# ‘Book Reservation System project’

OLLSCOIL TEICNEOLAÍOCHTA
BHAILE ÁTHA CLIATH

TU DUBLIN

TECHNOLOGICAL
UNIVERSITY DUBLIN

Name: ___Polina__Tiukavkina_________

Student number: ____D22124948________

Lecturer’s Name: ____Cindy Liu_____________

Programme: ___ CMPU2022_____________

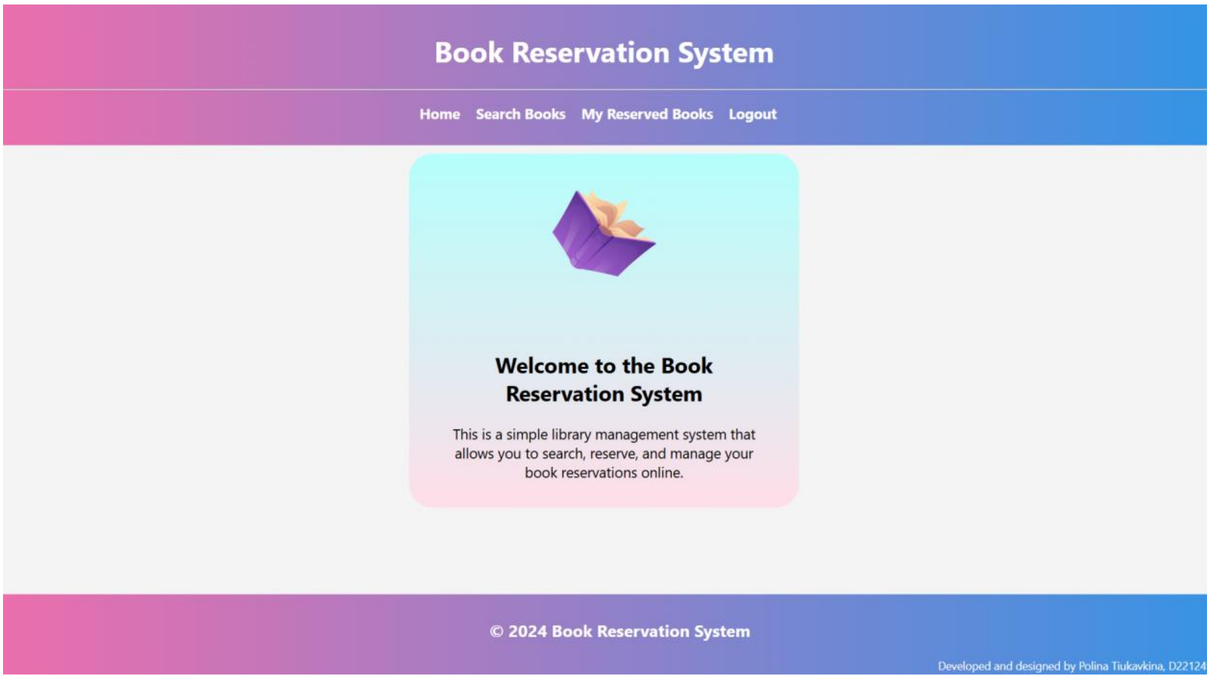
Module: ___‘Web Development’

Submission Date: ____06.12.2024_______________________

# Table of Contents

# Pages Description:

## Index.php



Book Reservation System

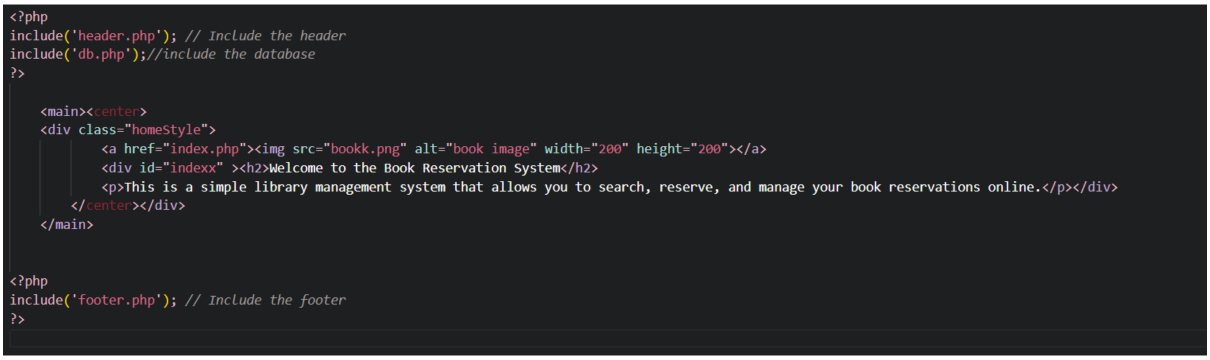
Welcome to the Book Reservation System

This is a simple library management system that allows you to search, reserve, and manage your book reservations online.

© 2024 Book Reservation System

Developed and designed by Polina Tiukavkina, D22124

The main (home) page of my website. Mostly descriptive to explain what functionality the website provides.

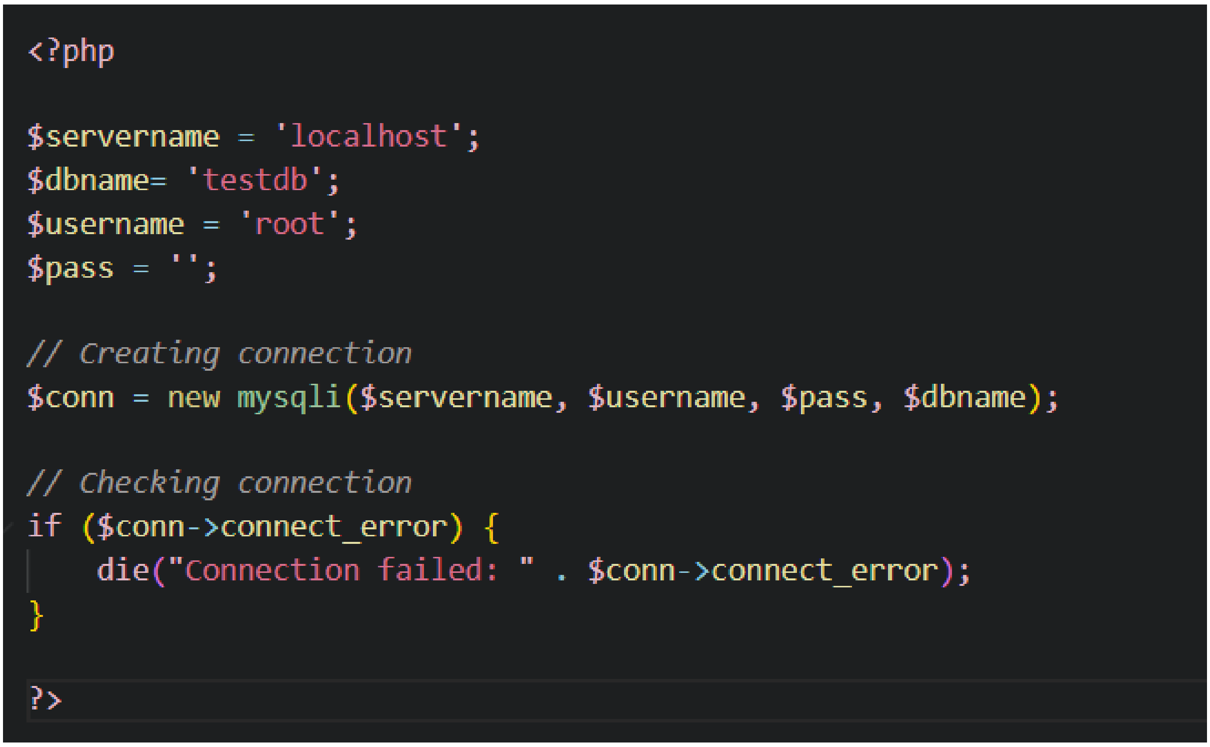


```
<?php
include('header.php'); // Include the header
include('db.php');//include the database
?>

    <main><center>
    <div class="homeStyle">
            <a href="index.php"><img src="bookk.png" alt="book image" width="200" height="200"></a>
            <div id="indexx" ><h2>Welcome to the Book Reservation System</h2>
            <p>This is a simple library management system that allows you to search, reserve, and manage your book reservations online.</p></div>
        </center></div>
    </main>


<?php
include('footer.php'); // Include the footer
?>
```

## db.php

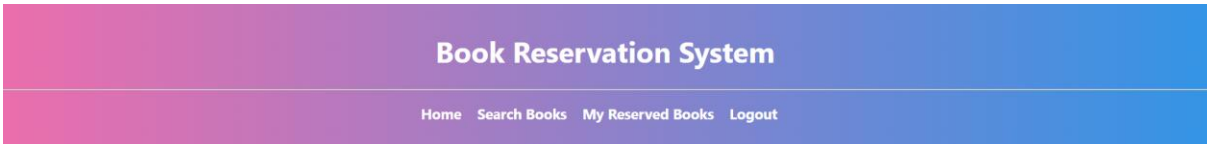
db.php handles connection between database on PHPmyAdmin and PHP files.

I included db.php in every PHP file to be able to retrieve objects from my database.

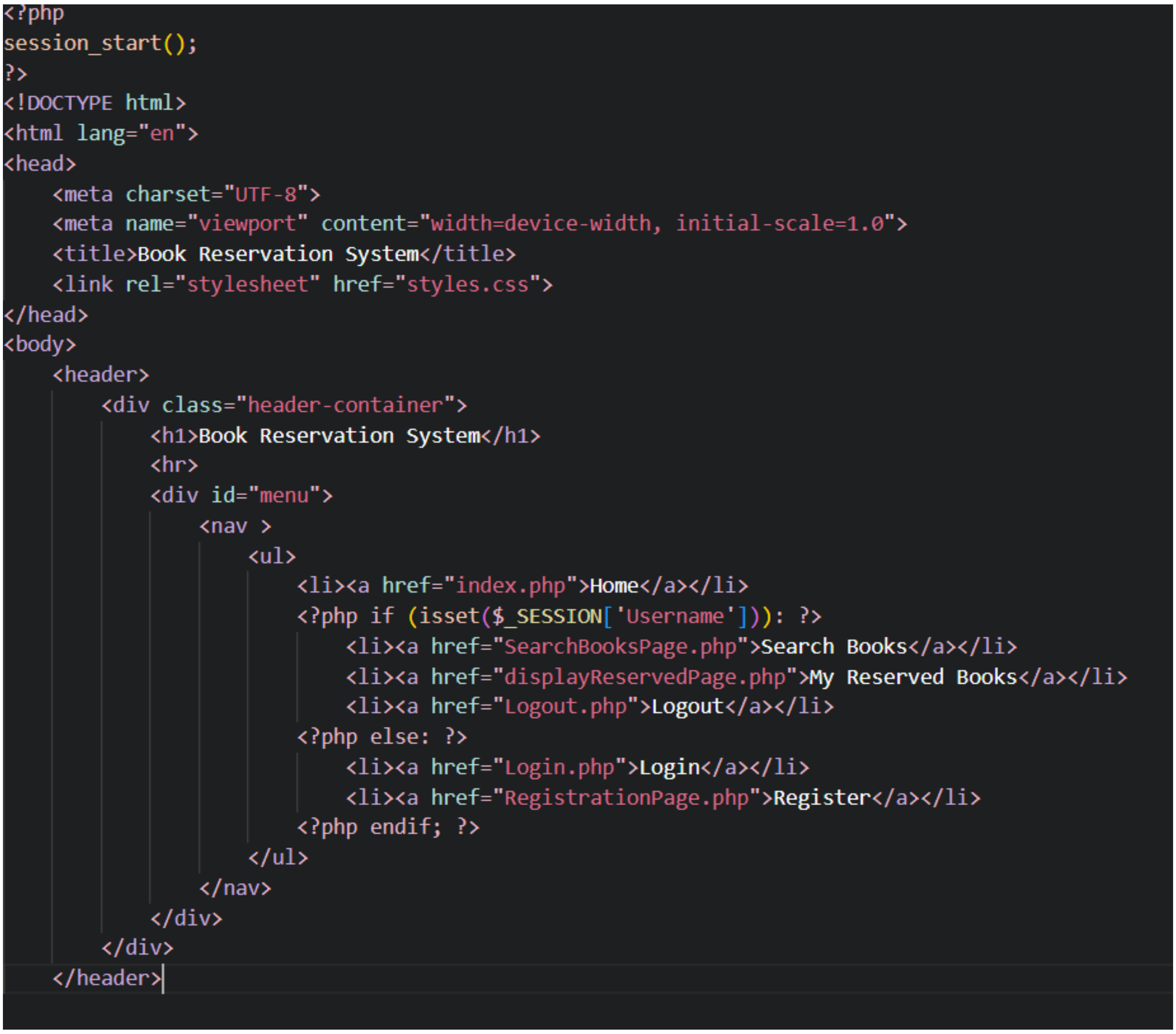


```php
<?php

$servername = 'localhost';
$dbname= 'testdb';
$username = 'root';
$pass = '';

// Creating connection
$conn = new mysqli($servername, $username, $pass, $dbname);

// Checking connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

?>
```

## header.php

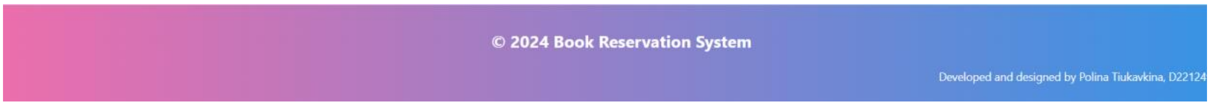


Header.php is the top section of the page and it contains links to other pages (using nav) which helps navigate through the website. It also has session_start() function to create a new session. It helps to not rewrite session_start() as header.php is included on every page. (**include('header.php'**);) It also uses php to see if the user is not logged in(if username is set or not in this session),and if not, links to Search Books page, Search Books and My Reserved Books won’t appear unless the user logs in.



```
<?php
session_start();
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Book Reservation System</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <header>
        <div class="header-container">
            <h1>Book Reservation System</h1>
            <hr>
            <div id="menu">
                <nav >
                    <ul>
                        <li><a href="index.php">Home</a></li>
                        <?php if (isset($_SESSION['Username'])): ?>
                            <li><a href="SearchBooksPage.php">Search Books</a></li>
                            <li><a href="displayReservedPage.php">My Reserved Books</a></li>
                            <li><a href="Logout.php">Logout</a></li>
                        <?php else: ?>
                            <li><a href="Login.php">Login</a></li>
                            <li><a href="RegistrationPage.php">Register</a></li>
                        <?php endif; ?>
                    </ul>
                </nav>
            </div>
        </div>
    </header>
```

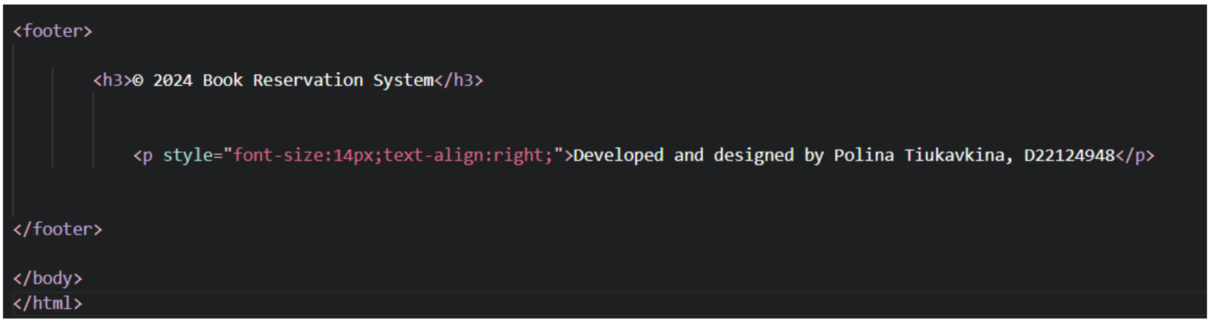
## footer.php



© 2024 Book Reservation System

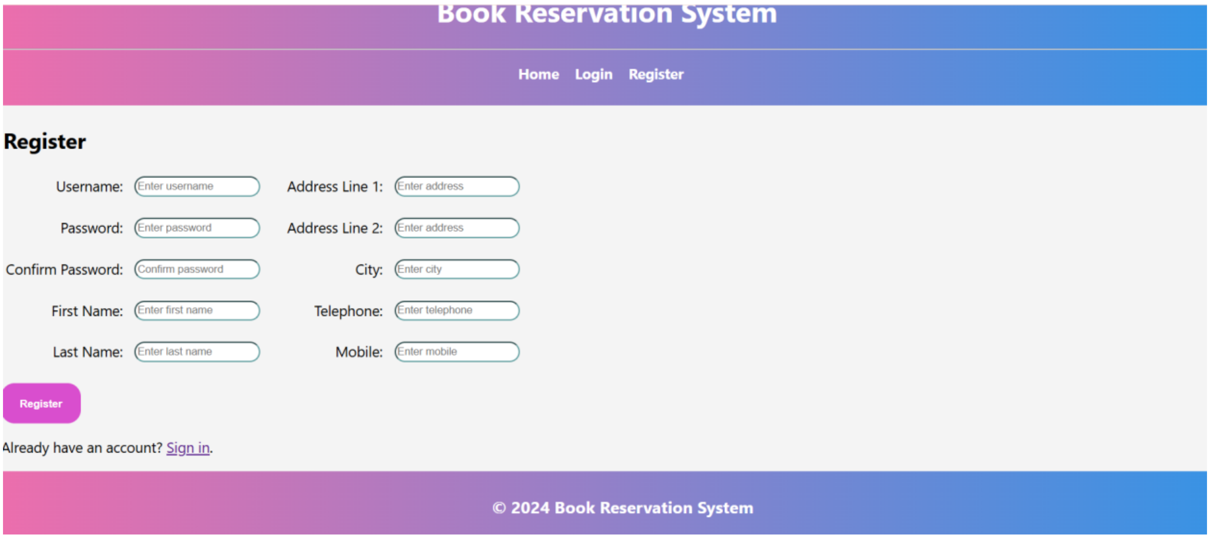
Developed and designed by Polina Tiukavkina, D22124

footer.php contains additional information which is just text content on the bottom of the page. Like header.php, it is also included on every page. We have body and html closing tags at the bottom of the code so we don’t need to use them on other pages as well. I also used inline CSS to make text neater.



```
<footer>

        <h3>© 2024 Book Reservation System</h3>

            <p style="font-size:14px;text-align:right;">Developed and designed by Polina Tiukavkina, D22124948</p>

</footer>

</body>
</html>
```

## RegistrationPage.php



Register

| | | | |
|---|---|---|---|
| Username: | Enter username | Address Line 1: | Enter address |
| Password: | Enter password | Address Line 2: | Enter address |
| Confirm Password: | Confirm password | City: | Enter city |
| First Name: | Enter first name | Telephone: | Enter telephone |
| Last Name: | Enter last name | Mobile: | Enter mobile |

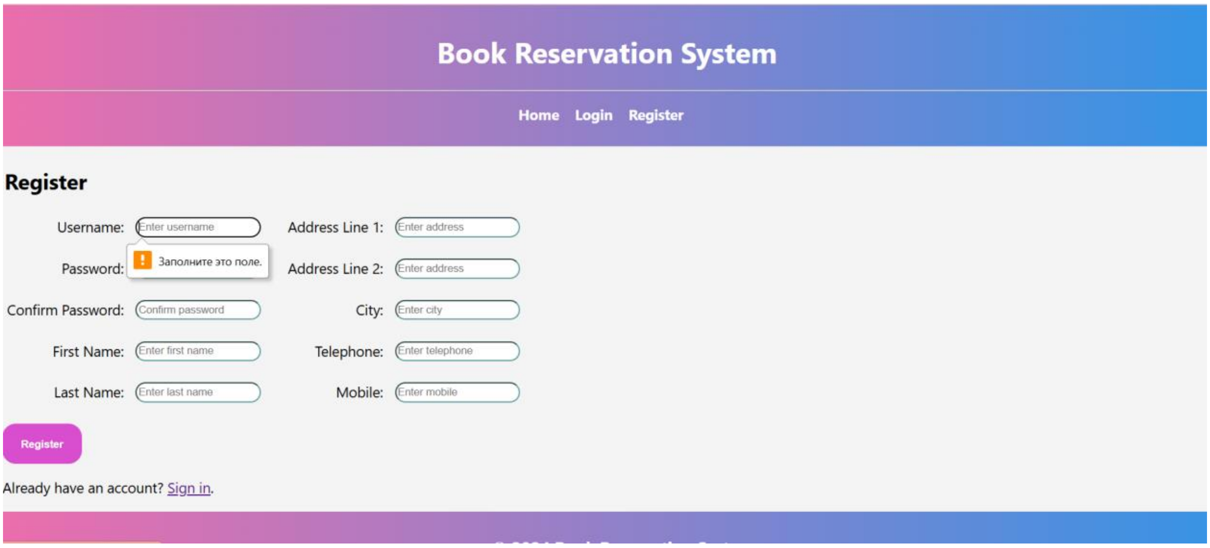
Already have an account? Sign in.

