

Python:

Python is a high-level, interpreted, Object-oriented programming, indented, versatile, dynamically typed programming language.

Features of python:

- Easy to learn and use
- Simple syntax
- Free and open source
- Platform independent
- Extensive standard library
- Large community support
- Suitable for diverse applications
- Database connectivity

Python Applications in real-world:

- Web development
- Data science/AI/ML
- Game development
- Embedded Systems and IoT
- Devops and Cloud Computing
- Cyber Security

Comments:

They are used to explain the code for the user, they make the code more readable, we can say it as they are part of the code but they won't be considered in execution time.

Types of comments in python:

- Single line comment: Indicated by “#”
- Multi line comment: Indicated by “” “” or “""" """

Keywords:

They are the reserved words in python which cannot be used as identifiers like variables, function names etc.

Eg: true, false, for, if, else, while, break, try, except, finally etc

Variables in python:

They are containers to store the value of data.

Rules of variables:

- Valid variable declaration:
 - *They are case sensitive means a=5, A=5 both are different
 - *They can be started with a letter or an underscore character (Eg: num=4, _num=4)
 - *They can be alpha-numeric characters and underscores (A-z, 0-9, _)
- Invalid variable declaration:
 - *They can not start with a number (eg: 1num=4)
 - *They can not use any special characters except “_”
 - *They can not use any reserved keywords

Declaring multiple variables in single line:

Eg: a=1 ; b=2 ; c=3 or a,b,c=1,2,3 or a=b=c =4

Datatypes:

They are built-in by default, they specify the data category.

Types of datatype:

- Numeric datatype: int, float, complex(5j)
- Text type: str
- Boolean type: bool (true or false)
- Special datatypes [Data structures]: list, tuple, set, dict

Input and Output Function:

*Anything we display to user its output function. [We use “ print() ”]

*Anything we take from user its input function, [We use “ input() ”]

Eg: O/P function: print(“hi”) [o/p: hi]

I/P function: a= input()

*To check for type of data use “type ()”

*To check for particular datatype we use :

Eg: a = int(input())

5

type (a)=>int

*When we don't mention datatype it does string concatenation

Eg: a=input()

5

b= input()

5

a + b => "55"

=>**Write a program to read 2 float no.s from user:**

x = float(input())

y = float(input())

print (x +y)

or print(x,"+",y,"=",x+y)

or print ("sum of:", x, " +", y, "=", x+y)

O/P:6

4

10.0

6.0+4.0=10.0

Sum of:6.0+4.0=10.0

=>**Since we can't combine strings and no.s python 3 introduced f-string or format()**

Eg: age = 36

txt = f " Hi, I'm {age} "

print(txt) =>o/p: Hi, I'm 36

* .format(): eg: print ("sum of {} + {} = {}".format(x, y, x+y))

*f-string method: eg: print (f"sum of {x} + {y} = {x+y} ")

Operators:

They are used to perform operations on variables and values.

=>Types:

- Arithmetic operator:

+ , - , * , / , % , // , **

Eg: write a program to calculate area of triangle and circle with given input.

```
b = float (input())
```

```
h = float (input())
```

```
r = float (input())
```

```
AOT = 0.5 * b *h
```

```
AOC = 3.142 * r*r
```

```
print (f"area of a triangle: {AOT} ")
```

```
print (f"area of a circle: {AOC} ")
```

- Relational or comparison operator

<, >, >=, <=, ==, != [checks the condition and returns Boolean value]

- Logical operator

AND, OR, NOT

- Assignment operator

= , += , -= , *= , /= , %= , //= , **=

Syntax: var=var +value

