

Unit 11 – Reflection:

During Week 11, I started to explore the lecture cast with the topic of *Pointers, references, memory management, and design patterns*, aimed to emphasise the fact that the lack of pointer use supports ease of Python development, understand the role played by design patterns in the development of object-oriented code; distinguish between the main categories of design patterns, and identify examples belonging to each; and appreciate the role to be played through sustainable code design and development, which provided as well valuable insights into the efficiency and organisation of code in Python and the benefits of using design patterns and writing sustainable code.

This kind of learning is so valuable for my understanding as long I progress in learning Python, which makes me spend much time structuring my codes and testing and experiencing different approaches to make my code run as expected.

I have also explored design patterns and the concept of design patterns with lots of research and reading. I focused on becoming more familiar because design patterns are related to UML, which I struggled with during this module. I have learned that it provides a structured approach to organising code and promoting modularity, flexibility, and reusability. I explored different categories of design patterns, including creational, structural, and behavioural patterns.

Additionally, I have been playing with a new tool my tutor shared, a lightweight UML designer for drawing static class or domain diagrams, called UMLBoard, which I am excited about in practice and testing new approaches.

Also, my main focus for this week is on the next assignment on System Implementation, which I am nervous about and afraid to do something differently than expected. Still, I am also excited to apply all my learnings and make it work.