# Sample Application I: Book Store

**Please download resources from:** [**https://smu.sg/hpy**](https://smu.sg/hpy)

**Given:**

* q1/model
  + common.php (**complete**)
  + ConnectionManager.php (**complete**)
  + Book.php (**complete**)
  + BookDAO.php (**partial**)
  + User.php (**partial**)
  + UserDAO.php (**complete**)
* q1/
  + add.php
  + edit.php
  + index.php
  + list.php
  + login.php
  + logout.php
  + process\_add.php
  + process\_delete.php
  + process\_edit.php
  + process\_login.php
  + bookstore.sql *(****run this before you start)***

This is a bookstore application. Upon logging onto the application successfully, the user will be able add, edit, delete a book from the database. The following is the sequence of flow for the various php programs.

index.php 🡪 login.php 🡪 process\_login.php 🡪 list.php

🡪 add.php 🡪 process\_add.php

🡪 edit.php 🡪 process\_edit.php

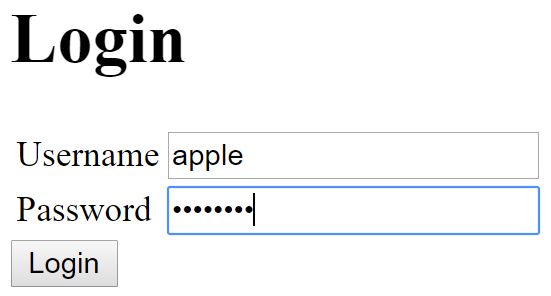
🡪 process\_delete.php

The application uses two classes - Book and User. The Book class has three properties – title, isbn13 and price. The User class similarly has three properties – username, hashed password and name.

The BookDAO and UserDAO are the corresponding DAO which provides connection to the database. The data stored in the database can be viewed via PHPMyAdmin.

## Part A - Login

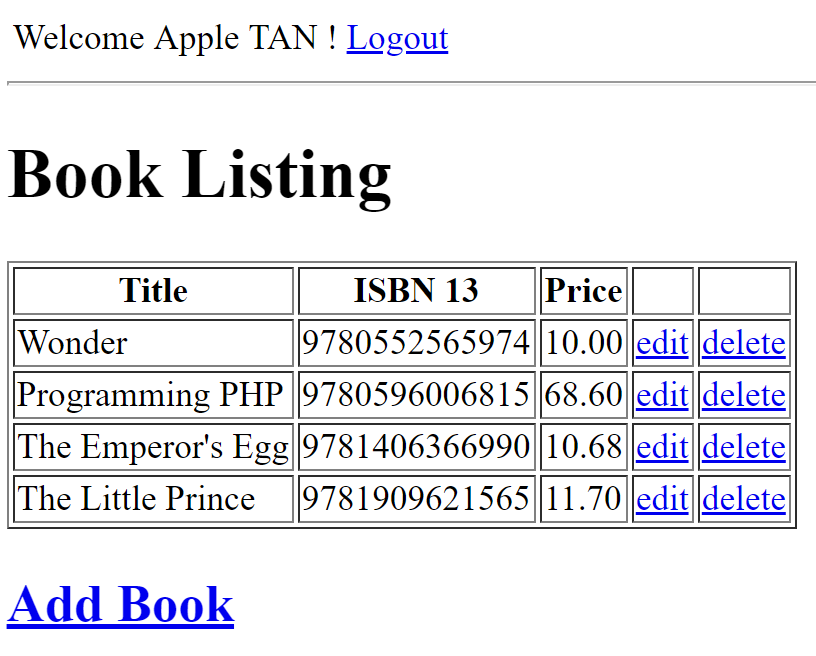
Complete authenticate method of User.php. The method needs to check if the password entered by the user matches the hashed password stored the object property, $this-> hashed\_password. After you have completed the method implementation, you would be able to login with username = apple, and password = apple123.



**login.php**

**Part B – List Books**

After successful authentication, complete list of books will be displayed via list.php. Complete list.php to make use of BookDAO to retrieve all books and print the details of each book in a table. The web browser will display the following.



