

TASK ONE: NUMBERS AND VARIABLES

- 1.1) Create three variables in a single line and assign different values to them and make sure their data types are different. Like one is int, another one is float and the last one is a string.

```
a,b,c = 1, 12.45, 'Akshay'  
print(a,b,c)
```

- 1.2) Create a variable of value type complex and swap it with another variable whose value is an integer

```
a=10  
b=50+1j  
print("a =",a)  
print("b =",b)  
a,b=b,a  
print("a =",a)  
print("b =",b)
```

- 1.3) Swap two numbers using the third variable as the result name and do the same task without using any third variable.

```
a=20  
b=30  
res = a  
a = b  
b = res  
print("a =",a)  
print("b =",b)
```

- 1.4) Write a program to print the value given by the user by using both Python 2.x and Python 3.x Version.

```
#python2  
a= raw_input("Enter No. ")  
print(a)
```

```
#python3
a = input('Enter No. ')
print(a)
```

1.5) Write a program to complete the task given below:

- i) Ask the user to enter any 2 numbers in between 1-10 and add both of them to another variable call z.**
- ii) Use z for adding 30 into it and print the final result by using variable results.**

```
x = eval(input('Enter a number between 1-10: '))
y = eval(input('Enter a number between 1-10: '))
z = x + y
result = z + 30
print(result)
```

1.6) Write a program to check the data type of the entered values. HINT: Printed output should say - The input value data type is: int/float/string/etc.

```
user_input = eval(input('Enter something: '))
print("The input value data type is : ", type(user_input))
```

1.7) If one data type value is assigned to 'a' variable and then a different data type value is assigned to 'a' again. Will it change the value? If Yes, then Why?

If one data type is assigned to 'a' variable and then a different data type value is assigned to 'a' variable again, it will change the value and give the latest change when asked.

This is because of one feature of python which is imperative. Python is imperative and thus any variable which is assigned a value can be updated ahead in the program without any error.

```
a = 10
print(type(a))
a = 'abc'
print(type(a))
```