Date : / /
Page No.

7/8/23

Python

By Guido Van Rossum in 1991

- It is a case sensitive.

* Uses :-
→ AI [Artificial Intelligence], ML [Machine Language]
→ Data Analytics.
→ Web Development, Django.
→ Search Engine Optimization [Abstracting large amount of data].
→ Block Chain.
→ Game Development.
→ Automation.
→ Visualization.
→ Linear Regretion [one variable depend on another variable].

* Features of Python :-

→ Easy to learn.
→ Simple language.
→ High level language [User Friendly / Human Readable]
→ Dynamically Type [Interpreter assign the variable during run time].
→ Portable
→ Interpreted [line by line check the Program].
→ Translation happen at the same time as program is being executed.
→ Extensible.
→ Database connectivity available.

Date : / /
Page No.

→ Procedural and Object Oriented.
[Design medium size to larger size webpage].
→ Platform Independent.
→ Open Source [Publically accessible]
→ Scalable
→ Scripting language.
[Programming language that employ high level construct to interpret and execute one command at a time].
→ Huge Library.

9/8/23

1.) First Python Program
→ Variables
→ Tokens
→ Data Types

INDENTATION [Space]
i.e, if ()
This space ←→ is Indentation

`i=10` → It is a cell where Program write.

`print("____")` → Input.

```
a = int(input("Enter first no."))
```

Date : / /
Page No.

* **Variables** :- Data and values can be stored in temporary storage spaces called variables.

* **Data Types in Python** :-

- **Numeric**
  - → Integer
  - → Float
  - → Complex Type
- **Dictionary** — It is a data structure used for mapping (Relationship) the key-value pairs.
- **Boolean** (True / False)
- **Set**
- **Sequence Types**
  - → List
  - → Tuple
  - → String

Sequence of character which can be written in single quote or double quotes

⇒ To find the data type of any variable it can be written as type (string) :- O/P = Str
a = 10, type (a) :- O/P = Int.

* **Token** :- It is the smallest meaningful component of a program is called token.

- Keywords
- Identifiers
- literals
- Operators

Date : / /
Page No.

Keywords

- And
- As (To create alias)
- Assert (used to Debugging to find error)
- break
- Class
- Continue
- def (To define function)
- del (deleting an object)
- elif [only used in Python] (used for Conditional statement)
- except (used for exception and only used in python).
- False [F]
- True [T]
- Finally
- global
- Import
- In (used in string)
- Lambda
- None (representing NULL value)
- Nonlocal
- Not
- Or
- Pass [When we don't want to execute then pass the value]
- raise (Exception)
- return
- try
- While
- With (for exception handling)
- Yield (generate return values).

10/8/23

Date : / /
Page No.

* Extracting Individual Elements :-

str = "Hello"
-5 -4 -3 -2 -1
0 1 2 3 4

str [2] = l
or str [-3] = l

"This is BCIIT"
-9
-8 -7 -6 -5 -4 -3 -2 -1

len(str) = leng(I)
[-8] = I

str.lower( ) :- ~~Convert~~ Find the string into lower.
str.upper( ) :- ~~Convert~~ Find the string into upper.

⇒ Str.lower( ) :- Convert string into lower.
⇒ Str.upper( ) :- Convert string into upper.

⇒ Str.is lower( ) :- To check all the strings should be lower and give true or false.
⇒ str.is upper( ) :- To check all the strings should be upper.

String [1:3] H E L L O = O/P = e l
0 1 2 3 4 1 2
Include Exclude

Date : / /
Page No.

⇒ Replacing a substring :-

str.replace ('i', 'h') =o/p=

⇒ Number of occurence

str.count ("Smriti")

⇒ First occurrence of that word

str.find ("Smriti")

∴ [To find the first word which is occurs on that paragraph]

* Identifiers :- They are names used for variables, functions or objects.

⇒ Rules :-

1) No special character other than "underscore" is allowed.
2) Identifiers are case sensitive.
3) First letter cannot be a digit.
4) Keyword cannot be an identifiers.

* Literals :- They are constants in Python.

[Assigned value cannot be changed throughout the program.]

⇒ str.split ('h').

Date : / /
Page No.

21/8/23

Escape Sequence

a = "this will Insert one \\ Backslash".

