

4.9-4.12: Hybrid Session 2

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

* Full PowerPoint Slides: [Week 4 Hybrid Session 2 Full Slides.pptx](https://winchester.instructure.com/files/1677437/download?wrap=1)
* Walkthroughs:
  + [4.9 If-Else-if-Else Statements.docx](https://winchester.instructure.com/files/1677441/download?wrap=1)
  + [4.11 Nested If Statements.docx](https://winchester.instructure.com/files/1677443/download?wrap=1)
  + [4.11 Using Boolean Operators.docx](https://winchester.instructure.com/files/1677446/download?wrap=1)

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

* Solve problems requiring selection-based control structures within Python

You will have the following learning opportunities:

* To implement the main types of control structure within Python
* To execute functions which require nested selection control structures
* To understand the pass keyword in Python



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

# 4.9: If-elif-else Statement

In this part of this week’s content, we will look at further extending our if statements to take into account more decisions which need to be made. We will be continuing to work with our example to further enhance it, using the if-elif-else statement. We will discuss the concept of elif in Python and how it is represented elsewhere.

In this element, we continued to use our example of even/odd numbers and extended it to include information about whether the number is divisible by 3.  To do this, we will start by looking at the if-elif-else statement.

# 4.10: The pass Keyword

In Python, the pass keyword represents a placeholder within our programs. In this section, we will discuss the pass keyword and how it can benefit us during the development process – especially when we are not quite ready to code something but need to include something within the construct.

In this section, we introduced the pass keyword and its use as a placeholder to support software development within Python.  We also discussed how the pass keyword is beneficial when we are trying to implement structures which require code where we don't want to include it.

# 4.11: Nesting If Statements

We will be continuing to develop our example which has been used this week to increase the complexity of the decisions being made. For this section we will look at two things: nesting and using Boolean operations. We will start by understanding the concept of nesting control structures – in this case, we will be nesting selection structures (we will continue to build on this next week). Secondly, we will look at how we can make our nested if statements more readable through using Boolean operators instead.

The final part of this week was focussed on how we can use nesting to increase the complexity of our control structures.  During this we looked at nesting an if statement inside another if statement.  We highlighted that this approach was not always readable, so we progressed to using boolean operators instead to improve the readability of our control structures.

# 4.12: 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.