© 2024 Book Reservation System

Registration Page allows a person to register if they don't have an account yet. It provides registration information for person  to fill in. After going through input validation, the database will be updated with new values (table called users) This form also handles input validation: 1)ensures that user fills in all fields(required), 2) user does not enter a username that already exists on database(by retrieving usernames from table users and checking),3)That password is 6 character long and matches with confirmation password,4)That telephone and mobile are numeric and 10 digits long.

The registration page interacts with other pages by creating a new user record in the database. It also has a link to login page if the user already has an account.

## If user doesn't fill in anything and clicks register button:



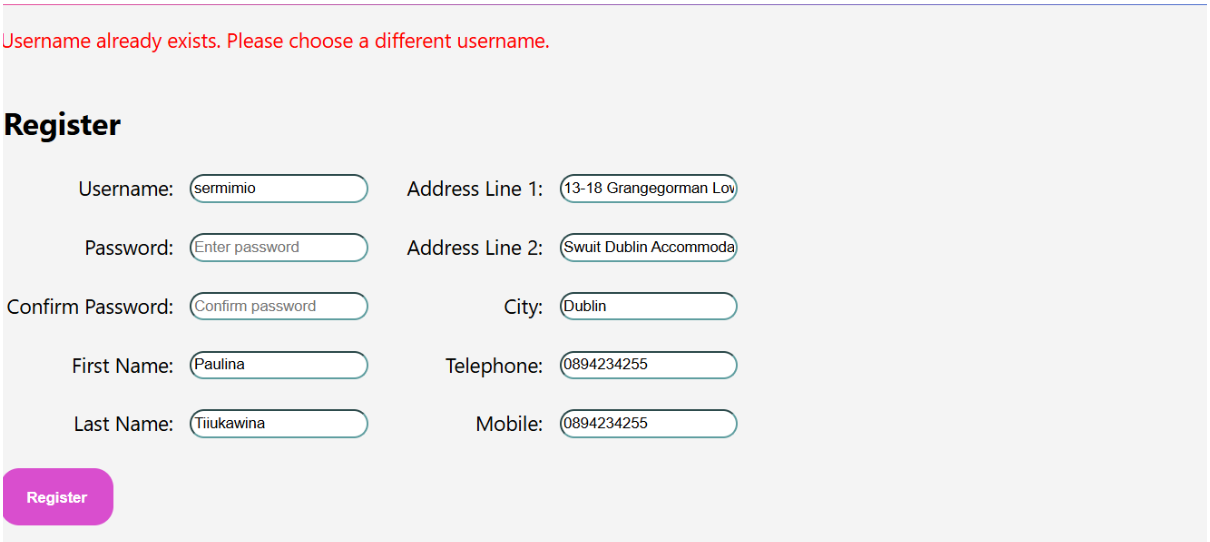
Register

| | | | |
|---|---|---|---|
| Username: | Enter username | Address Line 1: | Enter address |
| Password: | | Address Line 2: | Enter address |
| Confirm Password: | Confirm password | City: | Enter city |
| First Name: | Enter first name | Telephone: | Enter telephone |
| Last Name: | Enter last name | Mobile: | Enter mobile |

! Заполните это поле.

Already have an account? Sign in.

## After user enters a username which is already in database:

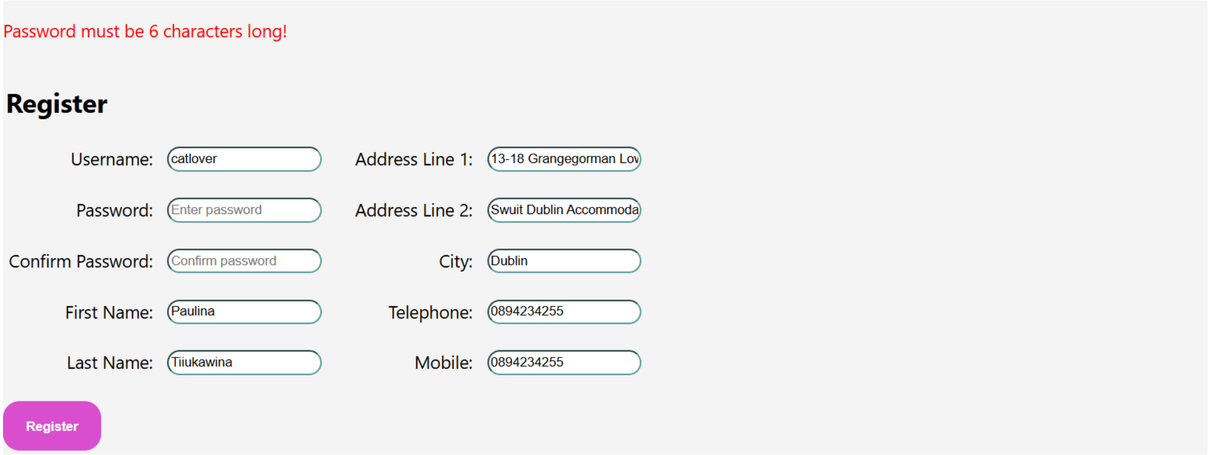


Username already exists. Please choose a different username.

**Register**

| | | | |
|---|---|---|---|
| Username: | sermimio | Address Line 1: | 13-18 Grangegorman Lov |
| Password: | Enter password | Address Line 2: | Swuit Dublin Accommoda |
| Confirm Password: | Confirm password | City: | Dublin |
| First Name: | Paulina | Telephone: | 0894234255 |
| Last Name: | Tiiukawina | Mobile: | 0894234255 |

## After user enter password that is not 6 characters long:

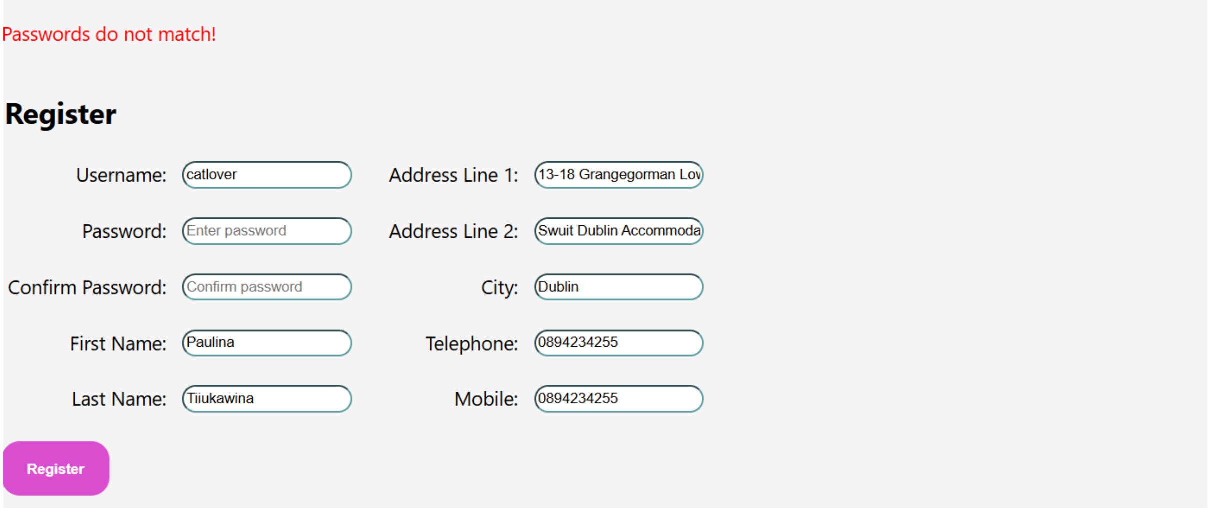


Password must be 6 characters long!

**Register**

| | | | |
|---|---|---|---|
| Username: | catlover | Address Line 1: | 13-18 Grangegorman Lov |
| Password: | Enter password | Address Line 2: | Swuit Dublin Accommoda |
| Confirm Password: | Confirm password | City: | Dublin |
| First Name: | Paulina | Telephone: | 0894234255 |
| Last Name: | Tiiukawina | Mobile: | 0894234255 |

## After user enters confirmation password and it doesn’t match with password:

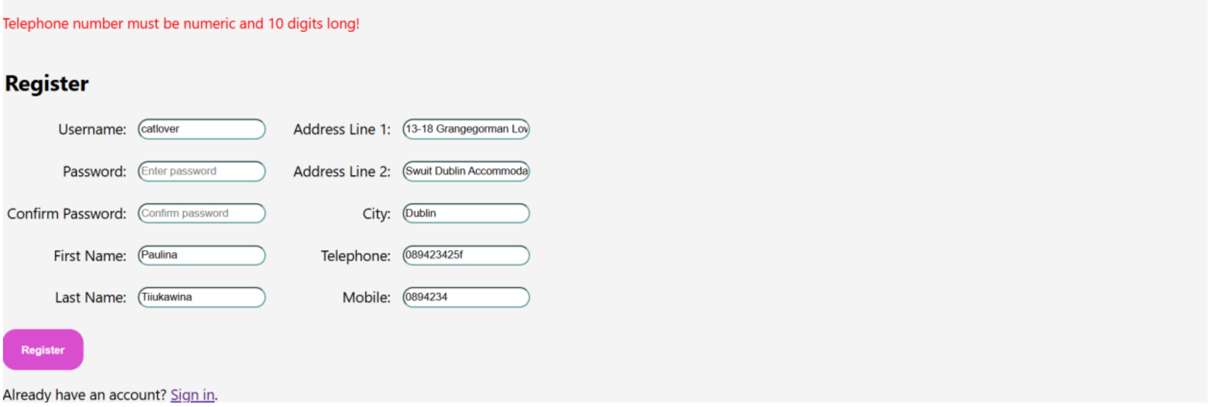


Passwords do not match!

**Register**

| | | | |
|---|---|---|---|
| Username: | catlover | Address Line 1: | 13-18 Grangegorman Lov |
| Password: | Enter password | Address Line 2: | Swuit Dublin Accommoda |
| Confirm Password: | Confirm password | City: | Dublin |
| First Name: | Paulina | Telephone: | 0894234255 |
| Last Name: | Tiiukawina | Mobile: | 0894234255 |

## After user enters telephone, mobile which is not 10 digits long and not numeric:



Telephone number must be numeric and 10 digits long!

### Register

Username: catlover

Password: Enter password

Confirm Password: Confirm password

First Name: Paulina

Last Name: Tiiukawina

Address Line 1: 13-18 Grangegorman Lov

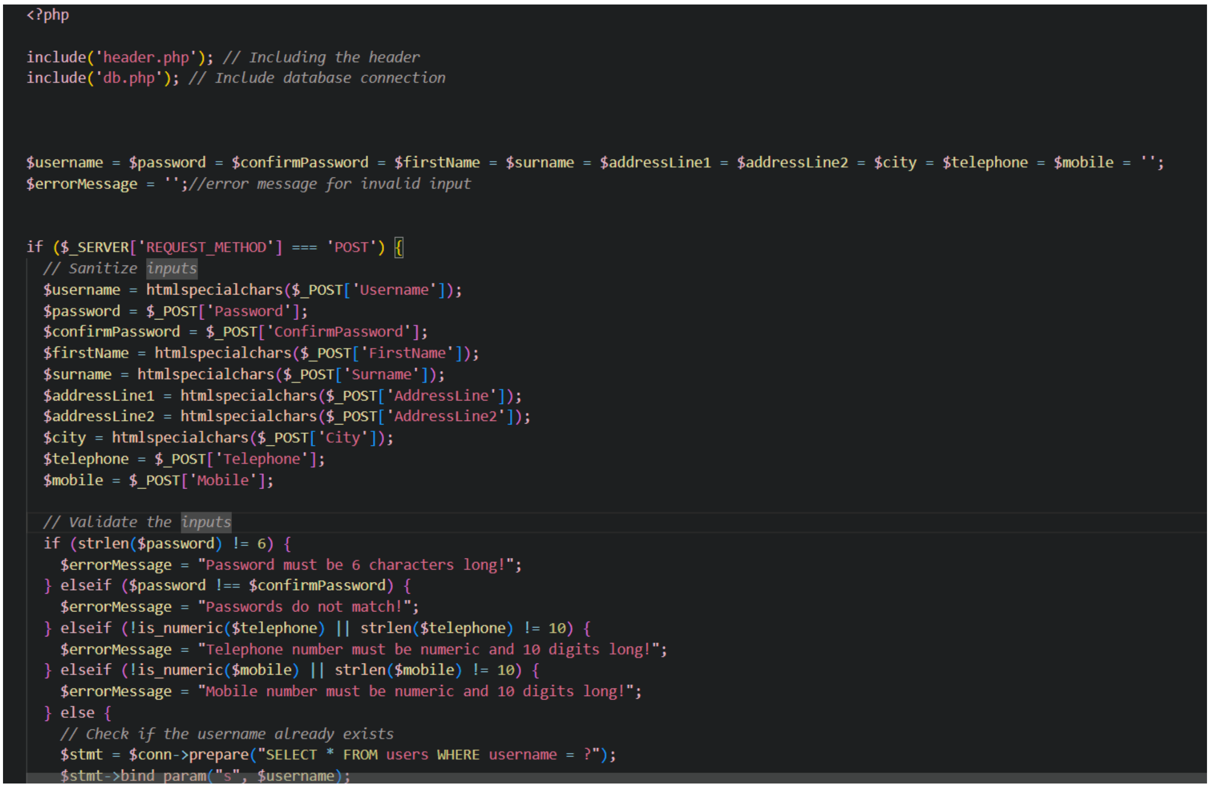
Address Line 2: Swuit Dublin Accommoda

City: Dublin

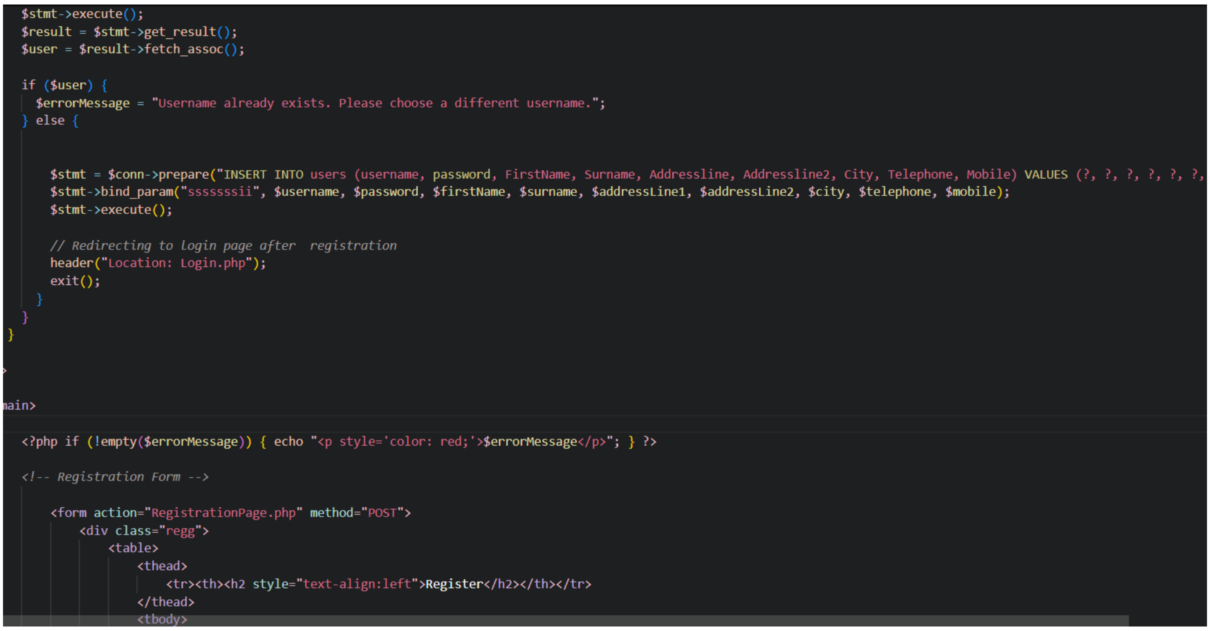
Telephone: 089423425f

Mobile: 0894234

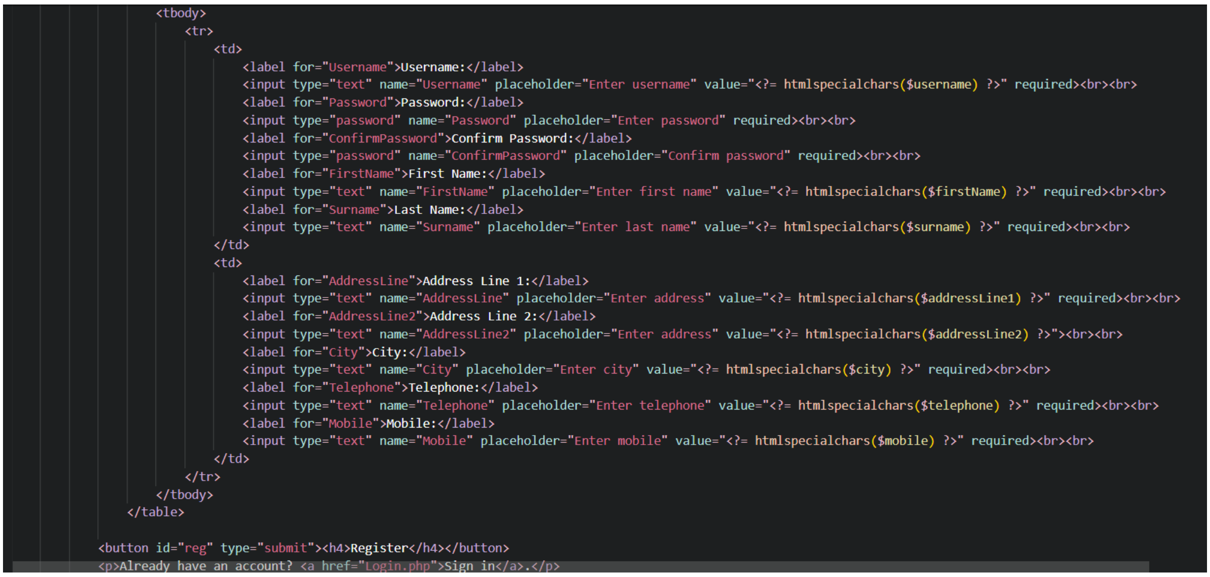
Already have an account? Sign in.



