3 September

Introduction to Python

Python is a high-level programming language that is widely used because of its simplicity and readability. It is an interpreted language, which means the code is executed line by line. This makes debugging easier. Python also supports multiple programming paradigms: procedural, object-oriented, and functional. Remember: Python uses indentation instead of braces, which makes code cleaner but also strict about spacing.

Basic Input and Output

To interact with users, Python provides input() and print(). Note: input() always returns data as a string. If we need numbers, we must convert it using int() or float(). print() can display values, variables, or even formatted strings. Python supports different formatting styles, such as f-strings, which are the most modern and readable.

```
# Example: Simple I/O
name = input('Enter your name: ')
age = int(input('Enter your age: '))
print(f'Hello {name}, you are {age} years old.')

# Arithmetic with formatted output
x, y = 5, 7
print(f'The sum of {x} and {y} is {x+y}')
```

Notes — continued

Variables and Data Types

A variable is simply a name given to a value stored in memory. In Python, you don’t declare the type explicitly—it is decided at runtime. Common types include int, float, str (string), bool, and complex. For collections, we have list (ordered, changeable), tuple (ordered, unchangeable), set (unordered, unique values), and dict (key-value pairs). Tip: Use type(variable) to check the datatype.

Operators

Operators are symbols that perform operations on values. Arithmetic operators (+, -, *, /, %, //, **) are used for mathematical calculations. Relational operators (>, <, >=, <=, ==, !=) are used for comparisons and always return True or False. Logical operators (and, or, not) help in combining conditions. Example: You can check if a number is divisible by both 2 and 3 using 'if n%2==0 and n%3==0:'.

Control Flow (if / else)

Control flow statements decide the direction of execution. The if-else structure allows decision making. Indentation is critical: all statements inside the same block must be aligned. You can also use elif for multiple conditions.

```
age = int(input('Enter age: '))
nat = input('Enter nationality: ')
if age >= 18 and nat.lower() == 'india':
    print('Eligible to vote in India')
else:
    print('Not Eligible')
```

Loops

Loops help in repeating tasks. for loops are commonly used to iterate through sequences or ranges. while loops continue until a condition becomes False. Example: factorial calculation using a loop, and generating a Fibonacci sequence.

```
# Factorial
n = int(input('Enter n: '))
fact = 1
for i in range(1, n+1):
    fact *= i
print(f'Factorial of {n} = {fact}')

# Fibonacci
terms = int(input('Enter number of terms: '))
a, b = 0, 1
for _ in range(terms):
    print(a, end=' ')
    a, b = b, a+b
```