

## Python assignment11

1) def statement is used to create a normal function whereas lambda expressions are used to create Anonymous functions which can be assigned to a variable and can be called using the variable later in function. Lambda's body is a single expression and not a block of statements like def statement.

```
x=lambda a,b : a+b  
print(x(2,3))
```

```
5  
def add(a,b):  
    print(a + b)  
add(4,5)
```

9

2) The following are some of the benefits of lambda expressions:

- 1.It can be used to create Nameless/Anonymous functions inside some complex functions if we are planning to use it only once.
- 2.Moderate to small functions can be created in a single line
- 3.Functions created using lambda expressions can be assigned to a variable and can be used by simply calling the variable

3) The differences between map, filter and reduce are:

- map(): The map() function is a type of higher-order. This function takes another function as a parameter along with a sequence of iterables and returns an output after applying the function to each iterable present in the sequence.
- filter(): The filter() function is used to create an output list consisting of values for which the function returns true.
- reduce(): The reduce() function, as the name describes, applies a given function to the iterables and returns a single value

```
from functools import reduce  
# map function  
print('Map ->',list(map(lambda x:x+x, [1,2,3,8])))  
# filter function  
print('Filter ->',list(filter(lambda x:x%2 !=0, [1,2,3,9])))  
# reduce function  
print('Reduce ->',reduce(lambda x,y:x+y, [1,2,3,4,5,7]))  
Map -> [2, 4, 6, 16] Filter -> [1, 3, 9] Reduce -> 22
```

4) Function annotations provide a way of associating various parts of a function with arbitrary python expressions at compile time.

Annotations of simple parameters `def func(x: expression, y: expression = 20):`

Whereas the annotations for excess parameters are as `-def func (**args: expression, **kwargs: expression):`

Purpose of Function Annotations:

- 1.Python supports dynamic typing and hence no module is provided for type checking.
- 2.String based annotations can be used by the libraries to provide better help messages at compile time regarding the functionalities of various methods, classes and modules.

7) Some of the ways in which a function can communicate with the calling function is:

- 1.print
- 2.return
- 3.yield