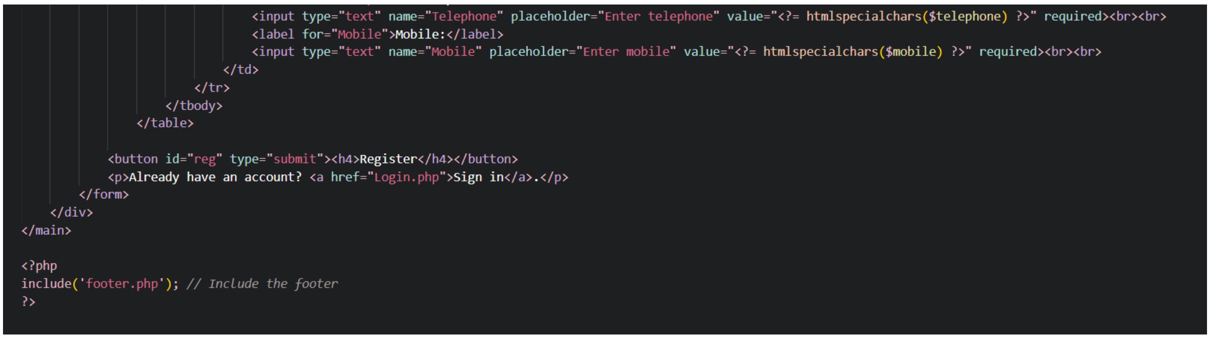
```php
<?php

include('header.php'); // Including the header
include('db.php'); // Include database connection



$username = $password = $confirmPassword = $firstName = $surname = $addressLine1 = $addressLine2 = $city = $telephone = $mobile = '';
$errorMessage = '';//error message for invalid input


if ($_SERVER['REQUEST_METHOD'] === 'POST') {
  // Sanitize inputs
  $username = htmlspecialchars($_POST['Username']);
  $password = $_POST['Password'];
  $confirmPassword = $_POST['ConfirmPassword'];
  $firstName = htmlspecialchars($_POST['FirstName']);
  $surname = htmlspecialchars($_POST['Surname']);
  $addressLine1 = htmlspecialchars($_POST['AddressLine']);
  $addressLine2 = htmlspecialchars($_POST['AddressLine2']);
  $city = htmlspecialchars($_POST['City']);
  $telephone = $_POST['Telephone'];
  $mobile = $_POST['Mobile'];

  // Validate the inputs
  if (strlen($password) != 6) {
    $errorMessage = "Password must be 6 characters long!";
  } elseif ($password !== $confirmPassword) {
    $errorMessage = "Passwords do not match!";
  } elseif (!is_numeric($telephone) || strlen($telephone) != 10) {
    $errorMessage = "Telephone number must be numeric and 10 digits long!";
  } elseif (!is_numeric($mobile) || strlen($mobile) != 10) {
    $errorMessage = "Mobile number must be numeric and 10 digits long!";
  } else {
    // Check if the username already exists
    $stmt = $conn->prepare("SELECT * FROM users WHERE username = ?");
    $stmt->bind_param("s", $username);
```



```
  $stmt->execute();
  $result = $stmt->get_result();
  $user = $result->fetch_assoc();

  if ($user) {
    $errorMessage = "Username already exists. Please choose a different username.";
  } else {


      $stmt = $conn->prepare("INSERT INTO users (username, password, FirstName, Surname, Addressline, Addressline2, City, Telephone, Mobile) VALUES (?, ?, ?, ?, ?, ?,
      $stmt->bind_param("sssssssii", $username, $password, $firstName, $surname, $addressLine1, $addressLine2, $city, $telephone, $mobile);
      $stmt->execute();

      // Redirecting to login page after  registration
      header("Location: Login.php");
      exit();
    }
  }
}

>

ain>

  <?php if (!empty($errorMessage)) { echo "<p style='color: red;'>$errorMessage</p>"; } ?>

  <!-- Registration Form -->

      <form action="RegistrationPage.php" method="POST">
          <div class="regg">
              <table>
                  <thead>
                      <tr><th><h2 style="text-align:left">Register</h2></th></tr>
                  </thead>
                  <tbody>
```



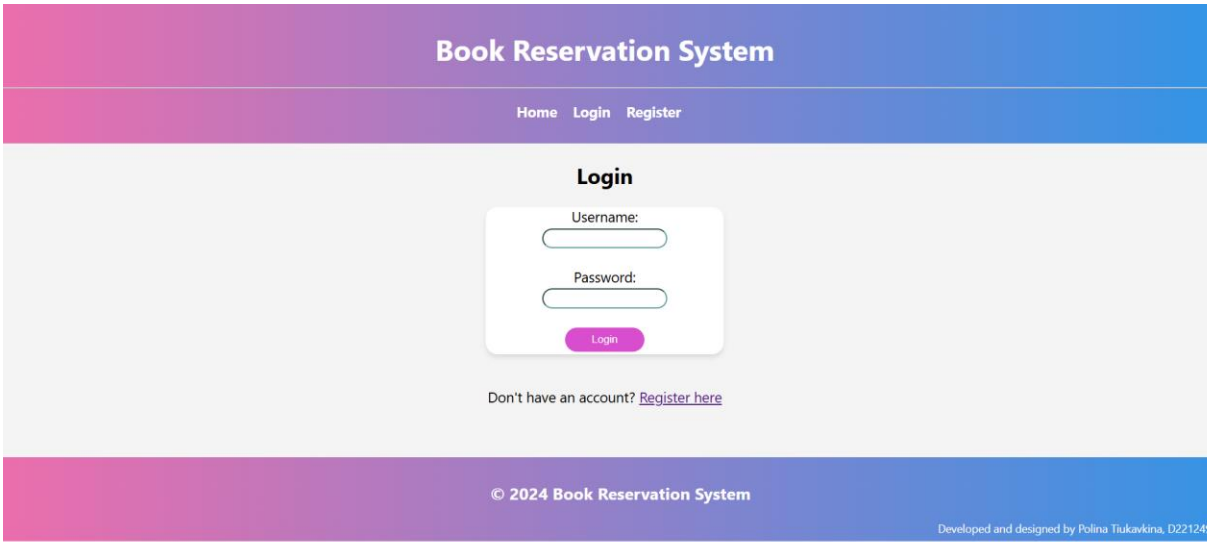
```
                    <tbody>
                        <tr>
                            <td>
                                <label for="Username">Username:</label>
                                <input type="text" name="Username" placeholder="Enter username" value="<?= htmlspecialchars($username) ?>" required><br><br>
                                <label for="Password">Password:</label>
                                <input type="password" name="Password" placeholder="Enter password" required><br><br>
                                <label for="ConfirmPassword">Confirm Password:</label>
                                <input type="password" name="ConfirmPassword" placeholder="Confirm password" required><br><br>
                                <label for="FirstName">First Name:</label>
                                <input type="text" name="FirstName" placeholder="Enter first name" value="<?= htmlspecialchars($firstName) ?>" required><br><br>
                                <label for="Surname">Last Name:</label>
                                <input type="text" name="Surname" placeholder="Enter last name" value="<?= htmlspecialchars($surname) ?>" required><br><br>
                            </td>
                            <td>
                                <label for="AddressLine">Address Line 1:</label>
                                <input type="text" name="AddressLine" placeholder="Enter address" value="<?= htmlspecialchars($addressLine1) ?>" required><br><br>
                                <label for="AddressLine2">Address Line 2:</label>
                                <input type="text" name="AddressLine2" placeholder="Enter address" value="<?= htmlspecialchars($addressLine2) ?>"><br><br>
                                <label for="City">City:</label>
                                <input type="text" name="City" placeholder="Enter city" value="<?= htmlspecialchars($city) ?>" required><br><br>
                                <label for="Telephone">Telephone:</label>
                                <input type="text" name="Telephone" placeholder="Enter telephone" value="<?= htmlspecialchars($telephone) ?>" required><br><br>
                                <label for="Mobile">Mobile:</label>
                                <input type="text" name="Mobile" placeholder="Enter mobile" value="<?= htmlspecialchars($mobile) ?>" required><br><br>
                            </td>
                        </tr>
                    </tbody>
                </table>

            <button id="reg" type="submit"><h4>Register</h4></button>
            <p>Already have an account? <a href="Login.php">Sign in</a>.</p>
```



```
                                <input type="text" name="Telephone" placeholder="Enter telephone" value="<?= htmlspecialchars($telephone) ?>" required><br><br>
                                <label for="Mobile">Mobile:</label>
                                <input type="text" name="Mobile" placeholder="Enter mobile" value="<?= htmlspecialchars($mobile) ?>" required><br><br>
                            </td>
                        </tr>
                    </tbody>
                </table>

            <button id="reg" type="submit"><h4>Register</h4></button>
            <p>Already have an account? <a href="Login.php">Sign in</a>.</p>
        </form>
    </div>
</main>

<?php
include('footer.php'); // Include the footer
?>
```

## Login.php

Login page collects user input for username and password. It searches for a user with that credentials in database(by fetching rows from database) and if it finds that record, the user will be successfully logged in and redirected to home page with access to reservation functionality. His username will be stored in that session to be used for reservation actions. If the user didn't provide right username or password, error message: Incorrect username or password will be displayed. Login page also has link to register page if the user doesn't have an account.



Book Reservation System

Login

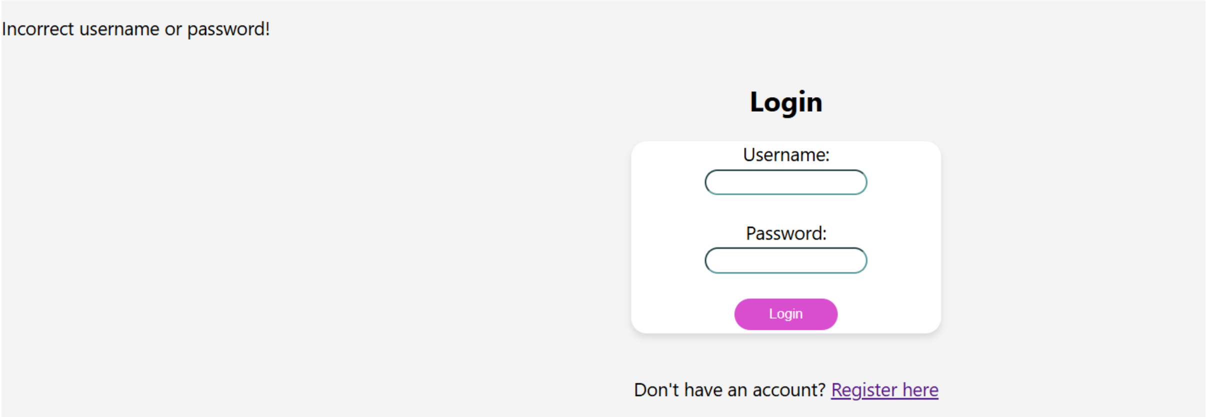
Username:

Password:

Don't have an account? Register here

© 2024 Book Reservation System

Developed and designed by Polina Tiukavkina, D22124



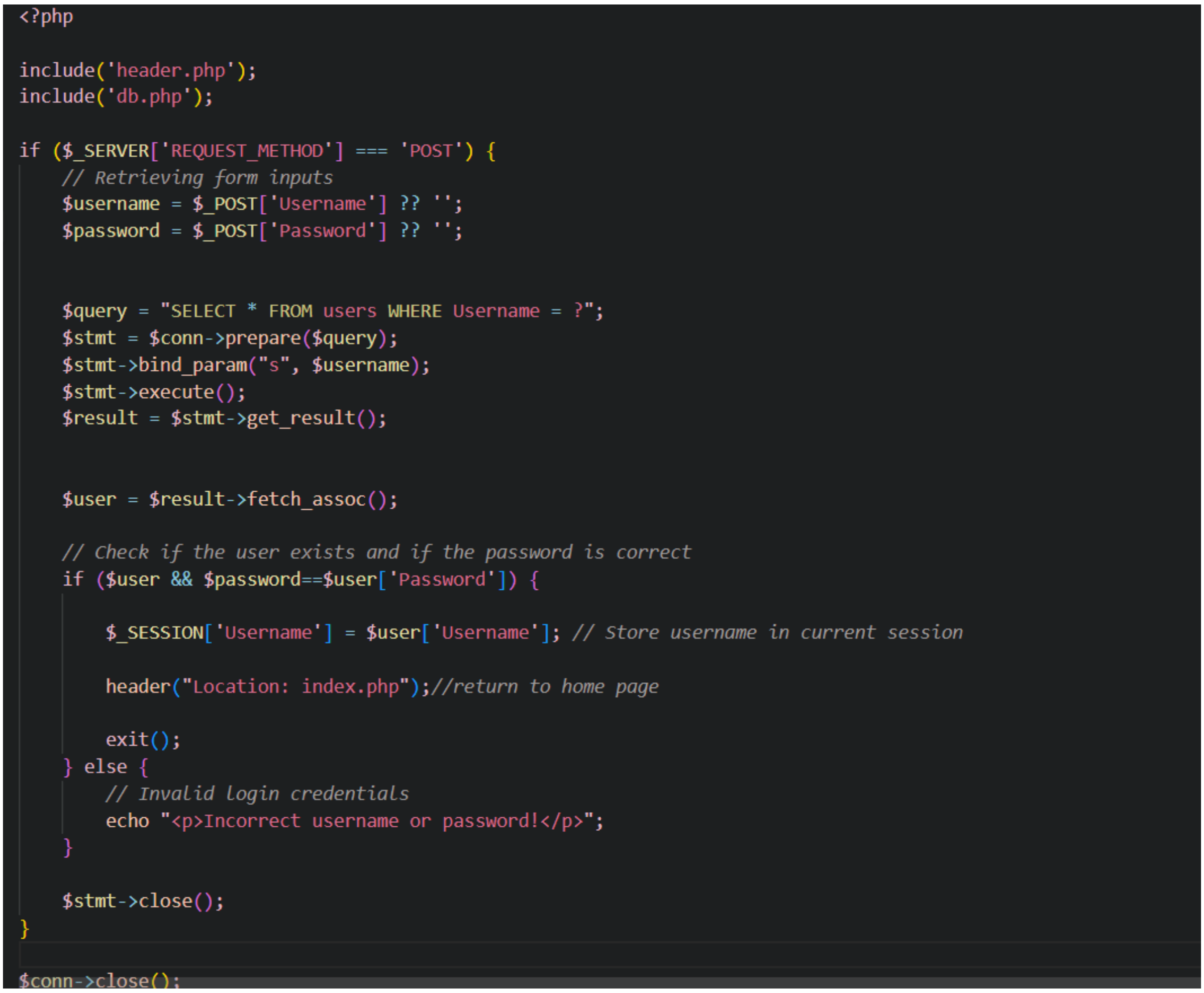
Incorrect username or password!

Login

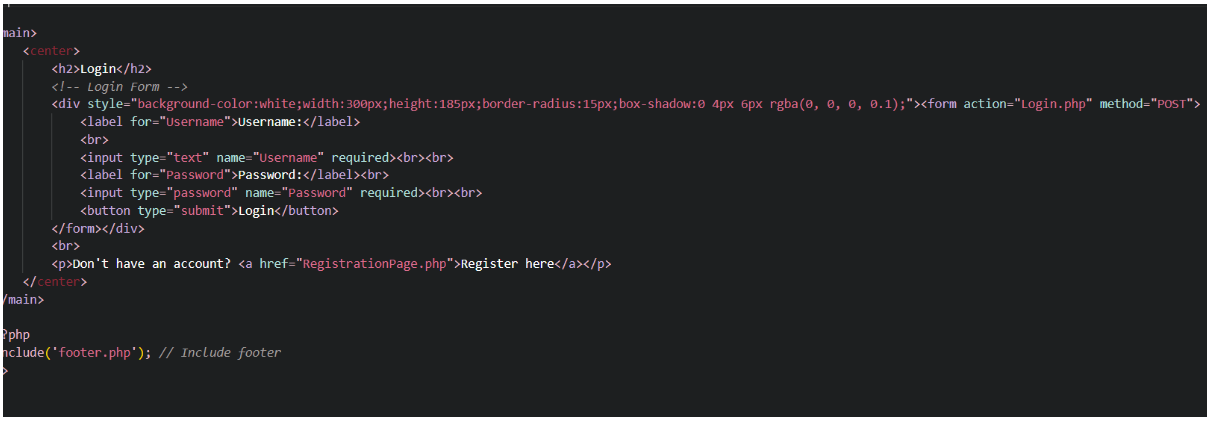
Username:

Password:

Don't have an account? Register here



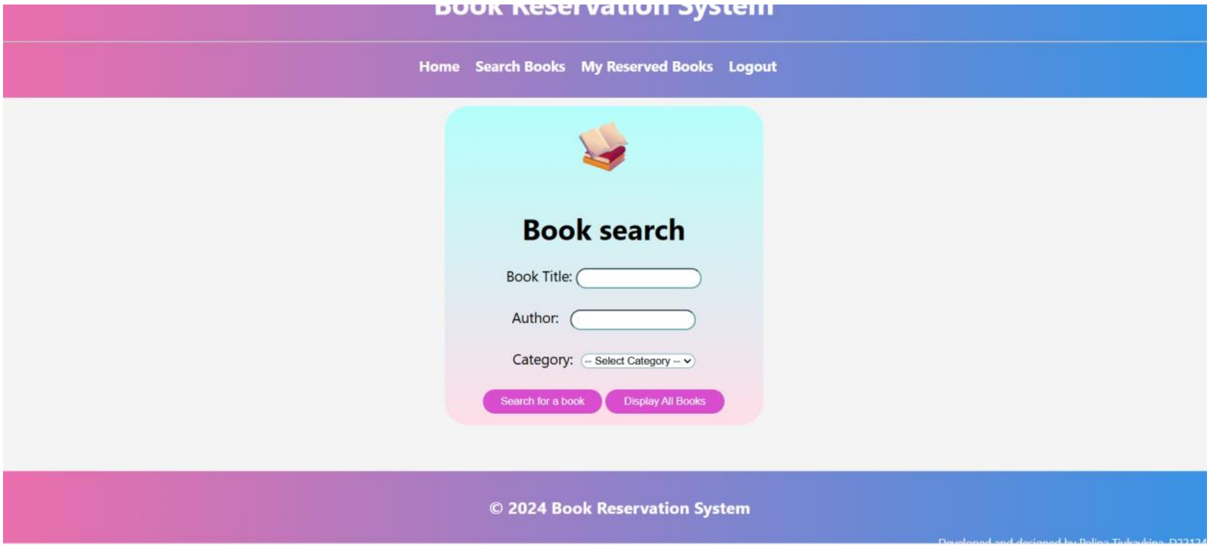
```
<?php

include('header.php');
include('db.php');

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Retrieving form inputs
    $username = $_POST['Username'] ?? '';
    $password = $_POST['Password'] ?? '';


    $query = "SELECT * FROM users WHERE Username = ?";
    $stmt = $conn->prepare($query);
    $stmt->bind_param("s", $username);
    $stmt->execute();
    $result = $stmt->get_result();


    $user = $result->fetch_assoc();

    // Check if the user exists and if the password is correct
    if ($user && $password==$user['Password']) {

        $_SESSION['Username'] = $user['Username']; // Store username in current session

        header("Location: index.php");//return to home page

        exit();
    } else {
        // Invalid login credentials
        echo "<p>Incorrect username or password!</p>";
    }

    $stmt->close();
}

$conn->close();
```



```
main>
   <center>
       <h2>Login</h2>
       <!-- Login Form -->
       <div style="background-color:white;width:300px;height:185px;border-radius:15px;box-shadow:0 4px 6px rgba(0, 0, 0, 0.1);"><form action="Login.php" method="POST">
           <label for="Username">Username:</label>
           <br>
           <input type="text" name="Username" required><br><br>
           <label for="Password">Password:</label><br>
           <input type="password" name="Password" required><br><br>
           <button type="submit">Login</button>
       </form></div>
       <br>
       <p>Don't have an account? <a href="RegistrationPage.php">Register here</a></p>
   </center>
/main>

?php
nclude('footer.php'); // Include footer
>
```

## SearchBooksPage.php

Search books Page allows logged in users to search for a book to reserve. We can search for a book by author, title and category. We can do research using one and more filters.  The results will be displayed page by page with a limit to 5 instances in one row (using pagination) When user fills in title/author/category they want to search for, it will be matched with values from database. Along with book information, user will be available to reserve the book if it’s not reserved by someone else yet. When user clicks button Reserve, the code inside ReservationPage will be executed. There’s also a button to display all books if the user wants to see all options.



