**Book Market Management System**

**Project Description**

Many book markets and academic bookshops struggle with efficiently managing their operations. The most common challenges are keeping track of book inventory, customer information, and monitoring orders. When these processes are handled manually or across disconnected systems, they quickly become tedious, inconsistent or redundant, which may lead to errors such as stock miscounts, lost customer records or delayed order processing. This not only reduces efficiency but also negatively impacts customers satisfaction.

A Book Market Management System addresses these issues by consolidating all critical functions into one centralized platform. This enables shop management to store and update book inventory details (such as titles, authors, categories, prices and quantities). It also allows the shop to maintain accurate customer records, as well as efficiently track orders.

Ultimately the Book Market Management System offers a structured solution to efficiently complete daily operations and reduce human error.

**Project Objectives**

The aim of this project was to build a **Windows Forms (WinForms) application in C#** that connects to a **SQL Server database**. The application serves as the graphical user interface (GUI), allowing the administrator to interact with the database without directly writing SQL queries. Through this system, the administrator can efficiently manage book records, categories, and orders.

The specific objectives of the project are:

1. **Database Integration**
   * Develop a SQL Server database with tables for administrators, books, categories, orders, and order details.
   * Establish relationships between the tables to ensure data consistency.
2. **Secure Login (Admin Only)**
   * Implement an authentication system where only the system administrator can log in and access the application.
3. **CRUD Functionality**
   * Provide full Create, Read, Update, and Delete operations for managing books, categories, and orders.
   * Ensure that all changes made through the application are reflected in the SQL Server database.
4. **Search Facility**
   * Implement a search feature that allows the administrator to quickly find book records by title, author, category, or other attributes.
5. **Error Checking and Validation**
   * Prevent invalid inputs (e.g., negative prices, blank fields).
   * Provide user-friendly error messages to guide the administrator.
6. **SQL Queries for Business Insights**
   * Demonstrate the use of SQL statements, including filtering (WHERE clauses), updates, subqueries, and aggregation functions.
   * Use these queries to extract useful information, such as the number of books in stock or the average book price.

**System Users**

This system will only have one primary user, that is the shop’s management with an administrative role in the system

**Responsibilities**

* **The administrator will be able to log in to the system**
* **They will be able to view all book, category, customer and order records.**
* **They will be able to make necessary changes to all records in the Books Table, Categories Table, Customers Table and Orders Table.**

**System Users**

* Admin login
* Book management (view, add, update, delete)
* Category management
* Order management
* Search books, orders and customers
* Data validation and error checking

**Entity Relation Diagrams**

**A diagram of a computer

AI-generated content may be incorrect.**

**Admin Table**

The admin (Administrator Table) stores the details used by system administrators to log in into the System. The SQL commands to create the Admin Table are as follows.

A close-up of a computer screen

AI-generated content may be incorrect.

**Categories Table**

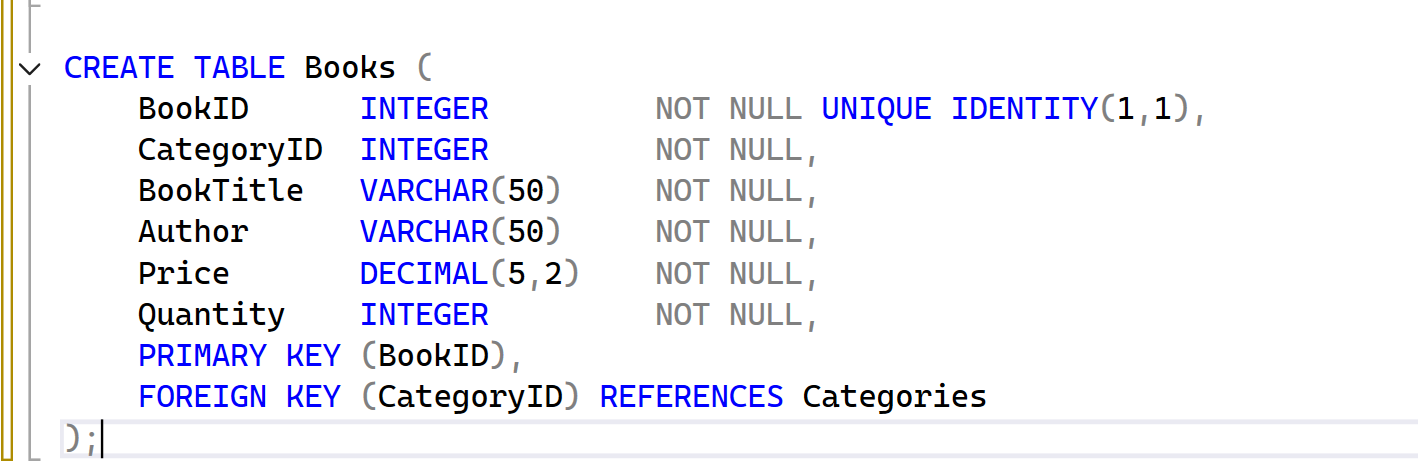
The Categories Table stores the different types of categories for books available, e.g. (History, Mathematics, Programming…)

**A close-up of a white background

AI-generated content may be incorrect.**

**Books Table**

The Books Table store the available books in the store as well as their details, such as who is the author, how many copies are available as well as the price for each copy among other important information.



**Customers Table**

The customers hold data for the customers who buy books in the shop, and plan to buy in the future.

A screenshot of a computer

AI-generated content may be incorrect.

**Orders Table**

The Orders Table holds data about the orders made, which book was ordered and who ordered the book.

A screenshot of a computer

AI-generated content may be incorrect.