหว่างจุดกับเส้น = absolute ของ $((A*x)+(B*y)+C)$ หารด้วย รากที่สองของ $(A^2 + B^2)$

Output:

distance:(0,0) = 1.6

distance:(3,1) = 4.2

distance:(0,-1) = 0.8

distance:(-2,1) = 1.2

distance:(2,-2) = 1.2

distance:(-1,-2) = 0.6

distance:(-2,-3) = 2.0

distance:(3,4) = 6.6

```
points = [  
    (0, 0), # x=0, y=0  
    (3, 1), # x=3, y=1  
    (0, -1), # x=0, y=-1  
    (-2, 1), # x=-2, y=0  
    (2, -2), # x=2, y=-2  
    (-1, -2), # x=-1, y=-2  
    (-2, -3), # x=-2, y=-3  
    (3, 4), # x=3, y=4  
]
```

Distance between line and points ต่อ

```
import math

✓ def distance_between_line_and_point(x, y):
    # คำนวณระยะห่างระหว่างจุด (x, y) กับ เส้นตรง L และส่งค่ากลับ
    global A, B, C
    pass

A = 3
B = 4
C = 8

✓ points = [
    (0, 0), # x=0, y=0
    (3, 1), # x=3, y=1
    (0, -1), # x=0, y=-1
    (-2, 1), # x=-2, y=0
    (2, -2), # x=2, y=-2
    (-1, -2), # x=-1, y=-2
    (-2, -3), # x=-2, y=-3
    (3, 4), # x=3, y=4
]

# Code here
```

Find days between date

Input:

- ค่า a เป็นวันที่ที่กำหนด
- ลิสต์ x เก็บค่า วันที่ในรูปแบบของสตริง (format '%Y-%m-%d %H')

Process:

- คำนวณหาจำนวนวันที่ต่างกันของแต่ละ item กับ a
- คำนวณหาจำนวนชั่วโมงต่างกันของแต่ละ item กับ a

Find days between date ต่อ

Output:

a = 2022-01-01 10:00:00

days difference:152 days hours:23.0 h. hours difference: 3671 h.

days difference:184 days hours:9.0 h. hours difference: 4425 h.

days difference:214 days hours:0.0 h. hours difference: 5136 h.

days difference:224 days hours:10.0 h. hours difference: 5386 h.

days difference:0 days hours:8.0 h. hours difference: 8 h.

days difference:-213 days hours:17.0 h. hours difference: -5095 h.

days difference:-201 days hours:10.0 h. hours difference: -4814 h.

days difference:-88 days hours:5.0 h. hours difference: -2107 h.

days difference:-89 days hours:6.0 h. hours difference: -2130 h.

days difference:-335 days hours:12.0 h. hours difference: -8028 h.

days difference:1 days hours:6.0 h. hours difference: 30 h.

```
x = [  
    "2021-08-01 11",  
    "2021-07-01 01",  
    "2021-06-01 10",  
    "2021-05-22 00",  
    "2022-01-01 02",  
    "2022-08-01 17",  
    "2022-07-21 00",  
    "2022-03-30 05",  
    "2022-03-31 04",  
    "2022-12-01 22",  
    "2021-12-31 04",  
]
```

Find days between date ต่อ

```
import datetime

a = datetime.datetime(2022, 1, 1, 10)

def find_days_sec(date_x):
    """คำนวณหาจำนวนวันที่ต่างกันของ ตัวแปรวันที่ date กับ a
    และ return ค่าส่วนต่างของ days, hours

    คำนวณชั่วโมงที่ต่างกันสามารถนำหน่วยวินาที มาหาร 3600 ได้
    เช่น
        a = 1 Jan 2022 เวลา 10.00 am
        date x = 2 Jan 2022 เวลา 1.00 AM
        วันที่จะต่างกัน -1 วัน กับ 9 ชั่วโมง
    หรือ
        a = 1 Jan 2022 เวลา 10.00 am
        date x = 31 Dec 2021 เวลา 4.00 AM
        วันที่จะต่างกัน +1 วัน กับ 6 ชั่วโมง

    Args:
        date_x (datetime): วันที่ วันที่ ที่ต้องการเปรียบเทียบกับ a

    Returns:
        int: จำนวนวัน ที่ต่างกันกับ a
        number: จำนวนชั่วโมง ที่ต่างกันกับ a

    """
    pass
```

