

Date
22/11/21

Chapter → 3

PYTHON FUNDAMENTALS

Introduction →

Computer language user तथा computer के बीच communication का माध्यम होता है।

कम्प्यूटर को निर्देश देने के लिए किसी न किसी language का प्रयोग किया जाता है।

Python HLL है जो Object Oriented programming को support करती है।

Python एक ऐसी HLL है जिसे आसानी से सीखा जा सकता है और प्रयोग किया जा सकता है।

Python 1991 में Guido Van Rossum के द्वारा विकसित की गई थी।

In Python	In C++
$x, y = 5, 7$	$\text{int } x = 5, y = 7$
$x, y = y, x$	$t = x;$ $x = y$ $y = t$

Advantage of Python :-

- पायथन इंटरप्रेटर, utility or library फ्री में उपलब्ध है।
- यह Portable है। (जैसे- Window, Linux, Android, Mac आदि प्लेटफॉर्म पर इसका प्रयोग किया जा सकता है।)
- यह interpreter पर आधारित है जो debugging को easy बनाता है।

पायथन के File handling System के द्वारा डेटा को easily स्टोर तथा मैनेज किया जाता है।

पायथन को सीखना तथा प्रयोग करना आसान होता है।

पायथन में बनाया गया प्रोग्राम अन्य language की उपेक्षा होता है।

Working with Python :-

www.python.org के द्वारा Python Software install किया जा सकता है। करने के बाद यह दो अलग-2 मोड में Python install की जा सकती है।
open

1. Interactive mode
2. Program mode

1. Interactive mode — इस मोड में कमांड्स पायथन प्रॉम्ट (">>>") पर दी जाती हैं।
Interactive mode को भी कह जाता है Command line mode

इसमें एक समय में केवल एक कमांड दी जाती है और पहले कमांड का रिजल्ट प्राप्त होने के बाद दूसरी कमांड दी जाती है।

NOTE: Interactive mode चलाने के लिए Python folder से विकल्प चुनना होगा।
Python 3.7 (32 bit)

2. Program mode - इसे स्क्रिप्ट मोड कहा जाता है। इसमें सभी कमांड्स एक साथ एक प्रोग्राम के रूप में दी जाती हैं तथा इस प्रोग्राम फाइल को (.py) extension के साथ सेव किया जाता है।

IDLE Python 3.7 Option को select करके program mode open किया जा सकता है।

Creating a Module / Program File —

- Step 1 → पायथन में कोरिप्ट मोड या प्रोग्राम मोड में ऑपन करे।
- Step 2 → File menu से कमांड के द्वारा नई फाइल Open में प्रोग्राम कोड टाइप करे।
- Step 3 → File से के द्वारा फाइल को सेव करे।
- Step 4 → File से Save option menu

सेव करते समय फाइल को कोई भी नाम दिया जा सकता है।

- Step 5 → प्रोग्राम फाइल हमेशा "py" extension के साथ सेव होती है।

Python Tokens —

पायथन में विभिन्न कोड्स ग्रुप को मिलाकर एक पायथन प्रोग्राम बनता है। ये कोड्स token कहलाते हैं।

या

पायथन प्रोग्राम की सबसे छोटी इकाई token कहलाती है।
Token कई प्रकार के होते हैं -

keywords, Identifiers, Literals, Operators, Punctuation

- (i) keywords :- Python में keyword pre-define होते हैं।

ये किसी विशेष उद्देश्य के लिए रिजर्व होते हैं।

Ex:- For while If True False
 none break global raise try
 catch def range is in
 and or Finally class return

(ii) Identifier :- Identifier developer के द्वारा define किये जाते हैं।

किसी भी Value को नाम देने के लिए Identifier का प्रयोग होता है। Identifier बनाने के लिए कुछ नियम follow किये जाते हैं।

- यह एक letter or underscore (-) से शुरू होना चाहिए।
- इसमें केवल letter या numbers हो सकते हैं।
- Underscore को छोड़कर अन्य किसी भी special character का प्रयोग इसमें नहीं किया जाता।
- keyword को Identifier के रूप में प्रयोग नहीं किया जा सकता।
- यह case sensitive है। इसलिए lower case और upper case अलग-अलग होते हैं।

Valid ex. → jno, jno, m5, y2k, FILE_NAME

Invalid ex. → 32 kb, def, nm\$, Math, marks

blank space not allowed

(iii) Literal :- Literal को डेटा के रूप में जाना जाता है। इसका निश्चित मान होता है।

Ex: String, Numeric, Boolean, etc. None

(A) String Literal :- Single या double कोर्स के अंदर लिखा गया डेटा String Literal होता है

Ex:- "Delhi" 'Rohan' "Rs. 25"

दो प्रकार से लिखे जा सकते हैं -
String Literal

1. Single Line String

2. Multi Line String

1. Single Line String - एक ही लाइन में होने वाली complete String Single Line String होती है।

Ex:- "Welcome to Programming"

2. Multi Line String - एक से अधिक लाइन में लिखी जाने वाली String Multi Line String होती है।
दो प्रकार से लिखी जाती है -
Multi Line String

(i) By adding backslash (\)

(ii) By using triple quotation

(i) By adding backslash :- Backslash का प्रयोग करके multi line string लिखी जाती है। यह प्रत्येक लाइन के अंत में लगाया जाता है।

Ex:-
"Hello world\
Welcome to Programming\
in Python"

(ii) By using tripple quotation :-

Triple quotation का प्रयोग करके लिखी जाती है। multi line string

Ex:
"Hello world
Welcome to Programming
in Python"

