MSC Classes | Techjaguar Academy Rewa
Savvy Mother School Azad Nagar Rewa M.P.
Admin@techjaguar.in | www.msc.techjaguar.in

PYTHON NOTES

TOPICS COVERED :-

- Introduction To Python
- Print() and Input()
- Statement and Comments
- Keywords and Identifiers
- Variable & DataTypes
- TypeCasting
- Operators
- Control Statements
- List

Introduction To Python :-

What Is Python ?

- Python is a high-level, general-purpose programming language.
- It Was Created By Guido Van Rossum And Intially Released In 1991.

It is used for :

- Web development (server side)
- Software development
- Mathematics
- System scripting

What can python do ?

- Python can used in server to create web application
- Python can connect to database system
- It can also read and modify file
- Pyhon can be used to handle big data and perform mathematics

Why python ?

- Python work on different platform like window mac linux raspberry pi etc.
- Python has a simple syntax similar to the english language
- Python has a syntax that allows developers to write program with fewer lines than some other programming language
- Python runs on a interpreter system meaning that code can be executed as it is written.
- This mean that the prototyping can be very quick.
- Python can be created in a processional way an object oriented way or a functional way.
- The most recent major version is python 3.
- Python will be written in a text editor.
- It is possible to written on pycharm, idle, netbean, vs code, jupyter, etc.
- Extension of python is .py

Print()And Input():-

print() :-

- print() Is Used To Print Anything.
- SYNTAX : print(_Data_) ➔ For Ex: print("Hello World")

NOTE :- PYTHON IS CASE SENSITIVE PROGRAMMING LANGUAGE

Input() :-

- input() Is an Inbuilt Function Which Is Used For Getting Input From User.
- SYNTAX : input(_Message_) ➔ For Ex: input("Your Input: ")

NOTE :- BY DEFAUL INPUT TAKEN BY INPUT() IS STRING.

Statements and Comments :-

Statements :-

- Simple Statements ➔ Any Line Of Code In Program [Single Line Of Code In Program]
- MultiLine Statements ➔ Statement In Python Which Can Be Extended To 1 or More Line Using Parantheses(), Brashes{}, Sq. Bracket [],"",;,\.

→ When We Need To Write Long Statement and Can't Fit Into 1 Line Then We Can Use MultiLine Statement's Character.

Example :

```
1. MyList = [ "Hello" , "A", "B" , "C" ,3 , 4 , 5 ]
```

FOR PRACTICE CLICK HERE

Comments :-

- Comments are used in program to explain code.
- Comments are ignored by python Interpreter.
- Comments can be used to make the code more readable.

1) Single Line Comment:-

- Comments starts with a #, and Python will ignore them
- It Can Only Extended upto Only Single Line
- Example:-

```
1. # This Is Single Line Comment
```

2) Multi-Line Comment:-

- Comments Written Between """ And """
- It Can Be Extended To More Than 1 Line
- Example:-

```
1. """
2. This
3. Is
4. Multi-line
5. Comment
6. """
```

FOR PRACTICE CLICK HERE

Keywords and Identifiers :-

Keywords →

- A python keyword is reserved word that can't be used to name variable, class, Function etc.
- Example:- if, elif, else, or, not etc.

Identifiers →

- Identifiers in python are word that we used to Specify and declare variable.

- Identifiers are Case Sensitive
- Identifiers Names start With A-Z, a-z, _ .
- Identifiers Names can't start With digits .
- Python Keyword cannot use as Identifiers. [Ex.: else=5]
- Special Symbol Like **@ # %** Cannot use in Idenfiers

Variable and Datatypes:-

Variable ➔

- Variable are Container use for store Data Value
- Example :- a=10 [Here a is Variable]

Assigning Value To Multiple Variable :

```
>> 1. x , y, z = "Python" , "HTML" , "CSS"
```

FOR PRACTICE CLICK HERE

Rules For Naming VariablesIn Python :

- The Name of variable must start with letter and _ .
- The variables names Can only Contain Alphanumerical characters and _
- Acceptable : Age, _age, B1, a_1 etc.
- Unacceptable. - 1age, 1_age, 11=2 etc.
- It's Name are Case Senstive.

Datatypes ➔

Data stored in the variable has datatype associated with it.

There are Three basic DataTypes :

- Integer [int] : Whole Number Without Decimals ➔ 23, 98, 378 etc.
- Float : Numbers With Decimal ➔ 2.4, 5.6, 3.0, 54.9 etc.
- String [str] : Group of Characters written inside double quotes ➔ "Python", "76", "Hello" etc.

Advance Data Types In Python :-

- Null/None
- Numerics → [Int, Float, Complex]
- List
- Set
- Tuples
- Range
- String

- Dictionary

NOTE : WE LEARN ALL THESE DATA TYPES LATER

Typecasting :-

Typecasting in Python is the process of converting one data type to another data type.

