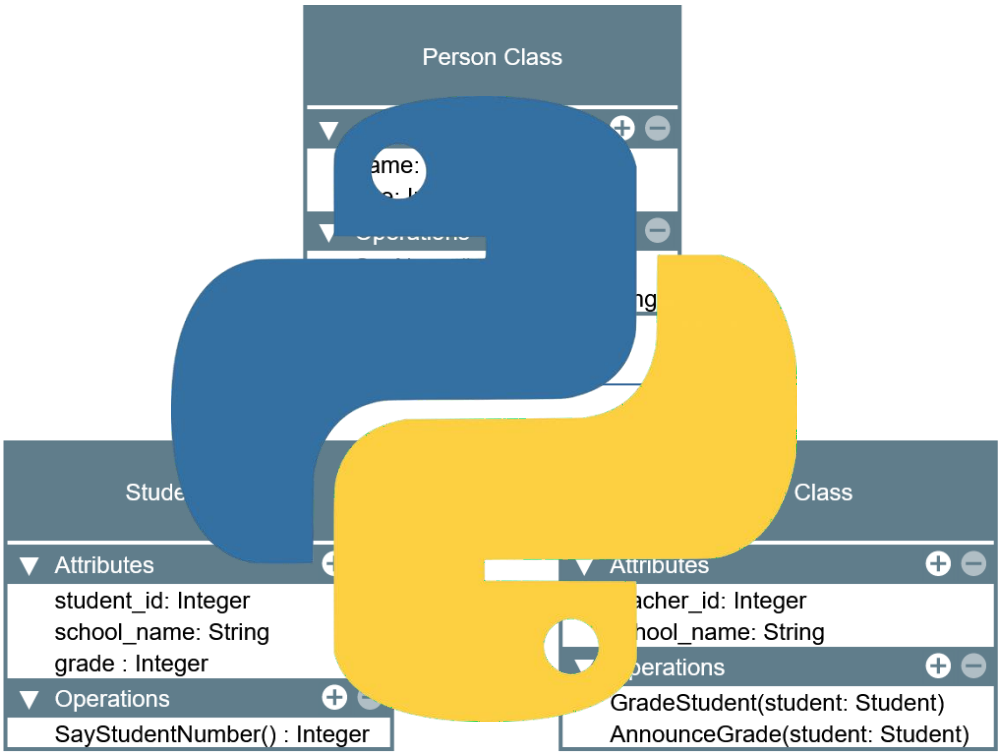
UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025



LABORATORY MANUAL

Object-Oriented Programming (CPE 103)

Laboratory Activity No. 2.1	
Literals, Operators, and Variables	
Course Code: CPE103	**Program:** BSCPE
Course Title: Object-Oriented Programming	**Date Performed:** 02/01/25
Section: 1-A	**Date Submitted:** 02/01/25
Name: Delinia, Filjohn B.	**Instructor:** Engr. Maria Rizette Sayo
1. Objective(s):	
Implement literals and variables in a python program.	

UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025

2. Intended Learning Outcomes (ILOs):

The students should be able to:
2.1 Write a simple program implementing literals and variables
2.2 Use comments and identify keywords from identifiers created by users.

3. Discussion:

1. Variables:

A **variable** is a name assigned to a storage location in memory, where data can be stored and modified during the execution of the program. In Python, you don't need to specify the type of data a variable holds; Python will automatically infer the type when a value is assigned to it.

- Variables can store different types of data, such as numbers, text, etc.
- The value in a variable can change throughout the program.
- Python uses dynamic typing, so you don't need to declare the type of the variable beforehand.

2. Constants:

A **constant** is a value that remains unchanged throughout the program. While Python doesn't have a built-in constant type, it's common practice to use a naming convention to indicate constants.

- Constants are meant to stay constant once assigned a value.
- By convention, constants are written in uppercase letters.

3. Literals:

A **literal** refers to a fixed value directly written in the code. These are the actual values like numbers or strings that appear in the program.

