**Python training Notes:**

**Course Name:** **SCRIPT 307: Basic Python**

**This is Part 1 of the whole training in the duration 17 to 31 July**

**This will be followed with next Part 2 session for Intermediate Python topics in the month of August.**

**Day 3: 19 Jul 2018 - Thursday (2 Hrs Session)**

**Expectation Setting ASL (Assisted Self-Learning) 2Hrs session daily**

**And then do self-study and hands on assignments from below learning course link and also the assignments given below here in this document:**

<https://knowledgecenter.persistent.co.in/ViewCourse/pmoc>

***Please visit the following URL to view the collaborative learning group***

<https://persistentuniversity.persistent.co.in/CollaborativeLearningGroup/view.aspx?SkillId=9144>

**Topics Covered:**

List, Tuple, Dictionary

Functions for List, tuple

Set

**\*\*\*\*\*To Do for Day 4:**

Nugget 1 : Introduction to Python & Python Fundamentals

Nugget 2 : Python Basics

Nugget 3 : Python Control Structures

Subjective Assignment for Nugget 1 to 3 : Only for self Practice

522

1. Complete reading these 3 Nuggets from <https://knowledgecenter.persistent.co.in/ViewCourse/pmoc>

2. Please execute all codes in these 3 Nuggets

3. Start solving assignment at the end of Nuggets

**Try Below Codes:**

**10\_Set\_demo.py**

#set is unordered collection of elements

listA=[1,2,3]

#create a set

s= set(listA)

print "Set = ", s

s1 ={1,2,3,3} #unique elements 1,2,3 taken

print "s1 = ", s1

print "One element = ", s1[0]

#set union

setOne = set([1, 2, 3])

setTwo = set([3, 4, 5])

print setOne | setTwo #returns me other set with unique elements set([1,2,3,4,5])

#set intersection

print setOne & setTwo #returns me other set with onlyu common elements set([3])

**11\_Dictionery\_Demo.py**

dictionaryOne = {}

dictionaryTwo = {'name': 'python', 'course': 'moc'}

print dictionaryOne, dictionaryTwo

print "name = ",dictionaryTwo['name']

print dictionaryTwo.has\_key('name')

#add new entry

dictionaryTwo['id']=100 #craetes key-value pair 'id':100

print dictionaryTwo

#delete pair

del dictionaryTwo['id']

print "after deletion = ", dictionaryTwo

dictionaryTwo.clear()

print "after claering = ", dictionaryTwo

dictionaryFour = {'name': 'python', 'course': 'moc'}

del dictionaryFour

#sprint dictionaryFour #not accissible as deleted, NameError if try to access

dictionaryTwo = {'name': 'python', 'course': 'moc'}

print dictionaryTwo.items()

#[('course', 'moc'), ('name', 'python')] list of tuple key-value pairs

print dictionaryTwo.keys() #list of keys

print dictionaryTwo.values() #list values

print "-----------------------------------------"

for i in dictionaryTwo.keys():

print i ,"\t = ",dictionaryTwo[i]

print "-----------------------------------------"

**13\_for\_demo.py**

#for loop

x = [1.0, 2.0, 3.0] #access first element x[0] o/p 1.0

for n in x:

print(n)

n=n\*\*2 #modifying n does not change X list element

print x

print "--------------------------------------------"

# for and range

x = [1, 3, -7, 4, 9, -5, 4]

for i in range(len(x)): #len(x) is 7 range(7) =[0,1,2,3,4,5,6]

#i is indexes of your list

print i ," = ", x[i]

"""

if x[i] < 0:

print("Found a negative number at index ", i)

x.append(1,2)

"""

**Assignments to do:**

1. Define dictionary as

emp ={'1a':30000,'2a':40000}

1. Add few more pairs by accepting it from keyboard.

1 line entry from keyboard can be like this for 1 pair – “3a:35000”

Accept such user defined number of pairs.

1. print all employee details in sorted keys order
2. Increment the salary of every person by 5000 and then print the updated emp
3. Print the total sal of all employees.

Save the solutions in a folder: **Assignments\Day3**

Script names should be Q1.py, Q2.py, Q3.py

**Assignments\Day3 --🡪**

**Q1.py**

**Q2.py**

**Q3.py**