Python Basics

Introduction

In Simple words Programming language is a way to communicate with computer and devices.

Why Python ?

- Python is easy to learn and understand because of easy syntaxs.
- Widely used and has community support across the internet (Stackoverflow, Discord Python server, Quora space for python enthusiast)
- Used widely mostly for data science , machine learning as well as general programming
- Lots of resources on internet.

Setting Up the environment

We will be using google collab for this project

Setting up the environment can be done in one of the following ways :-

1. Download python from microsoft windows store [recommended] or from python official web

   = = = = = O R = = = = = =

2. Go to https://replit.com/ click on start coding login or create an account. Click on the python logo below the create button name the file click create repl.

   = = = = = O R = = = = = =

3. Use google collab or any other online editor.

Code Reference [Syntax]

Try this code to get started .

```
# anything that starts with a hashtag is a comment
# It is written for progrmmer to describe code
# this output nothing special

# To display anything on the screen use print() function
# more on function later
print("Hello! World")

# similar to mathematics you can perform calculation
# numbers can be any real number
x = 7
y = 2
print( x + y ) #addition
print( x - y ) #subtraction
print( x * y ) #multiplication
print( x / y ) #divison
print ( x ** y ) # x to the power y
print( x // y ) #floor division ; it has only integer part
print( x % y ) # remainder

# Strings is the terminology used to refer text in programming
# it can be a word , set of word , sentences etc
a = "Hello "
b = "everyone "
print( a + b ) #add both of them as one
print( a * 2 ) #print this text multiple time
```

Operators

Operators are used in python to perform calculations and build expression.

Python operators are of three types

1. Arithematic Operators (+ , - , / , * , // , ** , %)
2. Relational Operators (> , < , == , >=, <= , !=)
3. Logical Operators (and , or , not , XOR)

```
# Relational operator Example
#Less than
print( 4 < 5 ) #True
print( 5 < 4 ) #False
```

```
print( 4 < 4 ) #False

#Less than or equal to
print( 4 <= 5 ) #True
print( 5 <= 4 ) #False
print( 4 <= 4 ) #True

# Greater than
print( 5 > 4 ) #True
print( 4 > 5 ) #False
print( 4 > 4 ) #False

# Greater than or equal to
print( 5 >= 4 ) #True
print( 4 >= 5 ) #False
print( 4 >= 4 ) #True

# Equal to
print( 4 == 4 ) #True
print( 4 == 5 ) #False

# Not equal to
print( 4 != 5 ) #True
print( 4 != 4 ) #False
```

```
# Logical Operators Example
# Logical AND
print( True and True ) #True
print( True and False ) #False
print( False and True ) #False
print( False and False ) #False

# Logical OR
print( True or True ) #True
print( True or False ) #True
print( False or True ) #True
print( False or False ) #False

# Logical NOT
print( not True ) #False
print( not False ) #True

# Logical XOR
print( True ^ True ) #False
print( True ^ False ) #True
print( False ^ True ) #True
print( False ^ False ) #False
```

Input

input() is used to take user input on python consoles . It is a mode of interaction between program and user on console based systems.

```
print("Enter your name to get started")
name = input()
print("Hello! " + name)
```

Input ( ) function takes a parameter that outputs on the console as instruction to user , similar to print.

```
name = input("Enter your name to get started")
print("Hello! " + name)
```

Data Types

Like numbers in mathematics, variables in programming can be of different type . These are some of the basic types used in programming.

1. Strings : Text , words , sentences , phrases , paragraphs or anything that is a combination of character is a string. A string is usually written with a pair of inverted commas (all called quotations)
2. Integer : These are similar to integers in mathematics.
3. floats or doubles : these are used for decimal number that is number with fractional part . E.g. 1.2

To change a data type to other , functions are used in python

To change a string to integer use int( ) function.

```
a = "15"
b = "16"
print( a + b ) #will print 1516 as output

c = int( a )
d = int ( b )
print( c + d ) #will print 31 as output
```

String

```
a = "I am String1" #declare variable with name a and type string
b = "I am String2" #declare variable with name b and type string
print(a + " " + b) #this is concatenation or addition of string
print(a * 3) #prints a for three times [string replication]
```

```
print(a[0]) #1st letter of string a
print(a[1]) #2nd letter of string a
print(a[-1]) #first character from last

# string slicing
print(a[1:5]) #print 2nd to 5th characters in string (index 5 excluded)
print(a[:5]) #print first 5 characters of string a
print(a[5:]) #print all characters after 5th character\

#strings can also be compared
print("abc" > "abd") #false. It works like index of english dictionary
```

> Note : Index in all modern programming language starts with 0 and not 1. so zero is the first element always.

Import

using some built infunction (or say predefined program) requires the use of library , library is like a collection of code snippet.

import keyword followed by name of library is used to import a library.

```
# here the name of library is random
# random is a python library, used for generating random numbers
import random
```

Practise program 1

make a program to generate random integer number when someone inputs a number between some range a and some range b for any a and b greater than 0.

Hint : [random.random() generate a random number between o and 1]

```
import random
a = int(input("Enter numbers lower limit"))
b = int(input("Enter numbers higher limit"))
# this lines uses function random from random library
# to generate a random number
c = random.random()
d = b - a  # here d is the range of number(i.e upper - lower limit )
e = int( a + d * c)
print(e)
```