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

Ans.1 # Creating variables

var1 = 15

var2 = 8

# Performing calculations

addition = var1 + var2

subtraction = var1 - var2

multiplication = var1 \* var2

division = var1 / var2

# Printing results

print(f"First variable is {var1} & second variable is {var2}.")

print(f"Addition: {var1} + {var2} = {addition}")

print(f"Subtraction: {var1} - {var2} = {subtraction}")

print(f"Multiplication: {var1} \* {var2} = {multiplication}")

print(f"Division: {var1} / {var2} = {division}")

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

(i) ‘/’ & ‘//’

(ii) ‘\*\*’ & ‘^’

Ans.2

(i) **'/' is the division operator that performs regular division and returns a floating-point result. '//' is the floor division operator that performs division and returns the largest whole number that's less than or equal to the division result.**

**(ii) '\*\*' is the exponentiation operator that raises a number to a certain power. '^' is not a standard arithmetic operator in Python. It's often confused with the exponentiation operator, but it's not used for that purpose in Python.**

Q.3. List the logical operators.

Ans.3 The logical operators in Python are:

* **and**: Returns True if both operands are True.
* **or**: Returns True if at least one operand is True.
* **not**: Returns True if the operand is False, and vice versa.

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

Ans.4 **The right shift (>>) and left shift (<<) operators are used to shift the bits of a binary number to the right or left, respectively. The right shift operator shifts the bits to the right by a specified number of positions, filling the vacant positions with zeros. The left shift operator shifts the bits to the left by a specified number of positions**.

Example of right shift:

**x = 16 # Binary: 10000**

**result = x >> 2 # Binary: 00100**

**print(result) # Output: 4**

Example of left shift:

**x = 5 # Binary: 00101**

**result = x << 3 # Binary: 101000**

**print(result) # Output: 40**

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.

Ans.5

# Creating a list

data\_list = [2, 5, 8, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65]

# Checking if 10 is present in the list

if 10 in data\_list:

print("10 is present in the list.")

else:

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