

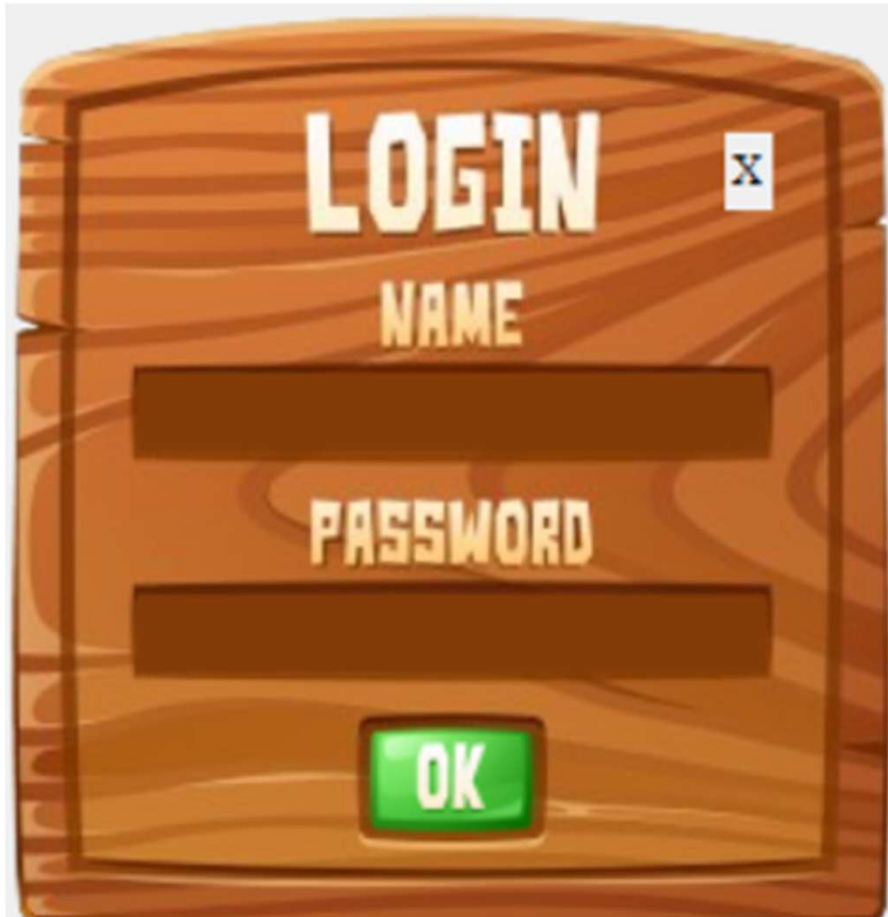
IT101-2L Machine Problem

2T 2024 - 2025



Objectives:

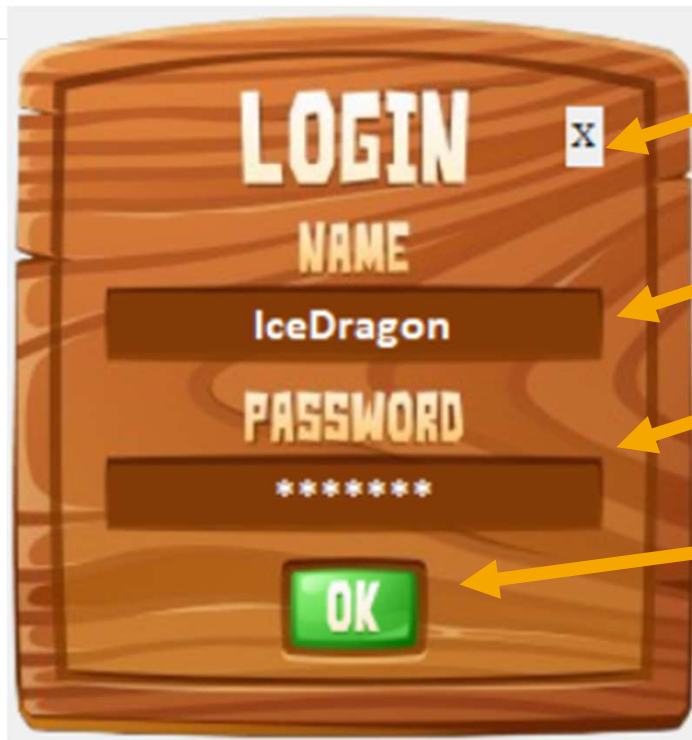
- Design and implement an interactive graphical user interface (GUI) using Tkinter, handling user inputs and events effectively
- Combine file handling and OOP principles to build a functional and event-driven application.



