คำสั่งทำซ้ำ while

Python

ตัวแปรชนิด list

list เป็นตัวแปรประเภทหนึ่ง สามารถเก็บข้อมูลได้หลายตัวพร้อมกัน และเป็นการเก็บข้อมูลแบบลำดับ (Sequence) โดยมี Index เป็นตัวระบุตำแหน่งในการเข้าถึงข้อมูล

ในการประกาศ list นั้นข้อมูลของมันจะอยู่ภายในเครื่องหมาย [ ] และคั่นสมาชิกแต่ละตัวด้วยเครื่องหมายคอมมา , เช่น

```
name = [“Gift”, “Pin”, “Kan”]
number = [1, 2, 3, 4, 5, 6]
mix = [0.1, 5, “Hello”, -9]
```

การเรียกใช้ข้อมูลใน list

list นั้นใช้ index สำหรับการเข้าถึงข้อมูล โดย index ของ list จะเป็นจำนวนเต็มที่เริ่มจาก 0 และเพิ่มขึ้นทีละ 1 ไปเรื่อยๆ ดังนั้น เราจึงสามารถเข้าถึงข้อมูลภายใน list เพื่ออ่านหรืออัพเดทค่าได้โดยตรงผ่าน Index ของมัน เช่น

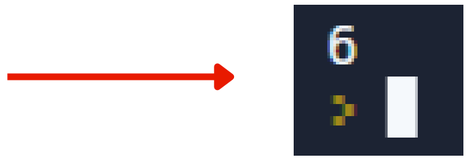
```
1  number = [1,2,3,4,5,6]
2  print(number[3])
```

ในตัวอย่าง เรามีตัวแปร list ที่ชื่อว่า number ดังนั้น เพื่อเข้าถึงสมาชิกตำแหน่งที่ 3 ภายใน list ซึ่งก็คือ 4 นั้นจะใช้คำสั่ง number[3] สังเกตว่า index จะลดลงหนึ่ง เพราะ index ของ List นั้นเริ่มต้นจาก 0

การเรียกใช้ข้อมูลใน list (ต่อ)

นอกจากนี้ เราสามารถเข้าถึงข้อมูลภายใน list โดยการใช้ index เป็นจำนวนลบได้ โดยเริ่มจาก -1 ซึ่งเป็นสมาชิกตัวสุดท้ายของ list และ -2 สมาชิกตัวถัดมาและลดลงไปทีละ 1 เช่น

```
1  number = [1,2,3,4,5,6]
2  print(number[-1])
```




```
1  number = [1,2,3,4,5,6]
2  print(number[-5])
```

คำสั่งทำซ้ำ while

while เป็นคำสั่งวนซ้ำที่มีการตรวจสอบเงื่อนไข (condition) ก่อนเข้าทำงานเสมอ เมื่อเงื่อนไขที่ทำการตรวจสอบเป็นจริง จึงจะประมวลผลคำสั่งหลังwhile แต่ถ้าเงื่อนไขเป็นเท็จจะยุติการทำงานทันที สำหรับงานที่นิยมใช้ while ในการแก้ปัญหาคือ ปัญหาที่ไม่ทราบจำนวนรอบการทำงานที่แน่นอนหรือปัญหาที่ไม่สามารถทราบได้ล่วงหน้าว่าจะต้องใช้เวลาในการประมวลผลนานเท่าใด ส่วนใหญ่มักจะหยุดการทำงานของ while ด้วยเงื่อนไขบางประการ