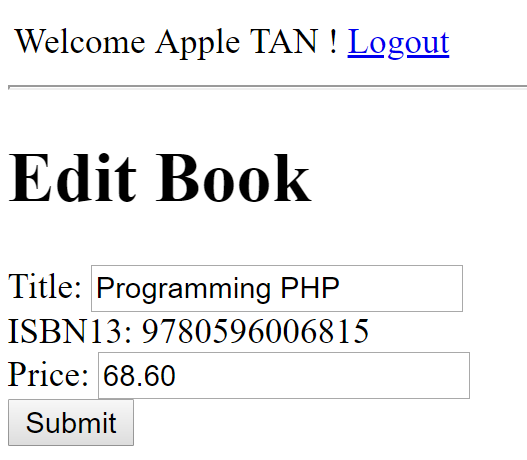
**list.php**

**In the** last two columns of each row in the table contains hyperlinks to edit.php and process\_delete.php. When one of these hyperlinks is clicked the **isbn13** number of the corresponding book should be passed to the next page using HTTP GET (e.g., edit.php?isbn13=9780552565974).

Below the table is a hyperlink to add.php, which will allow the user to create a new book.

**Part C– Edit a Book**

After one of the edit hyperlinks is clicked, the page will be redirected to edit.php. User can edit the title and price of the corresponding book.

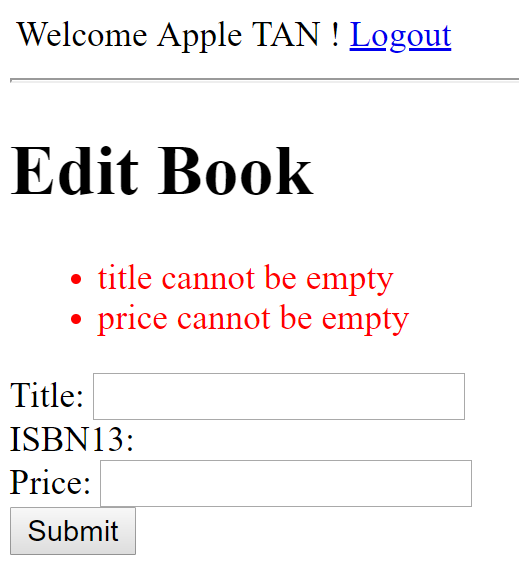


**edit.php**

Upon clicking **Submit** button, data would be sent to process\_edit.php. Implement the two methods in this PHP file:

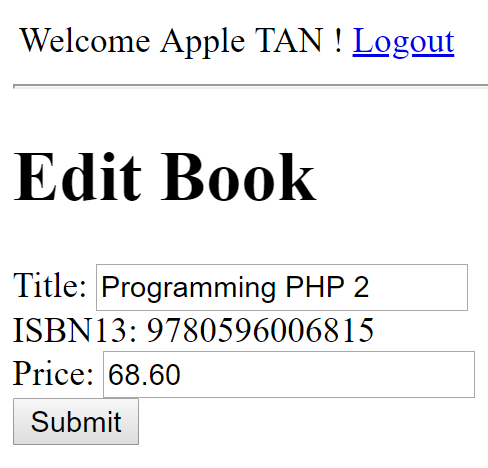
* validate\_inputs checks whether the inputted title, ISBN13 and price are empty strings. If any of them is an empty string, create a suitable error message. Use the isMissingOrEmpty method defined in common.php. Put the error messages as an array into $\_SESSION["errors"]. If any error occurs, redirect to edit.php passing in the inputted title, isbn13, and price using HTTP GET (i.e., by appending the data to the end of the URL). Otherwise, do nothing.
* perform\_edit creates a Book object with the updated details and calls a suitable method from BookDAO to modify the database. Redirect to list.php to display the updated list of books.

*Example 1*: Empty title and price.

