

# Python for Beginners: Functions in Python

---

Connect with me: [Youtube](#) | [LinkedIn](#) | [WhatsApp Channel](#) | [Web](#) | [Facebook](#) | [Twitter](#)

- [Download PDF](#)
- To access the updated lecture notes, please click on the following link:  
<https://yasirbhutta.github.io/python/docs/functions.html>

## What is a Function?

In Python, a function is a block of code designed to perform a specific task. It's reusable, which means you can call it multiple times in your program. This helps to organize your code, make it more readable, and avoid repetition.

## How to Write a Function

To define a function, you use the `def` keyword followed by the function name, parentheses for parameters, and a colon. The code block that defines the function is indented.

```
def function_name(parameters):  
    # Function body  
    # Code to be executed
```

### Example:

```
def greet(name):  
    print("Hello,", name + "!")  
  
# Calling the function  
greet("Ahmad") # Output: Hello, Ahmad!
```

### Explanation:

- `def greet(name):` defines a function named `greet` that takes one parameter, `name`.
- `print("Hello,", name + "!")` is the function body, which prints a greeting message using the provided name.
- `greet("Ahmad")` calls the function with the argument "Ahmad".

### Key Points:

- **Parameters:** These are variables passed to the function when it's called.
- **Return Value:** A function can optionally return a value using the `return` statement.
- **Docstrings:** It's good practice to include a docstring (a string that explains the function's purpose) after the function definition.

## More Examples:

### Function with a Return Value:

```
def add(x, y):  
    return x + y  
  
result = add(3, 5)  
print(result) # Output: 8
```

### Function with Default Parameters:

```
def greet(name="World"):  
    print("Hello,", name + "!")  
  
greet() # Output: Hello, World!  
greet("Alice") # Output: Hello, Alice!
```

### Function with Variable-length Arguments:

```
def multiply(*numbers):  
    result = 1  
    for number in numbers:  
        result *= number  
    return result  
  
print(multiply(2, 3, 4)) # Output: 24
```

## Practice Exercises:

- Write a function to calculate the factorial of a number.
- Write a function to check if a number is prime.
- Write a function to find the largest number in a list.
- [Video: How to Add Two Numbers and Print the Result](#)
- [Video: How to Find the Maximum Value in a List of Numbers](#)
- [Video: How to Write a Python Function to Find the Length of a Word](#)
- [Video: Learn How to Use Default Parameters in Function Definition](#)
- [Video: Understanding \\*args in Functions - How to Add Any Number of Arguments with \\*args](#)
- [Video: How to Use \\*args in Python Functions](#)

- [Video: How to use \\*\\*kwargs in Python](#)

## Parameters and Arguments

**Parameters** and **arguments** are often used interchangeably, but they have distinct meanings in the context of functions.

### Parameters

- **Definition:** Variables declared in a function's definition.
- **Purpose:** Act as placeholders for values that will be passed to the function when it's called.
- **Location:** Inside the function's parentheses.

### Arguments

- **Definition:** Actual values passed to a function when it's called.
- **Purpose:** Provide data for the function to work with.
- **Location:** Inside the function call parentheses.

### Example:

```
def greet(name): # 'name' is a parameter
    print("Hello,", name + "!")

greet("Alice") # "Alice" is an argument
```

In this example:

- `name` is a parameter in the function `greet`.
- `"Alice"` is an argument passed to the function when it's called.

### To summarize:

- Parameters are defined *before* the function is called.
- Arguments are provided *when* the function is called.

### Think of it like this:

- A parameter is like an empty box that expects a value.
- An argument is the value you put into the box.

