WHAT ARE PYTHON LAMBDA FUNCTIONS?

* Python lambda functions are:
  + Anonymous or nameless functions
  + ‘lambda’ is not a name but it is a keyword

WHY ARE THEY USED?

* One - time Use:
  + Also known as throw away functions as they are needed just once.
* I/O of other functions:
  + They are also passed as inputs or returned as outputs of other higher – order functions.
* Reduce Code Size:
  + The body of lambda functions is written in a single line.

HOW TO WRITE ANONYMOUS FUNCTIONS?

* A lambda function is created using the lambda operator.
* Syntax:
  + Lambda arguments: expressions
  + Lambda : “specify the purpose”
  + lambda a1 : “specify use of a1”
  + lambda a1…n : “specify use of a1…n”
* #lamda arguments : expression

x = lambda a : a\*a

x(3)

Output: 9

ANONYMOUS FUNCTIONS WITHIN USER DEFINED FUNCTIONS

* Lambda functions are best used within other higher – order functions.

USING LAMBDA FUNCTIONS WITHIN filter(), map(), reduce()

* Lambda within filter():
  + Used to filter the given iterables(lists, sets, etc) with the help of another function passed as an argument to test all the elements to be true or false.
  + mylist = [1,2,3,4,5,6]

newlist = list(filter(lambda a : (a/3==2), mylist))

print(newlist)

output: [6]

* Lambda within map():
  + Applies a given function to all iterables and returns a new list.
  + #map(func,iterables)

mylist = [1,2,3,4,5,6]

newlist = list(map(lambda a : (a/3!=2), mylist))

print(newlist)

Output: [True, True, True, True, True, False]

* Lambda within reduce():
  + Applies some other functions to a list of elements that are passed as a parameter to it and finally returns a single value.
  + #reduce(function, sequence)

from functools import reduce

reduce(lambda a,b : a + b, [10,20,30,40,50])

Output: 150

SOLVING ALGEBRAIC EXPRESSIONS USING LAMBDA

* Linear Equations
  + s = lambda a : a\*a

s(4)

Output: 16

* + s = lambda x,y : 3\*x + 4\*y

s(4,7)

Output: 40

* Quadratic Equations
  + #(a+b) ^ 2

s = lambda a,b : (a+b) \*\* 2

s(2,3)

Output: 25