Task 1 for Professional Developers: Gift Wishlist Manager

Description

Develop a command-line application in Python that manages a **gift wishlist**. Users should be able to add, view, update, and remove gift ideas for different people. The application should save the wishlist to a CSV file when exiting and load it when starting, preserving the data between sessions.

Requirements

1. Gift Model

Each gift entry should have the following attributes:

- ID (unique identifier)

- Recipient's Name

- Gift Description

- Occasion (e.g., Birthday, Christmas, Anniversary)

- Price

- Status (Planned/Purchased)

2. Functionalities

- **Add Gift Idea:** Input the recipient's name, gift description, occasion, and price. The status should default to 'Planned'. Each entry should have a unique ID.

- **List All Gifts:** Display all gift ideas, grouped by recipient. Include details like gift description, occasion, and status.

- **Update Gift Status:** Mark a gift as 'Purchased' or revert it to 'Planned' by providing its ID.

- **Delete Gift Idea:** Remove a gift by its ID.

- **Smart search:** Search the gifts in the list matching a given price range. Make the search efficient in terms of time complexity.

- **Save & Load:**

  - Save the wishlist to a CSV file (`wishlist.csv`) on exit.
  - Load the wishlist from `wishlist.csv` when the application starts, maintaining state.

3. Additional Details

- **Command-Line Interface:**
    - Text-based menu with the following options:
        * Add a new gift idea
        * List all gift ideas
        * Update gift status
        * Delete a gift idea
        * Search the gifts matching a given price range
        * Exit and save
    - User-friendly error messages for invalid inputs.

- **File Format and Persistence:**
    - Store the wishlist in a CSV file (`wishlist.csv`) with the following columns:
        * ID, Recipient's Name, Gift Description, Occasion, Price, Status
    - Ensure data persistence by loading this file at startup.

Checklist for Evaluation

☐ Add a new gift idea with recipient's name, description, occasion, and price. (10%)

☐ Input validation (e.g., non-empty name and description). (5%)

☐ User-friendly error messages and confirmation prompts. (5%)

☐ List all gifts grouped by recipient. (10%)

☐ Update a gift's status (Planned/Purchased). (10%)

☐ Delete a gift idea by its ID. (5%)

☐ Search gifts in a given price range. (15%)

☐ Search is more efficient than simple linear search $O(N)$. (10%)

☐ Save wishlist to `wishlist.csv` on exit. (10%)

☐ Load wishlist from `wishlist.csv` at startup. (10%)

☐ Exception handling (e.g., file not found, invalid input). (10%)