- Literals represent the direct values used in the code.
- They are often used to initialize variables or constants.

4. Materials

Google Colab
Microsoft Word

5. Procedure:

Calculating the Final Grade in a CpE Course
Materials Used:

- Microsoft Word: For documenting the procedure.
- Google Colab: For writing, testing, and running the Python program.

Step 1: Setup and Introduction

1. Open Google Colab: Go to Google Colab and create a new notebook.
2. Create a Program: Write a Python program that calculates the final grade based on the following formulas.

Step 2: Input Data

1. The program should prompt the user to enter the following grades:
    - Prelim Grades:
        - Prelim Exam Grade (out of 100)
        - Prelim Hands-on Activity Grade (out of 100)
        - Prelim Quiz Grade (out of 100)
        - Prelim Assignment Grade (out of 100)
    - Midterm Grades:
        - Midterm Exam Grade (out of 100)
        - Midterm Hands-on Activity Grade (out of 100)
        - Midterm Quiz Grade (out of 100)
        - Midterm Assignment Grade (out of 100)
    - Final Grades:
        - Final Exam Grade (out of 100)
        - Final Hands-on Activity Grade (out of 100)
        - Final Quiz Grade (out of 100)
        - Final Assignment Grade (out of 100)

Step 3: Calculating Intermediate Grades

1. Calculate Prelim Class Standing (CS):
    - Use a weighted average formula for Prelim CS: 50% Hands-on Activity, 30% Quiz, and 20% Assignment.
2. Calculate Prelim Grade:
    - Calculate the Prelim Grade using a weighted average: 50% Prelim Exam and 50% Prelim CS.
3. Calculate Midterm Class Standing (CS):
    - Use the same weighted average formula for Midterm CS: 50% Hands-on Activity, 30% Quiz, and 20% Assignment.
4. Calculate Midterm Grade:
    - Midterm Grade is calculated as: 1/3 of Prelim Grade + 2/3 of (50% Midterm Exam + 50% Midterm

CS).

5. Calculate Final Class Standing (CS):
    - Use the same weighted average formula for Final CS: 50% Hands-on Activity, 30% Quiz, and 20% Assignment.
6. Calculate Final Grade:
    - Final Grade is calculated as: 1/3 of Midterm Grade + 2/3 of (50% Final Exam + 50% Final CS).

Step 4: Display Final Grade

1. After calculating the final grade, the program should display it to the user, formatted to two decimal places.

Step 5: Adding Comments/Documentation

1. Add comments in the program to explain what each part of the code does.
2. Provide documentation strings (docstrings) for each function to clarify its purpose.

Step 6: Test the Program

1. Run the program and enter sample grades for each category to verify that the calculations are correct.
2. Ensure that the program correctly calculates and displays the final grade.

Step 7: Document the Procedure in Microsoft Word

1. Open Microsoft Word and create a new document.
2. Provide the problem statement, explaining the purpose of the program and the formulas used for calculation.
3. Outline the steps followed in the code, from input collection to final grade display.
4. Add screenshots or a description of the Python program running on Google Colab, if necessary.
5. Format the document neatly, including headings for each section.

Step 8: Save and Submit the Document

1. Save the Word document in the required format (e.g., .docx or .pdf).
2. Submit the document as per the requirements given by the instructor.

This procedure will help guide you in creating a program that calculates the final grade in a CpE course, using Python and Google Colab, while documenting the process in Microsoft Word.

