OPERATORS

1. Arithmetic Operators

Used for mathematical operations:

- `+` (Addition) `a + b`
- `-` (Subtraction) `a - b`
- `*` (Multiplication) `a * b`
- `/` (Division) `a / b` (float division)
- `//` (Floor Division) `a // b` (integer division)
- `%` (Modulus) `a % b` (remainder)
- `**` (Exponentiation) `a ** b` (power)

2. Comparison (Relational) Operators

Used to compare values and return `True` or `False`:

- `==` (Equal) `a == b`
- `!=` (Not Equal) `a != b`
- `>` (Greater than) `a > b`
- `<` (Less than) `a < b`
- `>=` (Greater than or equal to) `a >= b`
- `<=` (Less than or equal to) `a <= b`

3. Logical Operators

Used for logical operations (return `True` or `False`):

- `and` (Logical AND) `a and b`
- `or` (Logical OR) `a or b`
- `not` (Logical NOT) `not a`

4. Bitwise Operators

Used for bit-level operations:

- & (Bitwise AND) `a & b`
- | (Bitwise OR) `a | b`
- ^ (Bitwise XOR) `a ^ b`
- ~ (Bitwise NOT) `~a`
- << (Left Shift) `a << n`
- >> (Right Shift) `a >> n`

5. Assignment Operators

Used to assign values to variables:

- = (Assign) `a = 10`
- += (Add and assign) `a += 5` (same as `a = a + 5`)
- -= (Subtract and assign) `a -= 5`
- *= (Multiply and assign) `a *= 5`
- /= (Divide and assign) `a /= 5`
- //= (Floor divide and assign) `a //= 5`
- %= (Modulus and assign) `a %= 5`
- **= (Exponentiation and assign) `a **= 5`
- &= (Bitwise AND and assign) `a &= 5`
- |= (Bitwise OR and assign) `a |= 5`
- ^= (Bitwise XOR and assign) `a ^= 5`
- <<= (Left shift and assign) `a <<= 5`
- >>= (Right shift and assign) `a >>= 5`

6. Identity Operators

Used to compare memory locations of objects:

- `is` `a is b` (True if a and b refer to the same object)
- `is not` `a is not b` (True if a and b are different objects)

7. Membership Operators

Used to check if a value exists in a sequence:

- `in` `a in list` (True if a is in the list)
- `not in` `a not in list` (True if a is not in the list)