Loops (for, while)

1. Foor Loops:

The structure of a for loop is as follows:

```
for variable name in range ( number of times to repeat ) :
    statements to be repeated
```

The syntax is important here. The word ***for*** must be in lowercase, the first line must end with a colon, and the statements to be repeated ***must*** be indented. Indentation is used to tell Python which statements will be repeated.

Example 1: the following program will print the word ***Hello*** ten times:

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

- **The range function**: the value we put in the range function determines how many times we will loop. The way range works is it produces a list of numbers from zero the value minus one. For instance, range (5) produces five values: 0,1,2,3, and 4.

  If we want the list of values to start at a value other than 0, we can do that by specifying the starting value. The statement range (1, 5) will produce the list 1, 2, 3, 4. This brings up one quirk of the range function, it stops one short of where we think it should. If

E-mail: hasanein.y.m.alhrabi@uomus.edu.iq

we wanted the list to contain the number 1 through 5 (including 5), then we would have to do range (1, 6).

Another thing we can do is to get the list of values to go up by more than one at time. To do this, we can specify and optional step as the third argument. The statement range (1, 10, 2) will step through the list by twos, producing 1, 3, 5, 7, 9.

To get the list of values to go backwards, we can use a step of -1. For instance, range (5, 1, -1) will produce the values 5, 4, 3, 2, in that order. Note that the range function stops one short of the ending values. Here are a few more examples:

Statement	Values generated
`range(10)`	0,1,2,3,4,5,6,7,8,9
`range(1,10)`	1,2,3,4,5,6,7,8,9
`range(3,7)`	3,4,5,6
`range(2,15,3)`	2,5,8,11,14
`range(9,2,-1)`	9,8,7,6,5,4,3

- **The loop variable**: there is one part of a for loop that is a little tricky, and that is the loop variable. In the example below, the loop variable is variable i. the output of the following example will be the numbers 0,1, ……,99. Each printed on its own line.

```
for i in range(100):
    print(i)
```

When the loop first starts, Python set the variable i to 0. Each time we look back up, python increase the value of i by 1.

Exercises 1: what is the output of the following program:

```
for i in range(5,0,-1):
    print(i, end=' ')
print('Blast off!!')
```

```
5 4 3 2 1 Blast off!!!
```

Exercises 2: what is the output of the following program:

```
for i in range(4):
    print('*'*6)
```

```
******
******
******
******
```

2. While Loops: we have already learned about for loops, which allow us to repeat things about specified number of times. Sometimes, though, we need to repeat something, but we don't know ahead of time exactly how many times it has to be repeated. In this situation that would call for a while loop.