**Python training Notes:**

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

**Day 1: 17 Feb**

**Post the session from my side do self-study and hands on assignments form below learning course link:**

<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=10329>

**Topics Covered:**

Python Basic fundamentals

Python Interpreter

Python features

String

if, elif, while, break, continue

for

Python Interpreted lang

Open source – Python installation Windows open source/ Unix

Interpreter of Python

Top- Bottom way---line by lie execution – Interpreted lang

No intermediate obj file

Source File (First.py) ---🡪>python First.py ---🡪 o/p

Int x =100

Char c = ‘A’

Libraries in Python - modules/packages

Server side script

html + Java Script .jsp .servlet .asp

Web browser

Web Client ------------------------------>WebServer

http request html + Java Script .jsp .servlet .asp

.py

.CGI

<-------------------------------

http response

Executing Python Script:

1. By using Python software on Windows, use Python command prompt

**C:> python**

**>>>print ‘Hell0’**

**>>>Hello**

**>>>3+5**

**>>>8**

1. **File creation for Python program,**

**First.py**

**Text editor**

**Pre requisites : python software , text editor Notepad/Notepad++**

**OR IDE (Integrated Dev Env) - Eclipse/PyCharm/Idle editor**

**C:>Demos>python Hello.py**

**Python.exe in c:\python**

**Path env variable**

**Python is installed at C:\Python27\python**

**this executable command python is an interpreter for you**

**C:\Python27\python command is accessible in other folders as well due to 'path' system**

**environment variable setting**

**to check this path env variable--->**

**1)on dos propmt type path to see the current path content**

**c:\path ----->**

**to add path execute the command**

**c:\set path=%path%;c:\python27**

**2) in computer settings :**

**Windows explorer --->this PC ---->rt click---->properties ---->Advanced system settings---->**

**Environment Variable button ---->path ---->edit this (add c:\python27)**

**3.IDLE used**

**In some languages -**

**If(){**

**}**

**elif(){**

**}**

else

{

}

But in python:

if z==3.56:

Print’’

X+y

else:

Print ‘in else block’

print ‘Out of else block’

X =’Hello’s’

**Reserved Keywords**

elif except from in lambda pass raise

**Errors:**

**Errors:**

1. **SyntaxError**

**Example 1:**

**print “ABC**

**Example 2:**

**s3 = 'A'BC' #SyntaxError**

**s33 = "A"BC" #SyntaxError**

1. **Indentation Error**

**RunTime Errors: pre defined classes**

1. **NameError:**

**Example1:**

print a #NameError: name 'a' is not defined

#python interpreter on print statement execution **raised an object of predefined class error NameError**

1. **TypeError**

**Example1:**

print num1 + s1 #TypeError: unsupported operand type(s) for +: 'int' and 'str'

\

**Example 2:**

t1[1] = 999 #TypeError: 'tuple' object does not support item assignment

1. **IndexError**

t1 = (123, "abc", 3.14)

print t1[6] #IndexError: tuple index out of range

1. **ValueError**

**Example 1:**

s1 ="A"

num1 = int(s1)#ValueError: invalid literal for int() with base 10: 'A'

**Example 2:**

t1 = (123, "abc", 3.14)

print **t1.index("ABC")** #ValueError: tuple.index(x): x not in tuple

**\*\*\*\*\*To Do for Day1:**

Nugget 1 : Introduction to Python & Python Fundamentals

Nugget 2 : Python Basics

Nugget 3 : Nugget 3 : Python Control Structures

522

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

2. Please execute all codes in these Nuggets

3. Start solving assignment at the end of Nuggets

**Try Below Codes:**

**2\_DataType1.py**

x=100

z=3.45 #no data type

y='Hello'

#print "Started"

#y=y+1 run time error no concatenation of int to str object

if z==3.45 or y=='Hello':

x =x+1

y=y+' World' #+ is concatenation operator

else:

print 'In else block'

print "x = ",x

print 'y = \t value',y

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

print """a'b"c""" """xyz'm"a"""

print int(z)

print float(x)

print "Kinght's" "abc'z"

print 'Kinght"s' 'abc"z'

#try except

#print x+"AAAA" #not poosible GIVES U run time error TypeError

#raw string

print 'c\temp\new.txt' #\t and \n escape charaters are resolved

print r'c\temp\new.txt' #raw string it considers this as raw string as it is

str1 = r'c\temp\new.txt'

print "str1 = ", str

"""Doc string This is Data types in

Python

Interesting comment

"""

str ='ABCDE' #012 index

print str[0] #A

print str[0:3] #ABC

print str[2:] #CDE

print str[:4] #ABCD

s ='1234765675565567547646464'

print s[0]

**3\_DataType2\_String.py**

s = '012345' #every character is stored at index

print "oth index charater = ",s[0] #09

print "String from 1 to 4 index = ",s[1:4] #range 1:4 123

print "String after index 2 =",s[2:] #2345

print "String upto 4 index = ",s[:4] #0123

str ="Welcome to PSL. PSL Pune"

count = str.count('PSL',0,len(str))

print 'Cunt of PSL = ',count

print "one, %d, two"%2 # one, 2, two

print '%s two %s'%(1,'three')

print 'Hello\n'\*3 #printed thrice

print 'h' in "hello" #return True

"""

'AB"C' possible

"AB'C" error

"""A'B"C"""

