

CSE 1321L: Programming and Problem Solving I Lab

Assignment 2

Module 2

What students will learn

- o Problem Solving.
- o Basic Program Structure.
- o Input and Output with the user.
- o Write code that includes if/else/elif and match statements.
- o Structure program to include conditional logic.

Content

- o Overview
- o Assignment2A: Online Shopping Discount Calculator
- o Assignment2B: Loan Approval System
- o Assignment2C: Role Playing Game

Overview:

For this assignment, you're going to practice making decisions in your code. It's because of those decisions that your program can behave differently depending on the values in your variables; these decisions are called conditional statements. In practical terms, this means you're going to expand on the concepts from previous assignments, but also include things like IF, ELSE IF, ELSE and MATCH statements. Again, start early, practice, and ask a lot of questions.

Final note: **Do not cheat**

If your temptation is to look online, don't. Come see us instead and ask questions – we are here to help. Remember, you are going to have to write codes in your future job interviews, so learn it now to secure a high-paying job later.

Assignment2A: Online Shopping Discount Calculator

Write a Python program to calculate the discount on an online shopping order based on the customer's total purchase amount and their membership status.

For this task:

- o Prompt the user to enter their total purchase amount.
- o Prompt the user to enter whether they are a member of the online shopping club (yes or no).
 - o Users may enter Yes/YES/yes & No/NO/no.
 - o Program should be able to handle these cases.
- o Use if, elif, and else statements to calculate the discount based on the following rules:
 - o If the total purchase amount is less than \$50, there is no discount.
 - o If the total purchase amount is between \$50 and \$200 (inclusive):
 - The discount is 5% for non-members.
 - The discount is 10% for members.
 - o If the total purchase amount is greater than \$200:
 - The discount is 10% for non-members.
 - The discount is 15% for members.
- o Display the final discount amount and the total price after the discount is applied.

Below is the example run. The user input is shown in **red and bold** (notice the dollar sign is not part of the user input and round decimal points to 1 place).

Sample Output #1:

[Discount Calculator]

Enter your total purchase amount: **\$150**

Are you a member of the shopping club (Yes or No)? **Yes**

Your discount is: \$15.0

Your total price after discount is: \$135.0

Assignment2B: Loan Approval System

Write a Python program that accepts three inputs from the user:

- o Age (integer)
- o Income (integer)
- o Credit Score (integer between 300 and 850)

Based on the inputs, determine whether the user qualifies for a loan and what type of loan they qualify for. Use nested **if-elif-else** statements and the program must use a **match statement** to classify the person's **credit score** into categories.

For this task:

1)-Qualification Criteria:

- o The user's **age** must be at least **18 years old**.
- o The user's **credit score** must be:
 - o **Good**: 700–850
 - o **Fair**: 600–699
 - o **Poor**: 300–599
 - o Use **match statement** for this classification
- o The user's **income** determines the type of loan they qualify for:
 - o Income \geq **100,000** qualifies for a "Premium Loan".
 - o Income \geq **50,000** but $<$ **100,000** qualifies for a "Standard Loan".
 - o Income $<$ **50,000** qualifies for a "Basic Loan".

2)-Loan Decision:

- o If the user does not meet the **age criteria**, print "You do not qualify for a loan due to age."
- o If the user's credit score is **Poor**, print "You do not qualify for a loan due to poor credit."
- o Otherwise:
 - o If the user qualifies for a loan, print the type of loan based on their income:
 - "You qualify for a Premium Loan" (if credit is **Good** and income is \geq **100,000**).
 - "You qualify for a Standard Loan" (if credit is **Good** or **Fair**, and income is between **50,000 and 99,999**).
 - "You qualify for a Basic Loan" (if credit is **Fair**, but income is $<$ **50,000**).
 - o If income is too low for any loan, print "Your income is too low for a loan."

Note:

- o Use nested if-elif-else and match statements to handle the combinations of **credit score, income, and age**.
- o Validate that the credit score is in the correct range (e.g., credit score must be between 300 and 850).

Example runs are shown below. The user input is shown in **red and bold**(notice the dollar sign is not part of the user input).

