**Functions in Python**

A **function** is a reusable block of code designed to perform a specific task. It may:

* Take **inputs** (parameters/arguments)
* Return **outputs**
* Execute code without returning anything

Function Syntax:

def function\_name(parameters):

"""Docstring explaining what the function does"""

# function body

return result

**1. Functions with Arguments**

**Meaning:**  
The function takes input values when called.

**Purpose:**  
To make the function work with **different data** every time it is called.

**Example:**

def greet(name):

print(f"Hello, {name}!")

greet("Sowmya")

**Output:**

Hello, Sowmya!

**2. Functions without Arguments**

**Meaning:**  
The function does **not** take inputs when called.

**Purpose:**  
Useful when the task is **fixed** or uses **predefined values**.

**Example:**

def greet():

print("Hello, welcome to Python programming!")

greet()

**Output:**

Hello, welcome to Python programming!

**3. Parameters vs Arguments**

* **Parameter** → Name in the function definition (placeholder)
* **Argument** → Actual value passed during function call

Example:

def add(x, y): # x, y → parameters

return x + y

add(5, 3) # 5, 3 → arguments

**4. Types of Arguments in Python**

**a) Positional Arguments -** Values are passed **in the same order** as parameters.

def details(name, age):

print(name, age)

details("sowmya", 20) # Order matters

**b) Keyword Arguments -** Arguments are passed with **parameter names**, so order doesn’t matter.

details(age=20, name="sowmya")

**c) Default Arguments -** Parameters have default values that are used if no value is passed.

def greet(name="Guest"):

print(f"Hello, {name}!")

greet() # Uses default value

greet("Sowmya")

**d) Variable-length Arguments -**

1. **\*args** → Accepts multiple positional arguments as a tuple.

def add(\*numbers):

print(sum(numbers))

add(1, 2, 3, 4)

1. **\*\*kwargs** → Accepts multiple keyword arguments as a dictionary.

def show\_info(\*\*details):

print(details)

show\_info(name="sowmya", age=20)

**5. All Types of Parameters**

Python officially classifies parameters into **5 kinds**:

|  |  |  |
| --- | --- | --- |
| **Parameter Type** | **Example** | **Notes** |
| Positional-only | def func(x, /): | Must be passed by position |
| Positional-or-keyword | def func(x): | Can be passed by position or keyword |
| Keyword-only | def func(\*, x): | Must be passed by keyword |
| Variable positional | def func(\*args): | Collects extra positional args |
| Variable keyword | def func(\*\*kwargs): | Collects extra keyword args |