Find days between date ต่อ

```
56  ✓ x = [  
57      "2021-08-01 11",  
58      "2021-07-01 01",  
59      "2021-06-01 10",  
60      "2021-05-22 00",  
61      "2022-01-01 02",  
62      "2022-08-01 17",  
63      "2022-07-21 00",  
64      "2022-03-30 05",  
65      "2022-03-31 04",  
66      "2022-12-01 22",  
67      "2021-12-31 04",  
68  ]  
69  
70  print(f"a = {a}")  
71  
72  ✓ for item in x:  
73      # Code here  
74  
75      # ให้ใส่ตัวแปรที่ .format()  
76      print("days difference:{} days hours:{} h.\t hours difference: {} h.".format())  
77
```

Clean data

Input:

- อ่านข้อมูลจากไฟล์ "Raw_promotion_detail.txt" ที่อยู่ในโฟลเดอร์ "D:\Read_file"
 - บรรทัดที่ 1 เก็บ header 'date_start|date_end|Product_ID|Price|discount_rate|Quantity'
 - บรรทัดที่ 2 ถึง บรรทัดสุดท้าย เก็บข้อมูล date_start, date_end, Product_ID, Price, discount_rate, Quantity
 - *** discount_rate เป็นหน่วย % เช่น 10 คือ 10%
 - *** date_start, date_end เป็นสตริงในรูปแบบ format '%d %b %Y'
 - *** หมวดหมู่ของสินค้า คัดจาก 2 ตัวแรกของ Product_ID เช่น 'AG10044' จะได้หมวดหมู่คือ 'AG'

Process:

- คำนวณหาจำนวนวันที่ต่างกันระหว่าง date_start และ date_end แล้วแปลงเป็นหน่วยนาที่
- คำนวณหาส่วนลด จากราคาของสินค้า * discount_rate / 100 และปัดเศษทั้งหมดขึ้น เช่น ได้ส่วนลด 100.01 ให้ปัดเป็น 101
- คำนวณหาว่า ในปี 2021 ถ้าขายสินค้าหมดในช่วงจัดโปรโมชั่น จะมียอดขาย ต้นทุนสินค้า และกำไรเท่าไร (รวมทุกหมวดหมู่)
- คำนวณหาว่า ถ้าขายสินค้าหมดในช่วงจัดโปรโมชั่น ระหว่างวันที่ 1 Jan 2021 - 30 Jun 2021 (คัดจาก date_start) แต่ละหมวดหมู่จะมียอดขาย ต้นทุนสินค้า และกำไรเท่าไร
 - *** ให้ต้นทุนสินค้าอยู่ที่ 25% ของราคาสินค้าก่อนหักส่วนลด)

Clean data ต่อ

Output:

- แสดงผล เวลาทั้งหมดที่จัดโปรโมชั่นทั้งปีของทุก product (หน่วยวัน และ นาที)
- แสดงผล ยอดขาย ต้นทุนสินค้า และกำไรของทั้งปีทุกหมวดหมู่
- แสดงผล ยอดขาย ต้นทุนสินค้า และกำไรที่ได้ในช่วงวันที่ 1 Jan 2021 - 30 Jun 2021 ของแต่ละหมวดหมู่