DocString

"""

Type

“””

s=’ABC’

revStr = s[::-1] #reversed string

**4\_user\_input.py**

#keyborad input

x = raw\_input("Please enter ur name :") #input is "" string str = "55" num = int(str)

print "Hello ", x

print "Welcome to Python"

print 5\*x

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

y = input("Enter your name") #input is not "" string

print "Hello", y # error, so take input as "ABC" it will work use input for numeric input

"""

Case 1:

Enter your name : ABC

NameError: name 'ABC' is not defined

Case 2:

Enter your name : "ABC"

Value1 = ABC

Type of value1 = <type 'str'>

Case 3:

Enter your name : 99

Value1 = 99

Type of value1 = <type 'int'>

Case 4:

Enter your name : 3+4

Value1 = 7

Type of value1 = <type 'int'>

Case 5:

Enter your name : float(100)

Value1 = 100.0

Type of value1 = <type 'float'>

"""

**6\_List\_Tuple.py**

#List data tuple data

#list ---->mutable

listOne= [123, 'abc', 4.56]

print listOne

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

#Tuple --->immutable

tupleOne= (123, 'abc', 4.56)

print tupleOne

#inner list

listTwo= [4.56, ['inner', 'list']]

print listTwo

#inner Tuple

tupleTwo = (123, 'abc', 4.56, ['inner', 'tuple'], 7-9j,[10,20])#tuple having one of the elements as mutable list

print tupleTwo

print "First element of list = ", listOne[0]

print "First element of tuple = ", tupleOne[0]

print listTwo[0:1]

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

listFour= [123, 'abc', 4.56, ['inner', 'list']]

print "3rd index inner list = ", listFour [3]

print "3rd index inner list = ", listFour [6] #run time error

print "first index element of inner list = " , listFour [3][1]

**7\_List\_Update.py**

#Updating List data:

#Example 1: Replacing a value in List

#append insert remove pop extend reverse

listOne=[100,200,'hello']

print listOne

listOne[2] =200

print listOne

print "last element = ",listOne[-1]

listOne.append(500)

print listOne

listThree= [123, 'abc', 4.56]

listThree.insert(2,'c')

print listThree

listThree.remove(4.56)

print listThree

#pop

listSix= [123, 'abc', 4.56]

print listSix.pop()

print "Modified List =", listSix

listSix.pop(1)

#listSix.pop(5) run time error IndexError

print "Modified List =", listSix

listTwo= [123, 'abc', 4.56]

print 'Original list = ', listTwo

listTwo.reverse()

print "reversed list =", listTwo

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

listOne= [123, 'abc', 4.56]

listOne.append([1, 2])

print "Extended list = ",listOne

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

print 'abc' in listTwo

print 'abc' not in listTwo

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

number\_list = [43, -1.23, -2]

string\_list = ['hello', 'world']

print string\_list + number\_list

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

listX =[10,0,44,55,77]

print "Original X list = ",listX

listX.sort()

print "Sorted X list = ",listX

**8\_Tuple\_del.py**

"""Tuple example - immutable"""

tupleOne= (123, 'abc', 4.56) #tuple defination tupleOne

tupleTwo= (789, 'def', 2.24) #tuple defination tupleTwo

tupleThird= tupleOne[0], tupleTwo[1] #Tuples are immutable which means they cannot be updated or change values of

print "Count of 123 in tupleOne = ",tupleOne.count(123)

#tuple values.But they allow to take portions of Tuples to create a new Tuple.

print "tupleOne = ",tupleOne #tupleOne = (123, 'abc', 4.56)

print "tupleTwo = ",tupleTwo #tupleTwo = (789, 'def', 2.24)

print "New tupleThird = ",tupleThird #New tupleThird = (123, 'def')

del tupleThird #Removing individual tuple elements is not possible though it is possible to

#delete an entire tuple with del function

#print tupleThird #run time error

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

t1 =(1,2,3)

print "Original Tuple = ", t1

l1 = list(t1)

print "Converted list by list function = ", l1

l1.append([100,200])

print "Appended list = ", l1

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

newT1 = tuple(l1)

print "New tuple from tuple method = ",newT1

"""

len(list1)

id(listOne)

type(listOne)

del

"""

**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)

"""

"""Keyboard input raw\_input() input() """

name = raw\_input("Enter your Name : ")

print "Hello ", name #"name" is a string

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

name1 = input("Enter name = ")

print "Hello ", name1

"""

1) ABC

NameError: name 'ABC' is not defined

2) "ABC"

Hello ABC

3)100

Hello 100

4)3+4

7

"""

------------------------------------------------------------------------------------------------------------------------------------------

**Assignments to do:**

1. Accept a string, check whether it is palindrome

Eg. “madam”

1. Accept 2 numbers from user and print addition , subtraction.

Use input() function for this

1. Check whether entered string is lower case or uppercase and print the message respectively.
2. Validate the entered string for a valid password “psl” or “PSL”
3. Print addition of all elements of a list containing all numeric elements.
4. Accept the list elements from keyboard for user accepted number (one element per line )and then print all elements in ascending sorted order.

Save the solutions in a folder: **Assignments\Day1**

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

**Assignments\Day1 --🡪**

**Q1.py**

**Q2.py**

**Q3.py**

**Q4.py**

**Q5.py**

**Q6.py**