UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA
Faculty of Engineering, Built Environment and Information Technology

School of Information Technology

Department of Computer Science

COS326 – Database Systems

Practical 1 2025

Release Date:	29 August 2025
Submission Date and Time:	05 August 2025 @ 23:59Hrs
Lecturer:	Mr S.M Makura
Total:	50 Marks

A. Objectives

Demonstrate that you can use ObjectDB for Java to insert, modify, retrieve and delete objects from an ObjectDB database and that you understand ObjectDB.

In this practical you will have to write a Java program that:

1. Implements Java classes and makes them persistent by storing them in a ObjectDB database.
2. Executes queries on the ObjectDB database.

B. Submission Procedure:

1. Create a folder as follows:
   The folder name should be ***uXXXXXXXX_Surname_Initials***(*XXXXXXXX is your student number) e.g u12345678_Smith_JN* and should contain the project files (your.jar, .java and other necessary files needed for your project to run).
2. Create a zip file of the folder and upload it to ClickUP via the submission link provided for practical 1.

NO LATE submissions will be accepted after the submission date and time has lapsed**. Do not** wait till the last minute to submit and start giving excuses that you faced technical challenges when you tried to submit.

Question 1: ObjectDB Database Programming

Scenario

> You were recently hired as a graduate developer at Codealot pvt ltd. For your first task, they would like to test your proficiency in using OOBMSs. They would like you to develop a Java application that can be used to enter car details. The application must allow the user to enter the car details through a Java Swing or JavaFX GUI and perform the necessary CRUD operations through that GUI.
>
> Source: Makura S.M (2025)

Create a Java application using NetBeans or any Java IDE you are comfortable with. The Java application must interact with an ObjectDB database and must have the following capabilities:

(a) Must consist of a Java Swing or JavaFX GUI that must allow the user to enter the following details:

i. Car Registration Number
ii. Car Make
iii. Car Model
iv. Car year of manufacture
v. Top speed

(b) The Java application must allow the user to perform CRUD operations on the ObjectDB database. You will also need a main class where you will implement all the methods necessary to perform the CRUD operations. The CRUD operations expected are:

(i) **Create/store** Car objects with the details provided in (a) and save it in the

ObjectDB database through a “Save” button. Use an appropriate Java Swing/JavaFX GUI control to display a message to confirm if the operation has been done successfully. You will need to create an Entity class called **Car** with all the necessary variables, getters and setters.

(ii) **Read** one or all car details from the ObjectDB database and display it via the GUI interface, i.e have a search button to search the car details based on the car registration number then use an appropriate JavaSwing/Java FX GUI control to display the results.

(iii) **Update** specified car details i.e have an update button which when clicked, will update any of the car details specified in part (a). Please note, you need to update car details for already existing car details in the ObjectDB database. Use an appropriate Java Swing/JavaFX GUI control to display a message to confirm if the operation has been done successfully.

(iv) **Delete** the specified car details based on the car registration number i.e have a delete button which when clicked, will delete all the car details of the car registration number specified. Use an appropriate Java Swing/JavaFX GUI control to display a message to confirm if the operation has been done successfully.

(v) **Calculate Average**- have a "Calculate Average" button which when clicked will display the average speed of all the cars that are stored in the ObjectDB database. Use and appropriate Java Swing/JavaFX GUI control to display the average speed.

Ensure that your application handles any basic exceptions for example the application must display an error message via an appropriate Java Swing/JavaFX GUI if the user clicks the Calculate Average button without having any car details in the database or must display error messages via an appropriate Java Swing/JavaFX GUI if a user

clicks the any buttons without entering any car details. Full marks will be awarded for a fully functional Java application based on specification.

[Total Marks: 50]