print()

print() is standard output function. This function is used to print or display information/data on console/monitor.
It is a predefined function and exists in __builtins__ library and it is default library imported by any python program.

Syntax-1: print(data/info/values)
Syntax-2: print(values,sep=’ ‘)
Syntex-3: print(values,sep=’ ‘,end=’\n‘)

Example:

```
# Example of print function

print(10)
print(1.5)
print(1+2j)
print(True)
print(None)
print("Python")
```

Output:

```
10
1.5
(1+2j)
True
None
Python
```

Example:

```
# Example of print function

a=10
b=1.5
c=1+2j
d=True
e=None
f="Python"

print(a)
print(b)
print(c)
print(d)
print(e)
print(f)
```

Output:

```
10
1.5
(1+2j)
True
None
Python
```

Syntax-2: print(values,sep=’ ‘)

Print function print/display more than one value using default separator space.

Example:

```
# Example of print function to print multiple values


print(10,20,30,40,50)
print(1,1.5,1+2j,False,"Python")

print(10,20,30,40,50,sep=";")
print(10,20,30,40,50,sep=",")
print(10,20,30,sep="*")
print(10,20,30,sep="\n")
print(10,sep="%")
print(10,20,sep=':')
```

Output:

```
10 20 30 40 50
1 1.5 (1+2j) False Python
10;20;30;40;50
10,20,30,40,50
10*20*30
10
20
```

```
30
10
10:20
```

Example:

```
# Example of print function

a=10
b=1.5
c=1+2j
d=True
e=None
f="Python"

print(a,b,c,d,e,f)

print(a,b,c,d,e,f,sep=',')
print(a,b,c,d,e,f,sep=';')
```

Output:

```
10 1.5 (1+2j) True None Python
10,1.5,(1+2j),True,None,Python
10;1.5;(1+2j);True;None;Python
```

Syntex-3: print(values,sep=’ ‘,end=’\n‘)

After printing values, print function insert newline (using end). The default of end is \n.

```
# Example of print function

print(10)
print(100,200)
print(1,2,3,4,5)
print(10,20,30,40,50,end=';')
print("python","java")
print(3.12,end='\t')
print("nareshIT")
```

Output:

```
10
100 200
1 2 3 4 5
10 20 30 40 50;python java
3.12	nareshIT
```

Example:

```
# Find Output

print("PYTHON","LANGUAGE")
print("PYTHON","LANGUAGE",sep="\t")
print("PYTHON","LANGUAGE",end='3.12')
print("GENERAL PURPOSE")


# Line1: PYTHON LANGUAGE
# Line2: PTYHON	LANGUAGE
# Line3: PYTHON LANGUAGE3.12GENERAL PURPOSE
```

Print function receive 3 inputs

1. Values → empty
2. Sep → space
3. End → \n

Example:

```
print()
print()
print()
```

Output:

Three blank lines are inserted

Example:

```
# Write a program to add two integers
```

```
n1=10
n2=20
n3=n1+n2
print(n1,n2,n3)
```

Output:

```
10 20 30
```

input() function

input() is standard input function, this function is used to read data from keyboard.
A program read data from keyboard using input function.
End user input data to program using input function.
This function is exists within __builtins__ module.
Input function read values of type string.

Syntax1: input()
Syntax2: input(prompt)

Prompt is a string, which is displayed before reading value.

Example:

```
# input without prompt
n1=input()
n2=input()
print(n1,n2)
```

Output:

```
1
2
1 2
```

Example:

```
# input  prompt
n1=input("Enter N1 Value ")
n2=input("Enter N2 Value ")
print(n1,n2)
```

Output:

```
Enter N1 Value 10
Enter N2 Value 20
10 20
```

Example:

```
# Write a program to read rollno,name,course
# and print

rollno=input("Enter Rollno ")
studname=input("Enter Name ")
course=input("Enter Course ")
print(rollno,studname,course,sep="\n")
```

Output:

```
Enter Rollno 101
Enter Name naresh
Enter Course python
101
naresh
python
```

Example:

```
# write a program to input two integers and
# print sum of two integer

n1=input("Enter First Number ")
n2=input("Enter Second Number ")
print(type(n1),type(n2))
n3=n1+n2

print("Sum is ",n3)
```

Output:

```
Enter First Number 10
Enter Second Number 20
<class 'str'> <class 'str'>
Sum is  1020
```

Example:

```
>>> a="10"
>>> b="20"
>>> c=a+b
>>> print(c)
1020
>>> n1=10
>>> n2=20
>>> n3=n1+n2
>>> print(n3)
30
```

Type conversion functions/Type Casting

Type casting is process converting one type of value to another type. Python allows performing these typecasting using type conversion functions.