(B) Numerical Literal :-

यह positive (+), Negative (-) या दशमलव (.) वाले अंक होते हैं।

पायथन चार प्रकार के Numerical Literal को Support करता है।

int long float complex

(i) int (integer) - यह Positive या Negative पूर्ण संख्या होती है।
(+) (-)

Ex:
[123, +453, -123] → Decimal
[-45, +014, 235] → Octal
[0x142, 0xA2C, 0x9AB] → Hexadecimal

(ii) float - फ्लोट के साथ point होते हैं। real no. floating point

4.5, 4.567, 1.89

(iii) long - यह एक integer literal है जो integer से बड़ी संख्या को store करता है।

Ex:
33 → int
33L → Long

(iv) Complex :-

Complex no. "A+Bi" के रूप में लिखे जाते हैं।
जहाँ पर A real no. (वास्तविक)
B काल्पनिक कम्पोनेंट है।
i एक काल्पनिक संख्या है जो (-1) के वर्गमूल के बराबर होता है।

5+7i , 1.3+9j , 5+1.4j

Ex:

a = 1.3+9j
Print (a, real) -> 1.3
Print (a, imag) -> 9

(v) Boolean Literal :-

इसका मान True तथा False दो प्रकार का होता है।

Ex:

a = 5
b = 10
if (True)
Print (" b is greater")

Ex: a = True

(vi) None Literal :-

इसका अर्थ absent of Value है जिस Variable में कुछ भी मान न दिया जा रहा हो।
वहाँ पर none का प्रयोग किया जाता है।

Ex:

k = none
print k -> none

Operators :-

यह विशेष symbol होते हैं जो डेटा पर ऑपरेशन करते हैं। Symbol जिस डेटा पर ऑपरेशन किया जाता है, वह Operand कहा जाता है।

$$c = \overbrace{a + b}^{\text{Operand}}$$

↓
Operator

Operator कई प्रकार के होते हैं -

Arithmetic Operator -

इनका प्रयोग नम्बर पर किया जाता है। arithmetic Calculation के

Symbol	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Remainder
**	Exponent (raised to power)
//	Floor division

Relational Operator - Comparison (तुलना) करने के लिए इन Operator का प्रयोग किया जाता है।

Symbol	Description
>	Greater than
<	less than
=	Equals to
>=	Greater than equals to
<=	less " " "
!=	Not equal

Logical Operator :-

एक से अधिक condition के आधार पर कार्य करने के लिए logical operator का प्रयोग किया जाता है तथा यह boolean Value return करते हैं।

Symbol	Description
and	Logical AND
or	" OR
not	" NOT

Assignment Operator :-

इसका प्रयोग किसी वैरिअबल में Value स्टोर करने के लिए किया जाता है।

$$a = 5$$

Identity Operator :-

दो Objects की तुलना करने के लिए इसका प्रयोग होता है। यह boolean Value return करता है।

Symbol	Description
is	are both value same
is not	are both value not same

Membership Operator :-

यह Sequence में Object की उपस्थिति को check करता है।

Symbol	Description
in	is Object present in the Sequence
Not in	is " not " " " "

a = [5, 7, 9, 11, 13]

b = 13

b in a

Punctuation :-

ये वे Symbol हैं जिनका प्रयोग पाठ्यक्रम में प्रोग्राम लिखने के लिए किया जाता है।
 व्यवस्थित करने, फार्मेट करने के लिए ' किया जाता है।

() { } [] ' " # , @

Print Statement -

Information प्रिंट करने के लिए इसका प्रयोग किया जाता है। स्क्रीन पर information प्रिंट करने के बाद यह new line character automatic अंत में जोड़ देता है।

Syntax-1

Print (< information >)

Example

Print (" Welcome to Programming ")

Syntax-2

Print (< info 1 >, < info 2 >,)

Example

Print (" Area", 01)

Print ('A', 1, 2, 5, 6, " Delhi")

Program - सर्कल का एरिया Calculate करने के लिए प्रोग्राम लिखें। जिसकी त्रिज्या 7.9 है।

$$r = 7.9$$

$$a = 3.14 * r * r$$

Print ("Area of Circle = ", a)

Input Statement :-

User से डेटा प्राप्त करने के लिए इस Statement का प्रयोग किया जाता है।

Syntax-1 <Variable> = input ("Message")

Example nm = input ("Enter your name")

Input Statement के द्वारा डेटा प्राप्त होता है। String के रूप में

जिसके द्वारा Calculation नहीं की जा सकती।

Calculation के लिए इसे नम्बर फॉर्मेट में बदलना आवश्यक होता है।

int या Float Function का प्रयोग करके इस डेटा को input किसे गये no. format में बदला जा सकता है।

Syntax-2 <Variable> = int (input ("< Message >"))

Example nno = int (input ("Enter your roll no."))

1. Program: Write a Program to Calculate total, average and percentage of three Subject Marks. Marks are 45, 48 and 41 each out of 50.

```

h = int(input("Enter hindi "))
e = int(input("Enter english "))
m = int(input("Enter maths "))
add = (h + e + m)
aver. = (add) / 3
per. = (add * 100) / 150
Print (add, aver., per.)

```

$$h = 45$$

$$e = 48$$

$$m = 41$$

2. Program: Write a program to input base & height of a triangle. Calculate and print its area

```

h = int(input("Enter height of triangle "))
b = int(input("Enter base of triangle "))

```

$$\text{area} = (h * b) / 2$$

```

Print ("Area of triangle is", area)

```