Book Reservation System

Book search

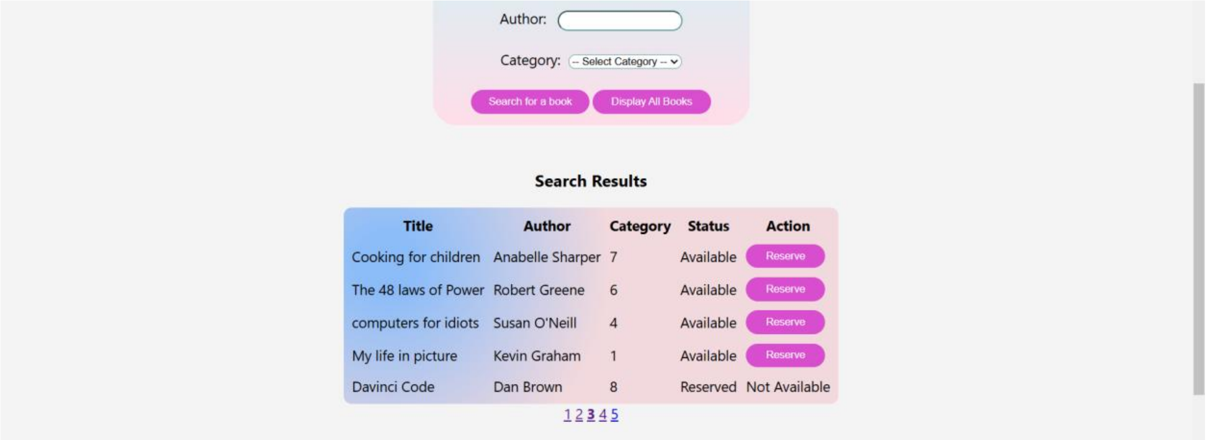
Book Title:

Author:

Category: -- Select Category --

© 2024 Book Reservation System

Developed and designed by Polina Tiskevkina D22124

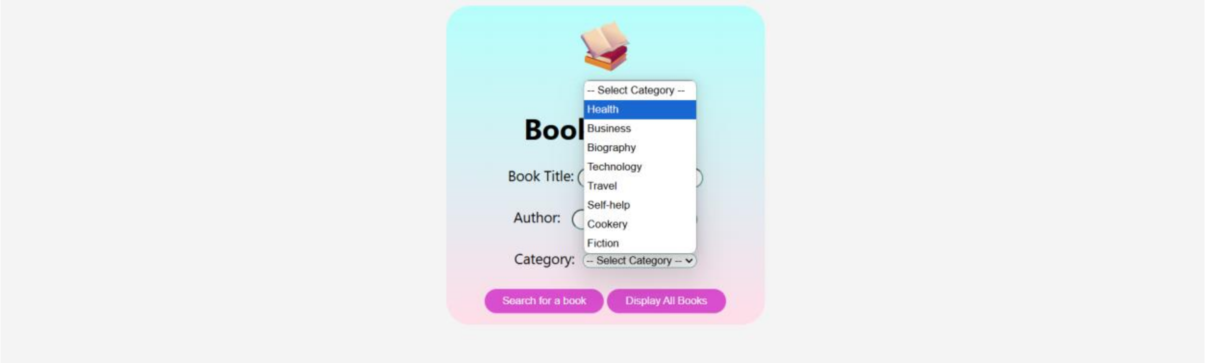


Author:

Category: -- Select Category --

Search Results

| Title | Author | Category | Status | Action |
|---|---|---|---|---|
| Cooking for children | Anabelle Sharper | 7 | Available | Reserve |
| The 48 laws of Power | Robert Greene | 6 | Available | Reserve |
| computers for idiots | Susan O'Neill | 4 | Available | Reserve |
| My life in picture | Kevin Graham | 1 | Available | Reserve |
| Davinci Code | Dan Brown | 8 | Reserved | Not Available |

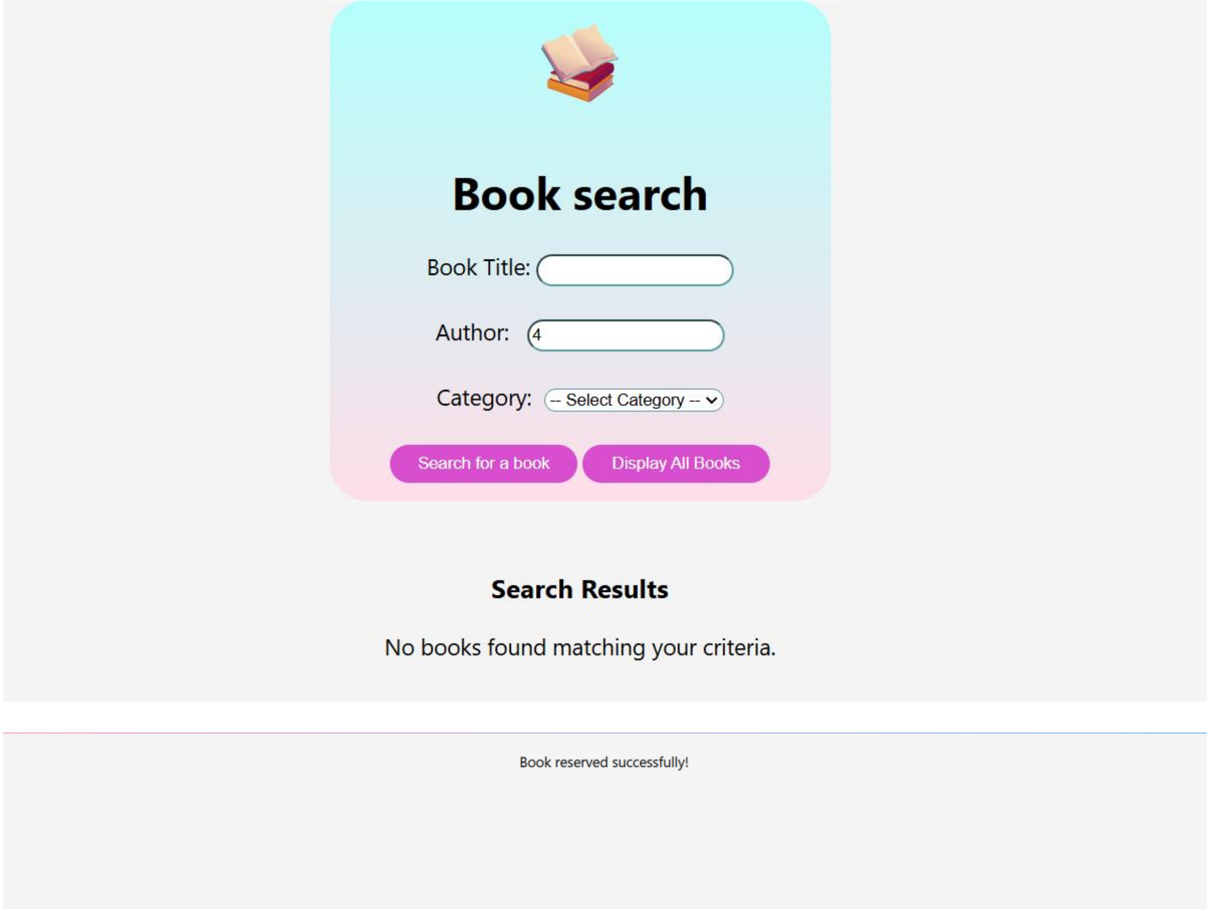


Book search

Book Title:

Author:

Category: -- Select Category --



Book search

Book Title:

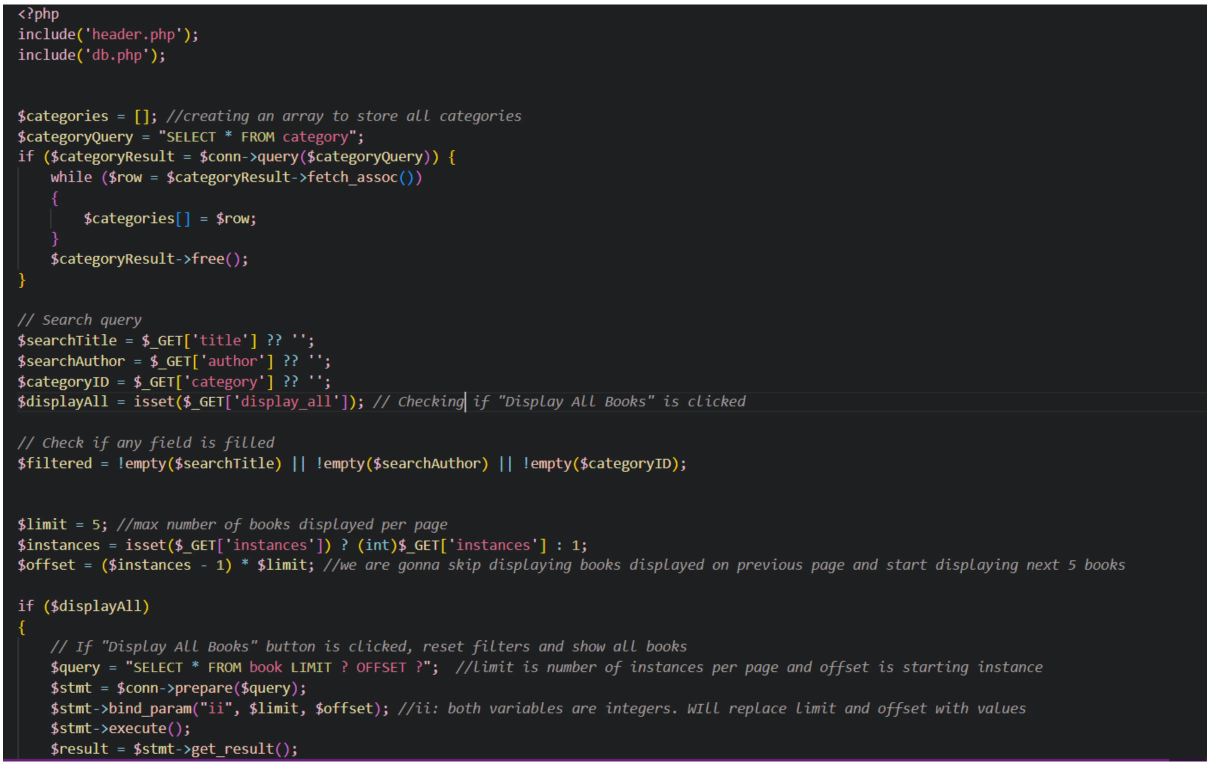
Author: 4

Category: -- Select Category --

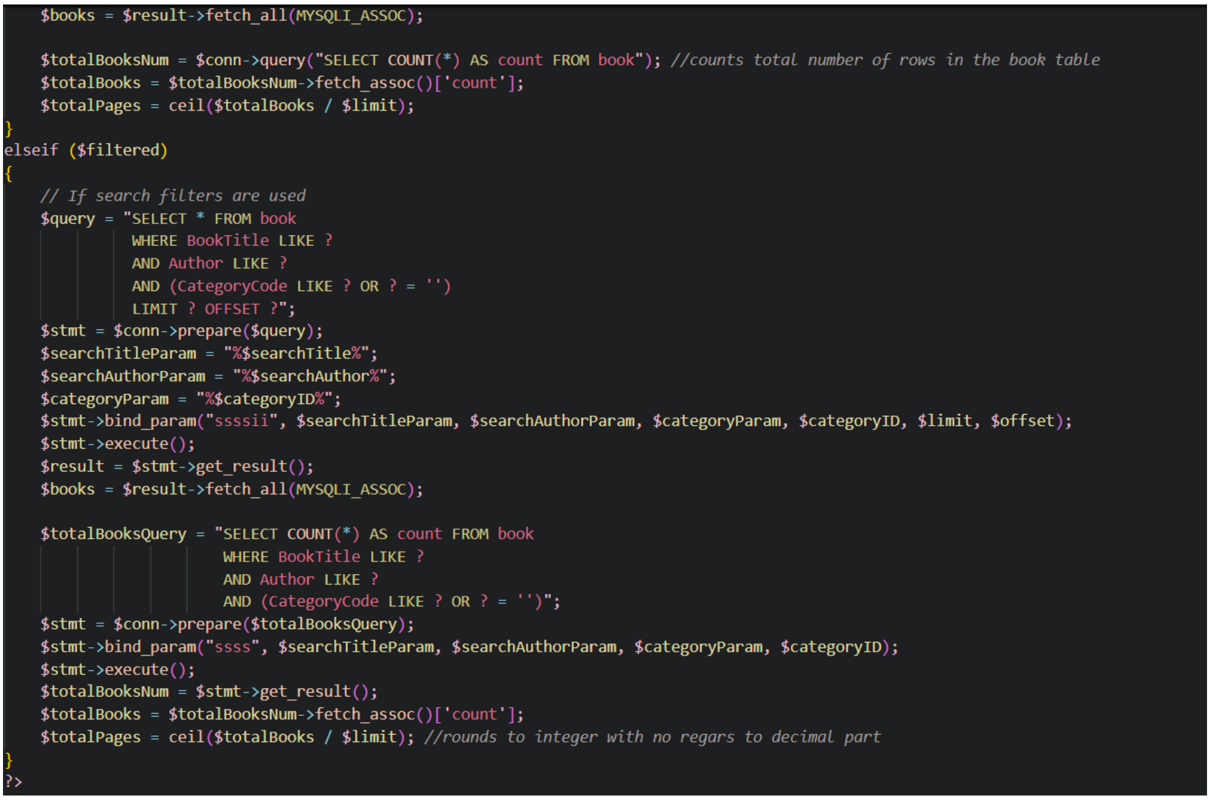
Search Results

No books found matching your criteria.

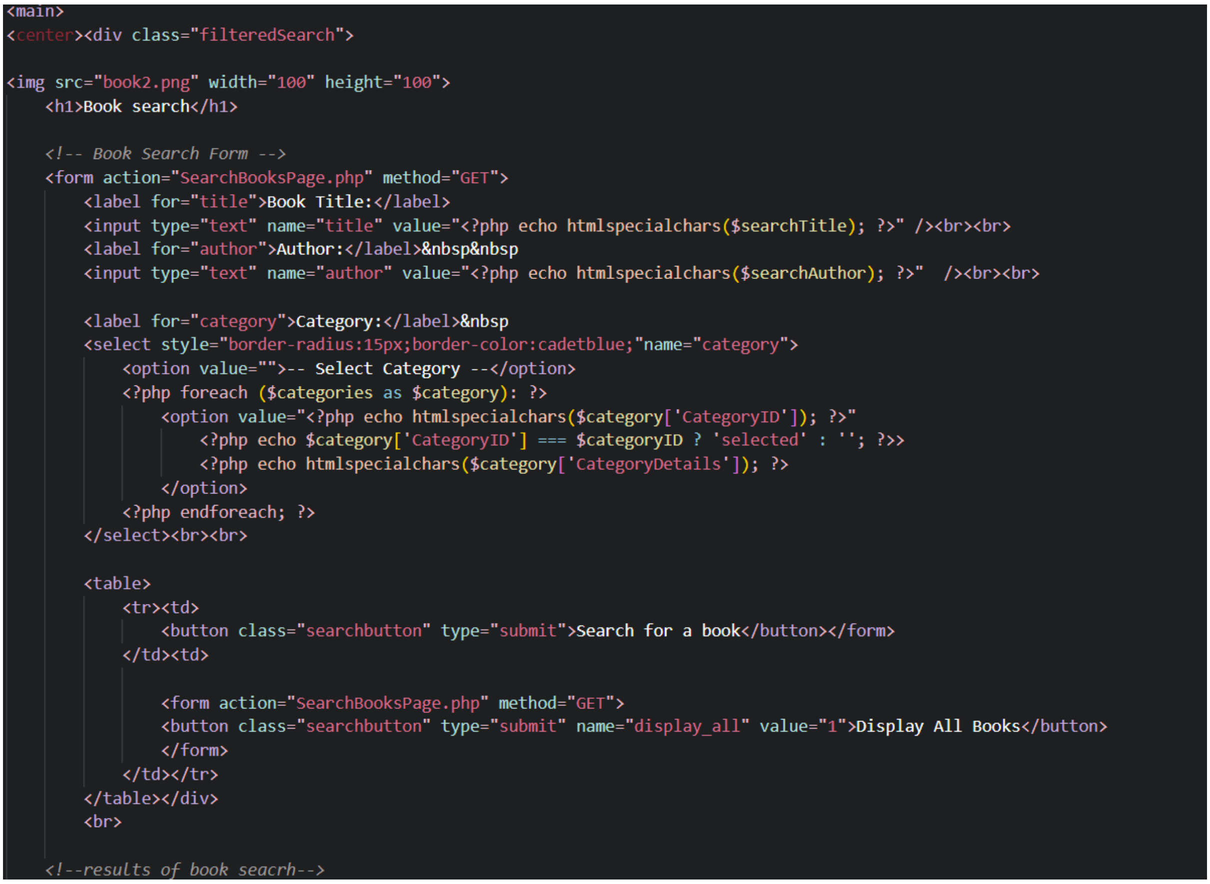
Book reserved successfully!



