LECTURE 5- LOOPS IN PYTHON

Thanks for staying with me so long, I am Akshansh (+91 8384891269, akshanshofficial@gmail.com) and I will be discussing loops in python. This is lecture number 5, thanks for overwhelming support for the documented lectures and video lectures on You Tube too. Let's get started, pythonistas!

Let's think about the example that can be realistic here. You don't have a girlfriend and you are on facebook. So what will you do? Try on girls, right? What if one says no? Will you stop trying? No, no no (being an indian I know it ha ha ha). You will say, let's try on another one and another one and so on until one says "YES" to you. Once a girl says yes, you should stop, right? (if you don't want to get beaten by her).

See keep trying on is a loop, when you do same thing again and agai until you get the desired result, process is called looping.

What you do, you set an initialization (like you don't have a girlfriend; gf == 0) and then you put a condition (like if someone says yes to you) then you write what to do (like stop trying or continue). This is looping.

Liked my example? You can applause later on my whatsapp he he he.

Here is an example of WHILE loop

```
>>>num=1
>>>while (num<5):
>>>      print(num)
>>>      num=num+1
>>>else:
```

```
>>>    print("you are out of the loop now")
```

EXPLANATION: **num=1** is initialization, you have to start from somewhere right? This means that at the beginning, value of num is 1.

while (num<5) :
this means you are going to use while loop and condition is whenever the value of number (num) is less than 5, loop needs to work. **Don't forget to put colon: in the end of the condition.**

print(num)
this print number, like 1, 2, 3...

num=num+1
this is called increamention (increase or decrease). At the beginning, num value was set to 1, now it has become 2 (added 1 in num). Since condition (it is still less than5, right?) is still true, loop will run once again until value becomes more than 5.

else:
what happens when condition becomes false? You are out of the loop now. Else statement will be executed.

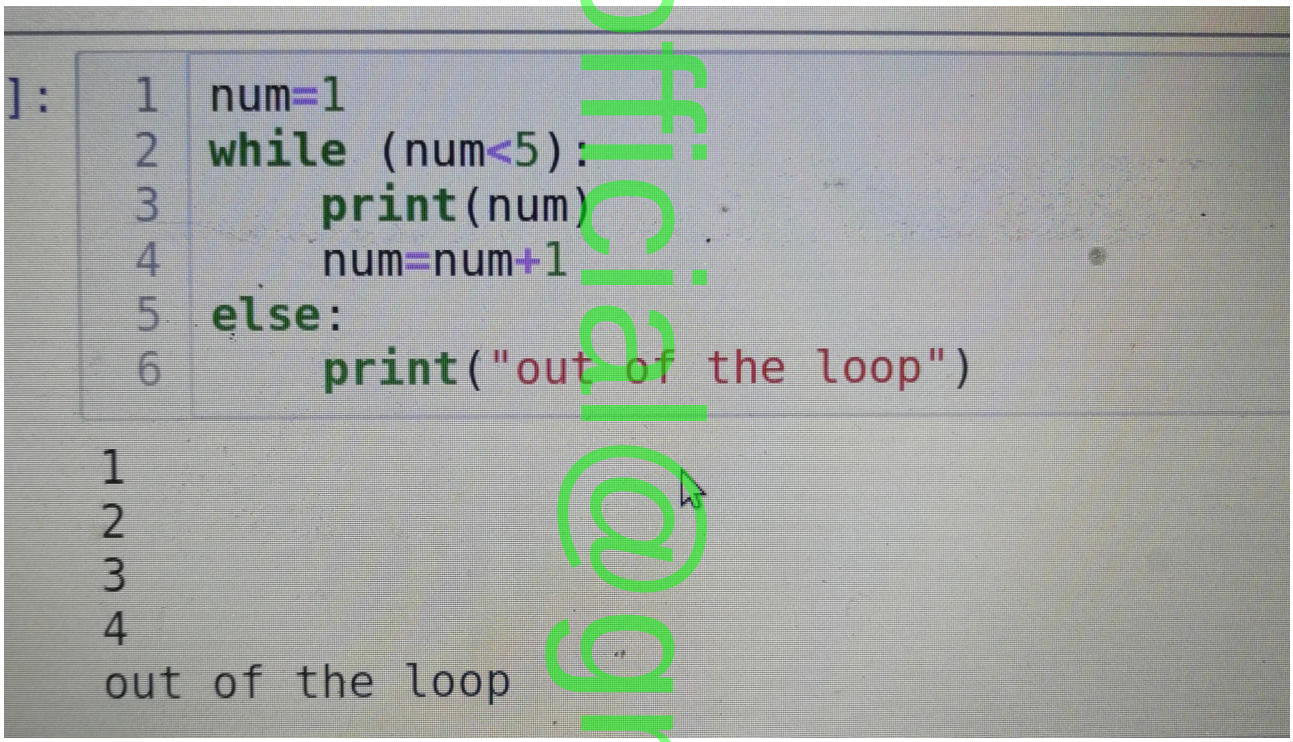
Example -

```
>>>num=1
>>>while (num<5):
>>>      print(num)
>>>      num=num+1
>>>else:
>>>    print("you are out of the loop now")
```

