Operators

Performing Calculations and Comparisons in Python

Operators and Expressions

Operators are symbols that perform operations on data **Expressions** combine values and operators to create new values

```
2 + 3        # Expression that evaluates to 5
name = "Bob" # Expression that assigns "Bob" to name
```

Arithmetic Operators

The basics - Python as a calculator

```
5 + 3   # 8   Addition
5 - 3   # 2   Subtraction
5 * 3   # 15  Multiplication
5 / 3   # 1.67 Division (always gives float)
```

More Arithmetic Operators

```
5 // 3  # 1    Floor division (whole number only)
5 % 3   # 2    Modulus (remainder)
5 ** 3  # 125  Exponentiation (power)
```

Common uses:

- `//` for splitting things evenly
- `%` for checking if numbers are even/odd
- `**` for squares, cubes, etc.

Order of Operations

Python follows **PEMDAS** (just like math class)

```
2 + 3 * 4      # 14 (not 20!)
(2 + 3) * 4    # 20
2 ** 3 + 1     # 9 (not 16!)
```

When in doubt, use parentheses!

Comparison Operators

Compare values - always return `True` or `False`

```
5 == 5   # True   Equal to
5 != 3   # True   Not equal to
5 > 3    # True   Greater than
5 < 3    # False  Less than
5 >= 5   # True   Greater than or equal
5 <= 3   # False  Less than or equal
```

Common Comparison Mistakes

```
# Assignment vs Comparison
x = 5      # Assignment (sets x to 5)
x == 5     # Comparison (checks if x equals 5)

# Strings vs Numbers
"5" == 5   # False (string vs number)
"5" == "5" # True (both strings)
```

Remember: `=` assigns, `==` compares!

Logical Operators

Combine `True` / `False` values

```
True and False   # False (both must be True)
True or False    # True  (at least one is True)
not True         # False (flips True/False)
```

Real example:

```
age = 16
has_license = True
can_drive = age >= 16 and has_license
print(can_drive)  # True if age is 16+ and has a license
```

Assignment Operators

Shortcuts for common operations

```
x = 10
x += 5    # Same as: x = x + 5    (x is now 15)
x -= 3    # Same as: x = x - 3    (x is now 12)
x *= 2    # Same as: x = x * 2    (x is now 24)
x /= 4    # Same as: x = x / 4    (x is now 6.0)
```

Use these to make your code cleaner!

Building Something "Useful"

Combining these concepts, we can create a simple (and useful!) program:

```python
# Calculate tip for a restaurant bill
cat_age = int(input("Enter your cat's age in years: "))
in_human_years = cat_age * 5
print(f"Your cat is {in_human_years} human years old.")
```

Note how we use:

- `input` to get user input
- `int` to convert input to an integer
- `*` to perform arithmetic
- `f"..."` (f-string) to format the output string
- `print` to display results

Exercise: Build a Calculator

bigd103.link/tip-calculator