คำสั่งทำซ้ำ while

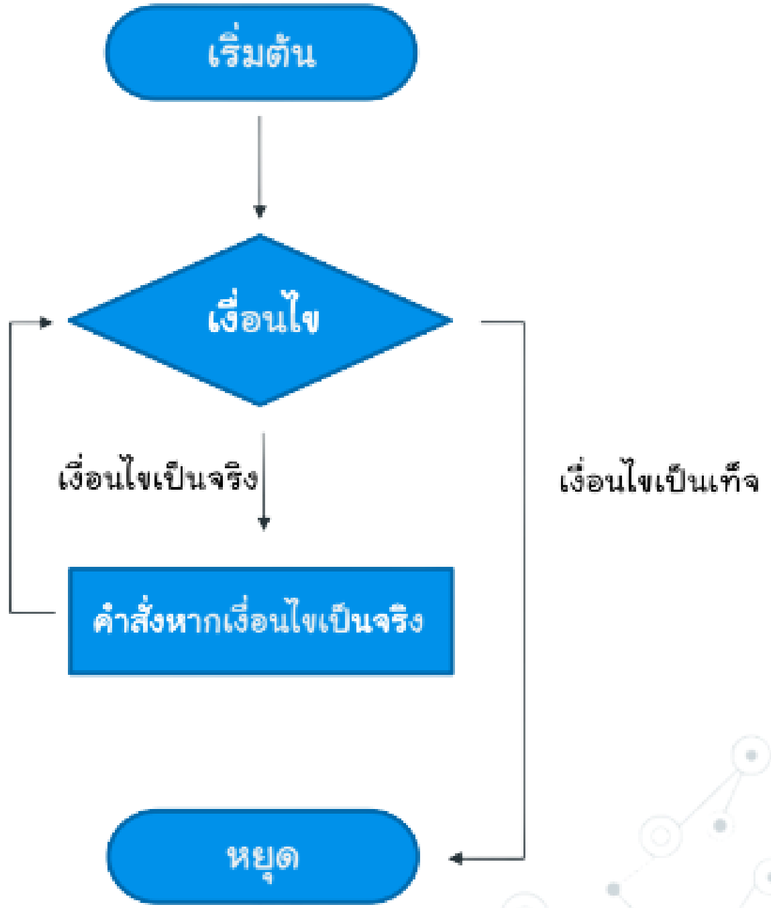
รูปแบบการเขียนคำสั่งทำซ้ำ while

while (เงื่อนไข) :

สั่งหรือชุดของคำสั่งที่จะให้ทำงานซ้ำ

(หากยังเป็นจริงอยู่จะทำงานไปเรื่อยๆ

แต่เมื่อ เงื่อนไขเป็นเท็จจะหยุดทำงาน)



ตัวอย่างคำสั่งทำซ้ำ while

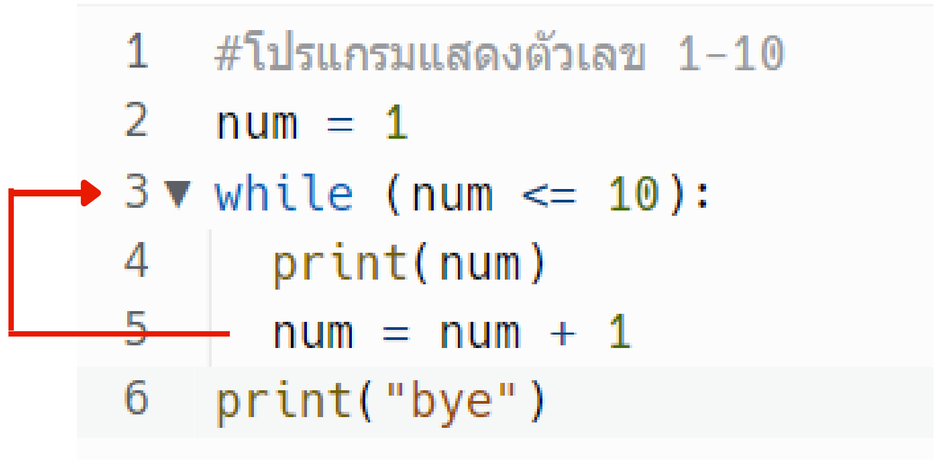
โปรแกรมแสดงตัวเลข 1 - 10

```
1  #โปรแกรมแสดงตัวเลข 1-10
2  num = 1
3  while (num <= 10):
4      print(num)
5      num = num + 1
6  print("bye")
```

```
1
2
3
4
5
6
7
8
9
10
bye
>
```