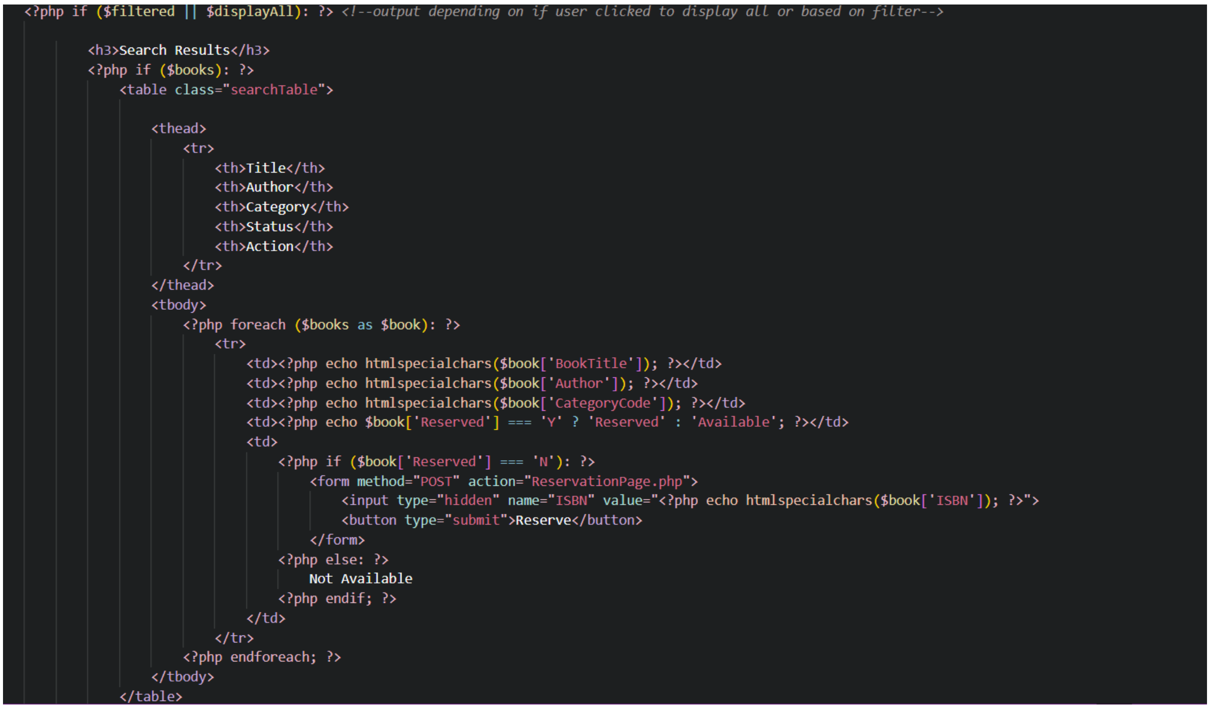
```php
<?php
include('header.php');
include('db.php');


$categories = []; //creating an array to store all categories
$categoryQuery = "SELECT * FROM category";
if ($categoryResult = $conn->query($categoryQuery)) {
    while ($row = $categoryResult->fetch_assoc())
    {
        $categories[] = $row;
    }
    $categoryResult->free();
}

// Search query
$searchTitle = $_GET['title'] ?? '';
$searchAuthor = $_GET['author'] ?? '';
$categoryID = $_GET['category'] ?? '';
$displayAll = isset($_GET['display_all']); // Checking if "Display All Books" is clicked

// Check if any field is filled
$filtered = !empty($searchTitle) || !empty($searchAuthor) || !empty($categoryID);


$limit = 5; //max number of books displayed per page
$instances = isset($_GET['instances']) ? (int)$_GET['instances'] : 1;
$offset = ($instances - 1) * $limit; //we are gonna skip displaying books displayed on previous page and start displaying next 5 books

if ($displayAll)
{
    // If "Display All Books" button is clicked, reset filters and show all books
    $query = "SELECT * FROM book LIMIT ? OFFSET ?";  //limit is number of instances per page and offset is starting instance
    $stmt = $conn->prepare($query);
    $stmt->bind_param("ii", $limit, $offset); //ii: both variables are integers. Will replace limit and offset with values
    $stmt->execute();
    $result = $stmt->get_result();
```



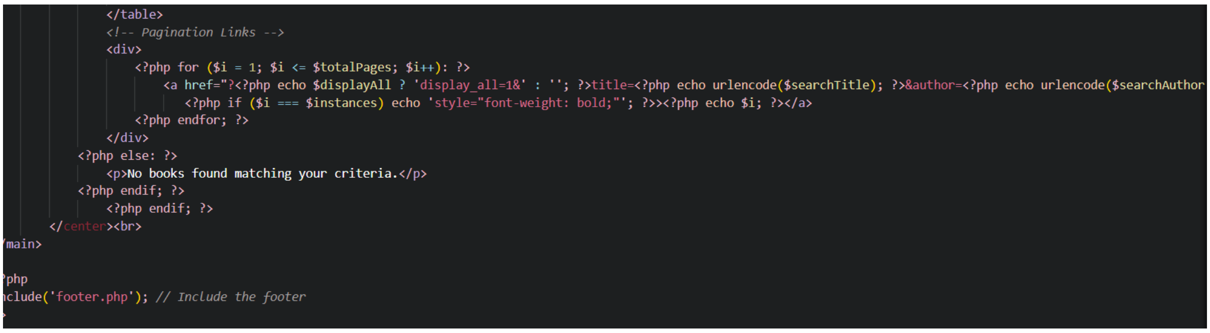
```
    $books = $result->fetch_all(MYSQLI_ASSOC);

    $totalBooksNum = $conn->query("SELECT COUNT(*) AS count FROM book"); //counts total number of rows in the book table
    $totalBooks = $totalBooksNum->fetch_assoc()['count'];
    $totalPages = ceil($totalBooks / $limit);
}
elseif ($filtered)
{
    // If search filters are used
    $query = "SELECT * FROM book
              WHERE BookTitle LIKE ?
              AND Author LIKE ?
              AND (CategoryCode LIKE ? OR ? = '')
              LIMIT ? OFFSET ?";
    $stmt = $conn->prepare($query);
    $searchTitleParam = "%$searchTitle%";
    $searchAuthorParam = "%$searchAuthor%";
    $categoryParam = "%$categoryID%";
    $stmt->bind_param("ssssii", $searchTitleParam, $searchAuthorParam, $categoryParam, $categoryID, $limit, $offset);
    $stmt->execute();
    $result = $stmt->get_result();
    $books = $result->fetch_all(MYSQLI_ASSOC);

    $totalBooksQuery = "SELECT COUNT(*) AS count FROM book
                        WHERE BookTitle LIKE ?
                        AND Author LIKE ?
                        AND (CategoryCode LIKE ? OR ? = '')";
    $stmt = $conn->prepare($totalBooksQuery);
    $stmt->bind_param("ssss", $searchTitleParam, $searchAuthorParam, $categoryParam, $categoryID);
    $stmt->execute();
    $totalBooksNum = $stmt->get_result();
    $totalBooks = $totalBooksNum->fetch_assoc()['count'];
    $totalPages = ceil($totalBooks / $limit); //rounds to integer with no regars to decimal part
}
?>
```



```
<main>
<center><div class="filteredSearch">

<img src="book2.png" width="100" height="100">
    <h1>Book search</h1>

    <!-- Book Search Form -->
    <form action="SearchBooksPage.php" method="GET">
        <label for="title">Book Title:</label>
        <input type="text" name="title" value="<?php echo htmlspecialchars($searchTitle); ?>" /><br><br>
        <label for="author">Author:</label>&nbsp&nbsp
        <input type="text" name="author" value="<?php echo htmlspecialchars($searchAuthor); ?>"  /><br><br>

        <label for="category">Category:</label>&nbsp
        <select style="border-radius:15px;border-color:cadetblue;"name="category">
            <option value="">-- Select Category --</option>
            <?php foreach ($categories as $category): ?>
                <option value="<?php echo htmlspecialchars($category['CategoryID']); ?>"
                    <?php echo $category['CategoryID'] === $categoryID ? 'selected' : ''; ?>>
                    <?php echo htmlspecialchars($category['CategoryDetails']); ?>
                </option>
            <?php endforeach; ?>
        </select><br><br>

        <table>
            <tr><td>
                <button class="searchbutton" type="submit">Search for a book</button></form>
            </td><td>

                <form action="SearchBooksPage.php" method="GET">
                <button class="searchbutton" type="submit" name="display_all" value="1">Display All Books</button>
                </form>
            </td></tr>
        </table></div>
        <br>

    <!--results of book seacrh-->
```



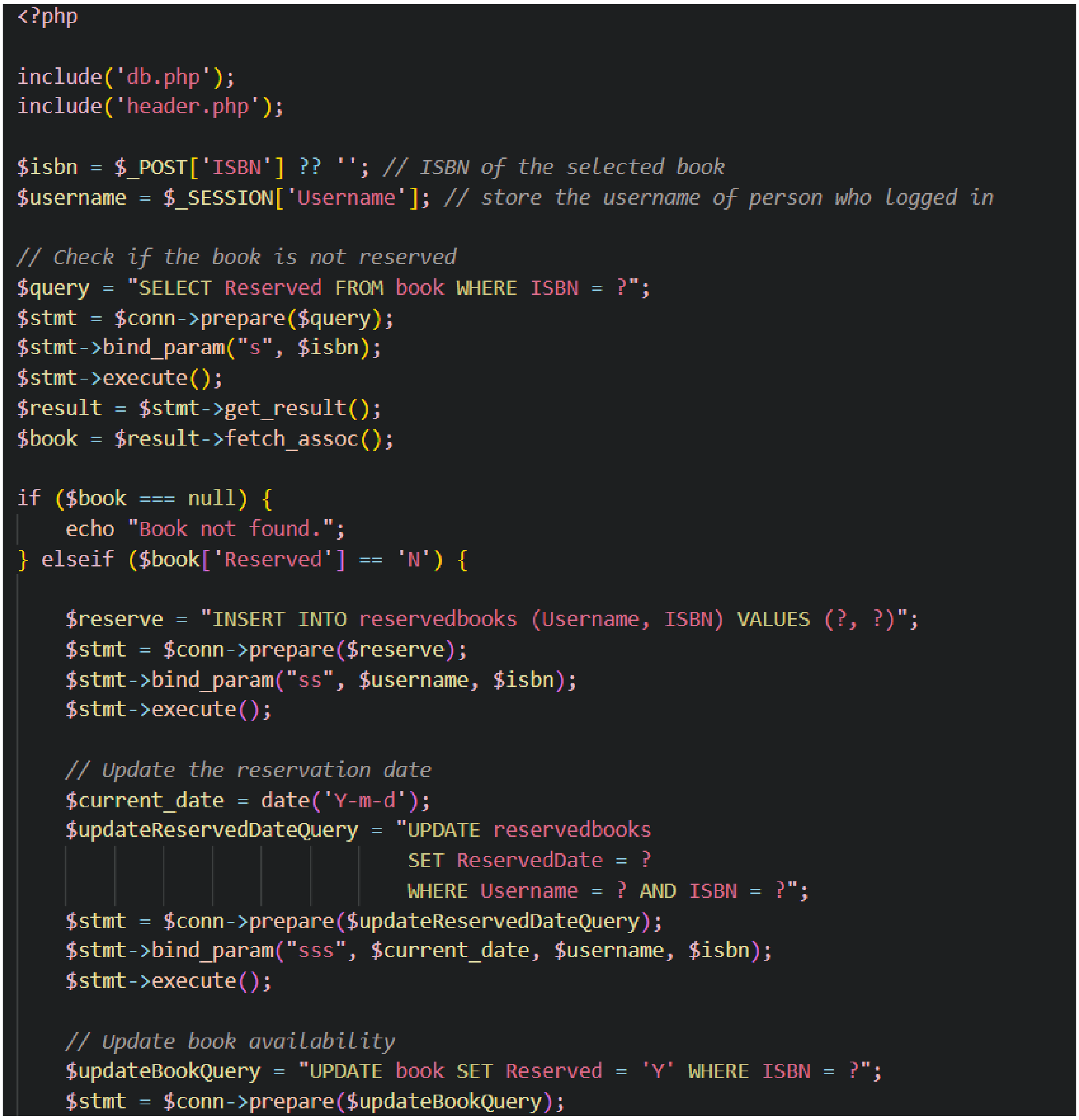
```
<?php if ($filtered || $displayAll): ?> <!--output depending on if user clicked to display all or based on filter-->

        <h3>Search Results</h3>
        <?php if ($books): ?>
            <table class="searchTable">

                <thead>
                    <tr>
                        <th>Title</th>
                        <th>Author</th>
                        <th>Category</th>
                        <th>Status</th>
                        <th>Action</th>
                    </tr>
                </thead>
                <tbody>
                    <?php foreach ($books as $book): ?>
                        <tr>
                            <td><?php echo htmlspecialchars($book['BookTitle']); ?></td>
                            <td><?php echo htmlspecialchars($book['Author']); ?></td>
                            <td><?php echo htmlspecialchars($book['CategoryCode']); ?></td>
                            <td><?php echo $book['Reserved'] === 'Y' ? 'Reserved' : 'Available'; ?></td>
                            <td>
                                <?php if ($book['Reserved'] === 'N'): ?>
                                    <form method="POST" action="ReservationPage.php">
                                        <input type="hidden" name="ISBN" value="<?php echo htmlspecialchars($book['ISBN']); ?>">
                                        <button type="submit">Reserve</button>
                                    </form>
                                <?php else: ?>
                                    Not Available
                                <?php endif; ?>
                            </td>
                        </tr>
                    <?php endforeach; ?>
                </tbody>
            </table>
```



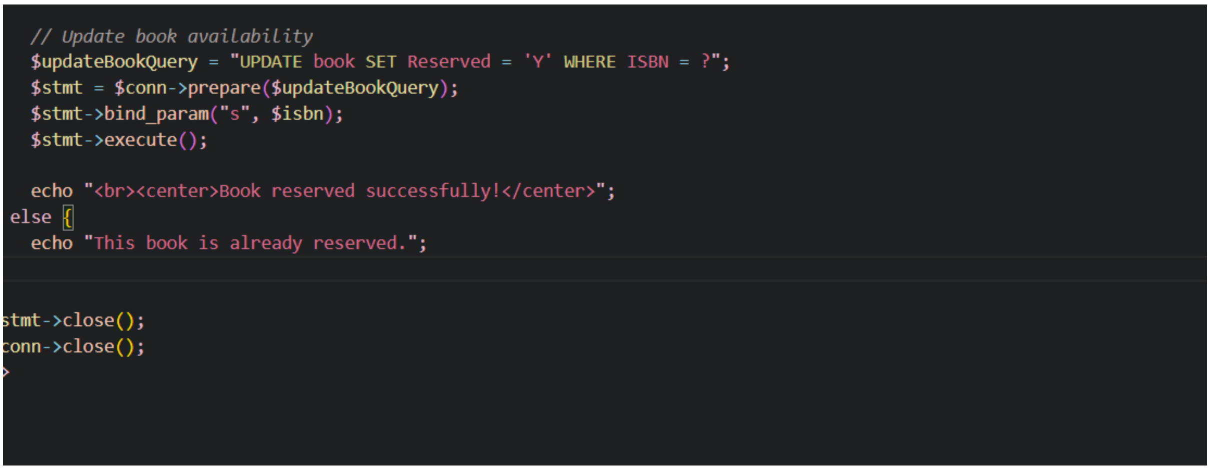
```
            </table>
            <!-- Pagination Links -->
            <div>
                <?php for ($i = 1; $i <= $totalPages; $i++): ?>
                    <a href="?<?php echo $displayAll ? 'display_all=1&' : ''; ?>title=<?php echo urlencode($searchTitle); ?>&author=<?php echo urlencode($searchAuthor
                       <?php if ($i === $instances) echo 'style="font-weight: bold;"'; ?>><?php echo $i; ?></a>
                <?php endfor; ?>
            </div>
        <?php else: ?>
            <p>No books found matching your criteria.</p>
        <?php endif; ?>
            <?php endif; ?>
    </center><br>
/main>

?php
nclude('footer.php'); // Include the footer
>
```

## ReservationPage.php

When a person reserves the book, database column reserved  in table book will be updated to ‘Y’ meaning it is reserved.  The username of the person who reserved the book and book ISBN will be added to reservedbooks table.



```
<?php

include('db.php');
include('header.php');

$isbn = $_POST['ISBN'] ?? ''; // ISBN of the selected book
$username = $_SESSION['Username']; // store the username of person who logged in

// Check if the book is not reserved
$query = "SELECT Reserved FROM book WHERE ISBN = ?";
$stmt = $conn->prepare($query);
$stmt->bind_param("s", $isbn);
$stmt->execute();
$result = $stmt->get_result();
$book = $result->fetch_assoc();

if ($book === null) {
    echo "Book not found.";
} elseif ($book['Reserved'] == 'N') {

    $reserve = "INSERT INTO reservedbooks (Username, ISBN) VALUES (?, ?)";
    $stmt = $conn->prepare($reserve);
    $stmt->bind_param("ss", $username, $isbn);
    $stmt->execute();

    // Update the reservation date
    $current_date = date('Y-m-d');
    $updateReservedDateQuery = "UPDATE reservedbooks
                                SET ReservedDate = ?
                                WHERE Username = ? AND ISBN = ?";
    $stmt = $conn->prepare($updateReservedDateQuery);
    $stmt->bind_param("sss", $current_date, $username, $isbn);
    $stmt->execute();

    // Update book availability
    $updateBookQuery = "UPDATE book SET Reserved = 'Y' WHERE ISBN = ?";
    $stmt = $conn->prepare($updateBookQuery);
```

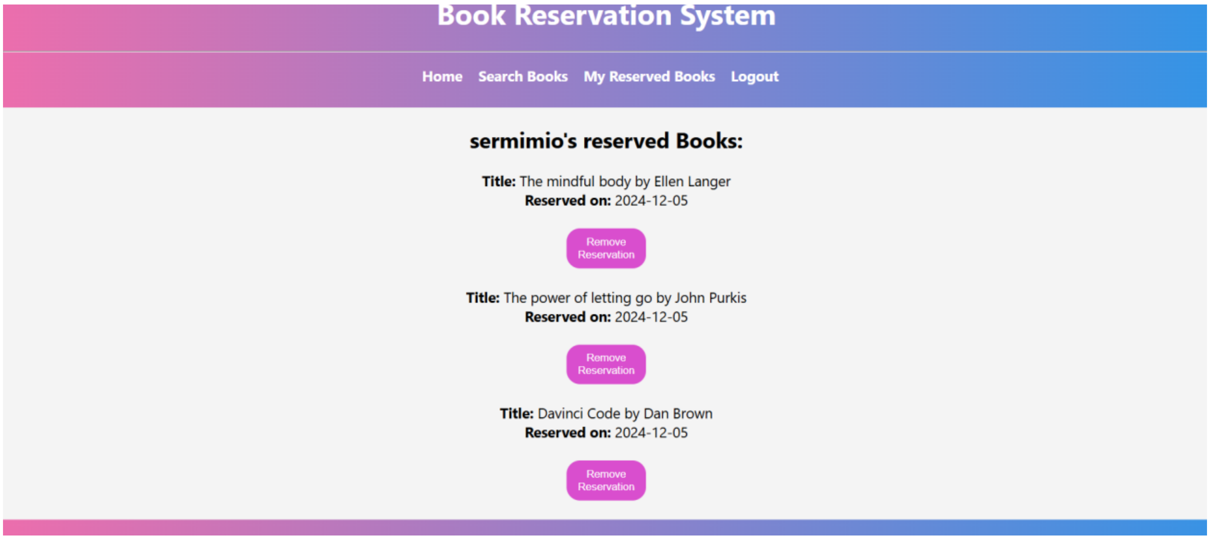


```
  // Update book availability
  $updateBookQuery = "UPDATE book SET Reserved = 'Y' WHERE ISBN = ?";
  $stmt = $conn->prepare($updateBookQuery);
  $stmt->bind_param("s", $isbn);
  $stmt->execute();

  echo "<br><center>Book reserved successfully!</center>";
else {
  echo "This book is already reserved.";


stmt->close();
conn->close();
>
```

## displayReservedPage.php

This page shows a list of user’s reserved books. It retrieves it from table reservedbooks and to retrieve book name, uses join  statement in sql query.

The page will show the title of reserved book and date it was reserved on by using current_date. There’s also the option to remove book reservation that will remove this record from reserved books table in database and reserved in Books will be updated to N.

