PYTHON PROGRAMMING

(BCC302 / BCC402/ BCC302H / BCC402H)

SYLLABUS

UNIT 2 :Python Program Flow Control Conditional blocks

if, else and else if, Simple for loops in python, For loop using ranges, string, list and dictionaries. Use of while loops in python, Loop manipulation using pass, continue, break and else. Programming using Python conditional and loop blocks

Gateway classes

PYTHON PROGRAMMING

UNIT 2

Python Program Flow Control Conditional blocks

LEC-2

By

Pragya Rajvanshi
M.Tech(CSE)

Gateway classes

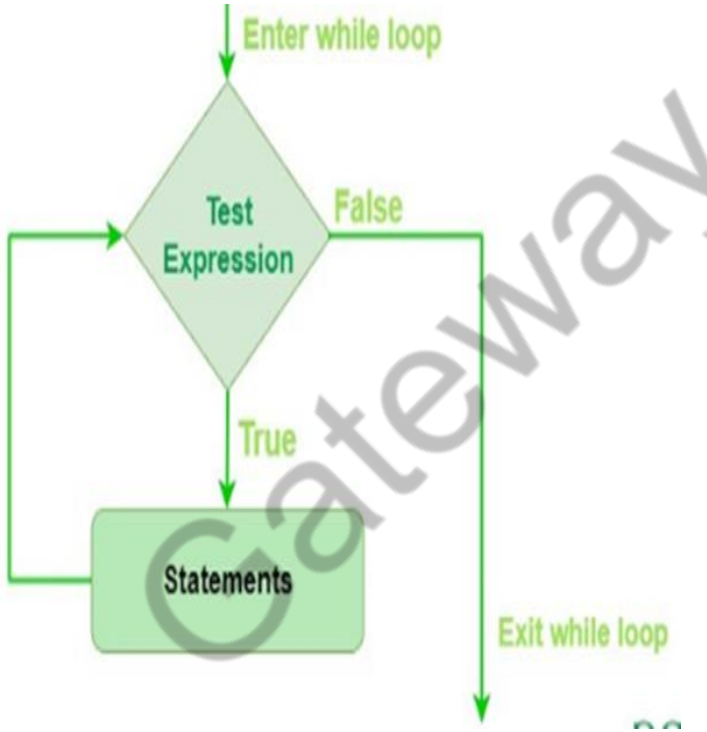
LOOPS

- Loops is used to execute a block of statements repeatedly until a given condition is satisfied.

1 while loop

Syntax

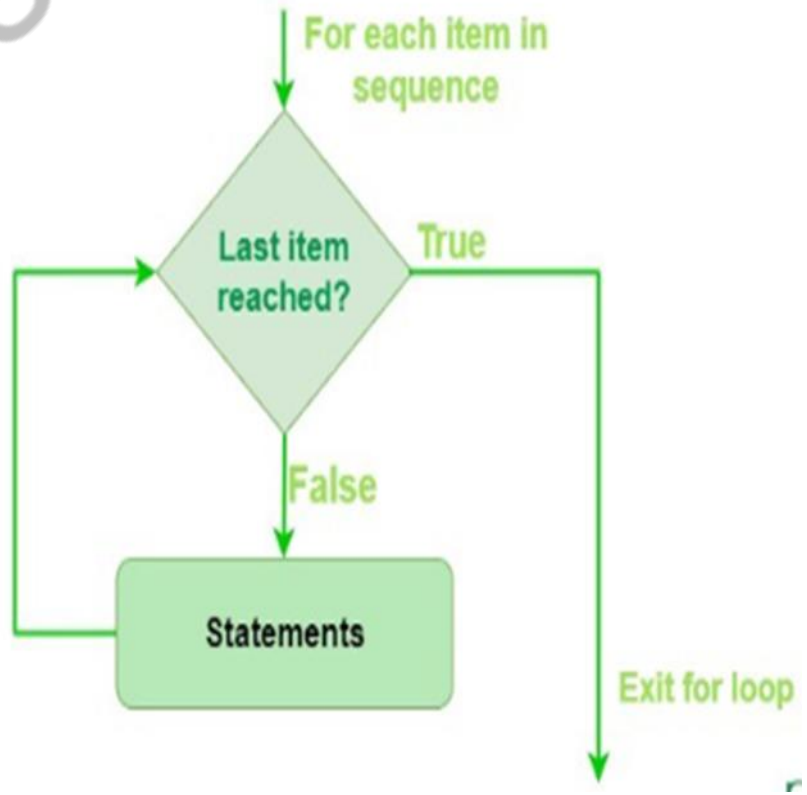
```
while expression:
    statement(s)
```



