WeThinkCode_ Student Progress Report

Student Name:	**Kganya Maleka**
Student Username:	**kmaleka023**
Cohort:	**2023**
Campus:	**Johannesburg**
Date:	**23 July 2024**

Performance Scale Descriptors

Performance Scale Descriptors					
Level	**5**	**4**	**3**	**2**	**1**
Outcomes	Exceeds Expectations	Above Expectations	Meets Expectations	Below Expectations	Expectations not Met

Interpreting the Outcomes

Exceeds Expectations: student is able to submit project requirements well in advance before the deadline. Grading was passed/correct on the first attempt.

Above Expectations: student attempted grading twice before the deadline, but passed on the second attempt before the deadline.

Meets Expectations: student is able to submit projects on time and pass at the required level.

Below Expectations: student is below expectations, attempted grading and submitted after the deadline, and passed.

Expectations not Met: student attempted to submit after the deadline passed/failed and this indicates that major development is required. Students should make use of the interventions available to improve performance.

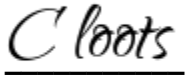
Performance Report

Module 2 - Object-Oriented Programming This module serves as an introduction to software design with a focus on Object-Oriented Programming (OOP). Java is introduced as the primary programming language, building upon the foundational programming principles initially covered in Python. This transition further reinforces and deepens the understanding of core programming concepts.		
Topic	**Learning Outcomes**	**Level**
Encapsulation	Understand that encapsulation in OOP involves bundling data and methods in a class and restricting direct access to some components.	**5**
Inheritance: (Toy-Robot Walk-through)	Explain that inheritance in OOP is a mechanism where a new class inherits properties and behaviours (methods) from an existing class, allowing for code reuse and extension.	**5**
Polymorphism: (Toy-Robot Enhanced)	Grasp that polymorphism in OOP allows different classes to be treated as instances of a common superclass, enabling a single function to work on various types of objects.	**5**
Robot Worlds	A Team Project used to build a client/server Robot World. A client/server architecture consists of a server program running on a computer on a network that is listening for incoming requests.	**4**

Assessment Scores		
Assessment 1: Coding	**Fundamentals of Java**	**86%**
Assessment 2: Formative 1	**Encapsulation**	**83%**
Assessment 3: Formative 2	**Composition & Polymorphism**	**47%**
Summative Assessment	**Fundamentals of Java and OOP**	**75%**

Final Score	
Final Score	**89%**

I hereby certify that these are the official results for Kganya Maleka for Semester 2.



Corban Loots
Performance Lead, Johannesburg
corban@wethinkcode.co.za