OPERATORS

Arithmetic operators

+:

* Used with minimum of 2 numbers to perform 🡺addition
* Used with a single number to represent a positive number

print(5 + 3 + 2) #addition

print(88.5 + 99.2)  #addition

print(5 + 8 + 9 + 4)#addition

print(+8)#positive

print(+88.5)#positive number

-:

print(67 - 32 - 7 -2)#subtration

print(45 - 67)#subtration

print(1000.8 - 86.6 - 77.9)#substration

print(-66.6)#negative number

print(-5)#negative number

* Used with minimum of 2 numbers to perform 🡺 subtraction
* Used with a single number to represent a negative number

\*:

print(3 \* 7 \* 6 \* 9 \* 77)#multiplication

print(4 \* 99)#multiplication

print(88.5 \* 99.2)#multiplication

print(\*8)#error

* Used with compulsorily with minimum of 2 numbers to perform 🡺 multiplication

\*\*(exponentiation):

print(4\*\*3)# 2\*2\*2 = 8

print((-6)\*\*2)

print(7.2\*\*2)

* Return exponent value

Division operation

1. Floor division //

Return non decimal output

1. Float division /

Return decimal output

1. Modulous %

Return quotient

#floor div

print(20 // 2)

print(22 // 7)

#float div

print(200 / 2)

print(22 / 7)

#modulus for remainder in return

print(10 % 2)

print(22 % 7)

print(3 % 77)

print(8 % 88)

a = 100

b = 100

c = 100

d = a

print(a == b == c == d)

#same shared memory

print(a is d)

print(b is c)

print(a is c)

print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

#same variable different memory

x = 10

print(x)

print(id(x))

x = 20

print(x)

print(id(x))

x = 30

print(x)

print(id(x))

a = 100

b = 100

c = 100

d = a

print(a == b == c == d)

#same shared memory

print(a is d)

print(b is c)

print(a is c)

print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

#same variable different memory

x = 10

print(x)

print(id(x))

x = 20

print(x)

print(id(x))

x = 30

print(x)

print(id(x))

s1 ="abbc123"

print("a" in s1)

print("bbc" in s1)

print("12 in s1")

#print(23 in s1)

print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

a = 10

print(1 in a) #error It id sungle valued data #error

 a = 10

print("1"in a) #error

a = ""

print("ab" in a)

**Control flow statements**

The statements that control the execution flow of a program

3 types:

1, Conditional statem:

* Simple if
* If – else
* elif / if – elif / elif ladder

Note: Combination of these above statm’s are “nested if conditions”

2. Looping statm’s:

* For loop

i. for with range() incre, decre

ii. for without range()

* While loop

i. incre

ii. decre

3. Jumping stmts:

* Break, continue, return

Conditional stamts:

Simple if :

age = int(input("enter your age"))

if (age >= 18):

    print("you can vote")

print("you are a citizen")

If – else:

Syx:

If bool\_cond

Logic

Else:

Alternate

Remain stmts

**Note: Divisibility or factor the first operator used is mod %**

num = int(input("enter the number"))

if (num % 2 == 0):

    print(f"{num} is even")

else:

    print(f"{num} is odd")

print("Prog exec")

**elif:**

It will be used when in a given scenario there will be multiple condi to be checked and for each condi there will be specific logic

Syntax:

If bool\_cond1:

#logic1

Elif bool\_cond2:

#logic2

Elif bool\_cond3:

#logic3

:

:

Else:

#alternate logic

Program to print statements accordingly

If the num is posit but even : posit even num

If num is neg but even : neg even num

If the num is posit but odd: posit odd num

If num is neg but odd: neg odd num