Python Programming

RGM College of Engineering & Technology (Autonomous)

Department of Computer Science & Engineering

Academic Year : 2020-2021

FLOW CONTROL STATEMENTS - 2

Guido Van Rossum

Learning Mantra

If you really strong in the basics, then remaining things will become so easy.

Agenda:

1. **Iterative Statements**
2.

2. Iterative Statements

- If we want to execute a group of statements multiple times then we should go for Iterative statements.
- Python supports 2 types of iterative statements.
    - i. for loop
    - ii. while loop

i) for loop:

If we want to execute some action for every element present in some sequence (it may be string or collection) then we should go for for loop.

Syntax:

```
for x in sequence:
    body
```

Where,

- 'sequence' can be string or any collection.
- 'body' will be executed for every element present in the sequence.

Eg 1: Write a Program to print characters present in the given string.

```
s="Sahasra"
for x in s:
    print(x)
```

Output:

```
S
a
h
a
s
r
a
```

Eg 2: Program to print characters present in string index wise.

```
S = input("Enter some String: ")
i = 0
for x in s :
        print("The character present at ",i,"index is :",x)
        i=i+1
```

Output:

```
Enter some String: Karthikeya
The character present at 0 index is : K
The character present at 1 index is : a
The character present at 2 index is : r
The character present at 3 index is : t
The character present at 4 index is : h
The character present at 5 index is : i
The character present at 6 index is : k
The character present at 7 index is : e
The character present at 8 index is : y
The character present at 9 index is : a
```

Eg 3: Program to print Hello 10 times.

```
s = 'Hello'
for i in range(1,11):
    print(s)
```

Output:

```
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
```

Eg 3: Program to print Hello 10 times. (Alternative Way)

```
s = 'Hello'
for i in range(10):
    print(s)
```

Output:

```
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
```

Eg 4: Program to display numbers from 0 to 10

```
for i in range(0,11):
    print(i)
```

Output:

```
0
1
2
3
4
5
6
7
8
9
10
```

Eg 5: Program to display odd numbers from 0 to 20

```
for i in range(21):
if(i%2!=0):
    print(i)
```

Output:

1
3
5
7
9
11
13
15
17
19

Eg 6: To display numbers from 10 to 1 in descending order.

```
for i in range(10,0,-1):
    print(i)
```

Output:

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

Eg 7: To print sum of numbers present inside list.

```
list=eval(input("Enter List:"))
sum=0;
for x in list:
    sum=sum+x;
print("The Sum=",sum)
```

Output:

```
Enter List:10,20,30,40
The Sum= 100
```

ii) while loop:

- If we want to execute a group of statements iteratively until some condition false, then we should go for while loop.

Syntax:

```
while condition:
    body
```

Eg 1: Program to print numbers from 1 to 10 by using while loop.

```
x=1
while x <=10:
    print(x)
    x=x+1
```

Output:

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

Eg 2: Program to display the sum of first n numbers using while loop.

Eg 3: write a program to prompt user to enter some name until entering Karthi.

```
name=""
while name!="Karthi":
    name=input("Enter Name: ")
print("Thanks for confirmation")
```

Output:

```
Enter Name: ramu
Enter Name: raju
Enter Name: Karthi
Thanks for confirmation
```

Infinite Loops

- Some times a loop can execute infinite number of times without stopping also.

Eg:

```
i = 1
while True: # The body of this while loop keep on executing because condition is always true
    print('Hello', i)          # This program never going to terminates
    i=i+1
```

Note: By pressing Ctrl + C we can stop this program.

- By mistake, if our program entered into an infinite loop, how we can solve this problem, where we have the above problem requirement.

```
while True:
    body
    if our required condition satisfied
        break
```

- If you are using break statement, you will come out from the loop.

Any question?

If you try to practice programs yourself, then you will learn many things automatically

Spend few minutes and then enjoy the study

Thank You