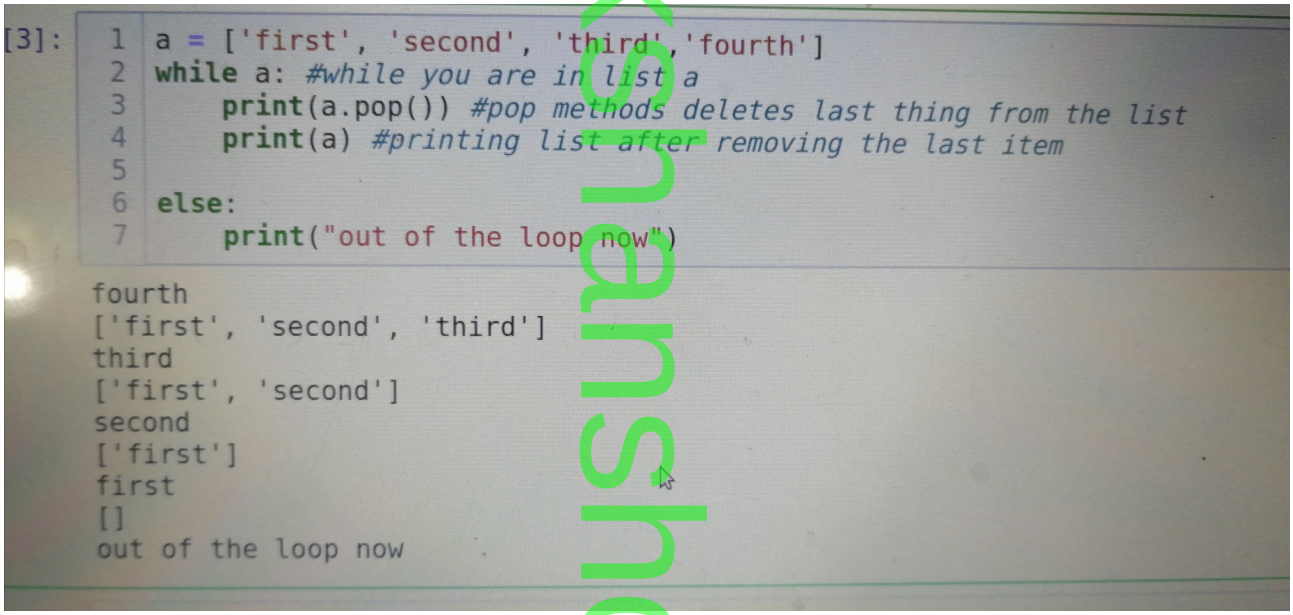
Let's talk about the output-
~num value is 1 at beginning.
~we are in loop and condition is true (num<5)
~print(num) will print value of num, that is 1 in this case.

~now we are on the next line, num=num+1 increases the value of number by one, 1+1=2
~again we are back at the top of the loop, since new value of num is 2 and it is still less than 5, condition (num < 5) is still true.
~print(num) will again run and it will print num value, that is 2 now

(3,4 will be printed like 2 was printed)
~now suppose value of num become 5, condition becomes false because 5 is not less than 5 (it is equal).
~condition has become false, you are not in the loop now and in this specific case else statement will be printed.



Example 2 – I've a list and I've to delete last item from the list. Condition is that as long as I am in the list, i've to delete last item from it.



```
[3]:
1  a = ['first', 'second', 'third','fourth']
2  while a: #while you are in list a
3      print(a.pop()) #pop methods deletes last thing from the list
4      print(a) #printing list after removing the last item
5
6  else:
7      print("out of the loop now")

fourth
['first', 'second', 'third']
third
['first', 'second']
second
['first']
first
[]
out of the loop now
```


Break and continue in while loop

Suppose you are in a while loop and in the middle of the condition, you want to quit. You'll use break here and if you want to continue, you'll use continue .