Sample Output #1:

[Loan Approval System]
Enter your age: **25**
Enter your income: **\$120000**
Enter your credit score: **750**
You qualify for a Premium Loan.

Sample Output 2:

[Loan Approval System]
Enter your age: **17**
Enter your income: **\$80000**
Enter your credit score: **720**
You do not qualify for a loan due to age.

Sample Output 3:

[Loan Approval System]
Enter your age: **30**
Enter your income: **\$45000**
Enter your credit score: **680**
You qualify for a Basic Loan.

Assignment2C: Role-Playing Game (RPG)

Your task is to develop a **text-based Role-Playing Game (RPG)** in Python. The game will allow the player to choose a character class and perform specific actions based on their selection. The program must provide clear instructions, handle input effectively, and display appropriate responses for valid and invalid choices.

To complete this assignment, you **must use the match statement** for decision-making.

For this task:

1. Welcome Message and Class Selection:

- o Display a welcome message introducing the game: `Welcome to the RPG Game!`
- o Prompt the player to select one of the following classes:
 - `Warrior: A strong and brave fighter.`
 - `Mage: A wielder of powerful magic.`
 - `Healer: A supportive and kind character.`
- o Use the **match** statement to process the player's selection and display a confirmation message:
 - `Warrior: You have chosen Warrior! You are strong and brave.`
 - `Mage: You have chosen Mage! You wield powerful magic.`
 - `Healer: You have chosen Healer! You are kind and supportive.`
 - `Invalid input: Invalid class choice. The game ends.`

2. Action Choices Based on Class:

- o After class selection, present the player with two unique actions based on their class:
 - `Warrior:`
 - `Attack your sword.`
 - `Defend with your shield.`
 - `Mage:`
 - `Cast a fireball.`
 - `Cast a healing spell.`
 - `Healer:`
 - `Heal your ally.`
 - `Attack with your staff.`
- o Use the **match** statement to process the player's action and display appropriate messages:
 - For valid actions:
 - **Warrior:**
 - `Attack: You swing your sword and defeat the enemy!`
 - `Defend: You raise your shield and block the enemy's attack!`
 - **Mage:**
 - `Cast Fireball: You cast a fireball and scorch the enemy!`
 - `Cast Healing Spell: You cast a healing spell and restore your energy.`
 - **Healer:**
 - `Heal Ally: You heal your ally and boost their morale!`
 - `Attack with Staff: You swing your staff and knock out the enemy!`
 - For invalid actions, display a specific error message:

3. End Game

- o At the end of the game, display: `Thank you for playing!`

4. Error Handling

- o Ensure the program does not crash for invalid inputs.
- o Handle invalid class and action inputs gracefully, displaying appropriate messages.

Example runs are shown below. The user input is shown in **red and bold**.

Sample Output #1:

Welcome to the RPG Game!
Choose your class: 1. Warrior 2. Mage 3. Healer
Enter the number of your class (1/2/3): **2**
You have chosen Mage! You wield powerful magic.
What would you like to do? 1. Cast a fireball 2. Cast a healing spell
Choose your action (1/2): **1**
You cast a fireball and scorch the enemy!
Thank you for playing!

Sample Output 2:

Welcome to the RPG Game!
Choose your class: 1. Warrior 2. Mage 3. Healer
Enter the number of your class (1/2/3): **5**
Invalid class choice. The game ends.
Thank you for playing!

Sample Output 3:

Welcome to the RPG Game!
Choose your class: 1. Warrior 2. Mage 3. Healer
Enter the number of your class (1/2/3): **3**
You have chosen Healer! You are kind and supportive.
What would you like to do? 1. Heal your ally 2. Attack with your staff
Choose your action (1/2): **2**
You swing your staff and knock out the enemy!
Thank you for playing!

Submission Instructions:

- o Programs must follow the output format provided. This includes each blank line, colons (:), and other symbols.
- o Programs must be working correctly.
- o Programs must be written in Python.
- o Programs must be submitted with the correct **.py** format.
- o Programs must be saved in files with the correct file name:
 - Assignment2A.py
 - Assignment2B.py
 - Assignment2C.py
- o Programs (source code files) must be uploaded to Gradescope by the due date.