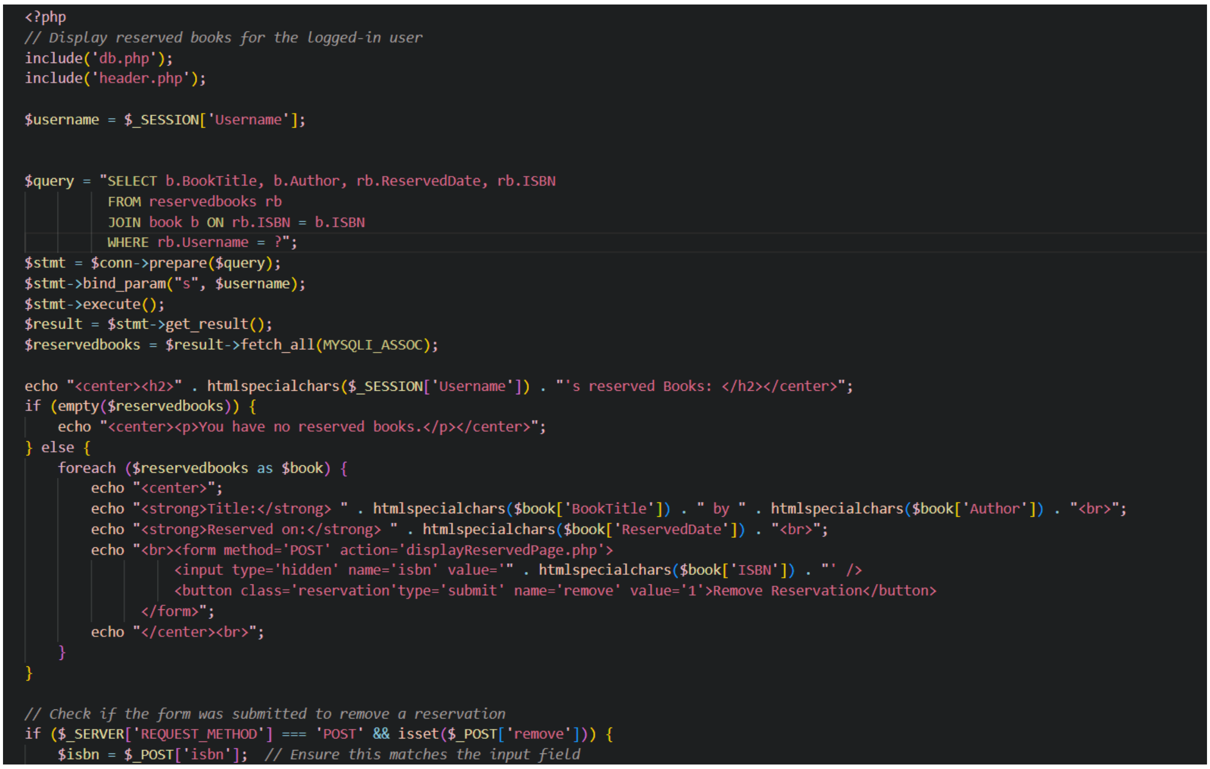


sermimio's reserved Books:

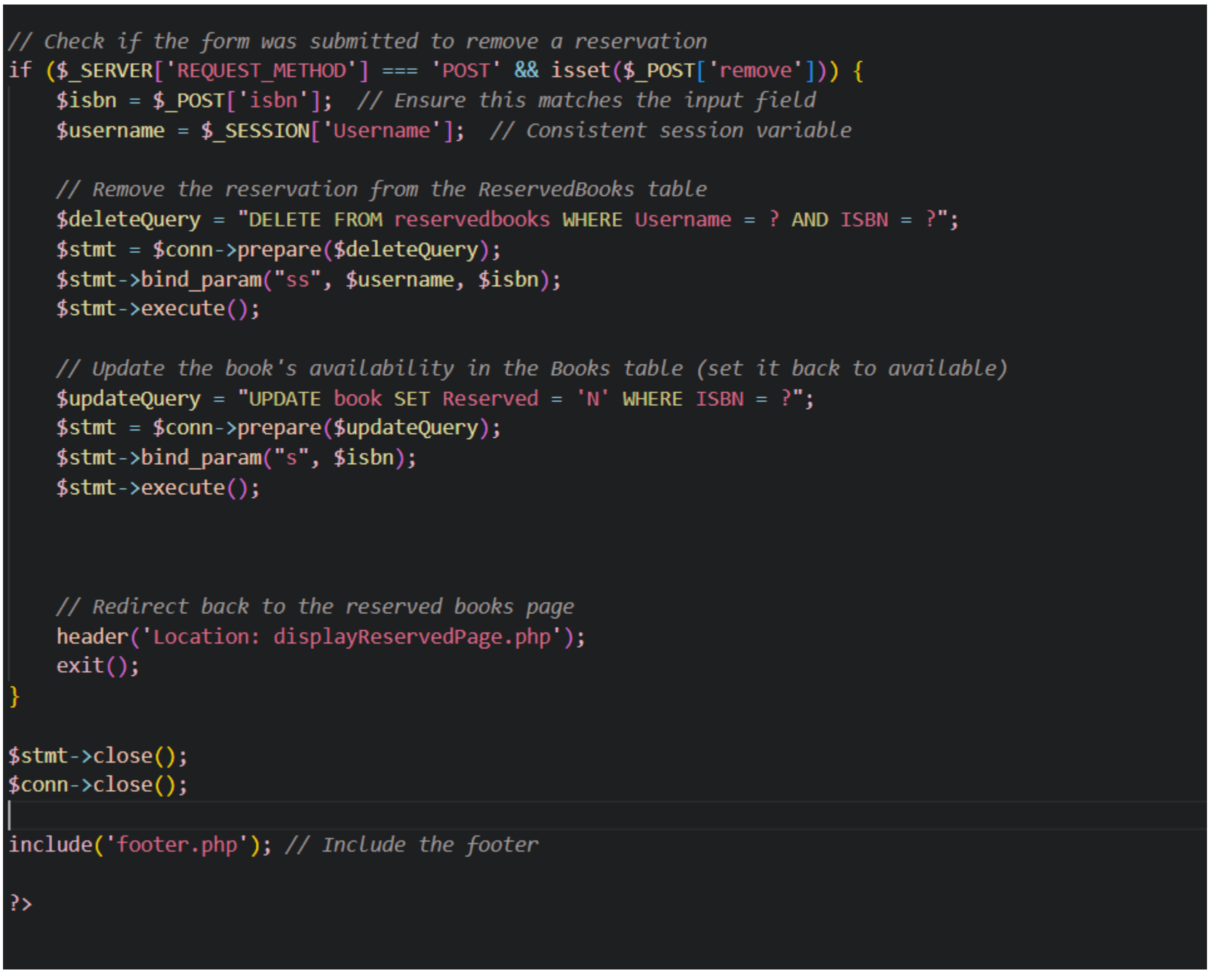
**Title:** The mindful body by Ellen Langer
**Reserved on:** 2024-12-05

**Title:** The power of letting go by John Purkis
**Reserved on:** 2024-12-05

**Title:** Davinci Code by Dan Brown
**Reserved on:** 2024-12-05



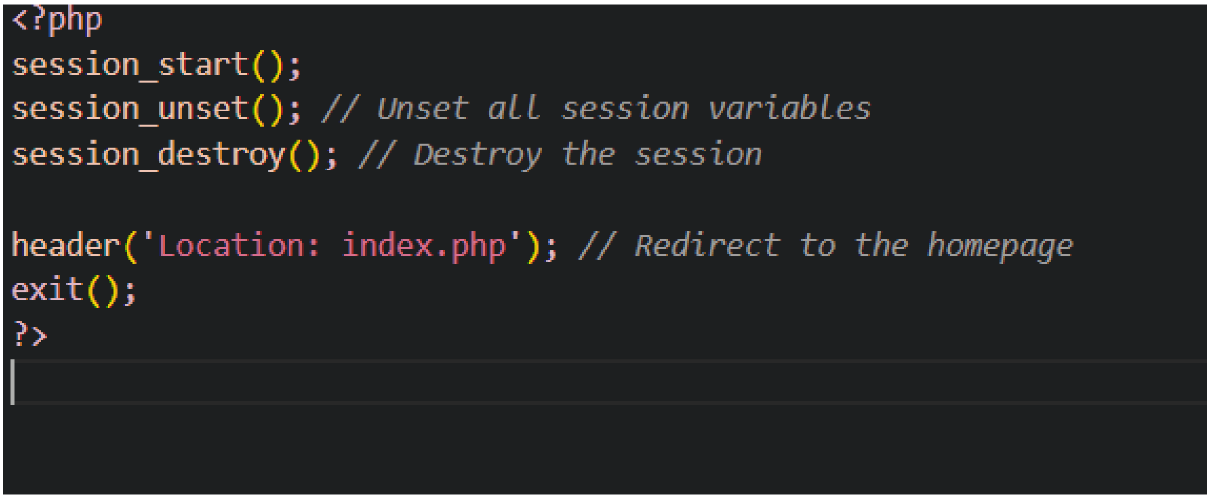
```php
<?php
// Display reserved books for the logged-in user
include('db.php');
include('header.php');

$username = $_SESSION['Username'];


$query = "SELECT b.BookTitle, b.Author, rb.ReservedDate, rb.ISBN
          FROM reservedbooks rb
          JOIN book b ON rb.ISBN = b.ISBN
          WHERE rb.Username = ?";
$stmt = $conn->prepare($query);
$stmt->bind_param("s", $username);
$stmt->execute();
$result = $stmt->get_result();
$reservedbooks = $result->fetch_all(MYSQLI_ASSOC);

echo "<center><h2>" . htmlspecialchars($_SESSION['Username']) . "'s reserved Books: </h2></center>";
if (empty($reservedbooks)) {
    echo "<center><p>You have no reserved books.</p></center>";
} else {
    foreach ($reservedbooks as $book) {
        echo "<center>";
        echo "<strong>Title:</strong> " . htmlspecialchars($book['BookTitle']) . " by " . htmlspecialchars($book['Author']) . "<br>";
        echo "<strong>Reserved on:</strong> " . htmlspecialchars($book['ReservedDate']) . "<br>";
        echo "<br><form method='POST' action='displayReservedPage.php'>
                    <input type='hidden' name='isbn' value='" . htmlspecialchars($book['ISBN']) . "' />
                    <button class='reservation'type='submit' name='remove' value='1'>Remove Reservation</button>
              </form>";
        echo "</center><br>";
    }
}

// Check if the form was submitted to remove a reservation
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['remove'])) {
    $isbn = $_POST['isbn'];  // Ensure this matches the input field
```



```
// Check if the form was submitted to remove a reservation
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['remove'])) {
    $isbn = $_POST['isbn'];  // Ensure this matches the input field
    $username = $_SESSION['Username'];  // Consistent session variable

    // Remove the reservation from the ReservedBooks table
    $deleteQuery = "DELETE FROM reservedbooks WHERE Username = ? AND ISBN = ?";
    $stmt = $conn->prepare($deleteQuery);
    $stmt->bind_param("ss", $username, $isbn);
    $stmt->execute();

    // Update the book's availability in the Books table (set it back to available)
    $updateQuery = "UPDATE book SET Reserved = 'N' WHERE ISBN = ?";
    $stmt = $conn->prepare($updateQuery);
    $stmt->bind_param("s", $isbn);
    $stmt->execute();



    // Redirect back to the reserved books page
    header('Location: displayReservedPage.php');
    exit();
}

$stmt->close();
$conn->close();

include('footer.php'); // Include the footer

?>
```

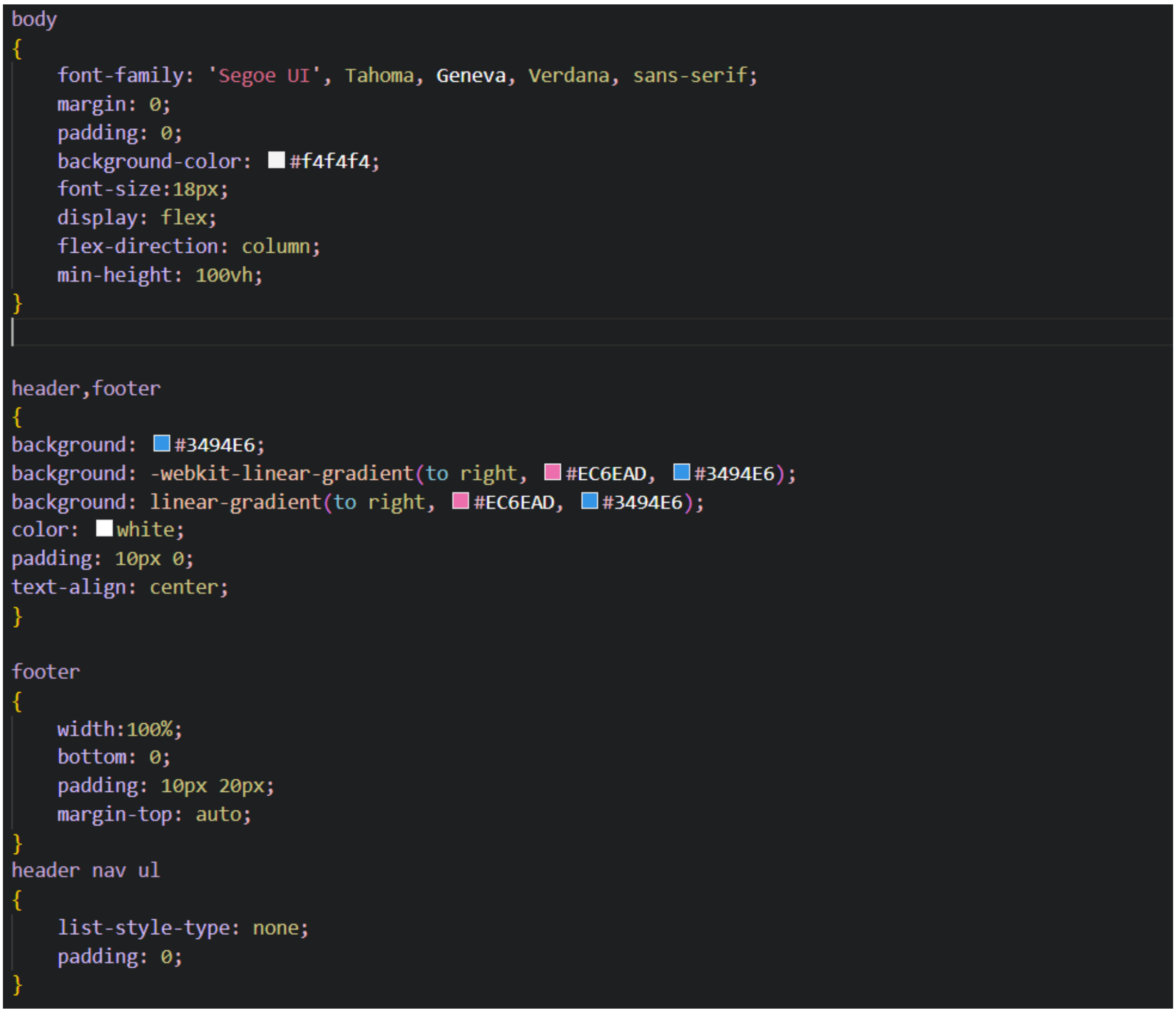
## Logout.php

This page will destroy the current session if the user clicks on Logout page in menu. The user will be redirected to home page (index.php) They no longer will be able to reserve books unless they log in.

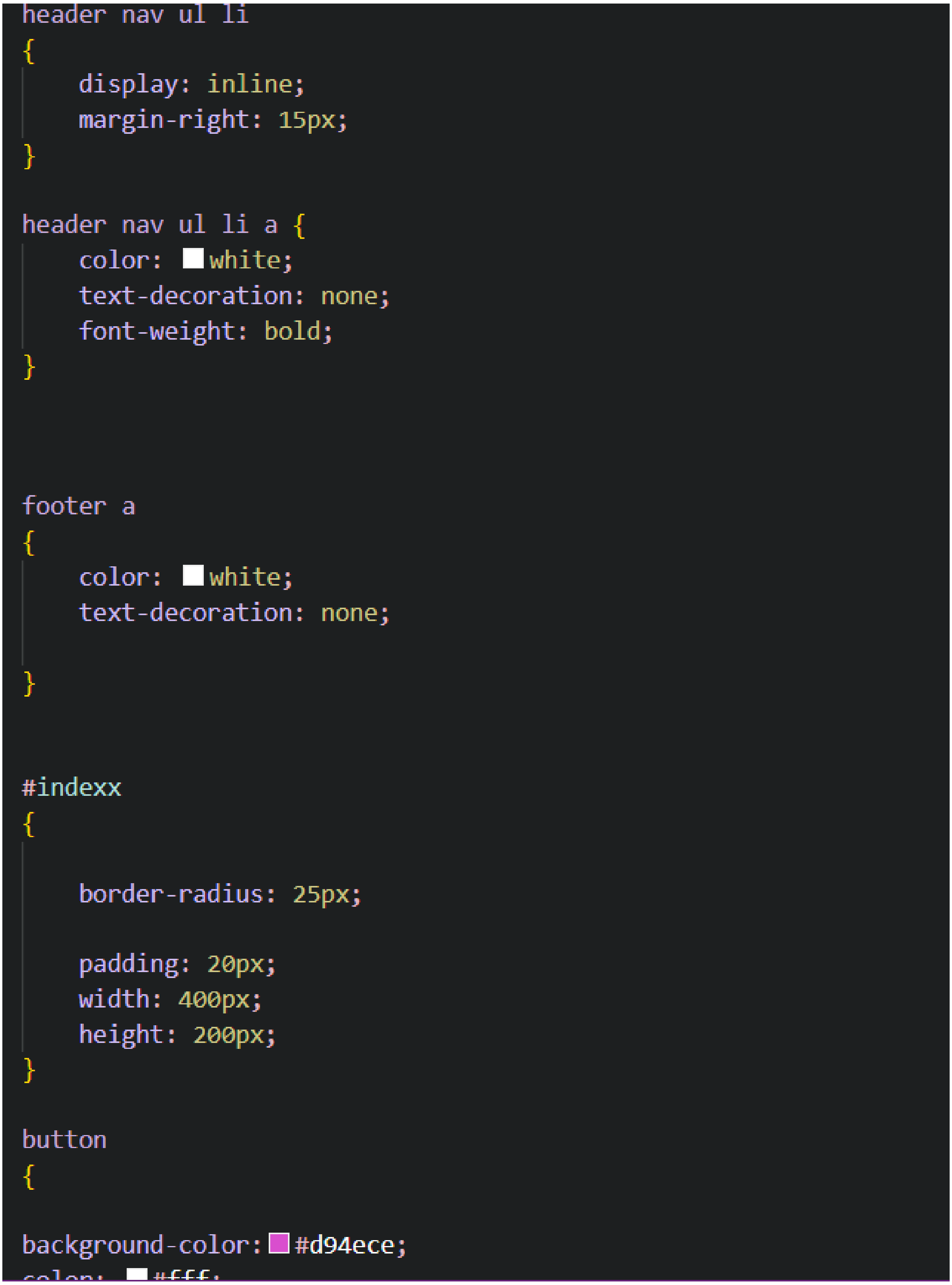


```
<?php
session_start();
session_unset(); // Unset all session variables
session_destroy(); // Destroy the session

header('Location: index.php'); // Redirect to the homepage
exit();
?>
```

