**🐍 Python Number Types (Clean Notes)**

In Python, numbers are objects of different **numeric types**.

The main numeric types are:

**✅ 1. Integers (int)**

* Whole numbers (positive, negative, or zero).
* No decimal point.
* No limit on size (Python’s int can handle very large numbers).

📌 Examples:

a = 10

b = -42

c = 0

🔹 Type:

print(type(a)) # <class 'int'>

# Python Arithmetic Operators (with Examples)

## ✅ 1. Addition (+)

Adds two numbers.

python

CopyEdit

a = 10

b = 5

print(a + b) # ✅ Output: 15

## ✅ 2. Subtraction (-)

Subtracts the second number from the first.

python

CopyEdit

a = 10

b = 5

print(a - b) # ✅ Output: 5

## ✅ 3. Multiplication (\*)

Multiplies two numbers.

python

CopyEdit

a = 10

b = 5

print(a \* b) # ✅ Output: 50

## ✅ 4. Division (/)

Divides the first number by the second (always returns a float).

python

CopyEdit

a = 10

b = 5

print(a / b) # ✅ Output: 2.0

## ✅ 5. Floor Division (//)

Divides and returns the **integer part** (floors the result).

python

CopyEdit

a = 10

b = 3

print(a // b) # ✅ Output: 3

## ✅ 6. Exponentiation (\*\*)

Raises the first number to the power of the second.

python

CopyEdit

a = 2

b = 3

print(a \*\* b) # ✅ Output: 8 (2³)

# 🔥 Bonus: Python accepts long numbers with \_

You can use underscores \_ to make large numbers easier to read. Python ignores them.

✅ Example:

python

CopyEdit

population = 1\_000\_000\_000

print(population) # ✅ Output: 1000000000

✅ It works for floats too:

python

CopyEdit

pi = 3.141\_592\_653

print(pi) # ✅ Output: 3.141592653

✅ These are **same** as:

python

CopyEdit

population = 1000000000

# 🏁 Summary Table

| **Operator** | **Description** | **Example** | **Result** |
| --- | --- | --- | --- |
| + | Addition | 10 + 5 | 15 |
| - | Subtraction | 10 - 5 | 5 |
| \* | Multiplication | 10 \* 5 | 50 |
| / | Division (float) | 10 / 5 | 2.0 |
| // | Floor Division | 10 // 3 | 3 |
| \*\* | Exponentiation | 2 \*\* 3 | 8 |

**✅ 2. Floating-Point Numbers (float)**

* Numbers with a decimal point.
* Represent **real numbers**.
* Approximation of real values (not always exact due to binary representation).

📌 Examples:

x = 3.14

y = -0.001

z = 2.0

🔹 Type:

print(type(x)) # <class 'float'>

**✅ 3. Complex Numbers (complex)**

* Numbers with a **real part** and an **imaginary part**.
* Written as a + bj where:
  + a = real part
  + b = imaginary part
  + j = imaginary unit (like √-1)

📌 Examples:

c1 = 2 + 3j

c2 = -1j

🔹 Type:

print(type(c1)) # <class 'complex'>

**✅ 4. Boolean (bool)**

* Special subtype of integers (True = 1, False = 0).
* Used for logical operations.

📌 Examples:

flag1 = True

flag2 = False

🔹 Behavior:

print(True + 5) # 6 (because True = 1)

print(False + 10) # 10 (because False = 0)

🔹 Type:

print(type(flag1)) # <class 'bool'>

**✅ 5. Decimal Numbers (decimal.Decimal)**

* Comes from the **decimal module**.
* Used for **high-precision floating-point arithmetic** (better than float for financial applications).

📌 Example:

from decimal import Decimal

price = Decimal("19.99")

print(price + Decimal("0.01")) # 20.00

🔹 Type:

print(type(price)) # <class 'decimal.Decimal'>

**✅ 6. Fraction (fractions.Fraction)**

* Comes from the **fractions module**.
* Represents rational numbers as exact fractions.

📌 Example:

from fractions import Fraction

f = Fraction(1, 3)

print(f + Fraction(1, 6)) # 1/2

🔹 Type:

print(type(f)) # <class 'fractions.Fraction'>

**🏁 Summary Table: Python Number Types**

| **Type** | **Example** | **Description** |
| --- | --- | --- |
| **int** | 5, -100, 0 | Whole numbers |
| **float** | 3.14, -0.5 | Decimal (approximate real numbers) |
| **complex** | 2+3j, -1j | Real + imaginary numbers |
| **bool** | True, False | Logical values (True=1, False=0) |
| **decimal** | Decimal("2.5") | High-precision decimal numbers |
| **fraction** | Fraction(1, 3) | Exact rational numbers |

**🔥 Bonus: Type Conversion Examples**

python

CopyEdit

int(3.9) # 3

float(5) # 5.0

complex(2, 3) # (2+3j)

bool(0) # False

bool(42) # True

**Python Logical Operators**

Python has **3 logical operators**:

| **Operator** | **Name** | **Description** |
| --- | --- | --- |
| and | Logical AND | True if **both** conditions are True |
| or | Logical OR | True if **any** condition is True |
| not | Logical NOT | Inverts the boolean value |

**✅ 1. and (Logical AND)**

Returns True if **both conditions** are True.

python

CopyEdit

x = 5

print(x > 2 and x < 10) # ✅ Output: True

* ✅ Because x > 2 is True AND x < 10 is True.

**✅ 2. or (Logical OR)**

Returns True if **any one** condition is True.

python

CopyEdit

x = 5

print(x < 2 or x < 10) # ✅ Output: True

* ✅ Because x < 10 is True even though x < 2 is False.

**✅ 3. not (Logical NOT)**

Reverses the result: True becomes False, and False becomes True.

python

CopyEdit

x = 5

print(not (x > 2)) # ✅ Output: False

* ✅ Because x > 2 is True, but not flips it to False.

**🏁 Summary Table**

| **Operator** | **Example** | **Result** |
| --- | --- | --- |
| and | 5 > 2 and 5 < 10 | True |
| or | 5 < 2 or 5 < 10 | True |
| not | not (5 > 2) | False |