ตัวอย่างคำสั่งทำซ้ำ while

แปลความหมายโปรแกรมแสดงตัวเลข 1 - 10



```
1  #โปรแกรมแสดงตัวเลข 1-10
2  num = 1
3  while (num <= 10):
4      print(num)
5      num = num + 1
6  print("bye")
```

บรรทัด 2 ประกาศตัวแปร num เก็บค่า 1 เอาไว้

บรรทัด 3 คำสั่งทำซ้ำ ในขณะที่ num น้อยกว่าหรือเท่ากับ 10 ให้ทำเงื่อนไขในบล็อก

บรรทัด 4 หากเงื่อนไขเป็นจริงเป็นให้แสดงข้อมูลในตัวแปร num

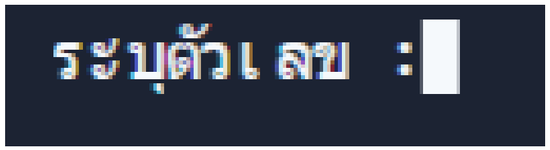
บรรทัด 5 เมื่อทำเงื่อนไขจากข้อ 3 แล้ว ให้ +1 ลงในตัวแปร num เมื่อทำเงื่อนไขในบล็อกครบแล้ว ให้กลับไปตรวจสอบเงื่อนไข ในบรรทัด 3

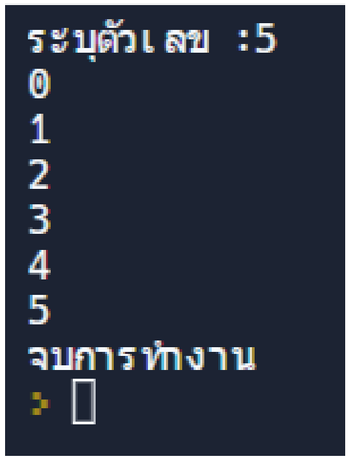
บรรทัด 6 หากเงื่อนไขบรรทัด 3 เป็นเท็จให้แสดงคำว่า bye

ตัวอย่างคำสั่งทำซ้ำ while ร่วมกับ input

โปรแกรมรับค่า และแสดงตัวเลข

```
1  num = int(input("ระบุตัวเลข :"))
2  i = 0
3  while (i <= num) :
4      print(i)
5      i = i + 1
6  print("จบการทำงาน")
```





ตัวอย่างคำสั่งทำซ้ำ while

แปลความหมายโปรแกรมรับค่า

และแสดงตัวเลข

```
1  num = int(input("ระบุตัวเลข :"))
2  i = 0
3  while (i <= num) :
4      print(i)
5      i = i + 1
6  print("จบการทำงาน")
```

ประกาศตัวแปร num รับค่าจำนวนเต็มจากผู้ใช้งาน

บรรทัด 2 ประกาศตัวแปร i เก็บค่า 0 เอาไว้

บรรทัด 3 คำสั่งทำซ้ำ ในขณะที่ข้อมูลในตัวแปร i มีค่าน้อยกว่า

หรือเท่ากับข้อมูลในตัวแปร num ให้ทำเงื่อนไขในบล็อก

บรรทัด 4 หากเงื่อนไขเป็นจริงเป็นให้แสดงข้อมูลในตัวแปร i

บรรทัด 5 เมื่อทำเงื่อนไขจากบรรทัด 3 แล้ว ให้ +1 ลงในตัวแปร i

เมื่อทำเงื่อนไขในบล็อกครบแล้ว ให้กลับไปตรวจสอบ

เงื่อนไข ในบรรทัด 3

บรรทัด 6 หากเงื่อนไขบรรทัด 3 เป็นเท็จให้แสดงคำว่า จบการทำงาน

ตัวอย่างคำสั่งทำซ้ำ while ร่วมกับ if

โปรแกรมตรวจสอบเลขคู่ตั้งแต่ 1 – 10

```
1  #โปรแกรมตรวจสอบเลขคู่ตั้งแต่ 1-10
2  num = 1
3  while (num <= 10) :
4    if (num % 2 == 0):
5      print(num)
6    num = num + 1
7  print("จบการทำงาน")
```

```
2
4
6
8
10
จบการทำงาน
>
```

