1. Construct a web application for managing a movie database. Begin by setting up a MySQL database named "movie\_db". Create a table named "movies" with the following attributes:

* ID (Primary Key)
* Title
* Director
* Genre
* Release Year
* Rating

Input sample data for at least 10 movies into the "movies" table using a web interface.

Perform the following tasks:

a) Search and display movies released in a specific year.

b) Update the director of a movie based on its ID.

c) Calculate and display the average rating of all movies in the database.

The results should be shown on the webpage only.

1. Create a web application for managing a movie database. Set up a MySQL database named "movie\_db". Define a table named "movies" with the following attributes:

* ID (Primary Key)
* Title
* Director
* Genre
* Release Year
* Rating

Insert sample data for at least 10 movies into the "movies" table using a web interface.

Perform the following tasks:

a) Display all records of the "movies" table in JSON format.

b) Display all records in ascending and descending order of the release year.

c) Delete a movie record based on its ID.

1. Develop a web application for an online bookstore. Start by creating a MySQL database named "bookstore\_db". Define a table named "books" with the following attributes:

* ISBN (Primary Key)
* Title
* Author
* Genre
* Price

Insert sample data for at least 10 books into the "books" table through a web form.

Perform the following operations:

a) Retrieve and display the details of a specific book based on its ISBN.

b) Update the price of a book using its ISBN.

c) Calculate and display the average price of all books in the database.

Results should be exhibited on the webpage exclusively.

1. Create a web application for browsing real estate listings. Set up the necessary databases for the application and display the results accordingly.
   * 1. Users should register on the portal by providing their personal information and preferences.
     2. Upon logging in, users are prompted to select a city.
     3. After selecting a city, users can choose an area and specify their budget for accommodation.
     4. Display the search results from the database based on the user's preferences, including details such as location, type of accommodation, and price range.
     5. Implement a search function to filter accommodations based on locality, such as city and sectors.
     6. Allow users to search for accommodations based on cost criteria.