SIT232

Object oriented Programming

Learning Summary Report

JASVEENA
224001588

Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

	Pass (P)	Credit (C)	Distinction (D)	High Distinction (HD)
Self-Assessment	✓			

Minimum Pass Checklist

	Included
Learning Summary Report	✓
Pass tasks completed (submitted and discussed)	✓

Declaration

I declare that this portfolio is my individual work. I have not copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person.

JASVEENA

Portfolio Overview

This portfolio showcases the work that reflects my achievement at a PASS level in the unit *Object-Oriented Development*. I successfully completed PASS-level tasks and applied key object-oriented programming principles such as encapsulation, inheritance, polymorphism, abstraction, and lambda expressions. The tasks were submitted punctually and discussed with my tutor during a face-to-face interview. These completed tasks demonstrate my ability to understand and apply object-oriented programming techniques at a satisfactory level. Through the course, I gained practical experience in problem-solving, particularly in projects like developing a banking system, which allowed me to apply theoretical concepts to real-world challenges. My individual work was complemented by discussions with my peers and tutor, which further strengthened my understanding of these programming techniques.

Reflection

The most important things I learnt:

One of the most valuable lessons from this unit was mastering the creation of code structure, which is central to object-oriented development. I learned to

decompose complex problems into smaller, manageable pieces by leveraging objects and classes. This approach taught me the importance of designing reusable software components, a key mindset for system design. I also learned how to use abstraction to simplify complex problems by creating abstract classes that hide unnecessary details while providing a functional framework for future development.

The things that helped me most were:

The resources in this unit were incredibly helpful. While the online lectures provided a foundation, it was the in-class discussions with my tutor that truly solidified my understanding, especially on more challenging topics like inheritance and error handling. My tutor's clarifications and examples showed me how these concepts are applied in real-world programming scenarios.

I found the following topics particularly challenging:

However, I did face challenges, particularly with the concept of inheritance. Initially, I found it difficult to determine when to use inheritance without overcomplicating the design. With practice and feedback, I became more confident in applying inheritance effectively alongside other object-oriented techniques.

I found the following topics particularly interesting:

I found polymorphism to be particularly engaging. Learning how it enhances software flexibility and adaptability was a highlight, especially when applied to manage diverse types of users or transactions in the banking system project. This reinforced the importance of building extensible software that can adapt to future changes without needing extensive rewrites.

I feel I learnt these topics, concepts, and/or tools well:

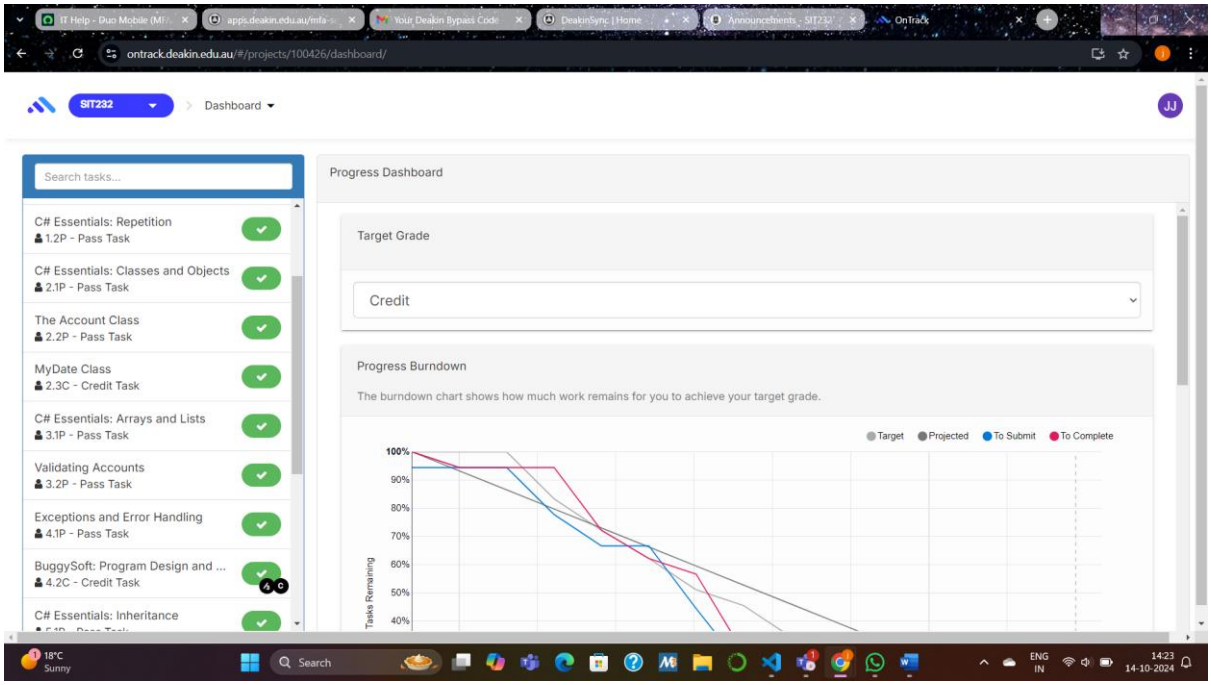
Throughout this unit, I gained a solid understanding of several important object-oriented programming concepts and tools. I became proficient in encapsulation, which allowed me to protect the internal states of objects and control access to class members. Polymorphism was another concept I grasped well, enabling me to write flexible and reusable code through method overriding and the use of interfaces. Additionally, abstraction helped me focus on essential features by simplifying complex systems, especially through the use of abstract classes and interfaces.

I still need to work on the following areas:

I think I understand important ideas like abstraction, polymorphism, and encapsulation well, there are certain things I still need to work on. Moving forward, I want to concentrate on improving my error handling strategies and learning more complex polymorphism concepts, particularly those that involve several levels of inheritance.

My progress in this unit was …:

My progress in this unit was steady and focused. I made a conscious effort to build a strong understanding of object-oriented principles from the start, completing some tasks well before their deadlines. If I were to take this unit again, I would aim for a higher grade by exploring more advanced topics beyond the minimum requirements, experimenting with complex design patterns, and delving deeper into object-oriented architecture.



If I did this unit again, I would do the following things differently:

If given the opportunity to retake this unit, I would push myself to explore beyond the core requirements and aim for a higher grade. I would dive deeper into more advanced topics, such as design patterns and data structures, which could significantly enhance my understanding of object-oriented programming. I'd also allocate more time to experimenting with complex

programming scenarios, allowing me to better grasp the intricacies of object-oriented design and architecture.

For future Career:

Finally, this unit has been highly relevant to my broader studies in computer science. The knowledge gained in building object-oriented systems will be invaluable in developing secure, reliable applications in industries such as finance. The hands-on experience with the banking system project has given me a deeper understanding of how to structure scalable and secure applications for the future.