2 For loop

Syntax

```
for var in iterable:
    # statements
```



Gateway classes

➢ Range ()

➢ The Python **range() function** returns a sequence of numbers, in a given range.

1. range (stop)

```
for i in range(6):
    print(i, end=" ")
```

```
0 1 2 3 4 5
```

2.range (start, stop)

```
for i in range(5, 20):
    print(i, end=" ")
```

```
5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
```

3.range (start, stop, step)

```
for i in range(0, 30, 4):
    print(i, end=" ")
```

```
0 4 8 12 16 20 24 28
```

Gateway classes

Write a for loop that prints number from 0 to 57 using range function

```
1 for i in range(58):
2     print(i)
3
```

write a program to check whether a number is Armstrong or not

```
1  n=int(input("enter the number"))
2  sum = 0
3  num=n
4  while n> 0:
5      digit = n% 10
6      sum += digit ** 3
7      n //= 10
8  if num==sum:
9      print("armstromg")
10 else:
11     print("not armstrong")
12
```

```
enter the number153
armstromg
```

Gateway classes

write a program to find out the whether a number is palindrome or not

```
n=int(input("enter n digit number\n"))
num=n
rev=0
while n>0:
    d=n%10
    rev=rev*10+d
    n=n//10
print("reverse",rev)
if rev== num:
    print("palindrome")
else:
    print("not palindrome")
```

```
enter n digit number
121
reverse 121
palindrome
```

write a program to find out the fibonnaci series upto n term

```

nterms = int(input("How many terms? "))
n1, n2 = 0, 1
count = 0
if nterms <= 0:
   print("Please enter a positive integer")
elif nterms == 1:
   print(n1)
else:
    print("Fibonacci sequence:")
    print(n1,n2)
    for i in range(3,nterms+1):
       nth = n1 + n2
       print(nth)
       n1 = n2
       n2 = nth

```

```
How many terms? 3
Fibonacci sequence:
0 1
1
```

Gateway classes

Write a program to find out the factorial of a number

```
num=int(input("enter the number\n"))
factorial = 1
if num < 0:
   print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
   print("The factorial of 0 is 1")
else:
   for i in range(1,num + 1):
       factorial = factorial*i
   print("The factorial of",num,"is",factorial)

```

input

```
enter the number
4
The factorial of 4 is 24
```

write a program to check whether a number is prime or not

```
num = int(input("Enter a number: "))
flag = False
if num == 1:
    print(num, "is not a prime number")

elif num>1:
    for i in range(2, num):
        if (num % i) == 0:
            flag = True
            break
    if flag:
        print(num, "is not a prime number")
    else:
        print(num, "is a prime number")
else:
    print("enter positive number")
```

input

```
Enter a number: 7
7 is a prime number
```

Gateway classes

write a program to print the reverse of a number

```
1 n=int(input("enter n digit number\n"))
2 num=n
3 rev=0
4 while n>0:
5     d=n%10
6     rev=rev*10+d
7     n=n//10
8 print("reverse",rev)
9
```

```
enter n digit number
1214
reverse 4121
```

