Python

Lesson-1 Modules, Comments & PIP

What is Programing ?

Just like we use Hindi or English to communicate with which (each) other, we use a Programming language like Python to communicate with the computer.

Programming is only way to instruct the computer to perform various tasks.

What is Python ?

Python is a simple and easy understandable language which feels like reading simple english. This Pseudo code nature of python makes it easy to learn and understandable by beginners.

Features of Python

- Easy to understand = Less development time
- Free & open source
- High level language
- Portable
- → Fun to work with

* Writing First Python Program

print("My name is Pranav")

→ Print is a function.

output: My name is Pranav.

[Python .\filename]

Modules

A module is a file containing code written by somebody else (usually) which can be imported and used in our programs

Pip

Pip is the package manager for python you can use pip to install a module on your system:

↳ pip install flask installs flask module

Types of Modules

There are two types of modules

1) Built in modules → Pre installed in Python
2) External modules → Need to install using pip.

Eg: of built in modules are OS, abc, etc

Eg of external modules are tensorflow, flask etc

* Using Python as a Calculator

We can use python as a calculator by typing "python" + ↵ on the terminal

↳ This opens REPL
(Read Evaluate Print Loop)

* Comments

Comments are used to write something which the programmer doesn't want to execute.

↳ Can be used to mark author name, date etc.

* Types of Comments

1.) Single line comments → written using '#'

2.) Multi line comments → written using
''' Comment '''

Q Install an external module & use it to perform an operation of your interest.

Sol. Google → Playsound module,
open terminal → type: "pip install Playsound"

copy path from module and
paste & provide the path using
'//' not single '/' because it is
an escape function. (Paste full path of
mp3.)

Q Write a python program to print the contents of a directory using os module. Search online for the function which does that.

Sol: Google → os module print content of a directory.

" (os.listdir()) " → stackoverflow

code

```
import os
Print(os.listdir())
```

Lesson-2

Variables and Datatype

A variable is the name given to a memory location in a program.

Eg:

a = 30 → Variables = Container to store a value

b = "Pranav"

c = 71.22 → Keywords = Reserved words in Python

→ Identifiers = Cls/function/ variable name.

* Data Types

Primarily there are following data types in Python.

1.) Integers
2.) Floating point numbers
3.) Strings
4.) Booleans
5.) None

→ under "()" string is given by default

→ Python is a language that automatically identifies the type of data for us.

a = 71 → identifies a as class <int>

b = 88.44 → identifies b as class <float>

name = "Hardy" → identifies name as class <Str>

Printing variables

Print(type(a)) → <int>

Print(type(b)) → <class 'float'>

Print(type(name)) → <class 'Str'>

* Rules for defining a variable name

→ A variable name can contain alphabets, digits and underscores

→ A variable name can only start with an alphabet and underscores

→ A variable name can't start with a digit

→ No white space is allowed to be used inside a variable name.

→ Variables are case sensetive
'A', 'a' are different variable names.

* Examples of a few variable names are :-
  Pranav, one8, seven_, _seven etc.

* Operators in Python

1.) Arithmetic operators => +, -, *, /
2.) Assignment operators => =, +=, -=
3.) Comparision operators => ==, >, >=, <, !=
4.) Logical operators => and, or, not.

1.) a = 3
    b = 4
    Print("The value of 3+4 is", 3+4)

    output: '7'

2.) a = 34
    a += 2
    Print(a)

    output: '36'

3.) b = (4 > 7)
    Print(b)

    output: 'False' (returns Boolean)

4.) bool1 = True
    bool2 = False
    Print("The value of bool1 & bool2 is", (bool1 & bool2))
    Print(" " " ", (bool1 or bool2))
    Print(" " " ", (not bool2))

output: False
True
True

not is ~~only~~ used only for one variable)

working: 'and' → returns 'true' only if both cases are 'true' else returns 'false'

'or' → returns 'true' if any one is 'true'

'not' → reverse the condition.

* type() function and typecasting

type function is used to find the data type of a given variable in python.

- a = 31
  type(a) ⇒ class <int>
- b = "31"
  type(b) ⇒ class<str>

eg: a = "3534"
a = int(a) → # trying to change it's class from str to int

Print(a+5)

A number can be converted into a string and vice versa (if possible)

There are many functions to convert one data type into another.

Str(31) ⇒ "31" → Integer to String Conversion

int("32") ⇒ 32 → String to Integer Conversion

float(32) ⇒ 32.0 → Integer to float Conversion.

... and so on.

Here "31" is a string literal and 31 a numeric literal.

* input() function

This function allows the user to take input from the keyboard as a string

a = input("Enter name") → If a is "Pranav", the user entered Pranav.

It is important to note that the output of input is always a string (even if the number is entered) ⇒ If a is "34", user entered 34

```
a = input("Enter name")
Print(a)
```

output: 'Name you entered'

Q1 Write a Python Program to add two nos.

Sol:
```
a = 30
b = 5
Print("The Sum of a & b is", a+b)
```

Q2 Write a Program to find remainder when a no. is divided by 2.

Sol:
```
a = 458
b = 15
Print("The remainder when a is divided b
      is", a%b)
```

Q3 Check the type of the variable assigned using input() function.

Sol:
```
a = Pranav
Print(type(a))
```

Q4 Use comparision operators to find out whether a given variable 'a' is greater than 'b' or not

a = 2 , b = 4

Sol:
```
a = 2 , b = 4
Print(a > b)
```

Q5 Write a python program to find avg. of two numbers entered by the user.

Sol.

```
a = input("Enter first no.")
b = input("Enter second no.")
a = int(a)
b = int(b)          # because value input is str, so change it to int
avg = (a+b)/2
Print("The average of a & b is", avg)
```

Q6 Write a python program to find average of two numbers entered by the user.

Sol:

```
a = input("Enter a number:")
# a = int(a)
Square = a*a
Print("The square of a is", square)
```