

```
In [0]: #This is a supplementary material to the lecture "Conditionals, Loops and Functions" to quickly revise, whenever needed
```

```
In [0]: #conditional statements executes set of instructions based on a condition is True or False
#conditional statement generally utilize arithmetic, logical and relational operators
#logical operators are 'and', 'or', 'not' in python
#relational operators are >, >=, <, <=, == same as in other programming languages like c++ or java
#Let's look at some of the examples
```

```
In [3]: if(5>2):
        print(True)
```

True

```
In [4]: #combining more than two expression
a = 10
b = 20
c = 30
if(a < c and b < c):          #indentation in python plays the same role as curly brackets in other
    languages like c++ or java
    print(c)
```

30

```
In [5]: #loops
#it works the same as in other languages, one thing to keep in mind is to have consistent indentation
        in python as we have curly brackets in other languages
i=1
while(i<=5):
    print(i, end = " ")
    i += 1
```

1 2 3 4 5

```
In [6]: #the same can be achieved as
for i in range(1, 6):
    print(i, end=" ")

#range(start, stop, step) function returns a sequence of numbers,
#starting from start (0 by default), and increments by step (1 by default), and ends at a stop - 1.
```

1 2 3 4 5

```
In [7]: #fast iterations
#particularly useful for iterating over collections of data
list1 = [1, 2.5, 4, 5, 6.0, 'end']
tuple1 = (1, 2.5, 4, 5, 6.0, 'end')
print('fast iteration over list')
for item in list1:
    print(item, end=' ')
print()
print('fast iteration over tuple')
for item in tuple1:
    print(item, end = ' ')
```

fast iteration over list
1 2.5 4 5 6.0 end
fast iteration over tuple
1 2.5 4 5 6.0 end

```
In [8]: #functions
        #let's understand with the help of an example
        #function to calculate the area and circumference of circle
        import math
        def my_func(radius):                                #def is the keyword for defining function
            area = math.pi * radius * radius
            circum = 2 * math.pi * radius
            return area, circum                             #more than one value can be returned in python

        area, circum = my_func(5)
        print('area: ', area, 'circumference: ', circum)

area: 78.53981633974483 circumference: 31.41592653589793
```

```
In [0]: #Thanks, Happy coding!
```

```
In [0]: #To download .ipynb notebook, right click the following link and click save as
        https://ninjasfiles.s3.amazonaws.com/0000000000003217.ipynb
```