

Lab Exam: Video Game Rental System

Task:

Create a video game rental system that includes functionalities for both admin and user roles.

Instructions:

1. Your task is to build a video game rental system that allows users to rent games from a library of available titles.
2. The system should have two main roles: admin and user.
3. The game library initially consists of three games: "Donkey Kong," "Super Mario Bros," and "Tetris." Each game has an initial quantity and rental cost as follows:
 - Donkey Kong: 3 copies, rental cost \$2
 - Super Mario Bros: 5 copies, rental cost \$3
 - Tetris: 2 copies, rental cost \$1
4. Feel free to add more games to the library as needed. You can represent the game library using a dictionary where the keys are game titles (strings) and the values are dictionaries containing the quantity (integer) and rental cost (float) of each game.
5. The rental system starts with no existing user accounts. You can represent user accounts using a dictionary where the keys are usernames (strings) and the values are dictionaries containing the password (string), balance (float), points (integer), and inventory (list of strings) of each user.
6. When a new user registers, they must top up their account balance before being able to rent a game.
7. Users should be able to view the available games, register as new users, log in, and rent or return games.
8. The admin has the authority to update game details, such as quantity and cost, in the game library.
9. The admin login credentials are as follows; they are in string type:
 - Username: admin
 - Password: adminpass
10. The system should allow users to cancel a transaction by leaving the input prompt blank.
11. Your program should be implemented using a functional programming approach and organized within a main function.

Instructions for Implementation:

1. Use the provided starter code template as a guide to build the program.
2. Implement each function according to its specified functionality.
3. Ensure input validation and error handling are in place to handle unexpected user inputs.
4. Provide clear prompts and messages to guide users through the rental process.
5. Test your program thoroughly to ensure it behaves as expected under various scenarios.
6. Organize your code logically and use meaningful variable names to enhance readability.
7. Document your code with comments to explain the purpose of each function and any complex logic.
8. Once completed, run your program to verify that it meets all the requirements listed above.

Best of luck with your implementation!

Point system is \$2 = 1 point

3 points = 1 free game