### See also:

- [Python Quiz - Functions](#)

True/False (Mark T for True and F for False)

Multiple Choice (Select the best answer)

What is the output of the following code? [Python Quiz #2]

```
def foo(x):  
    if x == 1:  
        return 1  
    else:  
        return x * foo(x - 1)  
  
print(foo(5))
```

- A) 5
- B) 15
- C) 120
- D) None

**Watch this video for answer:** <https://www.youtube.com/shorts/k50czTu7vao>

What is the output of the following expression? [Python Quiz #13]

```
def add(a,b,*parm):  
    total = 0  
    print(a+b)  
    for n in parm:  
        total += n  
    return total  
  
print(add(1, 2))
```

- A) 3 0
- B) 3
- C) 0
- D) Error

**Watch this video for answer:** <https://youtube.com/shorts/k4KVCxU5oMg>

What is the output of the following code? [Python Quiz #14]

```
def add(*args):  
    print(type(args))  
  
add(1, 2,8,9)
```

- A) set
- B) tuple
- C) list
- D) None

**Watch this video for answer:** <https://youtube.com/shorts/VQT4Cllpf9M>

**What is the output of the following code? [Python Quiz #30]****Code:**

```
def calculate_sum(n):  
    if n == 0:  
        return 0  
    else:  
        return n + calculate_sum(n-1)  
  
print(calculate_sum(4))
```

**Options:**

- A) 4
- B) 6
- C) 10
- D) 15

**Watch the video for the answer:** <https://youtube.com/shorts/LQEfGgJYIT4?si=MDvSvVHiBc6hCJ0W>

#23 What is the main purpose of a function in Python?

#python #pythonpoll #MCQsTest #yasirbhutta

a) To group a set of related code into a single unit b) To create a new type of data c) To write a program in a single line d) To change the value of global variables Answer: a) To group a set of related code into a single unit

#22 What is the purpose of the return statement in a function in Python?

#python #pythonpoll #MCQsTest #yasirbhutta

a) To print the output of the function b) To exit the function and return a value c) To execute the function without returning anything d) To stop the function and start a new one

Answer: b) To exit the function and return a value

How can you pass parameters to a function in Python? a) By listing the parameters after the function name, separated by commas b) By enclosing the parameters in square brackets c) By using the global keyword d) By using the input() function Answer: a) By listing the parameters after the function name, separated by commas

#19 What is the output of the following code?

```
def myfunction(val): return val % 4 == 0 print(myfunction (13) or myfunction (8))
```

Watch the Video Tutorial for the Answer: <https://youtu.be/laKpsLlq60I>

#python #pythonpoll #MCQsTest #yasirbhutta

0 13 False True 3.5

What is the correct way to define a function in Python? a) function my\_function(): b) def my\_function(): c) define my\_function(): d) my\_function() {

## Exercises

- Write a function `sum3(num1, num3, num3)` that takes three numbers as input and returns the sum.
- Write a function `SumNum(num1)` that takes a number as input and returns the sum of numbers from 1 to that number (num1).
- Write a function `sumSquares(x)` that takes a vector of numbers as input and returns the sum of their squares.
- Write a function `isEven(x)` that takes a number as input and returns true if it is even, and false otherwise.
- Write a program with three functions:
  1. `isEven(n)`: This function takes an integer `n` as input and returns `True` if `n` is even and `False` otherwise. You can use the modulo operator (%) to check for evenness.
  2. `printTable(n)`: This function takes an integer `n` as input and prints its multiplication table. The table should show the product of `n` with each number from 1 to 10, formatted like `n * i = n * i`, where `i` is the current number in the loop.
  3. `main`: The main program should:
    - Prompt the user to enter an integer.
    - Use the `isEven(n)` function to check if the entered number is even.
    - If the number is even, call the `printTable(n)` function to print its multiplication table.
    - If the number is odd, print a message indicating the number is odd and not eligible for printing a table.

### Example output:

```
Enter an integer: 4
4 is even! Here's its multiplication table:
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
...
4 * 10 = 40
```

- Write a function `avgPositive(data)` that takes a list of numbers as input and returns the average of all positive numbers in the list.

## Review Questions

## References and Bibliography

Which of the following will cause a syntax error due to incorrect indentation in Python?

A)

```
print("Hello World!")
```

B)

```
def my_function():  
    print("Hello World!")
```

C)

```
if x == 10:  
    print("x is 10")
```

D)

```
x = 10
```

*Answer: B*