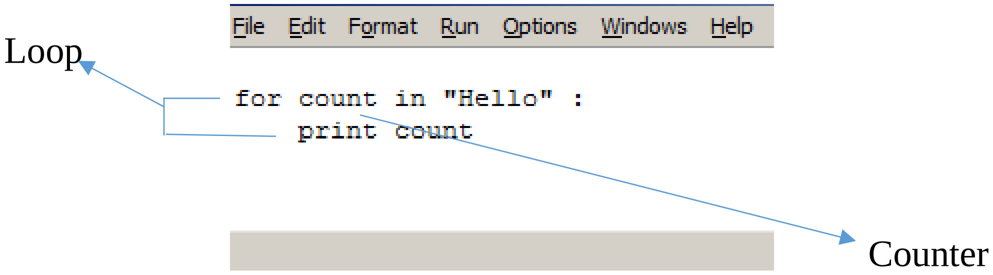
Python – For Loops

The **FOR** statement works a little differently than the **WHILE**. It has a built in counter. It also works with **text** or **numbers**:



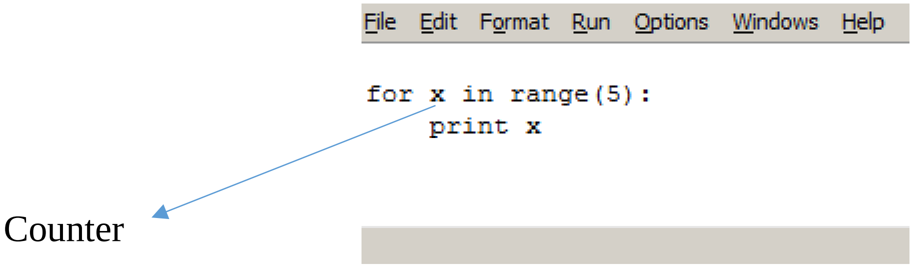
```
for count in "Hello" :
    print count
```

The Counter "count", will look at the text, 1 letter at a time and loop until it runs out of letters.

Output :

```
H
e
l
l
o
```

For numbers, you have to specify the **range ( ),** for the counter.



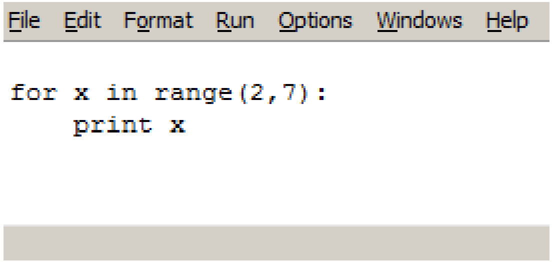
```
for x in range(5):
    print x
```

The Counter "x", automatically has a starting value of 0 and increments by 1's. **range(5)** tell it to loop until x=5 (it does not perform the loop when x=5)

Output :

```
0
1
2
3
4
```

To Specify a different starting value (besides 0) :



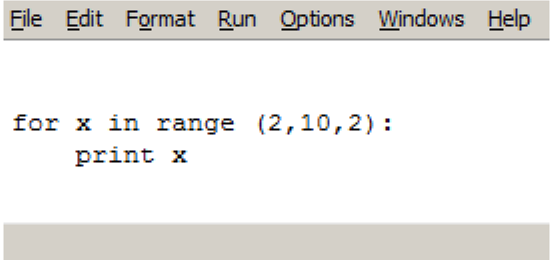
```
for x in range(2,7):
    print x
```

Range(2,5) - counts from 2 to 7 (not including 7)

Output :

```
2
3
4
5
6
```

To Increment by a different value than 1 :



```
for x in range (2,10,2):
    print x
```

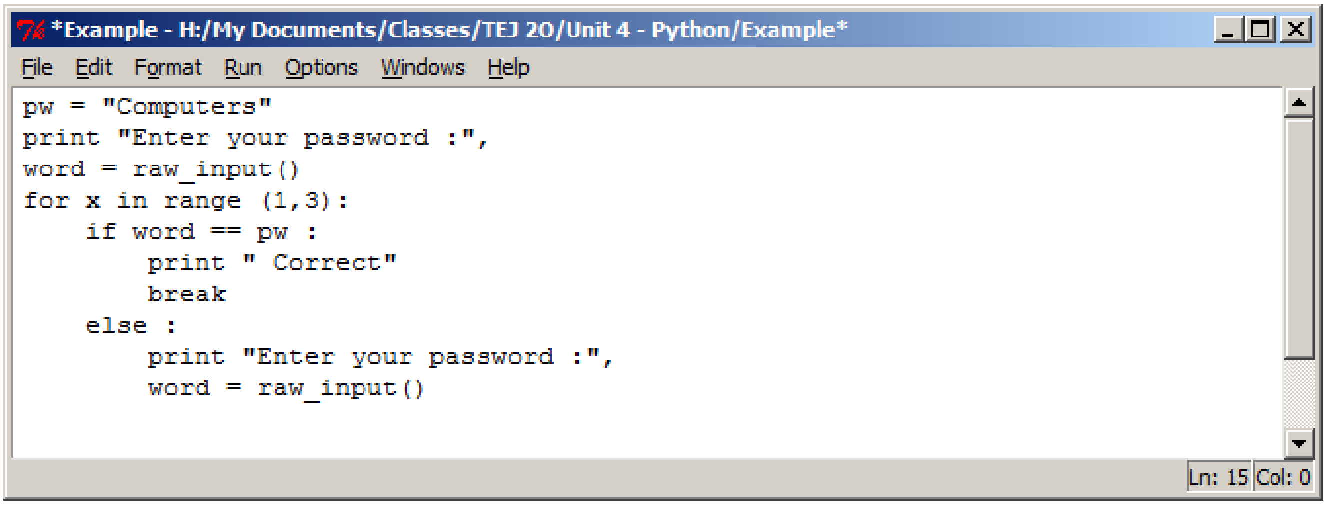
Range(2,10,2) - counts from 2 to 10 (not including) by increments of 2's

Output :

```
2
4
6
8
```

Exiting a Loop early :

* **Break** will exit a loop


Example - H:/My Documents/Classes/TEJ 20/Unit 4 - Python/Example
File Edit Format Run Options Windows Help
Ln: 15 Col: 0

```
pw = "Computers"
print "Enter your password :",
word = raw_input()
for x in range (1,3):
    if word == pw :
        print " Correct"
        break
    else :
        print "Enter your password :",
        word = raw_input()
```

* Ask for password (pw) if correct – Breaks loop, otherwise gives them 2 more attempt (loops from 1 to 2)