Python Basics

Before we begin

We won’t be concentrating on theory aspects but implementation only.

We won’t be diving deep into the syntax, only covering what is required to getting started with ml.

Don’t forget to indent your code! Use tab or four spaces but be consistent throughout.

Python is a case-sensitive language

Practice and explore rest.

HELLO WORLD

1. Open editor : gedit hello.py - .py is the extension for python scripts
2. Write the code

```
if __name__=="__main__":            #main function : execution starts from 0 indented block

        print("Hello world")     #indented block

                                 #print : used to print to console ,python3 mandates use of ()
```

3. Run the script : python3 hello.py

Lets change previous program to add two numbers

```
if __name__=="__main__":

    a=2                                  #no data type required like C

    b=3

    print(a+b)

    b=”this is a string now”

    print (b)                            #try and see what would happen
```

Let’s start with basic Data types

Every value in Python has a datatype. Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

1. Python Numbers

```
#lets try the below code

a = 5                             #Integers can be of any length, it is only limited by the memory available.

print(a, "is of type", type(a))   #type function returns the type of an obejct



a = 2.0

print(a, "is of type", type(a))
```

2. Lists

List is an ordered sequence of items. It is one of the most used datatype in Python and is very flexible. All the items in a list do not need to be of the same type.

Items separated by commas are enclosed within brackets [ ].

```
list1=['string',1,20.0, "hey"]        #lists are mutable

print(list1[0])

print(list1[1])        #accesing elements at particular index

print(list1[2])

print(list1[0:3])              #slicing: 0 to 2 , last element is not included

print(list1[:3])

print(list1[2:])
```

3. Tuples

Tuple is an ordered sequence of items same as list.The only difference is that tuples are immutable. Tuples once created cannot be modified.

Tuples are used to write-protect data and are usually faster than list as it cannot change dynamically.

```
tuple1=('string',1,20.0, "hey")      #tuples are enclosed within ( and )

print(tuple1[0])

print(tuple1[1])

print(tuple1[2])

print(tuple1[0:3])

tuple1[0]='Hey'      #error : tuples are immutable
```

4. Strings

String is sequence of Unicode characters. We can use single quotes or double quotes to represent strings. Multi-line strings can be denoted using triple quotes, `'''` or `"""`.

5. Set

Set is an unordered collection of unique items. Set is defined by values separated by comma inside braces { }. Items in a set are not ordered.

```
set1={1,2,3,1,4,1,3,1,4,5}

print("set=",set1)      #yes, you can print entire set like this
```

6. Dictionary

Dictionary is an unordered collection of key-value pairs.

It is generally used when we have a huge amount of data. Dictionaries are optimized for retrieving data. We must know the key to retrieve the value.

In Python, dictionaries are defined within braces {} with each item being a pair in the form `key:value`. Key and value can be of any type.

```
dict={1:"first",2:"second",3:"third",4:"fourth"}      #comma separated key value pairs

print("dict=",dict)

print(dict[1])                  #element whose key value is 1

print(dict[2])
```

Lets nest some datatypes

```
#dictionary or list ? or list of dictionary or dictionary of list ?

data={"movies":['rock on','range de basanti','raazi'],"songs":['pefect','shape of you'],"books":['milk and honey','alchemist']}



print(data)

print(data["movies"])

print(data["movies"][0])

print(data["songs"])
```

Okay one more..

```
#so many {‘s

data={"movies":{'shilpi':'rock on','shaini':'range de basanti','meenakshi':'raazi'},"songs":{'shilpi':'pefect','shaini':'shape of you'},"books":{'shilpi':'milk and honey','meenakshi':'alchemist'}}

#can you find which book do i like?

#which movie does shaini watch ?

#which book meenakshi reads?

#all the movie preferences of three of us ?
```

Decision making !!

```
num=10              #similiar to C, you can nest if-else constructs as well
if num>0:           #notice the colons “ : ” at the end , this is important
    print("positive num")
elif num==0:
    print("zero")
else:
    print("negative number")
```

LOOP

```
arr=[0,1,2,3,4,5,6,7]
print("first type")
for i in arr:              #i takes value of each element in arr
    print(arr[i])
print("second type")
for i in range(0,8):       #range function in python, notice the 8
    print(arr[i])
print("third type")
for i in [0,1,2,3,4,5,6,7]:     #another way
    print(arr[i])
```

Functions (talk modularity)

```
def printname():      #notice the colon at the end

    print("hey I am shilpi !")

def calculateSum(a,b):      #parameters

    print("sum is :",a+b)

def calculateSumR(a,b):    #return type not needed

    return a+b     #function returning value

def calculateSumProduct(a,b):

    return a+b,a*b

#function returning multiple values, no need of pointers to
do this :p
```

```
#don’t forget to call functions

printname()

calculateSum(5,10)

x,y=calculateSumProduct(12,13)

print(x," ",y)
```

Importing modules !

```
import math

a=99

print(math.sqrt(a))          #sqrt function is defined in math module



import math as m             #or you could do, to prevent rewriting module name again

a=99

print(m.sqrt(a))
```

Importing from another file

1. Lets put out dictionary in “hello1.py”

```
data={"movies":{'shilpi':'rock on','shaini':'range de basanti','meenakshi':'raazi'},"songs":{'shilpi':'pefect','shaini':'shape of
you'},"books":{'shilpi':'milk and honey','meenakshi':'alchemist'}}
```

2. Create another file “hello.py” and add the following code :

```
 import hello1 as h

print(h.data)

print(h.data['movies'])
```

3. And, Voila you can use content of one file into another ; you can import selected functions too; try at home

Taking Input

```
name = input("What's your name? ")            #input function is used in python3 to get input
print("Nice to meet you " + name + "!")
age = input("Your age? ")
print("So, you are already " + age + " years old, " + name + "!")
```

Classes

Pheww! That’s been a lot

Try and google classes at home !!