School of Information Technology
Department of Computer Science

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

COS326 – Database Systems
Practical 1 2024

Release Date:	01 August 2024
Submission Date and Time:	08 August 2024 @ 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***(*XXXXXXXX is your student number)* 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 software engineer at FinGuard Solutions. For your first task, they would like to test your proficiency in developing applications to manage financial transactions. They would like you to develop a Java application that can be used to enter and manage financial transaction details. The application must allow the user to enter transaction details through a Java Swing or JavaFX GUI and perform the necessary CRUD operations through that GUI.
>
> Source: Makura S.M (2024)

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 allows the user to enter the following details:

i. Transaction ID
ii. Transaction Date
iii. Amount
iv. Sender Account Number
v. Receiver Account Number
vi. Transaction Type (e.g., Deposit, Withdrawal, Transfer)

(b) The Java application must allow the user to perform CRUD operations on the 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 Transaction**: Allow the user to enter transaction details and save them in the 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 `Transaction` with all the necessary variables, getters, and setters.

(ii) **Read Transaction Details**: Allow the user to read one or all transaction details from the database and display them via the GUI. Have a search button to search transaction details based on the transaction ID and use an appropriate Java Swing/JavaFX GUI control to display the results.

(iii) **Update Transaction Details**: Provide an update button that, when clicked, will update any of the transaction details specified in part (a). Note that you need to update transaction details for already existing transactions in the database. Use an appropriate Java Swing/JavaFX GUI control to display a message to confirm if the operation has been done successfully.

(iv) **Delete Transaction Details**: Provide a delete button that, when clicked, will delete all the details of the specified transaction based on the transaction ID. Use an appropriate Java Swing/JavaFX GUI control to display a message to confirm if the operation has been done successfully.

(v) **Calculate Total Transactions**: Implement a "Calculate Total" button that, when clicked, will calculate the total amount of all transactions stored in the database. Use an appropriate Java Swing/JavaFX GUI control to display the total amount.

Ensure that your application handles any basic exceptions. For instance, the application must display an error message via an appropriate Java Swing/JavaFX GUI if the user clicks the Calculate Total button without having any transaction details in the database or must display error messages via an appropriate Java Swing/JavaFX GUI if a user clicks any buttons without entering the required details. Full marks will be awarded for a fully functional Java application based on the specification.

[Total Marks: 50]