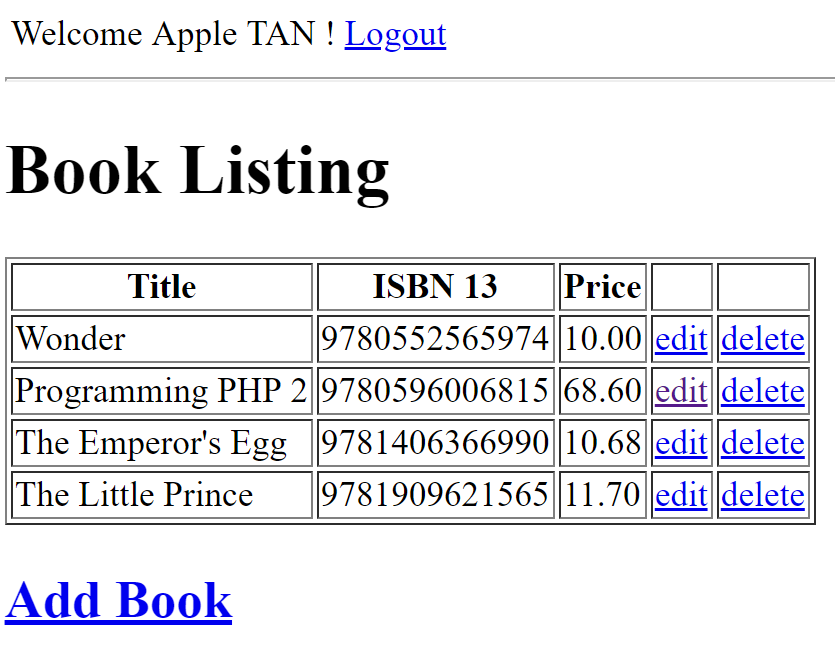


**edit.php**

*Example 2:* Success case. Change book title to "Programming PHP 2".



