**Python Assignment 1var2**

1. Create following variables in Python and print the type of each variable and explain if you face any errors
   1. a = 23
   2. var1 = 12.9
   3. var2 = ‘hello’
   4. var3 = “I’m Good”
   5. 5 = ‘python’

>>> var1= 12.9

>>> var1 = 12.9

>>> var2 = 'hello'

>>> print(var2)

hello

>>> 5 ='python'

File "<stdin>", line 1

SyntaxError: can't assign to literal

>>>

1. Write a program to take following user inputs using input() function, print it’s type to stdout and explain if you face any errors
   1. 256
   2. 2 + 3j
   3. Hello World
   4. ‘Hello World’

>>> a = input()

256

>>> print (type(a))

<type 'int'>

>>> a = input()

2 + 3j

>>> print (type(a))

<type 'complex'>

>>> a = input()

Hello Worls

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

File "<string>", line 1

Hello Worls

^

SyntaxError: unexpected EOF while parsing

>>> a = input()

Hello World

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

File "<string>", line 1

Hello World

^

SyntaxError: unexpected EOF while parsing

>>> a = input()

'hello world'

>>> print (type(a))

<type 'str'>

1. Repeat problem (2) using raw\_input() function

KeyboardInterrupt

>>> a = raw\_input()

256

>>> print (type(a))

<type 'str'>

>>> a = raw\_input()

2 + 3j

>>> print (type(a))

<type 'str'>

>>> a = raw\_input()

hello world

>>> print (type(a))

<type 'str'>

>>> a = raw\_input()

'hello world'

>>> print (type(a))

<type 'str'>

1. Write a program which takes 100, 256, ‘Hello’, ‘World’ inputs using input() function and print the following:
   1. Output of 100 + 256 using print statement
   2. Print Hello World using print statement (There is a space between Hello and World)
   3. Print HelloWorld using print statement (There is no space between Hello and World)
   4. Concatenate 100 and ‘Hello’ and explain the output

var1 = input()

var2 = input()

var3 = input()

var4 = input()

print (var1+var2)

print (str(var3)+''+str(var4))

print (str(var3)+str(var4))

print (var1+str(var3))

1. Repeat problem (4) using raw\_input() function
2. Write a program which accepts user input and then prints whether it is even or odd integer

num = int(input("enter a number"))

if (num % 2 )==0

print ("the given number is even")

else:

print ("the guven number is odd")

1. Write a program which takes user input and then does following:
   1. If value is greater than 0 then print ‘positive’
   2. If value is less than 0 then print ‘negative’
   3. If value is equal to 0 then print ‘zero’
   4. Else print ‘Error’

from somewhere import print

val = evl(input("enter the number:"))

if val > 0:

print('positive')

else:

if val < 0:

print('negitive')

else:

if val ==0:

print('zero')

else:

print('error')

1. Write a program to do the following:
   1. Create an empty string and print it to stdout
   2. Create a string variable and pass ‘pseudonymous’ using input()
   3. Print the length of value passed in (b)
   4. Iterate through the string created in (b) and print character and its corresponding index number onto stdout (like p, index is 0 – s, index is 1 etc.,)
2. What is slicing and membership in strings. Explain with examples

We can make a new list from a portion of an existing list using a technique known as slicing. A list slice is

an expression of the form

list [ begin : end ]

where

• list is a list—a variable referring to a list object, a literal list, or some other expression that evaluates

to a list,

• begin is an integer representing the starting index of a subsequence of the list, and

• end is an integer that is one larger than the index of the last element in a subsequence of the list.

lst = [10, 20, 30, 40, 50, 60, 70, 80]

print(lst) # [10, 20, 30, 40, 50, 60, 70, 80]

print(lst[0:3]) # [10, 20, 30]

print(lst[4:8]) # [50, 60, 70, 80]

print(lst[2:5]) # [30, 40, 50]

print(lst[-5:-3]) # [40, 50]

print(lst[:3]) # [10, 20, 30]

print(lst[4:]) # [50, 60, 70, 80]

print(lst[:]) # [10, 20, 30, 40, 50, 60, 70, 80]

print(lst[-100:3]) # [10, 20, 30]

print(lst[4:100]) # [50, 60, 70, 80]