Lesson 5: Functions

◇ Introduction

Functions are reusable blocks of code that perform specific tasks. They help you:

- Avoid repetition
- Make your code cleaner and easier to manage
- Organize logic into smaller pieces

In this lesson, you'll learn:

- How to define and call functions
- How to use parameters and return values
- Built-in vs. user-defined functions

◇ 1. Defining and Calling Functions

- A basic function in Python looks like this:

```
def greet():
    print("Hello, world!")

greet()  # Calling the function
```

- ✅ Use functions when you need to do the same task multiple times or logically group operations.

◇ 2. Parameters and Arguments

- You can pass values to functions using parameters.

```
def greet(name):
    print(f"Hello, {name}!")

greet("Sprinter")  # Output: Hello, Sprinter!
```

- You can even pass multiple arguments:

◇ 3. Return Statement

- Functions can send back results using return.

```
def square(number):
    return number ** 2
print(square(4))  # Output: 16
```

- If no return is used, the function returns None by default.

◇ 4. Default Parameters & Keyword Arguments

- Default values:

```
def greet(name="friend"):
    print(f"Hello, {name}!")

greet()          # Output: Hello, friend!
greet("Mona")    # Output: Hello, Mona!
```

- Keyword arguments:

```
def info(name, age):
    print(f"{name} is {age} years old.")

info(age=25, name="Ziad")  # Output: Ziad is 25 years old.
```

◇ 5. Built-in vs. User-Defined Functions

- Built-in: print(), len(), sum(), range()
- User-defined: Functions you create with def

◇ Mini Example: Calculator Function

```
def calculator(a, b, operation):
    if operation == "add":
        return a + b
    elif operation == "subtract":
        return a - b
    elif operation == "multiply":
        return a * b
    elif operation == "divide":
        return a / b
    else:
        return "Invalid operation"

print(calculator(10, 2, "multiply"))  # Output: 20
```

◇ Mini Challenge

Create a function that:

- Accepts a list of numbers.
- Returns the average.

◇ Outro

Nice work! 🎉 Today, you learned:

- How to define and call functions
- How to use parameters and return values
- The difference between built-in and user-defined functions

Functions make your code modular, reusable, and clean.
