\*\*\*\*\*\*\*\*\* Welcome to Python Programing Day 3 \*\*\*\*\*\*\*\*\*\*\*\*

Agenda of the day

Installation of Pycharm

Virtual Environment and its benefits

Arithmetic Operator

Addition (+)

sub (-)

multiplication (\*)

division (/)

> True # /

> floor # //

exponent(\*\*)

# 2\*\*3 ~ 2^3

# not applicable to string

modulus(%)

# this will give rem

Comparison Operator

1. Equal to ==

2. Not equal !=

3. Greater than >

4. less than <

5. Greater than equal to >=

6. less than equal to <=

eg:

a = 3

b = 2

print(a == b) # equal

print(a != b) # not equal

print(a > b) # greater than

print(a < b) # less than

print(a >= b) # greater than equal to

print(a <= b) # less than equal to

Note: == and != can be used with strings only and other operator can be used with string if we do operator overloading.

Logical Operator

and --->

print(True and True) # True

print(True and False) # False

or --->

print(True or True) # True

print(True or False) # True

not --->

print(not True) # False

print(not False) # True

Assignment Operator

= (is set to)

eg:

a = 4

print(a)

+=

eg:

a += 1 # a = a + 1

print("new value of a", a)

-=

eg:

a -= 1 # a = a - 1

print("new value of a", a)

\*=

eg:

a \*= 1 # a = a \* 1

print("new value of a", a)

/=

eg:

a /= 1 # a = a / 1

print("new value of a", a)

\*\*=

eg:

a \*= 1 # a = a \* 1

print("new value of a", a)

//=

eg:

a //= 1 # a = a // 1

print("new value of a", a)

%=

eg:

a %= 1 # a = a % 1

print("new value of a", a)

Bitwise Operator

AND Operator (&)

eg:

a = 10

b = 4

print(a & b)

OR Operator (|)

eg:

a = 10

b = 4

print(a | b)

Not Operator (~)

eg:

a = 10

print(~a)

Precedence of Operator

Associativity