

---

## Display the Result of Python Program

---

=>To display the result of any Python Program on the console (Monitor), we use a pre-defined called print()  
=>In Other words, print() is pre-defined function, which is used for displaying the result of Python program on the console (Monitor)

---

=>print() contains Various Syntaxes and They are

---

Syntax1:        print(value)  
                  (OR)  
                  print(Val1,Val2....Val-n)

=>This Syntax display only Values on the console.

---

Examples:

---

```
>>> sno=10
>>> sname="Ram"
>>> print(sno,sname)-----10 Ram
>>> print(sno)-----10
>>> print(sname)-----Ram
>>> a=10
>>> b=20
>>> print(a)-----10
>>> print(b)-----20
>>> print(a,b)-----10 20
```

---

Syntax2:    print(Message)

=>Here Message Represents any str data type.

=>This Syntax is used for displaying Only Messages on the console.

---

Examples:

---

```
>>> print("Hello World")-----Hello World
>>> print("Python Invented by Guido Van Rossum")----Python
                                                Invented by Guido Van Rossum
```

---

Syntax3:        print(Message Cum Values)  
                  (OR)  
                  print(Values Cum Message)

=>This Syntax displays Messages Cum Values.

-----  
Examples:

-----  
>>> print("Val of a=",a)-----Val of a= 10  
>>> print(a," is the value of a")-----10 is the value of a  
>>> a=10  
>>> b=20  
>>> c=a+b  
>>> print("sum=",c) -----sum= 30  
>>> print(c," is the sum")-----30 is the sum  
>>> print("Sum of ",a," and ",b,"=",c)---Sum of 10 and 20 = 30  
>>> a=10  
>>> b=20  
>>> c=30  
>>> d=a+b+c  
>>> print("Sum of",a," ",b," and ",c,"=",d)---Sum of 10 , 20  
and 30 = 60  
-----

Syntax4:            print(Message Cum Values with format())  
                          (OR)  
                          print(Values Cum Message with format())

=>This Syntax display Values cum Messages by using format().  
Here format() is supplying nth values for Empty Curly Braces  
on the basis First Cum First Served Basis.

-----  
Examples:

-----  
>>> print("Val of a={}".format(a))-----Val of a=10  
>>> a=100  
>>> b=200  
>>> c=a+b  
>>> print("Sum={}".format(c))-----Sum=300  
>>> print("Sum of {} and {}={}".format(a,b,c) )---Sum of 100  
and 200=300  
>>> print("{} is the sum".format(c))-----300 is the sum  
-----

>>> a=10  
>>> b=20  
>>> c=30  
>>> d=a+b+c  
>>> print("Sum of {},{} and {}={}".format(a,b,c,d))----Sum of  
10,20 and 30=60

```
>>> sno=10
>>> sname="Ram"
>>> print("My Roll Number is {} and Name is '{}'"
".format(sno,sname))-----My Roll Number is 10 and Name is 'Ram'
```

---

Syntax5:        `print(Message Cum Values with format Specifiers)`  
                  (OR)  
                  `print(Values Cum Message with format Specifiers )`

Here format specifiers represents %d for Integer data, %f for float data and %s for str data

=>If any other data types does not contains format specifiers then we must those into str type and display by using %s.

---

Examples:

---

```
>>>a=10
>>> print("Val of a=%d" %a)-----Val of a=10
>>> print("%d is the value of a" %a)-----10 is the value of a
>>> a=10
>>> b=20
>>> c=a+b
>>> print("Sum=%d" %c)-----Sum=30
>>> print("Sum of %d and %d=%d" %(a,b,c) )---Sum of 10 and 20=30
>>> print("Sum of %f and %f=%f" %(a,b,c) )---Sum of 10.000000
and 20.000000=30.000000
>>> print("Sum of %0.2f and %0.2f=%0.2f" %(a,b,c) )----Sum of
10.00 and 20.00=30.00

>>> a=23.45
>>> print("Val of a=%f" %a)-----Val of a=23.450000
>>> print("Val of a=%0.3f" %a)-----Val of a=23.450
>>> print("Val of a=%d" %a)-----Val of a=23
```

---

```
>>> sno=10
>>> sname="Rossum"
>>> print("My Roll No:%d and Name is '%s' " %(sno,sname))--My
Roll No:10 and Name is 'Rossum'
```

---

```
Syntax6: print(Message Cum Values with format() or format
           specifier with end option)
```

=>This Syntax displays result of python program in same Line or Horizontally by using "end" option.

```
=>Here Output values are printed Value ends with value...so
on(end to end value in same line)
```

## Examples

```
>>> lst=[10,"Ram",34.56,"Python"]
```

```
>>> for val in lst:
...     print(val,end=" ")-----10 Ram 34.56 Python
```

```
>>> r=range(10,21)
>>> for val in r:
...     print(val,end=" ")-----10 11 12 13 14 15 16 17 18 19 20
```

\_\_\_\_\_X\_\_\_\_\_

Special Case:

```
>>> s="PYTHON"
>>> s=3*s           # Here * operator is called Repetition Operator
>>> print(s) -----PYTHONPYTHONPYTHON
>>> s="Hyd"
>>> print(s*10) -----HydHydHydHydHydHydHydHydHydHyd
```

```
>>> print(s*2)-----HydHyd
```

```
>>> print("Hyd"*4) -----HydHydHydHyd
```

```
>>> print("="*30)-----=====
```

```
>>> print ("*" * 30) -----*****
```

[illegible]

=====X=====