**Q-Demonstrate the use of all types of operators**

1. Arithmetic Operators:

Arithmetic operators perform mathematical calculations on numerical values. They include addition (+), subtraction (-), multiplication (\*), division (/), modulo (remainder) (%), and exponentiation (\*\*).

1. Assignment Operators:

Assignment operators are used to assign values to variables. They include simple assignment (=), as well as compound assignment operators such as addition assignment (+=), subtraction assignment (-=), multiplication assignment (\*=), division assignment (/=), modulo assignment (%=), and exponentiation assignment (\*\*=).

1. Comparison Operators:

Comparison operators are used to compare two values and return a Boolean result (True or False). They include equal to (==), not equal to (!=), greater than (>), less than (<), greater than or equal to (>=), and less than or equal to (<=).

1. Logical Operators:

Logical operators are used to perform logical operations on Boolean values. They include logical AND (and), logical OR (or), and logical NOT (not). These operators are often used to combine multiple conditions and determine the outcome based on the truth or falsehood of the expressions.

1. Bitwise Operators:

Bitwise operators perform operations on individual bits of binary numbers. They include bitwise AND (&), bitwise OR (|), bitwise XOR (^), bitwise NOT (~), left shift (<<), and right shift (>>). These operators are commonly used in low-level programming or in scenarios where manipulation of individual bits is required.

1. Identity Operators:

Identity operators are used to compare the memory addresses of two objects to determine if they are the same object or not.

* is operator: It returns True if two variables point to the same object, and False otherwise.
* is not operator: It returns True if two variables point to different objects, and False if they point to the same object.

1. Membership Operators:

Membership operators are used to test whether a value or variable is a member of a sequence (such as a string, list, tuple, or set) or not.

* in operator: It returns True if a value or variable is found in the sequence, and False otherwise.
* not in operator: It returns True if a value or variable is not found in the sequence, and False if it is found.