Assignment 2:

Movie Booking System

Overview

You are tasked with developing a simple movie booking system, similar to BookMyShow, using Python. This system should allow customers to view popular movies, check showtimes based on location and city, and book tickets. The system will support movies in different languages, with varying ticket prices based on the language.

Objectives

1. **Implement a Movie Booking System**: Create a Python application where users can view and book movie tickets.

2. **Manage Movie and Show Details**: Include functionalities to list movies, showtimes, and theater locations.

3. **Handle Ticket Bookings**: Process ticket sales and store transaction data.

4. **Generate Sales Reports**: Provide insights on sales data, such as total revenue and tickets sold.

5. **Use Python and SQLite**: Utilize Python for the backend and SQLite for data storage.

Requirements

1. **Python Classes**: Employ object-oriented programming concepts to create classes for different system components.

2. **SQLite Database**: Utilize SQLite to manage all data, including movies, showtimes, and transactions.

3. **Ticket Pricing**: Set ticket prices based on the movie language (English: 500 Rs, Non-English: 300 Rs).

4. **Reporting Functionality**: Include features to generate sales and insights reports.

Tasks

1. **Data Modeling**:
   - Design a database schema to store information about movies, showtimes, theaters, bookings, and transactions.
   - Suggested tables: `Movies`, `Showtimes`, `Theaters`, `Bookings`, `Transactions`.

2. **System Development**:
   - Implement a class for movie management, capable of listing movies, filtering by language, and showing popular titles.
   - Create a class to handle showtimes and theater information, allowing users to find shows based on location and time.

- Develop a booking class for processing ticket sales and storing transaction details in the database.

3. **Database Integration**:
   - Develop a class to handle database operations, including creating tables, inserting data, and querying information.
   - Implement error handling for database operations.

4. **Report Generation**:
   - Create a reporting class to generate and display sales reports, such as total revenue, tickets sold per movie, and location-based sales data.

Evaluation Criteria

1. **Functionality**: The system should meet all the specified requirements.
2. **Code Quality**: The code should be well-structured, clean, and well-commented.
3. **Database Design**: Effective use of SQLite for data storage and retrieval.
4. **User Experience**: The system should be user-friendly and easy to navigate.
5. **Documentation**: Include detailed documentation on how to set up and use the system.

Submission

Provide your source code along with a README file containing setup instructions, usage guide, and any necessary documentation.

Bonus Features

Consider adding the following for extra credit:

- Advanced filtering options (e.g., by movie rating, genre)
- User interface improvements (CLI or GUI)
- Advanced reporting features (e.g., peak times, most popular locations)
- User reviews and ratings for movies
- Integration with an external API for real-time movie data