Point of Sale System Documentation

This Python script implements a simple Point of Sale (POS) System using the Tkinter library for the graphical user interface (GUI) and SQLite for data storage. The system allows users to add and delete products, make sales, and view sales details.

Table of Contents

1. Database Setup
2. Tkinter GUI
3. Functions
4. Widgets and Layout

1. Database Setup

- The script connects to an SQLite database named 'pos_system.db'.
- Two tables, 'products' and 'sales', are created to store information about products and sales, respectively.
- The 'products' table has columns: **id**, **name**, **price**, and **stock**.
- The 'sales' table has columns: **id**, **product_id**, **quantity**, and **total**

2. Tkinter GUI

- The main application window (**app**) is created with a title and dimensions.
- Two treeviews (**product_tree** and **sales_tree**) are used to display products and sales.
- Labels, entry fields, and buttons are used for various functionalities.

3. Functions

populate_product_tree()

- Retrieves all products from the database and displays them in the **product_tree** Treeview widget.

add_product()

- Adds a new product to the 'products' table in the database with the specified name, price, and stock.
- Updates the product_tree and clears the input fields.

delete_product()

- Deletes the selected product from the 'products' table in the database.
- Updates the **product_tree**.

make_sale()

- Processes a sale by updating the 'sales' table with the product ID, quantity, and total price.
- Updates the 'products' table by reducing the stock of the sold product.
- Updates both the **product_tree** and **sales_tree**.
- Clears the input fields.

populate_sales_tree()

- Retrieves all sales from the database and displays them in the **sales_tree** Treeview widget.

on_closing()

- Closes the SQLite database connection when the application is closed.

4. Widgets and Layout

- Entry fields for entering product details (name, price, stock) and sales details (product ID, quantity).
- Buttons for adding and deleting products, and making a sale.
- Labels for guiding the user in entering information.
- **product_tree** and **sales_tree** Treeview widgets to display lists of products and sales.

Additional Notes

- The script uses global variables for simplicity. In a larger application, it might be beneficial to use classes and methods for better organization and encapsulation.
- The script includes an event handler (**on_closing()**) to ensure the database connection is closed when the application is closed.