**edit.php**

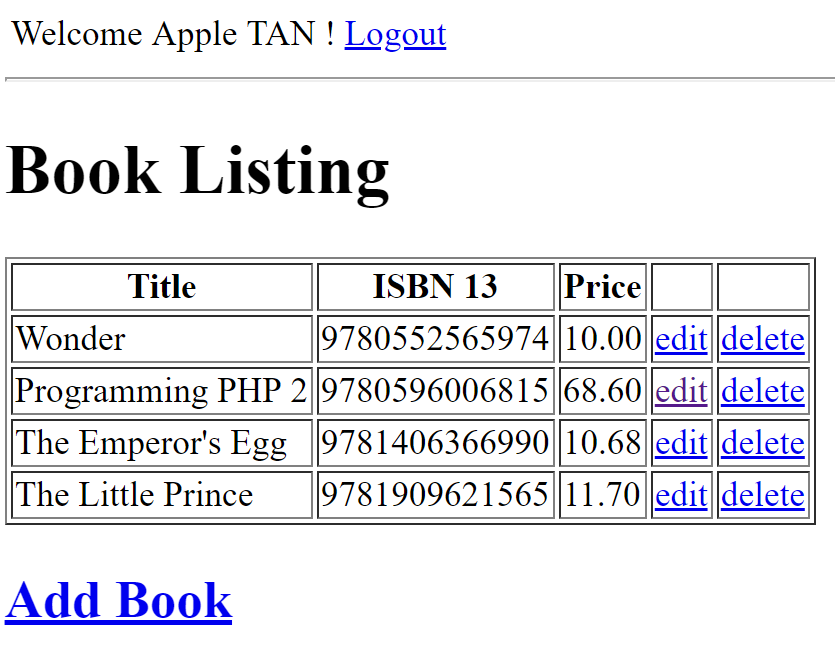


**list.php**

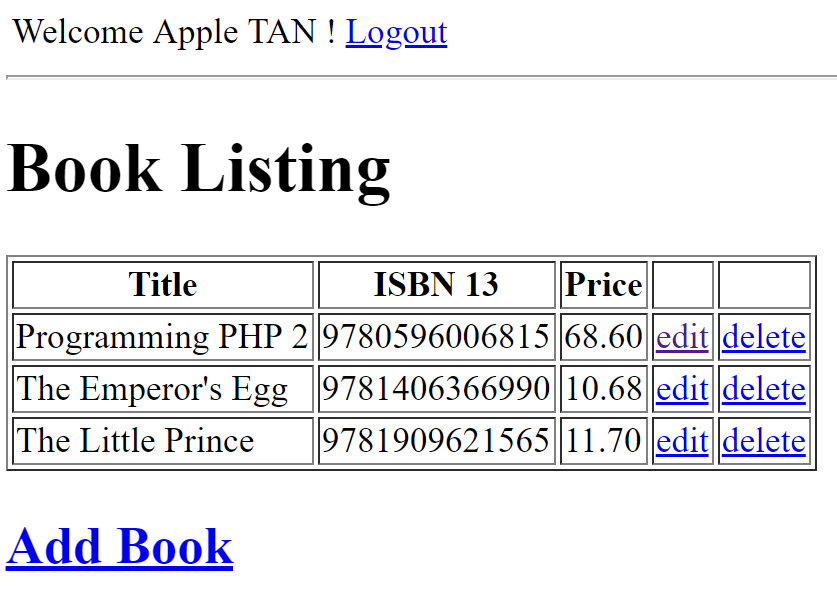
**Part D – Delete a Book**

Complete the remove method of BookDAO. After completing the code successfully, clicking one of the delete hyperlinks in list.php will result in the deletion of the corresponding book from the database.

*Example*: Delete first book from the list.



**list.php (before deletion)**

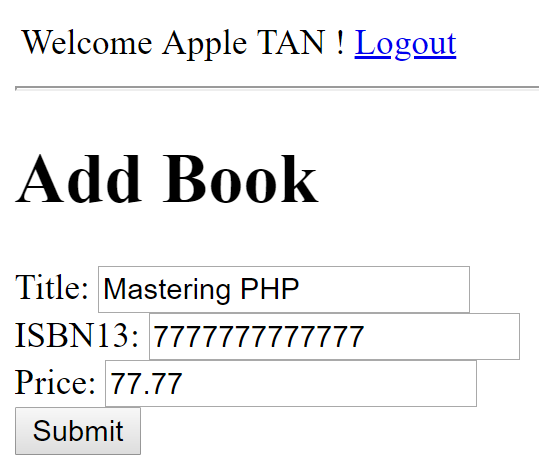


**list.php (after deletion)**

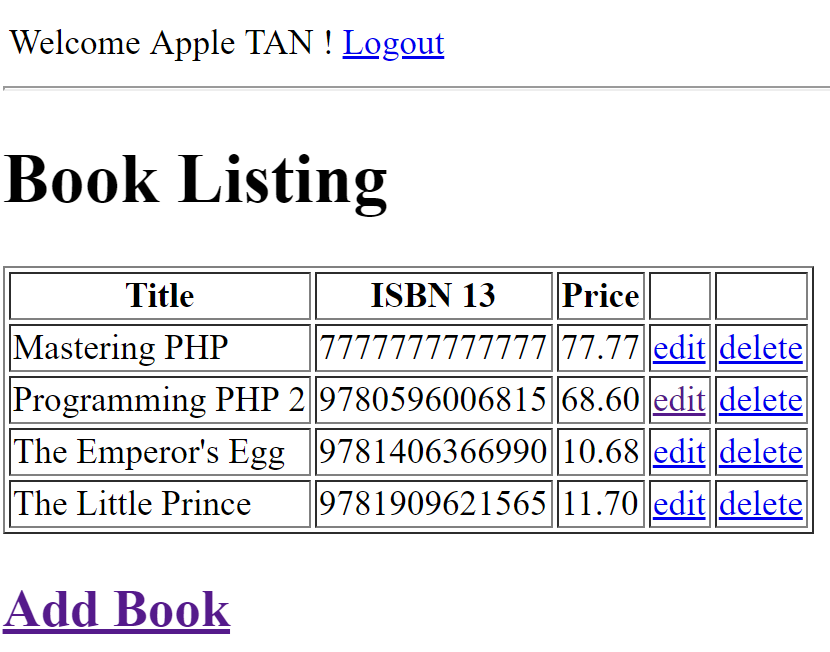
**Part E – Add a Book**

Complete the add method of BookDAO. After completing the code successfully, the add book functionality would work.

*Example*: Adding a new book



**add.php**



**list.php (after addition of new book)**