Methods To Change/Typecast Datatypes :-

- str() → To Change Data Into String Data Type
- int() → To Change Data Into Integers Data Type
- float() → To Change Data Into Float Data Type

Operators In Python:-

→ Operator are Special Characters they are used on operands which can be either Value or Variables.

Types of Operator In Python :-

Relational Operator :

→ It Is Comparing Operators.

- >
- <
- ==
- !=
- >=
- <=

Assignment Operator :

→ It Is Comparing Operators.

- = : Ex. x=2
- += : Ex. x+=2 [x=x+2]
- -= : Ex. x-=2 [x=x-2]
- *= : Ex. x*=2 [x=x*2]
- /= : Ex. x/=2 [x=x/2]
- %= : Ex. x%=2 [x=x%2]

Logical Operator :

- and : If All Operands Are True Then Result Is True Else Result Is False
- or : If All Operands Are False Then Result Is False Else Result Is True
- not : If Operand Are True Then Result Is False and Vice-Versa.

Control Statement:-

→ Control Statement are used to control The Flow Of Execution of Instruction of Program.

Conditional Statement :

→ Conditional Statement are design to add condition based Instruction To Program

→ They Can Be Implemented Using :

- **if** : It Is Used To Give Single Condition

```
SYNTAX :- if _condition_ :
                        Code
```

Example :

```
a = 10
b = 10
if a==b:
     print("It Is Equal")
```

FOR PRACTICE CLICK HERE

- **if…else** : It Is Used To Give Double Condition

```
SYNTAX :- if _condition_ :
                        Code
          Else:
              Code
```


Example :

```
a = 10
b = 10
if a==b:
     print("It Is Equal")
else:
     print("It Is Not Equal")
```

FOR PRACTICE CLICK HERE

- **if…elif…else** : It Is Used To Give Double Condition

```
SYNTAX :- if _condition_ :
                        Code
          Elif _condition_ :
                        Code
          Else:
              Code
```

Example :

```
a = 10
b = 10
if a==b:
     print("It Is Equal")
elif a==0:
      print("It Is Zero")
else:
     print("It Is Not Equal")
```

FOR PRACTICE CLICK HERE

- **Nested if** : It Is Used To Give Condition Under Existing Condition

SYNTAX :- if _condition_ :
If _condition_ :
Code

Example :

```
a = 10
b = 10
if a==b:
    if a==10:
        print("It Is Equal and 10 Both")
```

FOR PRACTICE CLICK HERE

Looping or Itterative Statement :

→ They are Use To repeat a block of statement or set of instruction to a specific number of time until condition is met.

→ They are Implemented Using For And While Loop.

- **For Loop** : To Itterate across the series, Such as list, dictionaries, sets, range or even string, a for loop is employed.

Example :

```
List1 =[ 1 , 2 , 3 , 4 , 5 , 6 ]
for item in List1:
        print("item")
''''
output  :  1
           2
           3
           4
           5
           6
'''
```

FOR PRACTICE CLICK HERE

- **While Loop** : It Checks the condition first. Run the conditional code If it is true.

SYNTAX : while(condition):
Code

FOR PRACTICE
CLICK HERE

Example :

```
a=1
while(a==1):
      print("It Is True")

# output : It Is True (“It will Print Again and Again Until Loop Get Break”)
```

List :-

→ List Are Used To Store Multiple Types Of Data In Single Variable.

How To Create List?

→ The List In Python are Created Using Square Bracket [].
→ Example : list1 = [1,”Hello”,9.5]

How To Access Element Of List ?

→ **Using List Index** : Each Item Of List has an Index Number and Python Allows Positive and Negative Indexing For Accessing Individual.

For Example : list[1] etc.

→ We Can Also Access Element Within Range By Giving Range

For Example : list[1:5]

```
List1 = ["a" , 2 , 3 , 5.7 ]
print(List1[2])
print(List1[2:4])
```

FOR PRACTICE
CLICK HERE

NOTE : IN PYTHON INDEXING START FROM 0.
FOR EXAMPLE : [”A”,”B”,2,7.9] → Here A is In 0 index.

List Methods :

- **append()** : It is Used To Add Element In Last Of List
- **insert()** : It Is Used To Add Element In Specific Index
- **pop()** : It Is Used To Remove Element By Index Number
  By Default pop() Remove The Last Element
- **remove()** : It Is Used To Remove Element Using By Value
- **extend()** : **It** Is Used To Add More Than One Element
- **clear()** : It Is Used To Clear The List
- **Replace Element In List** : list[1]= "New"
  → 1 is Index No. and New is Element

```
list1 = ["a","b",2,2.6]
list1.append("c")
print(list1)
list1.insert(2,"d")
print(list1)
list1.extend(["f","g"])
print(list1)
list1.remove("a")
print(list1)
list1.pop(2)
print(list1)
list1.clear()
print(list1)
list1[2]=6
print(list1)
```

FOR PRACTICE
CLICK HERE