

Lecture 25: Arguments and Parameters

Deep dive into function arguments.

Positional, keyword, default, and variable-length arguments.

Agenda

Parameters vs Arguments

Positional Arguments

Keyword Arguments

Default Parameters

*args (Variable Positional)

**kwargs (Variable Keyword)

Argument Ordering

Parameters vs Arguments

Parameter: A variable defined in the function declaration.

Argument: The actual value passed to the function when it is called.

Positional Arguments

Arguments passed to a function in the exact order they are defined.
The number of arguments must match the number of parameters.

Keyword Arguments

Arguments passed by explicitly stating the parameter name.
Order does not matter.

Default Parameters

Parameters can have default values. If no argument is provided, the default is used.

Default parameters must follow non-default parameters.

Mutable Default Parameters Pitfall

NEVER use mutable objects (like lists or dictionaries) as default arguments.

They are evaluated only once at definition time, leading to shared state.

Variable-Length Positional Arguments (*args)

Allows a function to accept any number of positional arguments.
Stored as a tuple inside the function.

Variable-Length Keyword Arguments (**kwargs)

Allows a function to accept any number of keyword arguments.
Stored as a dictionary inside the function.

Argument Ordering Rule

When combining argument types, they MUST follow this order:

1. Positional arguments
2. `*args`
3. Default arguments / Keyword arguments
4. `**kwargs`

Summary

Python offers immense flexibility in how functions accept data. Mastering `*args` and `**kwargs` is essential for writing generic, reusable functions.

Next Lecture

Next, we will explore Variable Scope: the difference between global and local variables.