Data types in Python

Variable: value that you are assigning

Variables starts with {a-z, A-Z, and \_} only symbol allow is \_

Do not use reserved keyword eg {for, if else, while etc}

E.g. cloud\_provider\_1 = “google”, Cloud\_provider\_2 = “aws”

Cloud\_provider\_1 and Cloud\_provider\_2 are all VARIABLES while google and AWS are values

String = collection of characters. When writing string you can use ‘ ’, “ ”, or “”” “””

“”” “”” is used when writing multiple lines codes

“”” my name is Lyonga

and I’m learning python

on this day”””

You can use number anywhere but not in the beginning

Numerical = int, or float

Boolean = True or False

Primitive = basic building blocks e.g String, Numericals (integer and float) and Booleans

Non-primitive = depends on primitive e.g list, tuple, dictionary set

Collection or group

1 List: To create a list use [] and when indexing a list or group use []. One can have a list within a list eg lyonga [‘python’, 1, 2, 2.5, [5, 7.9, 10]]

The len(lyonga) = 5

To call a number in the list in the list e.g if I want to print 10

Lyonga[4][2] This is called indexing

The list in a list is considered one entry eg [5, 7.9, 10]

2 Tuple: To create a tuple, start with ()

e.g tup\_var = (‘python’), 20, 1.5, [1,2], (5,6))

3 Set: Set start with {} and its unordered meaning printing will rearrange you set from small to big

NB: because of the unordered nature of SET, you can’t index it.

4 Dictionary: start with {} and used key and value

Key: create key only with immutable data types

Value: can e any data type

Mutable: can be changed e.g., List, Dictionary, Set

Immutable: cannot be changed once it is created e.g., string, integer, float tuple, Boolean

With Dictionary you can only index the key to get the value and NOT the value to get the key

Mutable.

CURD:

C = Create – add one object at the end of the list

U = Update

R = Read – use [] indexing

[single] -index

[two] -start: end+1

Three -start: end+1:step

D = Delete: pop -remove the item by passing its index number

FUNCTIONS

Function name is: dir(var or data structure) list all the functions

Id(var) gives the memory location of the var

NB: for any function if you want to call it

Func\_name(optional based on the func: input/inpusts)