→ to make our prog. dynamic.

Input() function:-

- input given by us/user/keyboard/or many sources. that's why it is dynamic.
- Input is a standard input function, the function is used to input/read data from keyboard.
- input() fn is used to input/read "string" value-only.
- Input() function allowes to input/read one value.

Syntax:-

Variable = Input([prompt])

Eg:- a = input("input value/enter the value a")

b = input("Input value/enter the value b")

c = input("enter value of c")

print(a, b, c)

print(type(a), type(b), type(c))

O/p ⇒ class<'str'> class<'str'> class<'str'>

02/11/23 Thur

Ex:- W.A.P to print sum of two no.

n1 = input("enter 1st value")

n2 = input("enter 2nd value")

n3 = n1+n2

print(f'sum of {n1} and {n2} is {n3}')

O/p:-

enter 1st value 10

enter 2nd value 20

sum of 10 and 20 is 1020.

Type conversion function or Type casting.

⇒ Type conversion is a process of converting one type of object to another type.

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

⇒ All these are in-built functions & available in default module imported by any python — builtins

Eg:- import cmath → complex math
cmath.sqrt(-9) } for negative sqrt.

Eg import math
math.sqrt(9) } for +ve sqrt.

if we want to check how many function are avilable we can see like:- dir(__builtins__)
↓ module name.
this fn we can use directly / by default it is inbuilt.

Eg: import os
dir(os) → it will should all fn.

Eg- import calendar.
dir(calendar)
↳ we can see content of the module / library

1> int() function:

This fn is used to create integer object (OR) this fn performs the following conversions.

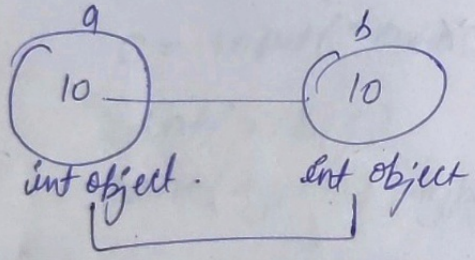
a) Int to Int.
b) Float to Int.
c) String to Int.
d) Bool to Int.

Synt1: int(value)
Synt2: int(value, base=10)

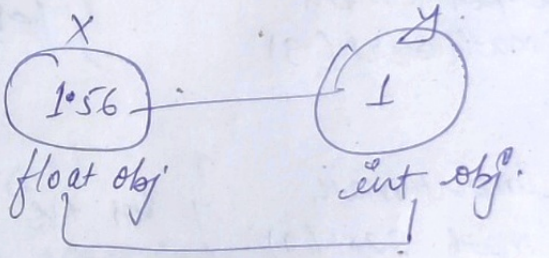
Synt1:- is used to convert float, int, bool, to integer object.

Synt2:- is used to convert string to integer object.

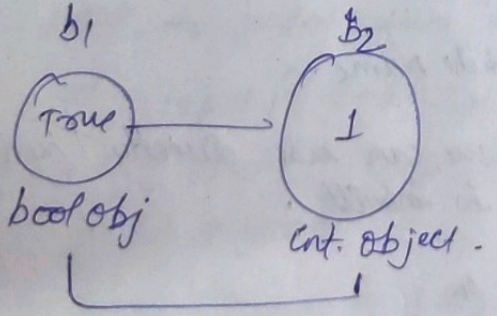
a) a=10 b=int(a)



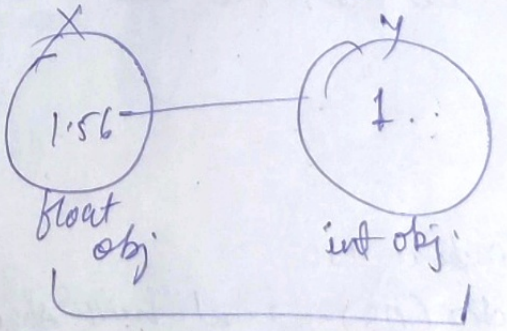
b) X=1.56 y=int(x)



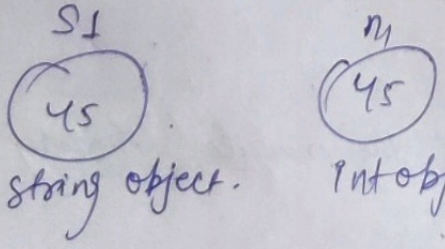
c) b1=true b2



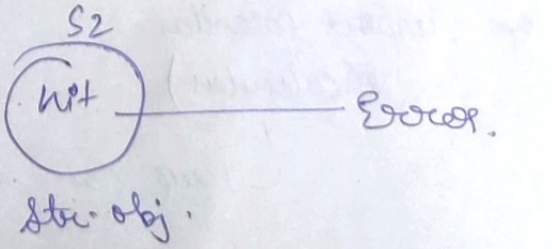
d) X=1.56 y=int(x)



e) S1="45" n1=int(S1)

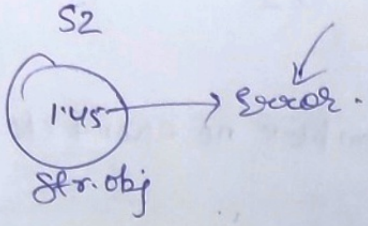


f) S2="nit" n2=nit(S2)



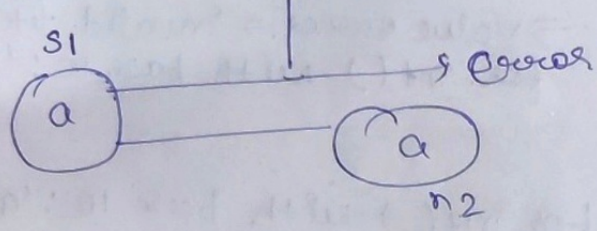
S2 = "1.45"

n3 = int(S2)



g) S1 = "a"

n1 = int(S1)