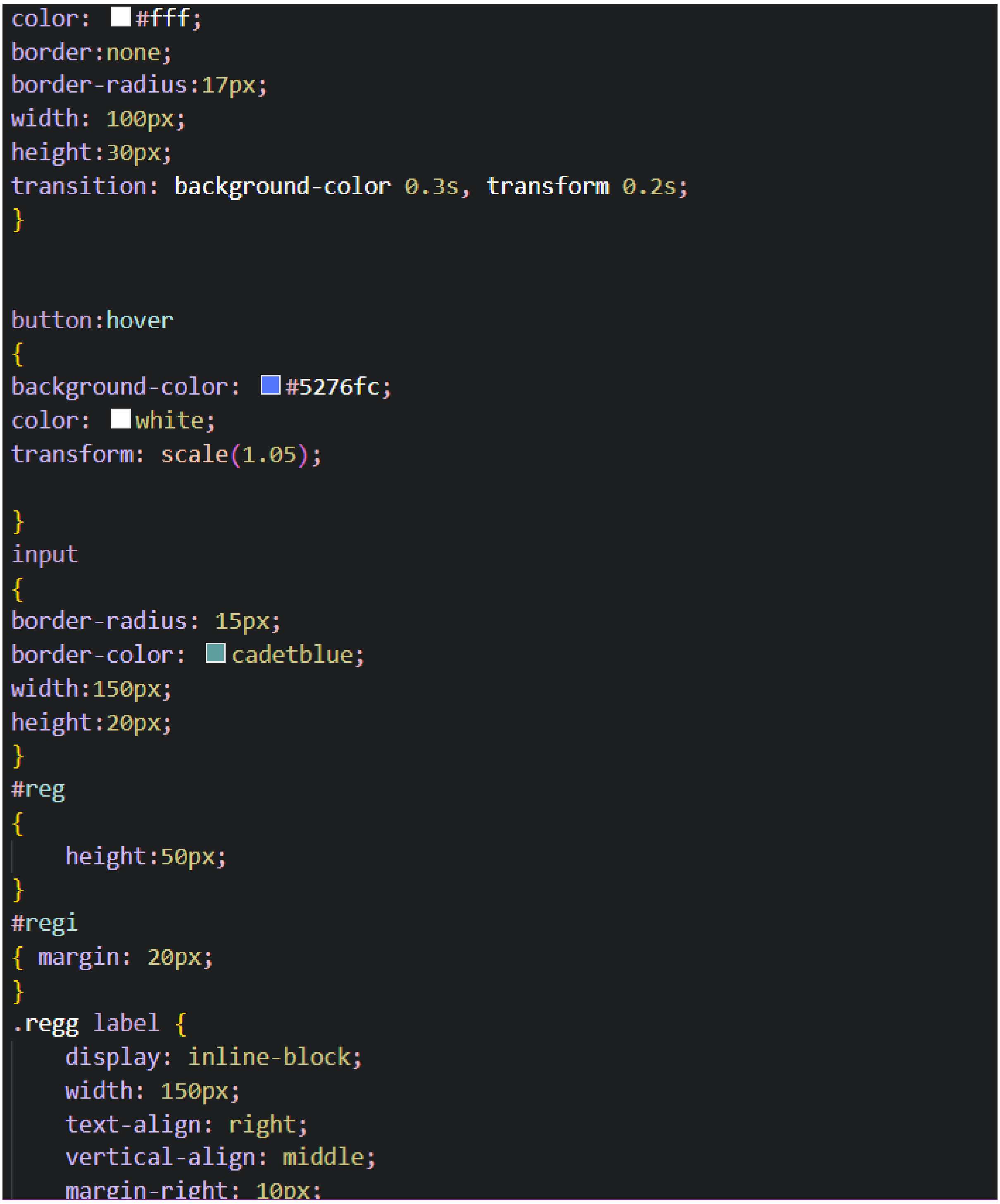
## Styles.css



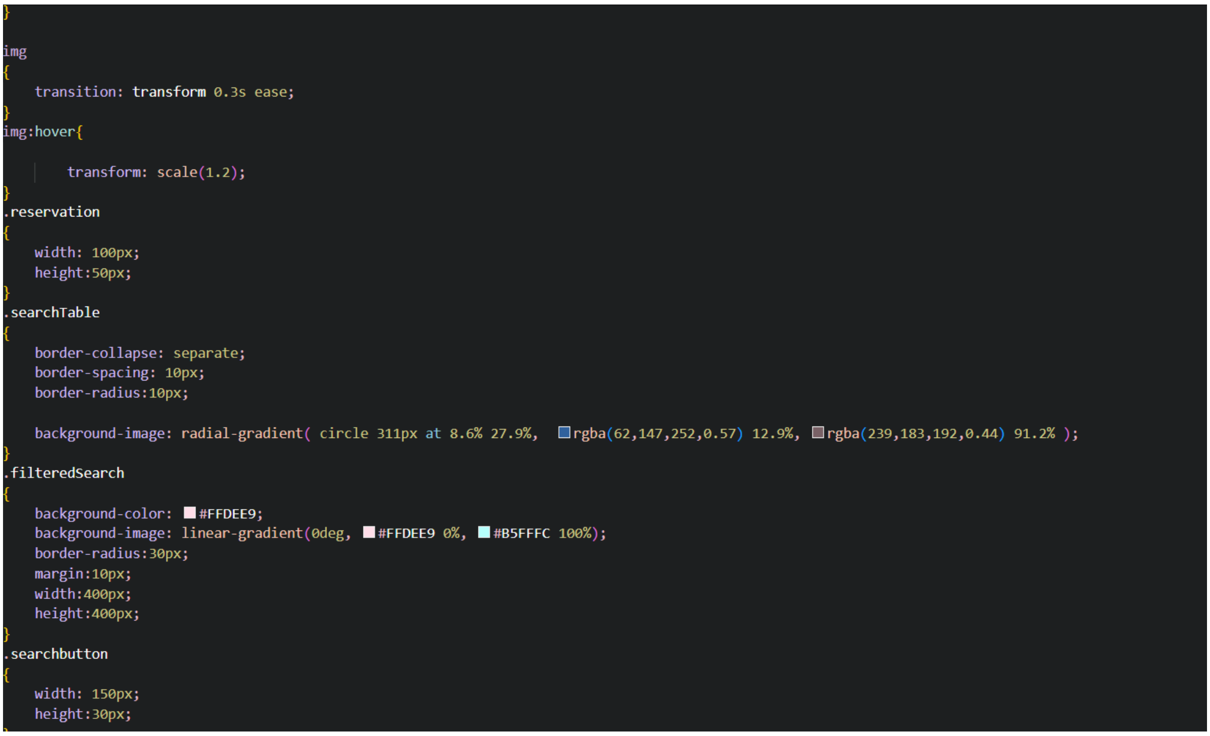
```
body
{
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f4f4f4;
    font-size:18px;
    display: flex;
    flex-direction: column;
    min-height: 100vh;
}


header,footer
{
background: #3494E6;
background: -webkit-linear-gradient(to right, #EC6EAD, #3494E6);
background: linear-gradient(to right, #EC6EAD, #3494E6);
color: white;
padding: 10px 0;
text-align: center;
}

footer
{
    width:100%;
    bottom: 0;
    padding: 10px 20px;
    margin-top: auto;
}
header nav ul
{
    list-style-type: none;
    padding: 0;
}
```



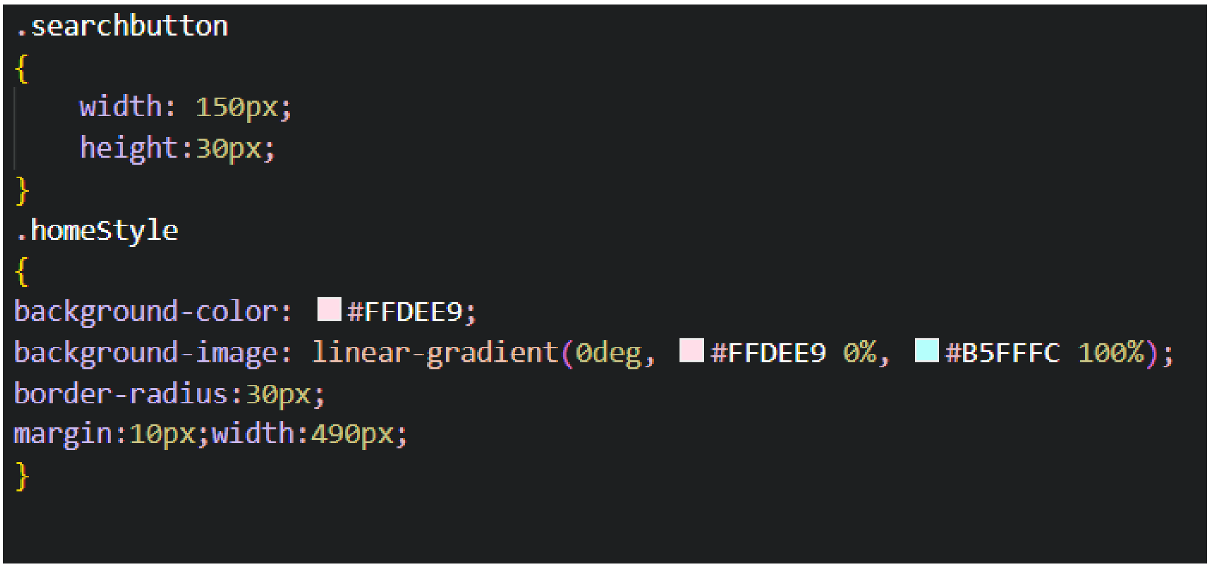
```
header nav ul li
{
    display: inline;
    margin-right: 15px;
}

header nav ul li a {
    color: ⬜white;
    text-decoration: none;
    font-weight: bold;
}



footer a
{
    color: ⬜white;
    text-decoration: none;

}


#indexx
{

    border-radius: 25px;

    padding: 20px;
    width: 400px;
    height: 200px;
}

button
{

background-color:🟪#d94ece;
color: ⬜#fff;
```



```
color: #fff;
border:none;
border-radius:17px;
width: 100px;
height:30px;
transition: background-color 0.3s, transform 0.2s;
}


button:hover
{
background-color: #5276fc;
color: white;
transform: scale(1.05);

}
input
{
border-radius: 15px;
border-color: cadetblue;
width:150px;
height:20px;
}
#reg
{
    height:50px;
}
#regi
{ margin: 20px;
}
.regg label {
    display: inline-block;
    width: 150px;
    text-align: right;
    vertical-align: middle;
    margin-right: 10px;
```



```
}
img
{
    transition: transform 0.3s ease;
}
img:hover{

        transform: scale(1.2);
}
.reservation
{
    width: 100px;
    height:50px;
}
.searchTable
{
    border-collapse: separate;
    border-spacing: 10px;
    border-radius:10px;

    background-image: radial-gradient( circle 311px at 8.6% 27.9%,  rgba(62,147,252,0.57) 12.9%, rgba(239,183,192,0.44) 91.2% );
}
.filteredSearch
{
    background-color: #FFDEE9;
    background-image: linear-gradient(0deg, #FFDEE9 0%, #B5FFFC 100%);
    border-radius:30px;
    margin:10px;
    width:400px;
    height:400px;
}
.searchbutton
{
    width: 150px;
    height:30px;
```



```
.searchbutton
{
    width: 150px;
    height:30px;
}
.homeStyle
{
background-color: #FFDEE9;
background-image: linear-gradient(0deg, #FFDEE9 0%, #B5FFFC 100%);
border-radius:30px;
margin:10px;width:490px;
}
```

## Design Document

**Project Overview**:

The project is a Book Reservation System where a user can:

- Search for a book
- Reserve a book
- Remove a reservation

The user should be able to register and log in

**Purpose of the project:**

Create a convenient, user-friendly website that allows users to search for the specific book providing search filters and reserve, if available.

---

## Functional Requirements

- **Book reservation Management**:
  - Users can add and remove reservations
  - Book can only be reserved by one person
  - Should show the date of reservation
- **User Authentication**:
  - Users must log in with a valid username and password.
  - Do not allow unauthorized user access  book reservation management pages
- **Session Management**:
  - Maintain user login status with session variables.
  - Redirect unauthorized users to the login page.
- **Error Handling**:
  - Data validation to ensure only valid inputs are accepted when registering new user.
  - Ensuring database connection
  - Display meaningful error messages for invalid input or missing address details.

---

## The system's performance, security, and usability expectations.

- **Performance**:
  - The system should respond to user action quickly (1-2 seconds)
  - Database should be instantly updated with new values
- **Security**:
  - Protect against SQL injection by using prepared statements and HTML special chars.
- **Maintainability**

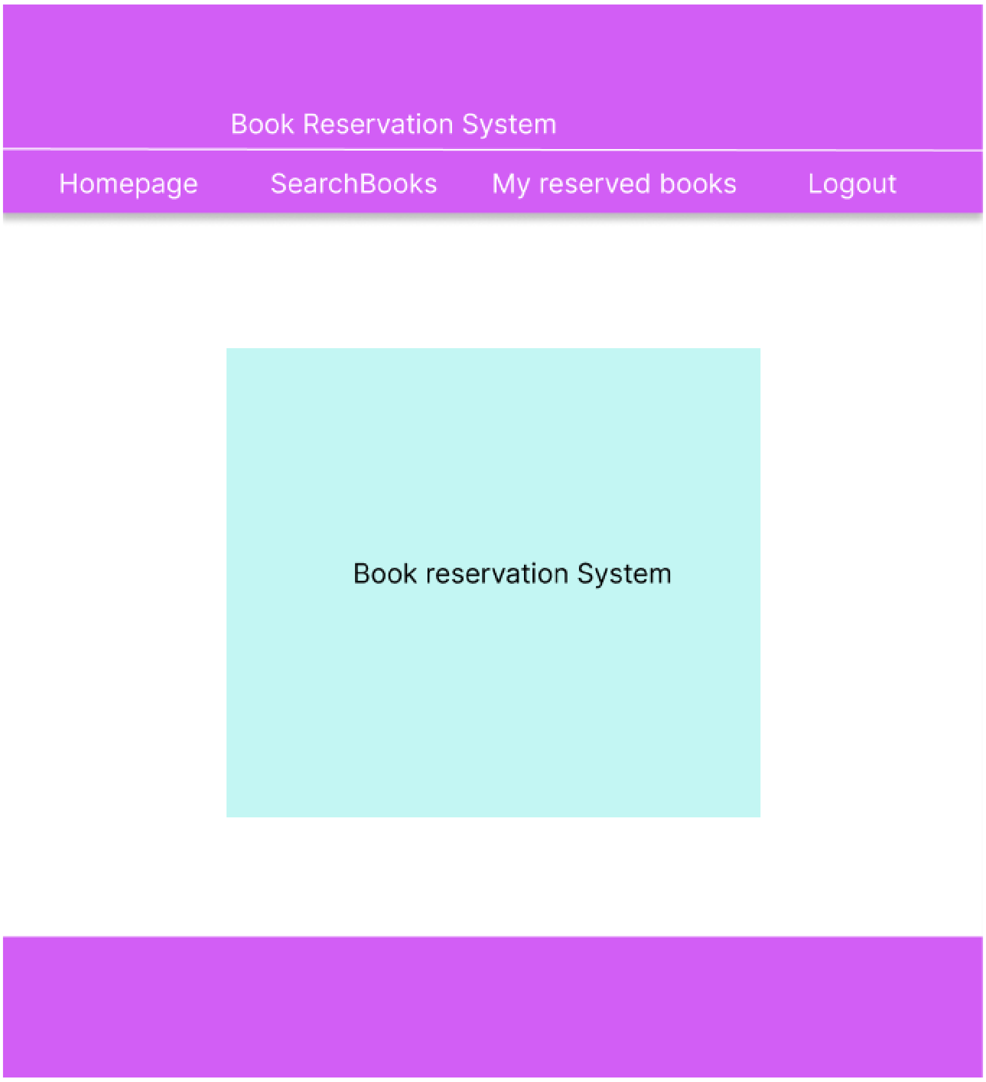
- Storing reusable code inside one file
- Meaningful variable names
- Avoiding hard coding
- Consistent comments on code functionality for other developers view

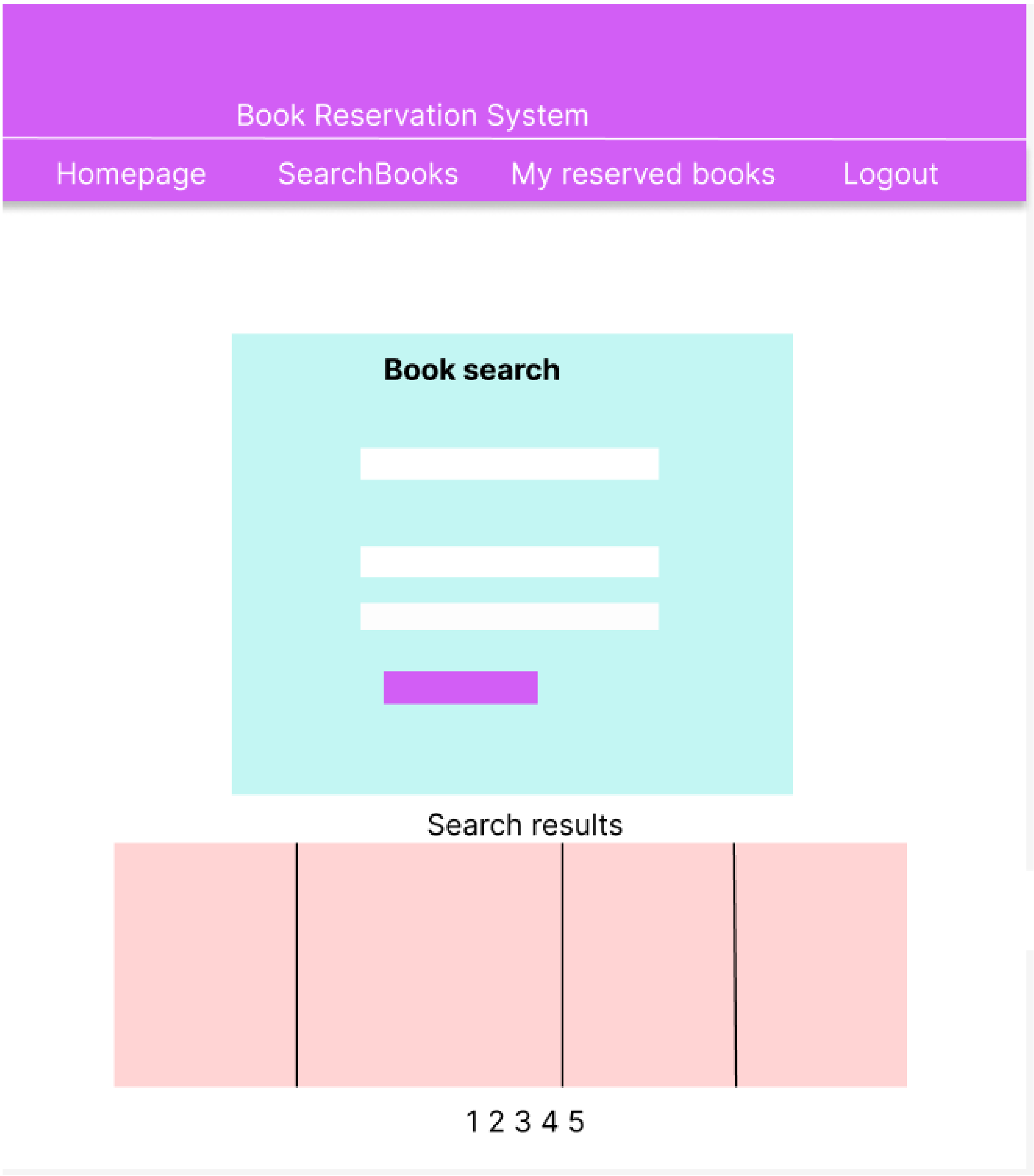
- **User-friendliness**:
  - Redirect user to related pages
  - Auto-fill form fields with existing session data if chosen.
  - Responsive design (contrast colors, interactive icons, intuitive page navigation)