Login Window

- Create a **login window** that allows the user to enter a **username** and **password**. The program should:
 1. **Validate Input:** Ensure that both the username and password fields are not empty.
 2. **Check Credentials:**
 - Verify if the **username exists** in the userAccount text file.
 - If the username exists, check if the **password matches**.
 3. **Display Appropriate Messages:**
 - If the credentials are correct, show a "**Login Successful**" message box and proceed to the **game screen**.
 - If the credentials are incorrect, show a "**Username and/or password is incorrect**" message box.

Log In Window



btnQuit : Button

txtUname : Entry

txtPw : Entry, show = "*"
 - Use option show to mask password

btnOK: Button

Game Window: IT Quiz Bee GUI application that presents **50 questions** randomly selected from the QandA text file. The game should follow these rules:

1.Game Mechanics:

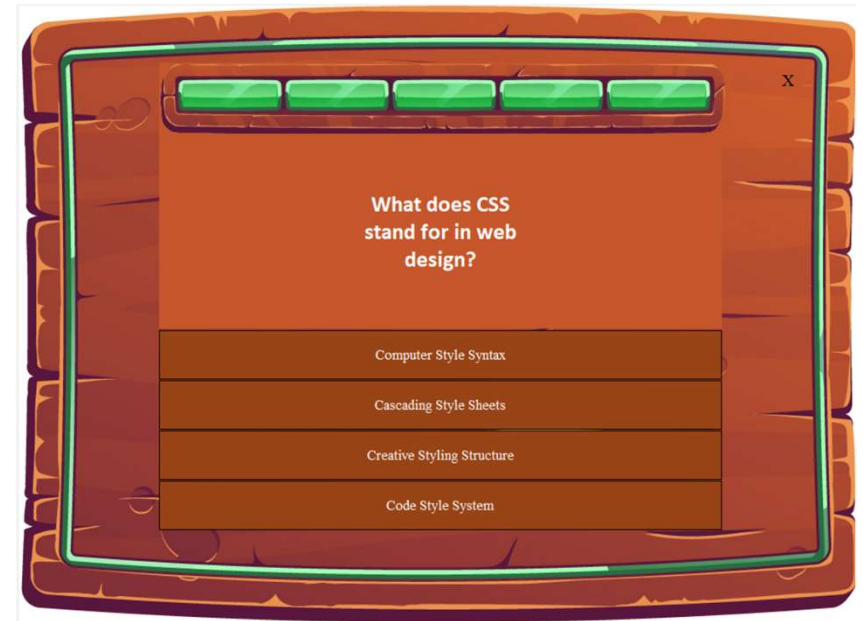
- The player starts with **five (5) lives**.
- For every incorrect answer, one life is deducted.**
- For every correct answer, one life is added**, but the maximum remains **five (5) lives**.
- The game ends when the player **loses all lives** or **answers all 50 questions**.

2.Question and Answer Format:

