

Day 10



Functions in Python:

What is a Function?

A function is a block of reusable code that performs a specific task. Instead of writing the same code again and again, we put it inside a function and reuse it.

Why do we need Functions?

- To avoid repetition
- To make code clean and organized
- To reuse code
- To make programs easy to understand

Types of Functions:

1. **Built-in functions:** Examples: `print()`, `len()`, `input()`
2. **User-defined functions:** Functions created by the programmer

Example: Geometric mean without using the functions.

A screenshot of a Jupyter Notebook terminal window. At the top, there is a code cell containing the following Python code:

```
1 a = 9
2 b = 8
3 gmean = (a*b)/(a+b)
4 print(gmean)
```

Below the code cell, there is a navigation bar with tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is underlined), and PORTFOLIO. Under the TERMINAL tab, there is a command-line interface showing the output of the code execution:

```
PS E:\Python\Day1-10\Day10> python main.py
4.235294117647059
```

Creating a Function (`def`): Syntax

```
def function_name():
    # code
```

Example: calculating the geometric mean using the functions in python.

```
6  def gmean(a, b):
7      mean = (a * b) / (a + b)
8      print(mean)
9
10 a = 9
11 b = 8
12 gmean(a,b)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\Python\Day1-10\Day10> python main.py
4.235294117647059
```

Example: code to compare the two numbers without using functions.

```
14 a = 8
15 b = 9
16 if(a>b):
17     print("First number is greater")
18 else:
19     print("Second number is greater")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
● PS E:\Python\Day1-10\Day10> python main.py
Second number is greater
```

Example: code to compare the numbers using functions.

```
21 def isGreater(a, b):
22     if(a > b):
23         print("First number is greater")
24     else:
25         print("Second number is greater")
26 a = 8
27 b = 9
28 isGreater(a, b)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
● PS E:\Python\Day1-10\Day10> python main.py
Second number is greater
```

Example:

```
21 def isGreater(a, b):  
22     if(a > b):  
23         print("First number is greater")  
24     else:  
25         print("Second number is greater")  
26 c = 8  
27 d = 9  
28 isGreater(c,d)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

• PS E:\Python\Day1-10\Day10> **python main.py**
Second number is greater

Example: we uses “pass” for things we don’t want to write as of now.

```
30 def isLove():  
31     pass
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

• PS E:\Python\Day1-10\Day10> **python main.py**
❖ PS E:\Python\Day1-10\Day10>

Example:

```
33 def greet():  
34     print("Hello, Welcome to Python")  
35 greet()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

• PS E:\Python\Day1-10\Day10> **python main.py**
Hello, Welcome to Python

--The End--