

การควบคุมแบบวนทำซ้ำ

ผศ.ดร.สายัญญ์ สายยศ

ภาควิชาวิทยาการคอมพิวเตอร์
วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น

วัตถุประสงค์การเรียนรู้

- เข้าใจหลักการวนซ้ำในรูปแบบต่าง ๆ
- สามารถวิเคราะห์เลือกรูปแบบของการวนซ้ำที่เหมาะสมกับปัญหาได้

หัวข้อที่นำเสนอ

- บทนำ
- การควบคุมแบบวนทำซ้ำแบบต่างๆ
- การเขียนโปรแกรมควบคุมแบบวนทำซ้ำ

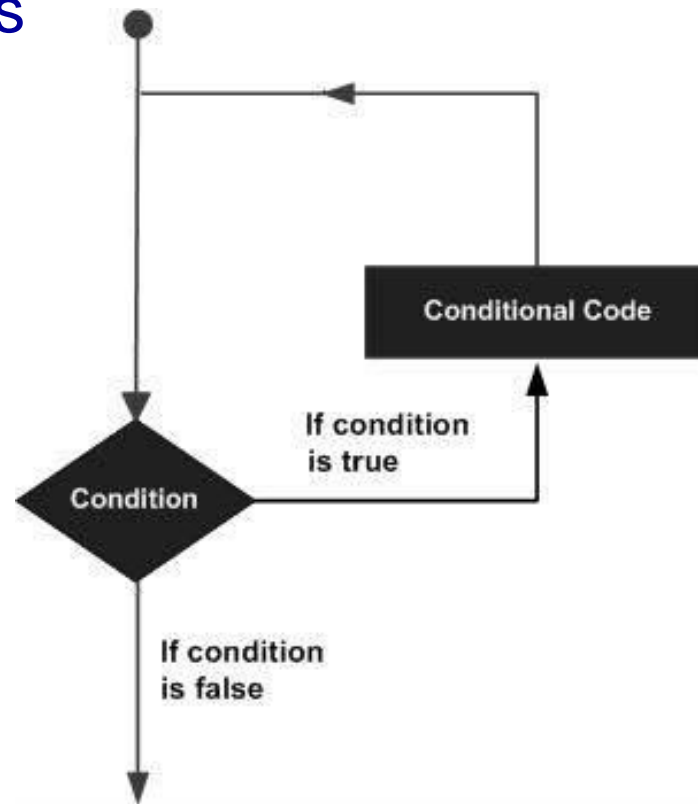
บทนำ



ที่มา: <http://internship19.wmu.psu.ac.th/2017/01/16/php-loop/>

การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python - Loops



การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python - Loops

Sr.No.	Loop Type & Description
1	<p>while loop</p> <p>Repeats a statement or group of statements while a given condition is TRUE. It tests the condition before executing the loop body.</p>
2	<p>for loop</p> <p>Executes a sequence of statements multiple times and abbreviates the code that manages the loop variable.</p>
3	<p>nested loops</p> <p>can use one or more loop inside any another while, for or do..while loop.</p>

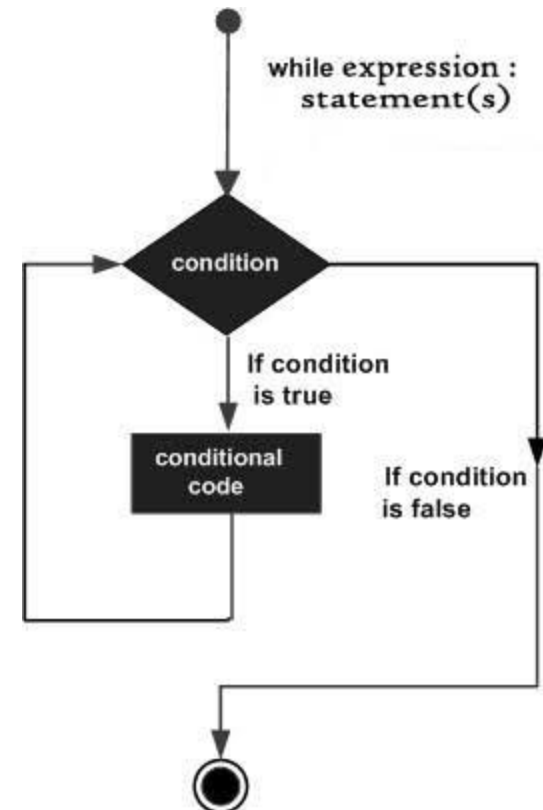
การควบคุมแบบวนซ้ำแบบต่างๆ

Python while Loop Statements

Syntax

```
while expression:  
    statement(s)
```

```
count = 0  
while (count < 9):  
    print 'The count is:', count  
    count = count + 1  
  
print "Good bye!"
```