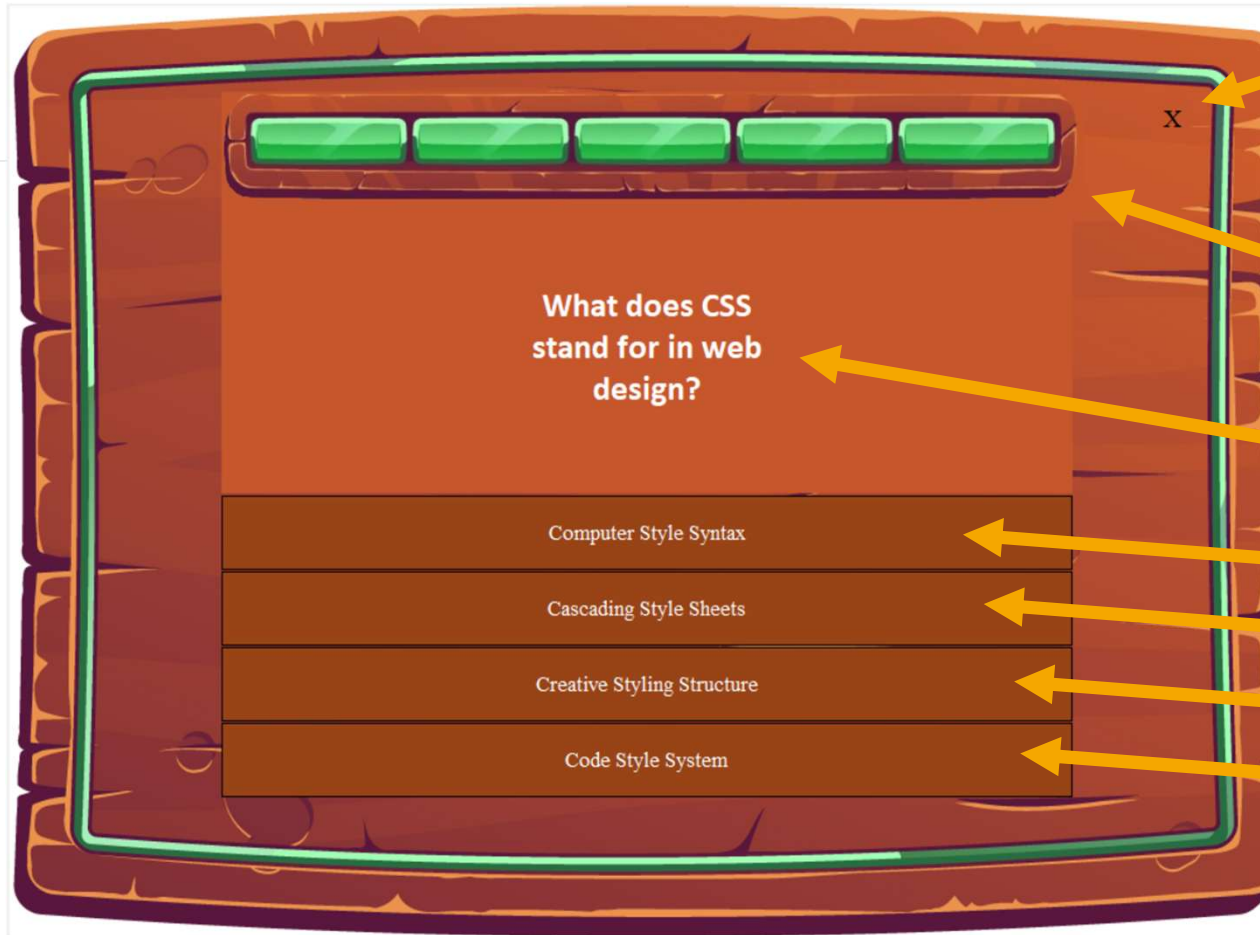
- Questions must be displayed in **random order**.
- Each question follows the format in the text file:
question;optionA;optionB;optionC;optionD
 - The **first option (optionA) is always the correct answer**.
- The app should **shuffle answer choices** before displaying them to prevent predictable patterns.

3.End of Game:

- Display the **player's final score** once the game ends.
- Provide an option to **restart the game or exit**.



Game Window



btnQuit : Button

lblHP : Label

lblQuestion : Label

btnA : Button

btnB : Button

btnC : Button

btnD : Button

Rubrics

- **Functionality** 10 pts. (The app works flawlessly, correctly handling login authentication/game mechanics with no errors or crashes.)
- **User Interface** (UI) 10 pts. (UI is well-designed, intuitive, and visually appealing with clear labels and organized layout.)
- **Error Handling & Validation** 10 pts. (Proper input validation and error handling (e.g., empty fields, incorrect inputs, file handling errors).)
- **Implementation of OOP** 10 pts.
- **Randomization & Logic** (Quiz Game) 10 pts. (Questions and answers are randomized correctly; game follows all mechanics perfectly.)
- **Endgame & Scoring System** 10 pts. (Endgame mechanics work perfectly; score is correctly displayed and updated.)
- **Group Attendance** (20 pts.)
- **Presentation** (20 pts.) Create a 20-min (max.) recorded presentation demonstrating the output of your program and code explanation