\' Single Quote
\\ Backslash
\n New line
\r Carriage Return (It will print after \r statement first. "Hello \r world".)
First Print World

\t tab
\ooo octal \110 \145
\xhh Hex Value
\b * 48 * 68 [For Backspace]

23/8/23

Operators

1) Arithmetic Operators :- +, -, *, /, %, //
2) Logical Operators :- and, or, not
3) Assignment Operator :- =, +=, -=, *=, /=
4) Comparison Operator :- ==, !=, >, <, >=, <=.
5) Identity Operator :- is, is not (~)
6) Membership Operator :- not in, in
7) Bitwise Operator :-

Date : / /
Page No.

Python Identity Operator :-

It is used to compare the objects, not if they are equal but if they are actually the same object with the same memory location.

Operators :- is , is not (~)

Syntax :- (x is y) (x is not y)

Membership Operator -

in :- It is used to test if sequence is present in object. (x in y).

not in :- If sequence with the specified value is not present in object.
(x not in y).

Python Bitwise Operator :-

⇒ Used to compare binary numbers.

→ AND - &
→ OR - |
→ XOR - ^ [When only two of the bits is one.]
→ Shift right - >>
→ Shift left - <<
→ NOT - ~

Right shift '>>' :- Shift right by pushing copies of left most bit in from the left and let the rightmost fall off.

Right fmost bit :- Value half ho jati hai right mai.

Date : / /
Page No.

Left Shift '<<' :-
Shift left by ~~putti~~ pushing zeros in from the right and let left most bit ~~follow~~ off.

left most bit :- Value half ho jati hai left mai.

25/8/23

5: → Fifth ko chodkar uske aage ka
:5 → Fifth se phela wala.

Lab Work

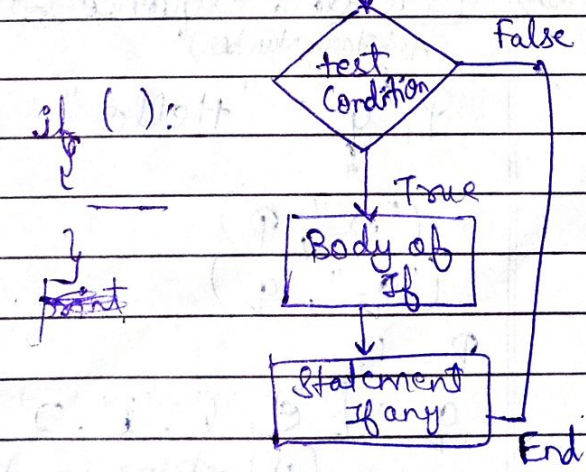
''' ---
--- ''' } Multiline Comment

Q1. Swaping two numbers.
⇒ ~~a = int(i~~

Q2. Swaping two string.
Q3. Slicing in a string.

⇒ String = "Hello World"
new_string = String[5:] + String[:5] + String[3] + String[2]
print(new_string).

If Flow chart



if ( ):
{
}
~~print~~

Date : / /
Page No.

22/8/23

Assignment Statement form

1) Basic form :-

a = 10

⇒ Student = "Sam"

2) Tuple Assignment :-

⇒ tup1 = (1, 2, "Ram")

tup1 → Enter O/P = (1, 2, "Ram")

3) List Assignment :-

L1 = [1, 2, 3, 4]

4) Sequence Assignment :-

* a, b, c = "Hey"

Enter - a = H
b = E
c = Y

print("a =", a)
print("b =", b)
print("c =", c)

O/P :- a = H
b = E
c = Y

5) Extended Sequence unpacking :-
(Arbitrary Number)

P, *q = "Hello"

("P=", P)
("q=", q)

P = H
q = ['e', 'l', 'l', 'o']

(Unpacking :- All the letters of string will be separate).

Date : / /
Page No.

6) Multitarget Assignment :-

x = y = 75 [Jab ek se zyda variable ko same value deni hai]

7) Augumented Assignment :- / [Short Hand Assignment / Compound Assignment]

x += 2 [a += b ⇒ a = a + b]

a &= b

a = a & b

Conditional Statements [Decision-making statement]

→ if
→ if-else
→ if elif else
→ Nested if

→ if Syntax :- `if ( ):`

Eg :-
```
if (x>0):
    print("Number is Positive")
```
Indentation

→ if else

Syntax :-
```
if ( ):
    print("____")
else:
    print("____")
```

Date : / /
Page No.

→ if elif else

Syntax :-

```
if (x = 0):
    print("____")
elif (x > 0):
    print("____")
else:
    print("____")
```

→ Nested if

Syntax :-

```
if ( )
```