---Result---

เวลาทั้งหมดที่จัดโปรโมชั่นทั้งปี = 234 วัน หรือ 336960 นาที

ในปี 2021 มียอดขายรวม: 13652270.0 บาท

ในปี 2021 มีต้นทุนรวม: 5115500.0 บาท

ในปี 2021 มีกำไรรวม: 8536770.0 บาท

ในช่วงครึ่งปีแรก 1 Jan 2021 - 30 Jun 2021

AE มียอดขายรวม: 425450.0 บาท, มีต้นทุนรวม: 121875.0 บาท, มีกำไรรวม: 303575.0 บาท

AS มียอดขายรวม: 1141060.0 บาท, มีต้นทุนรวม: 402550.0 บาท, มีกำไรรวม: 738510.0 บาท

AQ มียอดขายรวม: 1267200.0 บาท, มีต้นทุนรวม: 485000.0 บาท, มีกำไรรวม: 782200.0 บาท

AT มียอดขายรวม: 4819500.0 บาท, มีต้นทุนรวม: 1721250.0 บาท, มีกำไรรวม: 3098250.0 บาท

AG มียอดขายรวม: 245000.0 บาท, มีต้นทุนรวม: 87500.0 บาท, มีกำไรรวม: 157500.0 บาท

Clean data ต่อ

```
38  ✓ import datetime
39      import os
40      import math
41
42  ✓ def get_discount(discount_rate, price):
43  ✓     """คำนวณหา discount พร้อมพิเศษทั้งหมดขึ้น
44      เช่น discount_rate = 15
45          price = 150
46          ส่วนลดก่อนพิเศษ = 22.5
47          ส่วนลดหลังพิเศษขึ้น = 23
48
49      Args:
50          discount_rate (int): ส่วนลด (เปอร์เซ็นต์)
51          price (float): ราคาสินค้า
52
53      Returns:
54          int: ส่วนลดหลังพิเศษขึ้น
55      """
56      pass
```

Clean data ต่อ

```
57
58  ✓ def find_difference_days(date_start, date_end):
59  ✓     """คำนวณหาจำนวนวันที่ต่างกันของ ตัวแปรวันที่ date_start กับ date_end
60     และ return ค่าส่วนต่างของ days
61
62     Args:
63         date_start (datetime): วันที่
64         date_end (datetime): วันที่
65
66     Returns:
67         int: จำนวนวัน ที่ต่างของ date_start กับ date_end
68     """
69     pass
70
```

Clean data ต่อ

```
70
71 ✓ def check_half_year(date_start):
72 ✓     """ตรวจสอบว่า date_start อยู่ในช่วงวันที่ 1 Jan 2021 - 30 Jun 2021 หรือไม่
73
74     Args:
75         date_start (datetime): วันที่
76
77     Returns:
78         bool: True ถ้า date_start อยู่ในช่วงวันที่ 1 Jan 2021 - 30 Jun 2021
79     """
80     pass
81
82     # add function
83
84     total_discount = 0 # ส่วนลดรวมทั้งปี
85     total_sales = 0 # ยอดขายรวมทั้งปี
86     total_cost = 0 # ต้นทุนรวมทั้งปี
87     total_profit = 0 # กำไรรวมทั้งปี
88     total_minute = 0 # จำนวนนาทีที่จัดโปรโมชั่นทั้งปี
89     total_day = 0 # จำนวนวันที่จัดโปรโมชั่นทั้งปี
90
91     # Code here
92
```


Clean data ต่อ

```
91  # Code here
92
93
94  print('---Result---')
95  print("เวลาทั้งหมดที่จัดโปรโมชั่นทั้งปี = {} วัน หรือ {} นาที".format(total_day, total_minute))
96  print("ในปี 2021 มียอดขายรวม: {} บาท".format(total_sales))
97  print("ในปี 2021 มีต้นทุนรวม: {} บาท".format(total_cost))
98  print("ในปี 2021 มีกำไรรวม: {} บาท".format(total_profit))
99  print()
100 print("ในช่วงครึ่งปีแรก 1 Jan 2021 - 30 Jun 2021")
101 # show result here
```