What will be the output?

```
1 i=0
2 while i<3:
3     print(i)
4     i+=1
5 else:
6     print(0)
```

```
0
1
2
0
```

teeway classes

write a program to convert time from 12 hour to 24 hour format

```
str1="12:05:45 AM"
if str1[-2:]=="AM" and str1[:2]=="12":
    print("00"+str1[2:-2])
elif str1[-2:]=="AM":
    print(str1[:-2])
elif str1[-2:]=="PM" and str1[:2]=="12":
    Print(str1[:-2])
else:
    print(str(int(str1[:2])+12)+str1[2:8])
```

```
00:05:45
```

Gateway classes

break statement

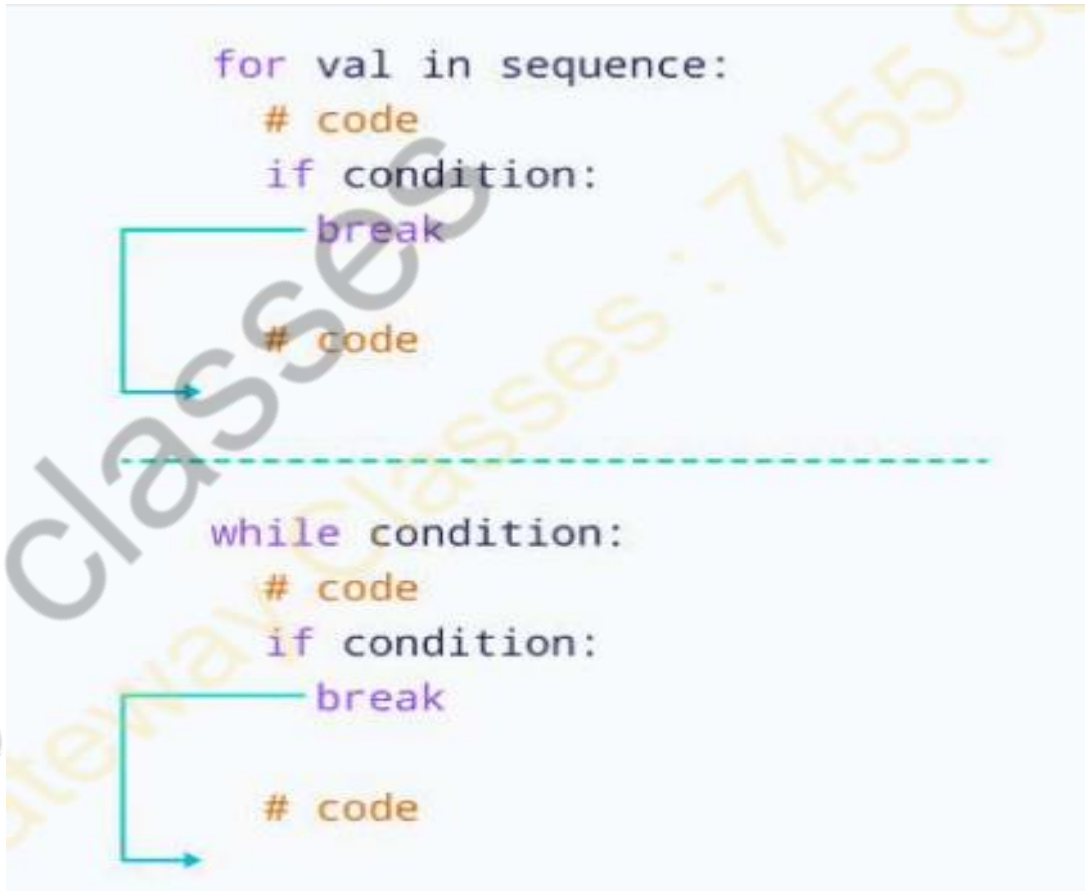
The break statement is used to terminate the loop immediately when it is encountered.

The syntax of the break statement is:

break

```
for i in range(5):
    if i == 3:
        break
    print(i)
```

```
0
1
2
```



Gateway classes

continue statement

The continue statement is used to skip the current iteration of the loop and the control flow of the program goes to the next iteration.