Place all the output of the procedures here

#1st Student
Student Information

Enter Student Name: Jhustine Bron

Enter Prelim Hands-on Activity grade: 94
Enter Prelim Quiz grade: 90
Enter Prelim Assignment grade: 85

Prelim CS is: 91.0
Enter Prelim Exam grade: 86
Prelim Grade is: 88.50

Enter Midterm Hands-on Activity grade: 90
Enter Midterm Quiz grade: 94
Enter Midterm Assignment grade: 83
Midterm CS is: 89.80000000000001
Enter Midterm Exam grade: 95
Midterm Grade is: 92.40

Enter Finals Hands-on Activity grade: 94
Enter Finals Quiz grade: 91
Enter Finals Assignment grade: 70
Finals CS is: 88.3
Enter Finals Exam grade: 99
Finals Grade is: 93.65

#2nd Student
Student Information

Enter Student Name: Joshua Catahan

Enter Prelim Hands-on Activity grade: 88
Enter Prelim Quiz grade: 81
Enter Prelim Assignment grade: 89
Prelim CS is: 86.1
Enter Prelim Exam grade: 90
Prelim Grade is: 88.05

Enter Midterm Hands-on Activity grade: 91
Enter Midterm Quiz grade: 92
Enter Midterm Assignment grade: 94
Midterm CS is: 91.89999999999999
Enter Midterm Exam grade: 88
Midterm Grade is: 89.95

Enter Finals Hands-on Activity grade: 90
Enter Finals Quiz grade: 89
Enter Finals Assignment grade: 95
Finals CS is: 90.7
Enter Finals Exam grade: 91
Finals Grade is: 90.85

#3rd Student
Student Information

Enter Student Name: Hannah Directo

Enter Prelim Hands-on Activity grade: 90
Enter Prelim Quiz grade: 91
Enter Prelim Assignment grade: 95
Prelim CS is: 91.3
Enter Prelim Exam grade: 93
Prelim Grade is: 92.15

Enter Midterm Hands-on Activity grade: 87
Enter Midterm Quiz grade: 89
Enter Midterm Assignment grade: 90
Midterm CS is: 88.2
Enter Midterm Exam grade: 97
Midterm Grade is: 92.60

Enter Finals Hands-on Activity grade: 90
Enter Finals Quiz grade: 99
Enter Finals Assignment grade: 94
Finals CS is: 93.5
Enter Finals Exam grade: 85
Finals Grade is: 89.25

6. Supplementary Activity:

- Test 3 students from the program you created.

UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025

- The program should show the name of the student, the PRELIM, MIDTERM and FINAL grades.
- Convert the final grade into the UCCs numerical grade. Please refer to the grading system.

PLEASE REFER TO THIS LINK:

https://colab.research.google.com/drive/1sDodPf683w25UTOi2waNOuifEeGffAXk?usp=sharing

UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025

Questions:

1. Why is it important to calculate the Prelim Class Standing (CS) using a weighted average of the hands-on activity, quiz, and assignment, rather than treating them equally?

Answer:
Using a weighted average reflects the different importance of each assessment type. Hands-on activities and quizzes are often considered more critical to understanding practical skills, so they have higher weights. This approach ensures that the student's real-time performance is valued more than just assignments.

2. How does the inclusion of the Prelim Grade in the calculation of the Midterm Grade affect a student's overall performance in the course?

Answer:
Including the Prelim Grade in the Midterm Grade gives a more comprehensive picture of a student's performance over time. It ensures that both early and mid-course efforts contribute to the final evaluation, helping to reflect consistent progress.

3. If a student consistently performs well in quizzes and assignments but struggles with exams, how might this affect their final grade calculation in the course?

Answer:
Good performance in quizzes and assignments can help boost the final grade, even if exam scores are low. Since non-exam components are weighted, a student’s overall grade can still be decent, but poor exam results will likely lower their final grade due to the significant weight of exams in the grading system.

7. Conclusion:

The objective of using literals and variables in a Python program was successfully met. In the process of creating the grade calculator for the CpE course, we made use of variables to store different types of data, such as exam scores, class standing, and activity grades. We also used literals, like the fixed percentages (50%, 30%, 20%), to help perform the calculations for each grading component.

By incorporating both variables and literals, the program was able to dynamically calculate the final grade based on the user's input, making it functional and aligned with the grading system. In the end, we were able to achieve the goal of effectively using literals and variables in the program to handle grade calculations.

8. Assessment Rubric: