21 May

**Python Basic - 2**

Q.1. Create two int type variables, apply addition, subtraction, division and multiplications and store the results in variables. Then print the data in the following format by calling the variables:

First variable is \_\_ & second variable is \_\_.

Addition: \_\_ + \_\_ = \_\_

Subtraction: \_\_ - \_\_ = \_\_

Multiplication: \_\_ \* \_\_ = \_\_

Division: \_\_ / \_\_ = \_\_

First variable is 10 & second variable is 5

Addition: 10 + 5 = 15

Subtraction: 10 - 5 = 5

Multiplication: 10 \* 5 = 50

Division: 10 / 5 = 2

Q.2. What is the difference between the following operators:

(i) ‘/’ & ‘//’

(ii) ‘\*\*’ & ‘^’

The operators / and // are both used for division, but they have different results. The / operator performs floating-point division, which means that the result is a floating-point number. The // operator performs integer division, which means that the result is an integer. If the quotient of the division is not an integer, the // operator will round down to the nearest integer.

The operators \*\* and ^ are both used for exponentiation, but they have different results. The \*\* operator raises the first operand to the power of the second operand. The ^ operator performs bitwise XOR, which is a logical operation that returns 1 if the corresponding bits of the two operands are different, and 0 if they are the same.

Q.3. List the logical operators.

There are three logical operators in Python:

AND (&&)

OR (||)

NOT (!)

# Check if both x and y are greater than 0

x = 1

y = 2

if x > 0 and y > 0:

print("Both x and y are greater than 0")

# Check if either x or y is greater than 0

if x > 0 or y > 0:

print("Either x or y is greater than 0")

# Check if x is not equal to y

if x != y:

print("x is not equal to y")

Q.4. Explain right shift operator and left shift operator with examples.

# Shift the bits of the number 10 to the right by 1 place

x = 10

y = x >> 1

print(y) # 5

The right shift operator (>>) and the left shift operator (<<) are bitwise operators that are used to shift the bits of a number to the right or left, respectively.

Q.5. Create a list containing int type data of length 15. Then write a code to check if 10 is present in the list or not.

# Create a list containing int type data of length 15

list\_of\_ints = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

# Check if 10 is present in the list or not

if 10 in list\_of\_ints:

print("10 is present in the list")

else:

print("10 is not present in the list")