The syntax Continue;

```
for i in range(5):
    if i == 3:
        continue
    print(i)
```

```
0
1
2
4
```

```
for val in sequence:
    # code
    if condition:
        continue

    # code
```

```
while condition:
    # code
    if condition:
        continue

    # code
```

Gateway Classes : 7455 96

Continue Example

main.py

```
num = 0
while num < 10:
    num += 1
    if (num % 2) == 0:
        continue
    print(num)
```

```
1
3
5
7
9
```

Break example

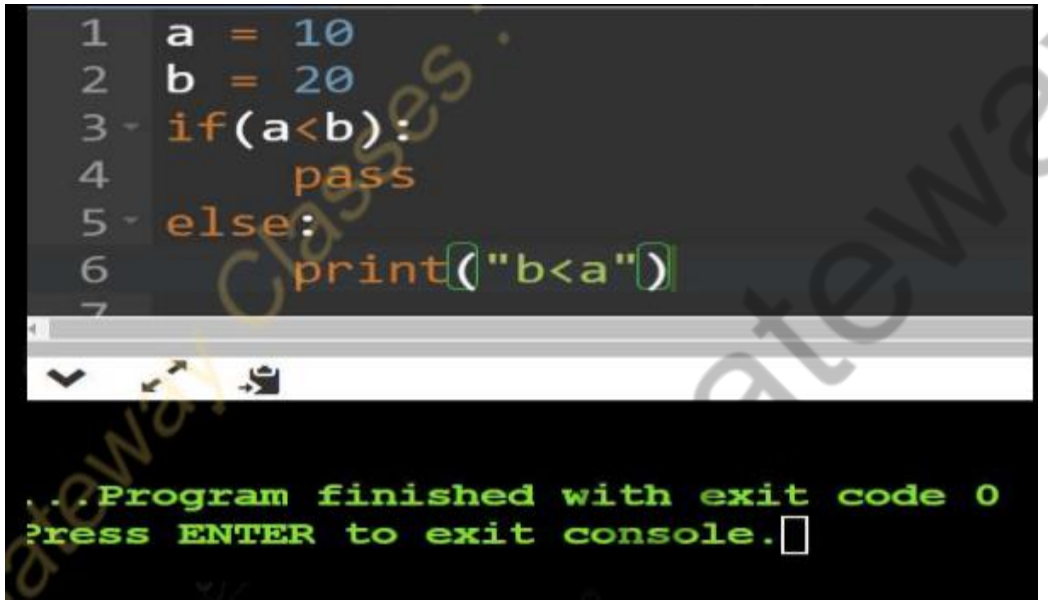
```
i = 1
while i <= 10:
    print('6 * ',(i),'=',6 * i)
    if i >= 5:
        break
    i = i + 1
```

```
6 *  1 = 6
6 *  2 = 12
6 *  3 = 18
6 *  4 = 24
6 *  5 = 30
```

Gateway Classes : 7455 9612

Pass example

- python pass statement is a null statement. But the difference between pass and comment is that comment is ignored by the interpreter whereas pass is not ignored.
- SYNTAX pass



What is pass statement in Python?

- When the user does not know what code to write, So user simply places a pass at that line.
- Sometimes, the pass is used when the user doesn't want any code to execute.
- So users can simply place a pass where empty code is not allowed, like in loops, function definitions, class definitions, or in if statements. So using a pass statement user avoids this error.

Gateway Classes

```
1  li =['a', 'b', 'c', 'd']
2  for i in li:
3      if(i =='a'):
4          pass
5      else:
6          print(i)
7
```

```
b
c
d
```

Gateway classes

Write a program to print the prime number in range

main.py

```
1  lower = int(input("enter the lower range"))
2  upper = int(input ("enter the upper range"))
3  print("Prime numbers between", lower, "and", upper, "are:")
4  for num in range(lower, upper + 1):
5      if num > 1:
6          for i in range(2, num):
7              if (num % i) == 0:
8                  break
9          else:
10             print(num)
11
```

input

```
enter the lower range2
enter the upper range7
Prime numbers between 2 and 7 are:
2
3
5
7
```

way classes