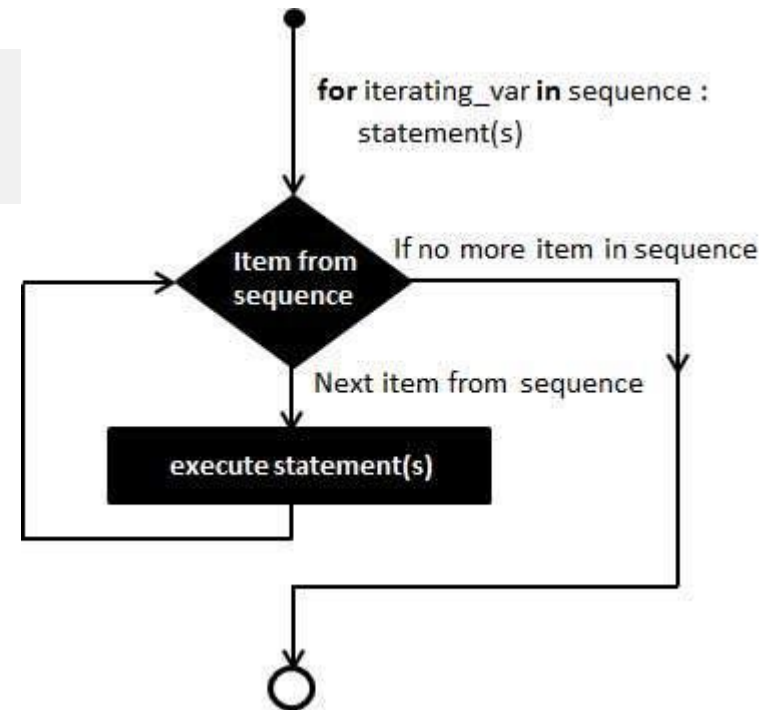
การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python for Loop Statements

Syntax

```
for iterating_var in sequence:  
    statements(s)
```

```
for x in range(0, 3):  
    print "We're on time %d" % (x)
```



การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python for Loop Statements

Python's range() Parameters

The range() function **has two sets** of parameters, as follows:

range(stop)

stop: Number of integers (whole numbers) to generate, starting from zero.

eg. range(3) == [0, 1, 2].

range([start], stop[, step])

การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python break and continue

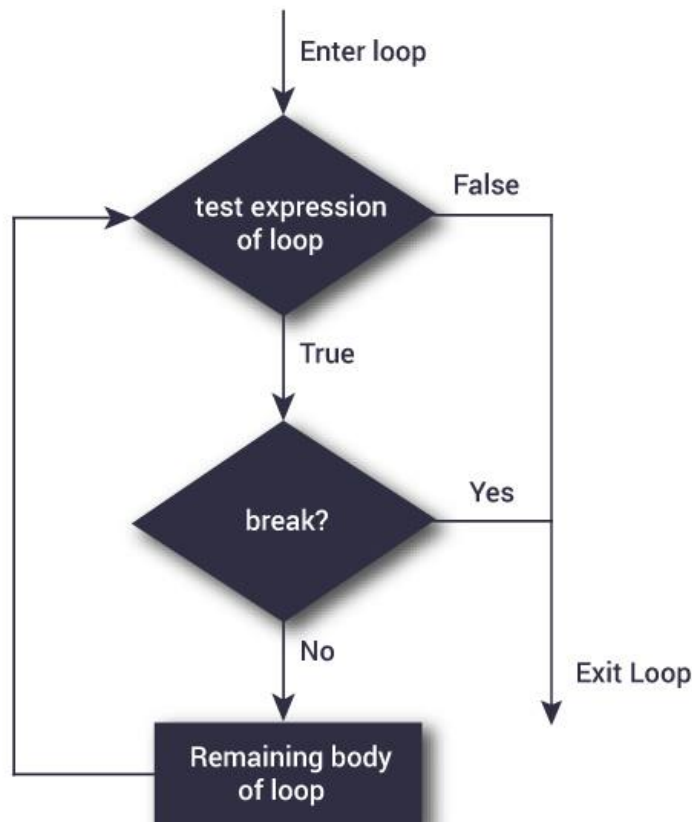
Python break statement

The break statement **terminates** the loop containing it. Control of the program flows to the statement immediately after the body of the loop.

