Documentation:

Author: Darren Huynh
Credits: SeaofBTC

Overview:

This inventory program is written in Python 2.7, and uses SQLite3 for database management, and TkInter for user interface. Functions are written in Python to execute SQL commands to communicate with the SQL database, which are called upon button press in the GUI. The program includes a self created Database upon initial usage, and generates a changelog as well.

Technical Description:

The program consists of multiple frames (pages henceforth) that have data entry fields for accessible user interaction. User input is collected upon user interaction(entering data, selecting options, button presses), and used in functions that execute included functions (create, edit , view) by passing said input into SQL execute statements (Insert, Update, Select). Widgets are appropriately placed and labeled for easy navigation via Grid/Pack widget management. A summary page is dynamically created upon request, such that it includes all changes made to the database upon creation. An admin delete function is included for deleting entries, which is username and password protected. All transactions are logged within the included changelog.txt file.

For modification purposes:

1) **Modules used**
Tkinter, Sqlite3, tkMessageBox, Datetime, Time, Sys,Ttk, os.path

2) **Dialog Boxes**
Majority of the confirmation/message dialog boxes are created using tkMessageBox. Editing these dialog boxes should follow tkMessageBox syntax. If necessary, replace tkMessageBox with module of choice. Because these dialog boxes are called by functions, further editing should not be necessary.

3) **Summary/Admin Toplevel Windows**
The summary and admin pages operate using Python's Toplevel function. Editing the parameters for the summary and admin pages are local. The summary page may be edited by replacing the entire page with a listbox, such that you may include a scrollbar for easier access.

4) **Changelog Writing**
Changelog writing was implemented using the os.path module. Each of the changelog functions write a different type of event to the changelog. You may modify the elements written to the changelog.

5) **Container Modification**
In the inventory management class, there is a list of pages that are preloaded upon program start. By modifying this list, you may edit the pages that are displayed. It is important to include other class instances within this list, or they will not be loaded and thus cause bugs.

6) **Page Modification**
Each of the pages are separate instances. Each of the pages contain their own labels,entry fields, and frames, which are managed by grid. You may create new pages if necessary, but remember to add it into the list in the container object.

7) **SQL Function Modification**
Majority of the parameters in SQL execute statements are written to accept user input, which precedes the actual function call. User entry is collected using .get(), and stored into a local variable. These variables are then passed into the execute statements. You may modify the parameters that are entered, and also modify SQL commands as well.

8) **GUI Modification**

Majority of the program is written with Grid widget management. You may modify the parameters of specific buttons. Feel free to modify grid placement, this is up to personal discretion

9) **Admin Modification**

The admin username and password are statically set within the program for simplification purposes. You may edit the admin delete page to not need admin verification at all. You may also change the statically set username and password to a dictionary with keys, such that you may manage multiple users with different abilities. It would be highly suggested that the changelog include the “username” of any activity for consistency purposes.