

3.9-3.12: Hybrid Session 2

This document has been updated to reflect the materials which were covered and shared within the session.  Anything new is highlighted in purple.

In this topic, we will be focussing on the following learning outcome for this week:

* Summarise the difference between an argument and a parameter
* Apply functions to problems with varying quantities of parameters

You will have the following learning opportunities:

* To discuss the differences between parameters and arguments
* To implement and use functions with varying numbers of parameters
* To understand the purpose and implementation of default parameters

Prefer to complete this activity offline? - download the following documents:

* Full PowerPoint Slides: [Week 3 Hybrid Session 2 Full Slides.pptx](https://winchester.instructure.com/files/1671749/download?wrap=1)
* Walkthrough Documents:
  + [3.9 Parameters Walkthrough.docx](https://winchester.instructure.com/files/1671744/download?wrap=1)
  + [3.10 Default Parameters Walkthrough.docx](https://winchester.instructure.com/files/1671745/download?wrap=1)
  + [3.11 Local Variables Walkthrough.docx](https://winchester.instructure.com/files/1671738/download?wrap=1)
  + [3.11 Global Variables Walkthrough.docx](https://winchester.instructure.com/files/1671747/download?wrap=1)



During this online session, you will be engaging in acquisition, discussion, and practice learning activities.

# 3.9: Parameters vs Arguments

During the pre-session content, you were asked to find definitions of Parameters and Arguments. In this part of the third week, we will be revisiting these definitions and highlighting the difference between an argument and a parameter. Once we have completed this, we will be looking at how we can make use of parameters within our functions. Finally, we will look at the errors which are produced when insufficient arguments are passed to a function.

We started this section by revisiting our definitions of parameters and arguments with a view to establishing the differences between the two - as they are often incorrectly used interchangeably.  We also looked at how we can incorporate parameters within our functions and call them with arguments.

# 3.10: Default Parameters

In this section we will be focussing on the concept of default parameters, and how they can help us to use a function to carry out a few different operations. In addition, they can be used to help us to overcome the error produced in the previous section. We will demonstrate the implementation of default parameters within Python as part of this section.

As an extension to parameters, in this section we talked about the concept of default parameters and how they can be useful when trying to develop multiple possible implementations of a function.  We adapted our multiply\_four\_numbers() function to have default parameters allowing us to multiply 2, 3, and 4 numbers together.

# 3.11: Scope, Local Variables, and Global Variables

In the last section of new content for this week, we will be looking at scope – particularly in relation to variables. We will demonstrate the implications of variable scope within Python and how we can utilise the global variable within our programs.

In this section, we discussed the concept of scope and defined both local variables and global variables. During the demonstration, we encountered a situation where we were trying to access a local variable outside of its scope. Finally, we looked at the read-only access to global variables within functions and investigated how we can overcome the issue.

# 3.12: Week Summary and Overview

In this final part of this session, we will be reiterating the topics we have covered during the pre-session materials, and the online session. We will talk about what you need to do next in terms of finishing the content for this week - which includes the practice questions and stretch tasks. Finally, we will talk about the topics we will be covering next week in the pre-session content and timetabled session.