**Difference between Methods and Functions:**

**✅ Functions**

* **Defined using** def (or lambda).
* **Standalone** blocks of code.
* Can be called independently.

**Example:**

def greet(name):

return f"Hello, {name}!"

print(greet("Maya"))

# Output: Hello, Maya!

**✅ Methods**

* Also defined using def, **but inside a class**.
* Always associated with an object (instance of a class).
* The first parameter is usually self, which refers to the instance.

**Example:**

class Greeter:

def greet(self, name):

return f"Hello, {name}!"

g = Greeter()

print(g.greet("Maya")) # Output: Hello, Maya!

*For Linked in post:*

🔍 **Functions vs. Methods in Python — What’s the Difference?**

In Python, both **functions** and **methods** let us write reusable blocks of code, but there’s a key distinction:

✅ **Functions**  
– Defined using def or lambda  
– Standalone (not part of a class)  
– Can be called on their own

Example (Python code):

def greet(name):

return f"Hello, {name}!"

✅ **Methods**  
– Also defined with def, **but inside a class**  
– Called on an object (an instance of the class)  
– Always take self as the first parameter

Example (Python code):

class Greeter:

def greet(self, name):

return f"Hello, {name}!"

📌 **In short:**  
A **method** is just a **function that belongs to an object**.

Hope this helps someone learning Python! 🐍  
#Python #Coding #Functions #OOP #ProgrammingTips