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 \_5\_ & second variable is \_8\_.

Addition: 5\_\_ + \_8\_ = \_13\_

Subtraction: \_5\_ - \_8\_ = -3\_\_

Multiplication: \_5\_ \*8 \_\_ = \_\_45

Division:5\_\_ / \_8\_ = \_0.625\_

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

(i) ‘/’ & ‘//’

(ii) ‘\*\*’ & ‘^’

= In Python, the single forward slash (/) is the division operator, used for regular division. The double forward slash (//) is the floor division operator, which performs division and rounds down to the nearest whole number. For example, 5 / 2 would result in 2.5, while 5 // 2 would result in 2.

For numeric data types, double-asterisk (\*\*) is defined as an Exponentiation Operator:

a \*\* b is a raised to the b power. The same \*\* symbol is also used in function argument and calling notations, with a different meaning (passing and receiving arbitrary keyword arguments).

a ^ b will return a value with only the bits set in a or in b but not both.

&; It is bitwise operator.It performs a bitwise "AND" operation on two integers.

Q.3. List the logical operators.

= They perform Logical AND, Logical OR, and Logical NOT operations.

The logical AND operator returns True if both the operands are True else it returns False.

Logical OR operator returns True if either of the operands is True.

The logical not operator works with a single boolean value. If the boolean value is True it returns False and vice-versa.

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

= right-shift operator ( >> ), which moves the bits of an integer or enumeration type expression to the right, and the left-shift operator ( << ), which moves the bits to the left.

Right shift operator shifts least significant bits to left by the number on the right side of the ">>" symbol. Hence, "x >> 2" causes two bits of the binary representation of to left.

Eg; a=60

print ("a:",a, "a<<2:", a<<2)

a: 60 a<<2: 240

Left shift operator shifts most significant bits to right by the number on the right side of the "<<" symbol. Hence, "x << 2" causes two bits of the binary representation of to right.

Eg; a=60

print ("a:",a, "a>>2:", a>>2)

a: 60 a>>2: 15

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.

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

#checking if element 10is present

# in the given list or not

i=10

# if element present then return

exist, otherwise not exist

if i in lst:

print("exist")

else:

print("not exist")