If break statement is **inside a nested loop** (loop inside another loop), break will **terminate the innermost** loop.

การควบคุมแบบวนทำซ้ำแบบต่างๆ

Flowchart of break



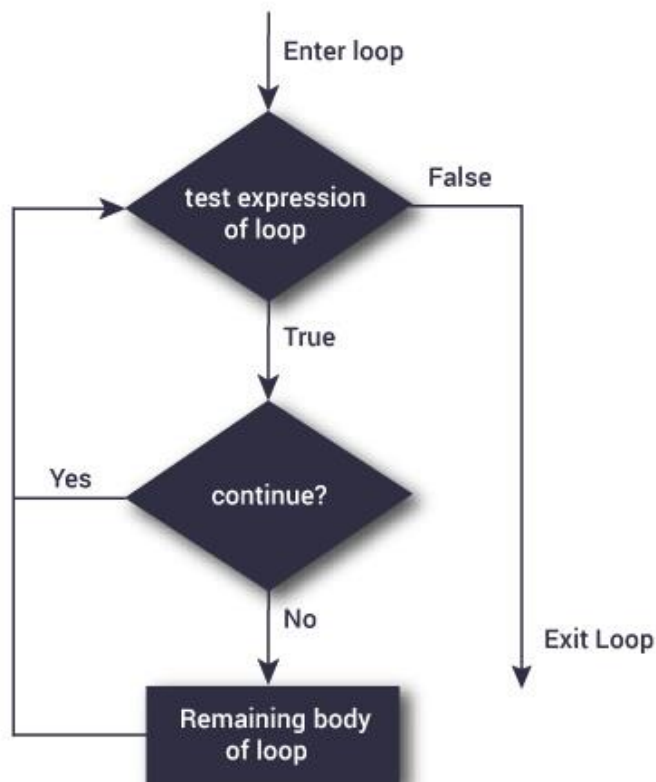
```
for var in sequence:
    # codes inside for loop
    if condition:
        break
    # codes inside for loop
# codes outside for loop
```

```
while test expression:
    # codes inside while loop
    if condition:
        break
    # codes inside while loop
# codes outside while loop
```

การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python continue statement

The continue statement is used to **skip** the rest of the **code inside a loop** for the **current iteration only**. Loop **does not terminate** but continues on with the next iteration.



```
for var in sequence:
    # codes inside for loop
    if condition:
        continue
    # codes inside for loop

# codes outside for loop
```

```
while test expression:
    # codes inside while loop
    if condition:
        continue
    # codes inside while loop

# codes outside while loop
```

การควบคุมแบบวนทำซ้ำแบบต่างๆ

Python pass statement

In Python programming, **pass** is a null statement. The difference between a **comment** and **pass** statement in Python is that, while the interpreter ignores a comment entirely, **pass** is not ignored.

ตัวอย่างการทำวนซ้ำ

```
sum=0
num= int( input("Please input
number:"))

for i in range(1,num):
    sum=sum+i
    print(i)

print(sum)
```

รอบที่	i	sum
1	1	1
2	2	3
3	3	6

สรุป

- บทนำ
- การทำวนซ้ำ



โจทย์

1. จงเขียนโปรแกรมตัดเกรด โดยโปรแกรมสามารถตัดเกรดได้หลายรอบ จะวนจนกว่าผู้ใช้ตอบ 'n'
2. จงเขียนโปรแกรมเพื่อหาผลรวมของตัวเลขที่ผู้ใช้ป้อนเข้าไป
3. จงเขียนโปรแกรมเพื่อหาค่าแฟคทอเรียล ของตัวเลขที่ผู้ใช้ป้อนเข้าไป
4. จงเขียนโปรแกรมเพื่อนับจำนวนเลขคู่ ตามช่วงที่ผู้ใช้ป้อนเข้าไป
5. จงเขียนโปรแกรมเพื่อตรวจสอบว่า ตัวเลขที่ผู้ใช้ป้อนเข้าไปเป็นจำนวนเฉพาะหรือไม่ (คะแนนพิเศษ)

พัก 10 นาที

Reference

<https://www.programiz.com/python-programming/break-continue> สืบค้น
สิงหาคม 2562