



**Data Science
Bootcamp**

Hyperiondev

Beginner Programming with Functions – Defining Your Own Functions

WELCOME TO THE EVENT HANDLING TASK

Your Lecturer for This Session



Christiaan Joubert

Objectives

- Understand what functions are and how to define your own
- Understand how to call functions
- Understand the scope of functions

What is a Function?

- Reusable and Organised block of code.
- Sometimes called a 'method'.
- Similar to functions in maths – $f(x)$ takes input x and produces some output.
- Useful for abstraction.
 - For example, "make a cup of tea" vs "boil water, add tea bag, add sugar, add milk, stir".

Functions in Python

- Python comes bundled with in-built functions.
- Examples:
 - **print(string)** – prints string to console.
 - **input(string)** – prints string to console, then reads input as string.
 - **len(array)** – finds the length of an array.
 - **int(data)** – converts the value to an integer.

Why Functions?

- **Reusable code** – Sometimes you need to do the same task over and over again.
- **Error checking/validation** – Makes this easier, as you can define all rules in one place.
- **Divide code up into manageable chunks** – Makes code easier to understand.
- **More rapid application development** – The same functionality doesn't need to be defined again.
- **Easier maintenance** – Code only needs to be changed in one place.

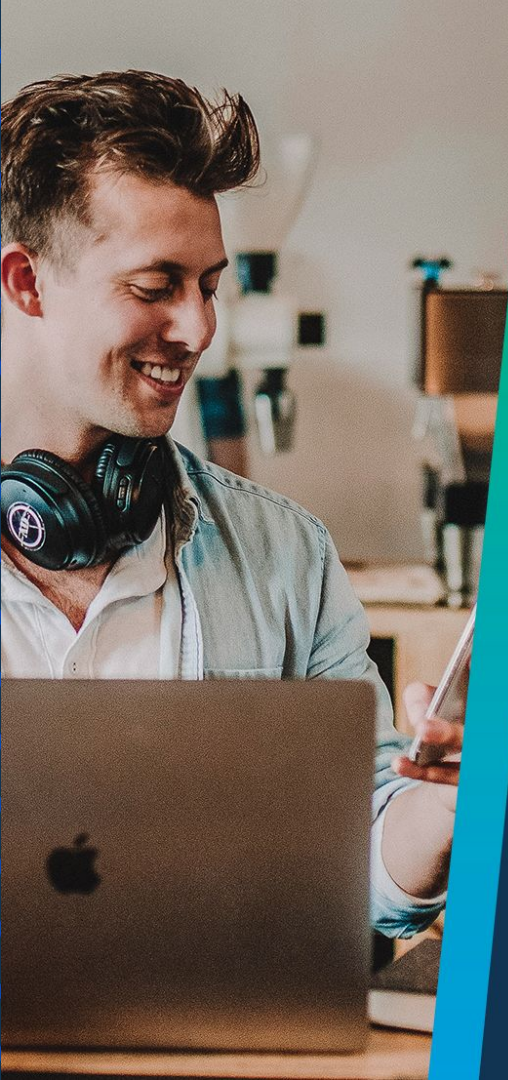
Some Important Terms

- **Function** – A block of code that performs an action.
- **Method** – A function defined on or owned by an object. Not quite the same thing as a function but very similar for our purposes at this stage of learning.
- **Parameters** – The defined input of a function.
- **Arguments** – The values passed to parameters.

Hyperiondev

Q & A Section

Please use this time to ask any questions relating to the topic, should you have any.



Hyperiondev

Thank You for Joining Us