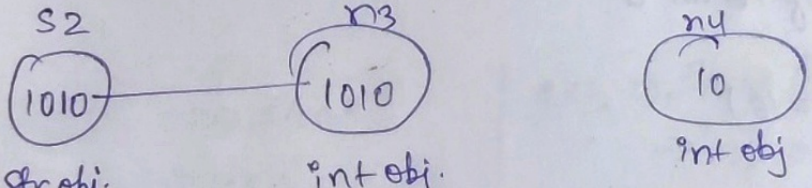


n2 = int(S1, base = 16)

h) S2 = "1010"

n3 = int(S2)

n4 = int(S2, base = 2)



Ex: # W.A.P to print sum of integers.

input two integers from keyboard.

```
n1 = input("Enter first value")
n2 = input("  "  2nd value")
n3 = int(n1) + int(n2)
print(f'sum of {n1} and {n2} is {n3}')
```

O/p:- Enter first value 100

Enter 2nd " 200

sum of 100 and 200 is 300.

Ex1 >>> int(10) → 10 | Eg2: >>> int(1.5) → 1 | Eg3: >>> int(true) → 1

Ex:4 >>> int(False) → 0 | Eg5 >>> int(1+2j) → Type Error, complex no can't change.

Eg:-6 >>> int("65") → 65.

Eg:-7 >>> int("1.25") → value error → invalid literal for int() with base 10: '1.25'.

Eg:-8 >>> int("a") → value Error → invalid literal for int() with base 10: 'a'.

Eg:-9 >>> int("a", base=16) → 10 ← Hexa. value of 10 is 'a'.

Eg:- int("ff", base=16) → 255

Eg:- int("ffff", base=16) → 65535

Eg:-12 >>> int("1010", base=2) → 10.

Eg:13 → >>> int("1010") → 1010.

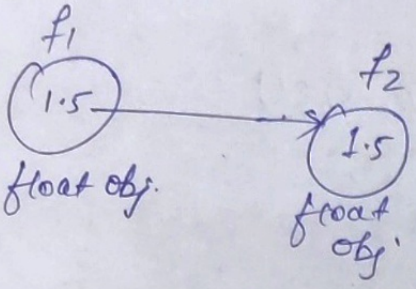
2.) float() function:-

This fn is used to create float obj or to perform the following conversions.

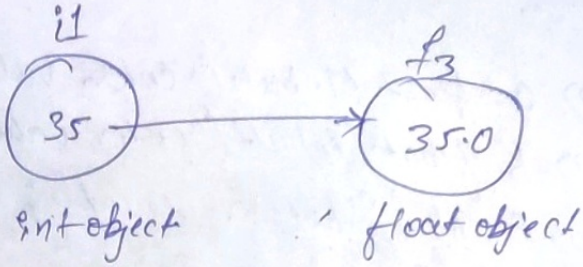
1) float to float
2) int to float
3) string to float.
4.) bool to float.

Syntax:- float(value)

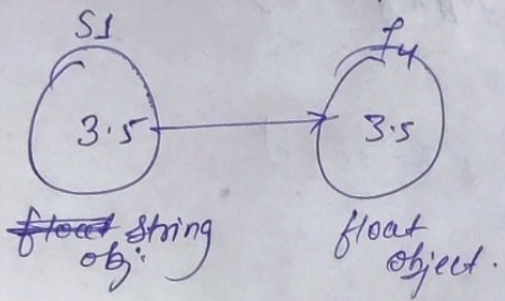
(a) f1 = 1.5 f2 = float(f1)



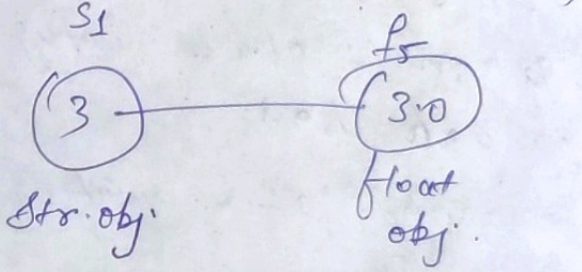
(b) i1 = 35 f3 = float(i1)



(c) S1 = "3.5" f4 = float(S1)



(d) S1 = "3" f5 = float(S1)



Examples:- >>> float
→ 1.5.

Ex:2- >>> float(1)
→ 1.0.

Ex3 → >>> float(True)
→ 1.0

Ex4 → >>> float(False)
→ 0.0

Ex5 → float("1.5")
→ 1.5

Ex6: float("15e-1")
→ 1.5

Ex7:- >>> float("15")
→ 15.0

Ex8:- >>> float("abc")
→ value error: could not convert string to float: 'abc'.

Examples:- # WAP to swap / interchange two float values input from stdin.

```
>>> a=float(input("enter value of a"))
>>> b=float(input("enter value of b"))
>>> print(f'before swaping a={a} and b={b}')
>>> c=a
>>> a=b
>>> b=c
>>> print(f'after swaping a={a} and b={b}')
>>> a,b=b,a
>>> print(f'after swaping a={a} and b={b}')
>>> a=a+b
>>> b=a-b
>>> a=a-b
>>> print(f'after swaping a={a} and b={b}')
```

O/p:-

```
enter value of a = 1.5
enter " " b = 2.5
befor swaping a= 1.5 and b=2.5
after " a=2.5 and b=1.5
after " a = 1.5 and b=2.5
after " a = 2.5 and b=1.5
```