

Getting Start

การส่งงาน

1. ให้ตั้งชื่อไฟล์ในรูปแบบ LabXX_YY_ZZZZZZZZ.py เมื่อ XX หมายถึง หมายเลขปฏิบัติการ YY หมายถึง ลำดับข้อ และ ZZZZZZZZ คือรหัสประจำตัวนักศึกษา เช่น Lab01_1_630510999.py หมายถึงงาน ในปฏิบัติการที่ 1 ข้อที่ 1 และรหัสนักศึกษา คือ 630510999
2. ให้ตรวจสอบความถูกต้องในการทำงานของโปรแกรมและรับคะแนนที่ <http://202.28.248.55:10048> หรือ <https://gdr217.cs.science.cmu.ac.th/>
3. **การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข if `__name__ == '__main__'` : เพื่อความสะดวกในการ import จาก Script อื่น ๆ
4. ลักษณะ/ลำดับข้อความของการรับค่า/แสดงผล จะต้องเป็นไปตามที่ระบุในตัวอย่างการ run

จากตัวอย่างผลลัพธ์ดังที่ออกมาทางหน้าจอจดังรูป

Hello world

- 1) (Lab01_1_ZZZZZZZZ.py) ให้เขียนโปรแกรมเพื่อแสดงผลลัพธ์ดังรูปด้วย Template ต่อไปนี้
Template: [Lab01_1_ZZZZZZZZ.py](#)
- 2) (Lab01_1_ZZZZZZZZ.py) ให้เขียนโปรแกรมเพื่อแสดงผลลัพธ์ดังรูปด้วย Template ต่อไปนี้
Template: [Lab01_2_ZZZZZZZZ.py](#)

จากตัวอย่าง Code นี้

ให้นำหลักการ การเขียนฟังก์ชัน

การรับค่า/คืนค่า และหลักการหา

ค่าจำนวนเต็มบวก มาใช้เขียน function

จากโจทย์ต่อไปนี้

```

25 def isPositive(n):
26     result = (n > 0)
27     print("isPositive(", n, ") =", result)
28     return result
29
30
31 def isEven(n):
32     result = (n % 2 == 0)
33     print("isEven(", n, ") =", result)
34     return result
    
```

- 3) (Lab01_1_ZZZZZZZZ.py) ให้เขียนฟังก์ชัน `my_abs(x)` เพื่อรับค่าจำนวนเต็ม x และคืนค่า absolute ของ x ในลักษณะเดียวกันกับฟังก์ชัน built-in `abs()`

Template: [Lab01_3_ZZZZZZZZ.py](#)

Input

Output

2	2
-2	2

- 4) (Lab01_2_ZZZZZZZZ.py) ให้เขียนฟังก์ชัน `reverse_digits(x)` เพื่อกลับหลักจำนวนเต็มบวก x ในระบบเลข 4 หลัก โดยจะคืนค่าเป็นจำนวนเต็มที่กลับหลักแล้ว

Template: [Lab01_4_ZZZZZZZZ.py](#)

Input

Output

1234	4321
1	1000

