## Placeholder image

*Project Bookworm: A Bookstore GUI*

12/13/2024

Lilly Hampton & Marcin Brodowicz

# Overview

*The BookWorm Inventory Manager is a desktop application designed to help a small, local bookstore called BookWorm to manage its inventory, customers, and orders. The app allows users to easily add new books, update existing ones, and remove books from the inventory. It also enables users to manage customer profiles and track orders.*

*The application features a simple and user-friendly graphical interface, making it easy for new users to navigate. It securely stores all information, such as book details, customer information, and order histories, using SQLite, a reliable database system. The app uses Tkinter, a popular tool for creating clear and attractive interfaces. Overall, the BookWorm Inventory Manager is a valuable tool for any small business that wants to improve its operations and customer service.*

# User Interface Description

* *The graphical user interface (GUI) is structured around several key components, each designed to enhance user interaction and functionality:*
* *- The Header: This section prominently displays the title of the application, providing users with a clear identification of the program. Additionally, the current date and time are shown here, ensuring that users are always aware of the context in which they are operating.*
* *- The Menu: The menu is equipped with a variety of buttons that facilitate navigation throughout the application. Users can access different functionalities with ease, such as managing inventory, viewing customer details, and utilizing options to add new books, customers, or orders. This streamlined navigation is essential for efficient operation within the application.*
* *- The System Frame: Depending on the menu option selected, this area of the GUI dynamically displays relevant forms and lists. This could include detailed information on available inventory, customer records, or order details, allowing users to interact with the application’s core functionality seamlessly. The adaptability of the system frame ensures that users have immediate access to the information they need, tailored to their specific tasks.*

## Key Features

* *Book Management: This feature allows users to efficiently manage the inventory of books. Users can add new titles to the inventory, update existing book details such as author, genre, and price, and remove books that are no longer available.*
* *Customer Management: This component enables the management of customer information. Users can add new customer profiles, update existing details such as contact information and preferences, and remove customer records as necessary to maintain an accurate database.*
* *Order Management: This feature facilitates the entire order process. Users can place new orders, track their status, and manage previous orders, ensuring a streamlined experience from selection to delivery.*
* *Real-Time Clock: This tool displays the current time and updates every second, providing users with an accurate time reference for scheduling and managing tasks effectively.*

# *Step-by-step instructions*

## Setting up the database

*Before running the application, the user should ensure that the database tables are created.*

* *Start by opening either terminal or command prompt*
* *Run the following command to execute the bookstore.db (some may title it create\_tables.py) script:*

*python bookstore.db*

## *Running the application*

* *Open the terminal or command prompt*
* *Run the main application file:*

***python main.py***

### *Adding a book*

* *Click the Add Book button in the menu*
* *Fill in the details ( title, author, ISBN, price, and quantity) in the window that appears*
* *Click Add to save the book to the inventory*

#### 

#### *IIII. Adding a customer*

* *Click the Add Customer button in the menu.*
* *Fill in the details ( name, customer ID, email, and phone number) in the window that appears.*
* *Click Add to place the customer to the database.*

##### 

##### *V. Placing an order*

* *Click the Add Order button in the menu,*
* *Fill in the details ( order ID, customer ID, list of books and book IDs, and order date ) in the window that appears.*
* *Click Add to save the order.*

## *Technical Requirements*

* *Python: Ensure Python is installed*
* *Tkinter: Tkinter comes pre-installed with Python, but it is best to ensure it is available.*
* *SQLite: The project uses SQLite for the database, which is also included with Python.*

## *Project Structure:*

* *main.py: Entry point for running the application*
* *bookstore.db: Script to create the necessary database tables.*
* *GUI.py: Contains the Tkinter GUI setup and the BookstoreApp class.*