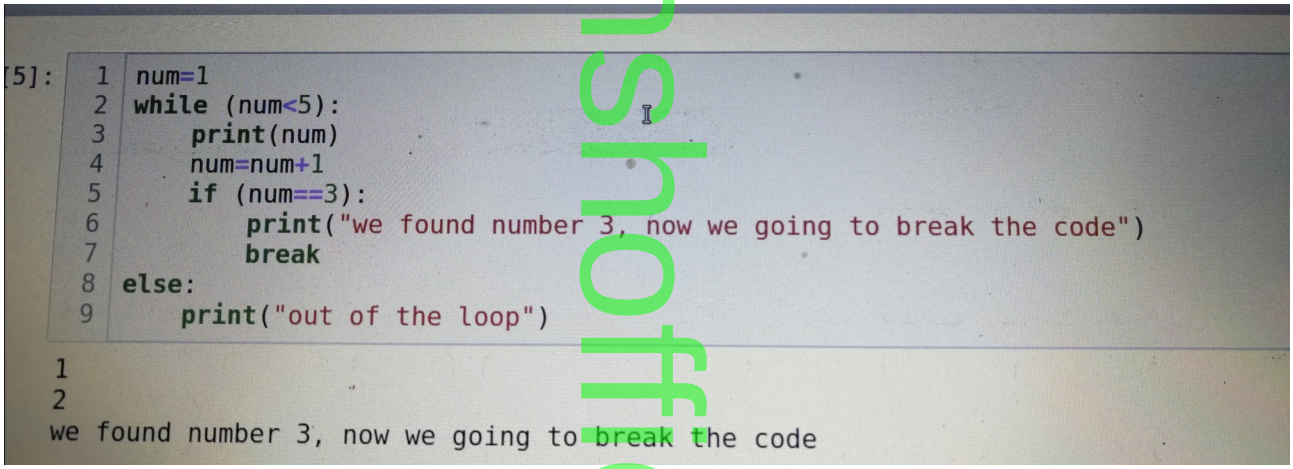
Using break keyword:

Suppose there is a condition that number should be less than 5, you have to print the number and then increase it by 1. Condition will be tested again for the same. But there is another condition, if you found number 3 in that loop. You don't want to continue, you'll be using "break" here.

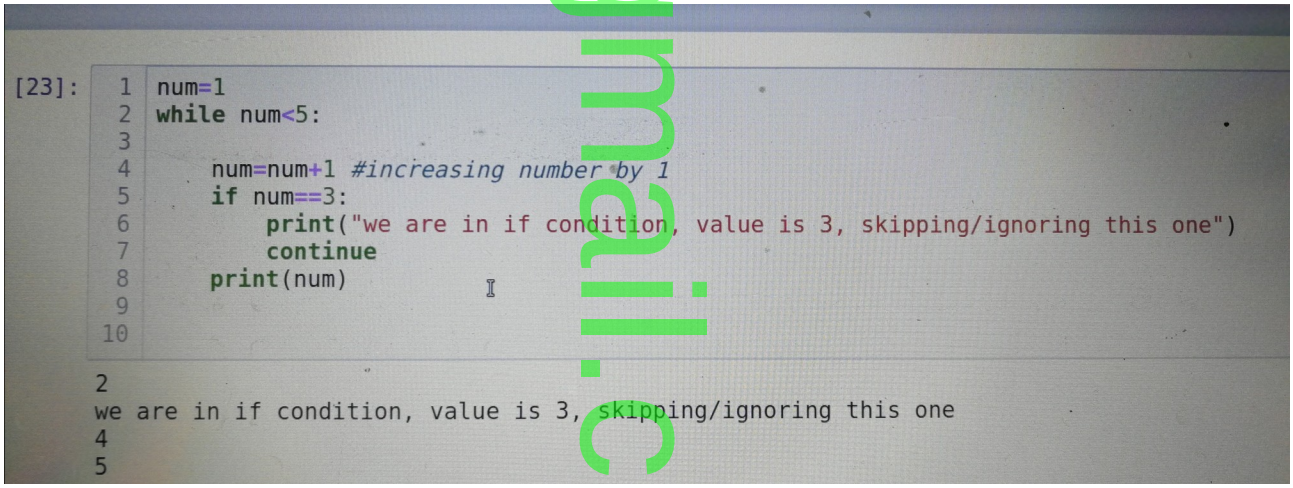
```
>>>num=1
>>>while(num<5):
>>>    print(num) #this will print num if it is less than 5
>>>    num=num+1 #this will increase the number by 1
>>>    if (num==3): #it will check if value of number is 3
>>>        print("we found number 3")
>>>        break
>>>else:
```

>>> print(“you are out of the loop now”)

O/P = 1 will be printed
2 will be printed
3 will not printed, as soon as number is equal to 3, it will print “we found number 3”
and then break will be executed



Using continue keyword: suppose if you find something in your code and you don’t want to print that result, what will you do, you will continue in that loop and ignore that particular result. Let’s take a realistic example, if you are looking for gf on facebook and you encounter with your ex. What is the choice? Ignore her/his and continue on searching. It is like that



Let’s have some questions-

Q1- Print numbers from 1 to 100 using while loops.
Q2- Print even number from 2 to 100 using while loop.
Q3- Print table of 3 using while loop.
Q4- print table of 10 and ignore 90 from the table.
Q5- Print table of 5 and break the table if you encounter 45 in the table.

Hope you are enjoying pythoning, this is all from my end. In next topic, we will be discussing “for” loop. Stay tuned.