ตัวอย่างคำสั่งทำซ้ำ while ร่วมกับ if

แปลโปรแกรมโปรแกรมตรวจสอบเลขคู่ตั้งแต่ 1 – 10

```
1  #โปรแกรมตรวจสอบเลขคู่ตั้งแต่ 1-10
2  num = 1
3  while (num <= 10) :
4    if (num % 2 == 0):
5      print(num)
6    num = num + 1
7  print("จบการทำงาน")
```

บรรทัด 2 ประกาศตัวแปร num เก็บค่า 1 เอาไว้

บรรทัด 3 คำสั่งทำซ้ำ ในขณะที่ข้อมูลในตัวแปร num มีค่าน้อยกว่าหรือเท่ากับ 10 ให้ทำเงื่อนไขในบล็อก

บรรทัด 4 หากข้อมูลในตัวแปร num หาร(นับเฉพาะเศษ) กับ 2 แล้วมีเศษเท่า 0 เป็นจริง ให้แสดงข้อมูลในตัวแปร num แต่ถ้าเป็นเท็จให้ข้ามเงื่อนไขในบล็อก และไปทำบรรทัด 5

บรรทัด 5 เมื่อทำเงื่อนไขจากบรรทัด 4 แล้ว ให้ +1 ลงในตัวแปร num

เมื่อทำเงื่อนไขในบล็อกครบแล้ว ให้กลับไปตรวจสอบเงื่อนไขในบรรทัด 3 (ทำซ้ำไปเรื่อยๆจนกว่าเงื่อนไขในบรรทัด 2 จะเป็นเท็จ)

บรรทัด 6 หากเงื่อนไขบรรทัด 3 เป็นเท็จให้แสดงคำว่า จบการทำงาน

คำสั่ง break ร่วมกับ while

คำสั่ง break เป็นคำสั่งที่ให้โปรแกรมออกจาก loop ทันที โดยไม่ทำคำสั่งที่เหลือต่อ ส่วนมากก่อนจะใช้คำสั่งนี้ ก็จะมีการตรวจสอบอะไรซักอย่างเสียก่อนการทำงานของคำสั่ง break ใน while loop แสดงได้ดังนี้

ตัวอย่างการใช้คำสั่ง break ร่วมกับ while

```
1  # โปรแกรมการใช้ break
2  num = 0
3  while (num <= 10):
4      if (num == 5):
5          break
6      print(num)
7      num = num + 1
8  print("จบการทำงาน")
```

```
0
1
2
3
4
จบการทำงาน
```

คำสั่ง break ร่วมกับ continue

คำสั่ง continue เป็นคำสั่งที่ให้โปรแกรม กลับไปทำงานที่ต้น loop โดยไม่ทำคำสั่งที่เหลือต่อ ส่วนมาก ก่อนจะใช้คำสั่งนี้ก็จะต้องมีการตรวจสอบอะไรซักอย่างเสียก่อน คล้าย ๆ กับค้าสั่ง break การทำงานของคำสั่ง continue ใน while loop แสดงได้ดังนี้

ตัวอย่างการใช้คำสั่ง continue ร่วมกับ while

```python
# โปรแกรมการใช้ continue
num = 0
while (num < 10):
    num = num + 1
    if (num == 5):
        continue
    print(num)
print("จบการทำงาน")
```

```
1
2
3
4
6
7
8
9
10
จบการทำงาน
>
```

ใบงานที่ 2 คำสั่ง while

1.เขียนโปรแกรม While loop นับเลขถอยหลัง 50-0

2.เขียนโปรแกรมแสดงตัวเลขที่หาร 3 ลงตัว โดยเริ่มจาก 12 , 15 , 18 ..... จำนวนเกิน 10 ตัว