Recursion in Python

What is Recursion?

- Recursion is when a function calls itself.
- Every recursive function must have a **base condition** to stop calling itself (otherwise infinite loop/error).

Why Use Recursion?

- Solves problems that can be broken into smaller sub-problems.
- Used in mathematics (factorial, Fibonacci), searching, sorting, and tree structures.

Syntax:

```
def function_name():
   # base condition
   if condition:
      return something
   else:
      return function_name()
```

Example: Factorial using Recursion

```
def factorial(n):
     if n == 0 or n == 1:
          return 1
     else:
          return n * factorial(n - 1)

print(factorial(5))
```

Key Points:

- Always write a **base case** to stop recursion.
- Recursive functions can be less efficient than loops (they use more memory).
- But recursion makes code **shorter and closer to natural problem-solving**