1. int()
2. float()
3. complex()
4. bool()
5. str()

int()

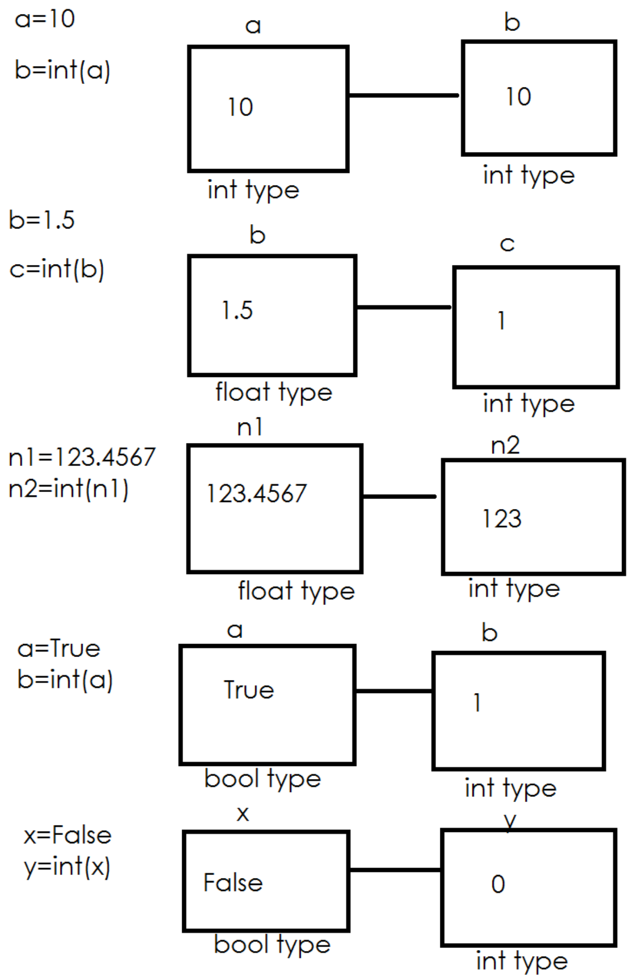
This function is used to perform the following conversion.

1. Int to int
2. Float to int
3. Bool to int
4. String to int

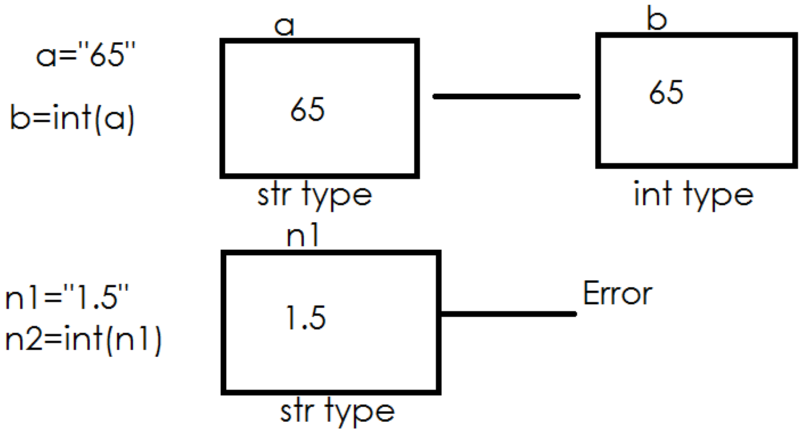
Syntax-1: int(value)
Syntax-2: int(value,base=10)

Syntax-1 is used to perform the following conversion

1. Int to int
2. Float to int
3. Bool to int



Syntax-2 is used to convert string representation of integer to integer.
Int(value,base=10)



```
>>> a=int(10)
>>> b=int(1.5)
>>> c=int(True)
>>> d=int(False)
>>> print(a,b,c,d)
10 1 1 0
>>> e=int("65")
>>> print(e)
65
>>> f=int("1.5")
Traceback (most recent call last):
  File "<pyshell#17>", line 1, in <module>
    f=int("1.5")
ValueError: invalid literal for int() with base 10: '1.5'
>>> f=int("0o25")
Traceback (most recent call last):
  File "<pyshell#18>", line 1, in <module>
    f=int("0o25")
ValueError: invalid literal for int() with base 10: '0o25'
>>> f=int("0o25",base=8)
>>> g=int("0xab",base=16)
>>> h=int("0b1010",base=2)
>>> print(oct(f),hex(g),bin(h))
0o25 0xab 0b1010
>>> print(f,g,h)
21 171 10
```

Example:

```
# write a program to input two integers and
# print sum of two integer

n1=input("Enter First Number ")
n2=input("Enter Second Number ")
n3=int(n1)+int(n2)
print("Sum is ",n3)
```

Output:

```
Enter First Number 100
Enter Second Number 200
Sum is  300
```