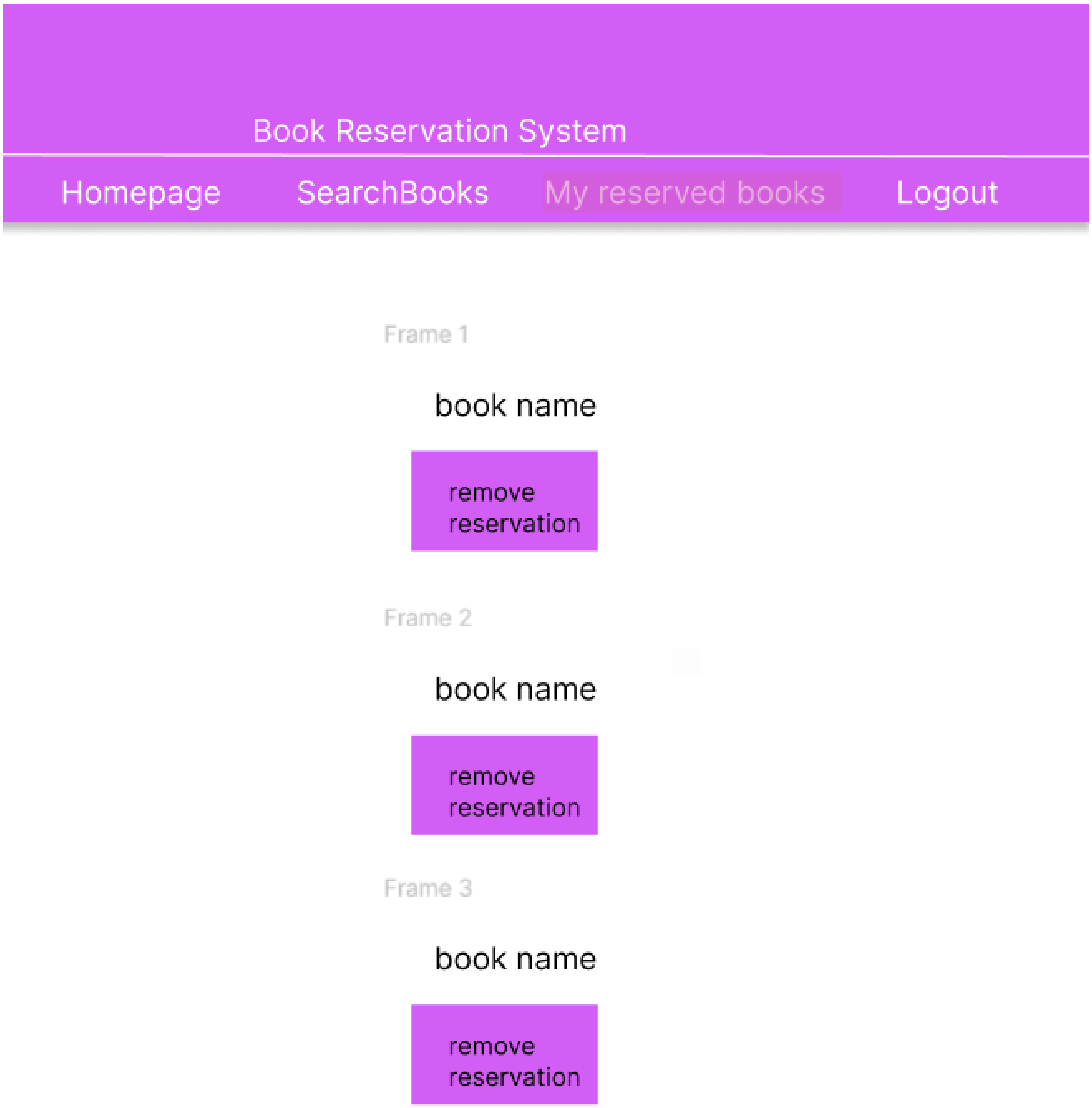
---

**Design**

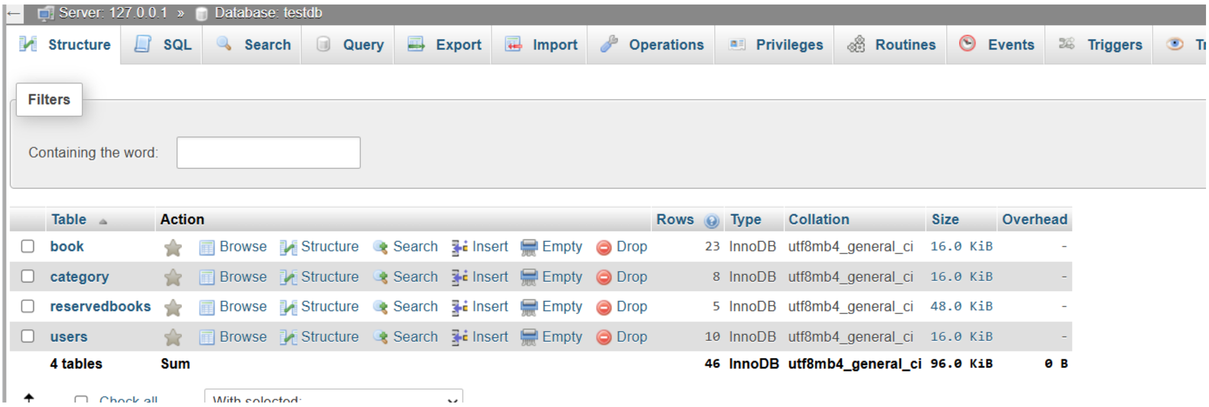
Low-fidelity prototypes of more functional pages:




Book Reservation System
Homepage
SearchBooks
My reserved books
Logout
Book search
Search results
1 2 3 4 5

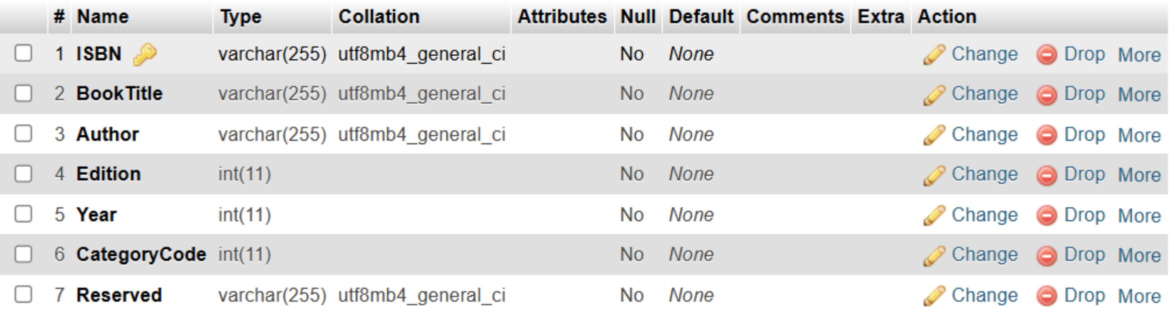

Book Reservation System
Homepage
SearchBooks
My reserved books
Logout
Frame 1
book name
remove reservation
Frame 2
book name
remove reservation
Frame 3
book name
remove reservation

## Database



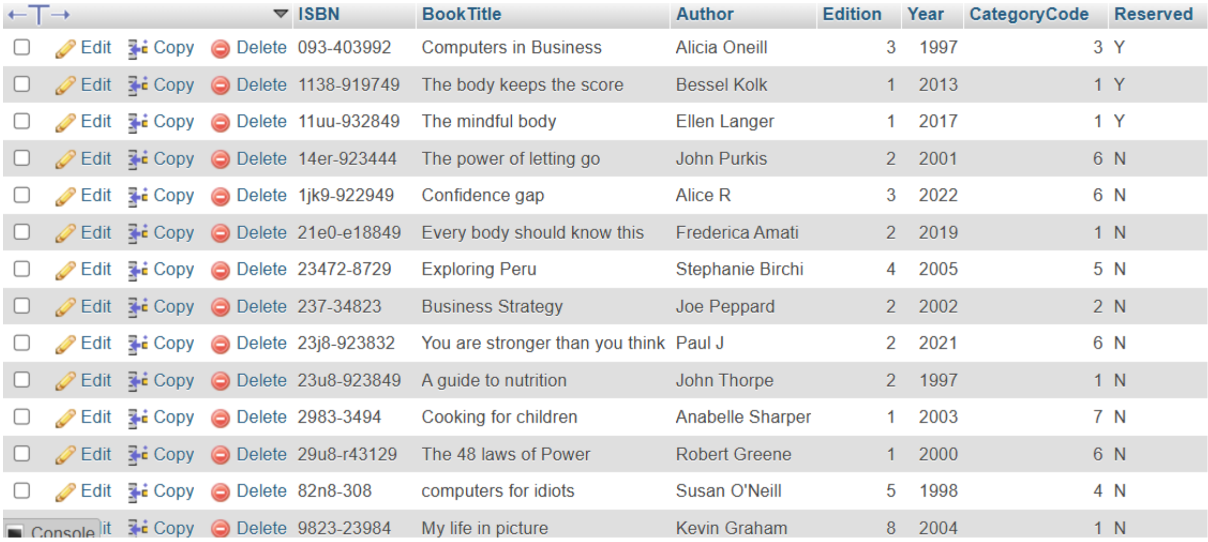
| | Table | Action | Rows | Type | Collation | Size | Overhead |
|---|---|---|---|---|---|---|---|
| □ | book | ☆ Browse Structure Search Insert Empty Drop | 23 | InnoDB | utf8mb4_general_ci | 16.0 KiB | - |
| □ | category | ☆ Browse Structure Search Insert Empty Drop | 8 | InnoDB | utf8mb4_general_ci | 16.0 KiB | - |
| □ | reservedbooks | ☆ Browse Structure Search Insert Empty Drop | 5 | InnoDB | utf8mb4_general_ci | 48.0 KiB | - |
| □ | users | ☆ Browse Structure Search Insert Empty Drop | 10 | InnoDB | utf8mb4_general_ci | 16.0 KiB | - |
| | 4 tables | Sum | 46 | InnoDB | utf8mb4_general_ci | 96.0 KiB | 0 B |

book structure:



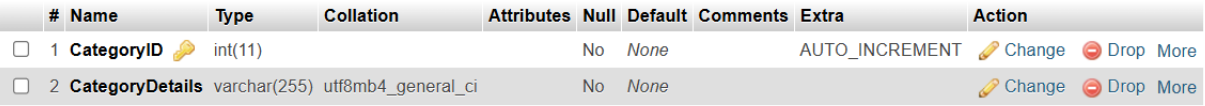
| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|---|
| □ | 1 | **ISBN** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 2 | **BookTitle** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 3 | **Author** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 4 | **Edition** | int(11) | | | No | *None* | | | Change Drop More |
| □ | 5 | **Year** | int(11) | | | No | *None* | | | Change Drop More |
| □ | 6 | **CategoryCode** | int(11) | | | No | *None* | | | Change Drop More |
| □ | 7 | **Reserved** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |

Book records:



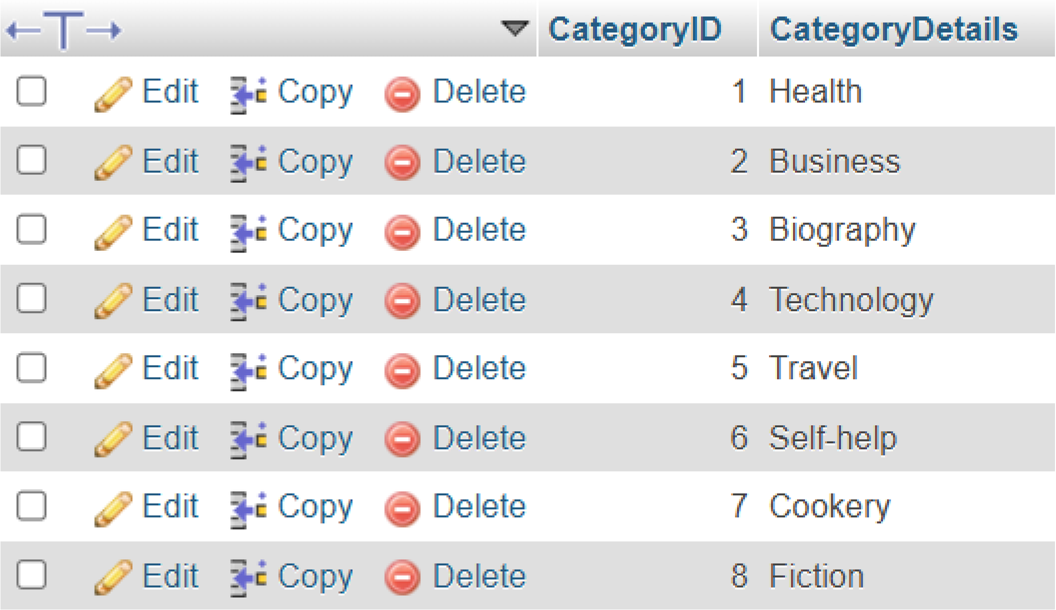
| ←T→ | ISBN | BookTitle | Author | Edition | Year | CategoryCode | Reserved |
|---|---|---|---|---|---|---|---|
| □ Edit Copy Delete | 093-403992 | Computers in Business | Alicia Oneill | 3 | 1997 | 3 | Y |
| □ Edit Copy Delete | 1138-919749 | The body keeps the score | Bessel Kolk | 1 | 2013 | 1 | Y |
| □ Edit Copy Delete | 11uu-932849 | The mindful body | Ellen Langer | 1 | 2017 | 1 | Y |
| □ Edit Copy Delete | 14er-923444 | The power of letting go | John Purkis | 2 | 2001 | 6 | N |
| □ Edit Copy Delete | 1jk9-922949 | Confidence gap | Alice R | 3 | 2022 | 6 | N |
| □ Edit Copy Delete | 21e0-e18849 | Every body should know this | Frederica Amati | 2 | 2019 | 1 | N |
| □ Edit Copy Delete | 23472-8729 | Exploring Peru | Stephanie Birchi | 4 | 2005 | 5 | N |
| □ Edit Copy Delete | 237-34823 | Business Strategy | Joe Peppard | 2 | 2002 | 2 | N |
| □ Edit Copy Delete | 23j8-923832 | You are stronger than you think | Paul J | 2 | 2021 | 6 | N |
| □ Edit Copy Delete | 23u8-923849 | A guide to nutrition | John Thorpe | 2 | 1997 | 1 | N |
| □ Edit Copy Delete | 2983-3494 | Cooking for children | Anabelle Sharper | 1 | 2003 | 7 | N |
| □ Edit Copy Delete | 29u8-r43129 | The 48 laws of Power | Robert Greene | 1 | 2000 | 6 | N |
| □ Edit Copy Delete | 82n8-308 | computers for idiots | Susan O'Neill | 5 | 1998 | 4 | N |
| Console it Copy Delete | 9823-23984 | My life in picture | Kevin Graham | 8 | 2004 | 1 | N |

Category structure:



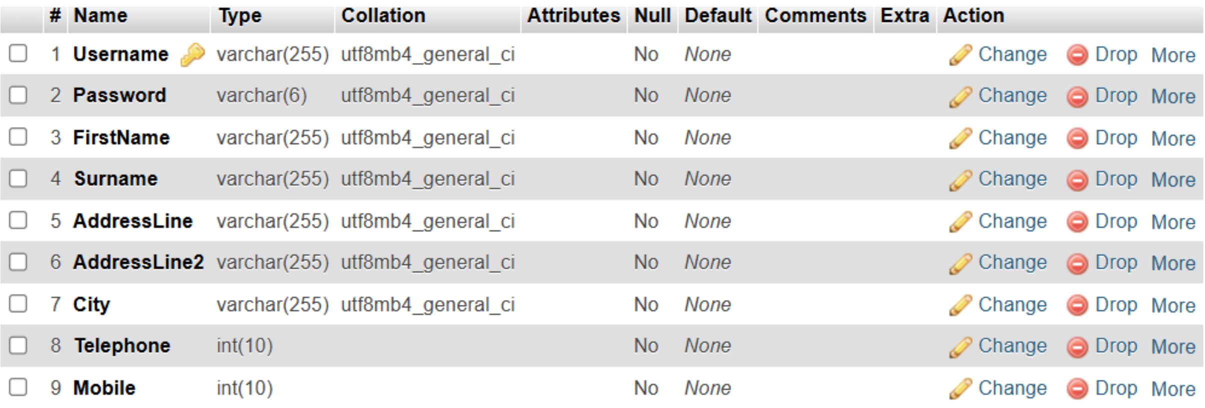
| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|---|
| □ | 1 | **CategoryID** | int(11) | | | No | *None* | | AUTO_INCREMENT | Change Drop More |
| □ | 2 | **CategoryDetails** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |

Category records:



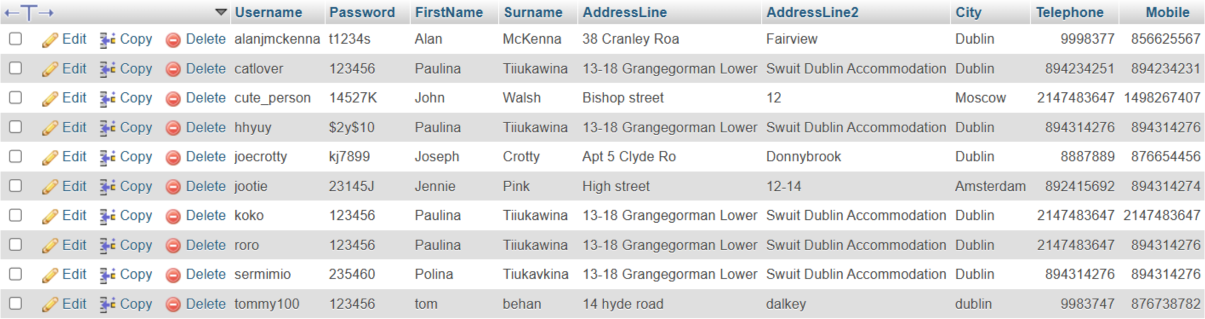
| ←T→ | | | | CategoryID | CategoryDetails |
|---|---|---|---|---|---|
| □ | Edit | Copy | Delete | 1 | Health |
| □ | Edit | Copy | Delete | 2 | Business |
| □ | Edit | Copy | Delete | 3 | Biography |
| □ | Edit | Copy | Delete | 4 | Technology |
| □ | Edit | Copy | Delete | 5 | Travel |
| □ | Edit | Copy | Delete | 6 | Self-help |
| □ | Edit | Copy | Delete | 7 | Cookery |
| □ | Edit | Copy | Delete | 8 | Fiction |

Users structure:



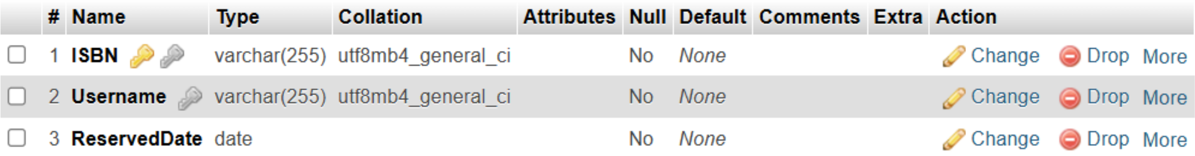
| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|---|
| □ | 1 | **Username** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 2 | **Password** | varchar(6) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 3 | **FirstName** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 4 | **Surname** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 5 | **AddressLine** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 6 | **AddressLine2** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 7 | **City** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change Drop More |
| □ | 8 | **Telephone** | int(10) | | | No | *None* | | | Change Drop More |
| □ | 9 | **Mobile** | int(10) | | | No | *None* | | | Change Drop More |

Users records:



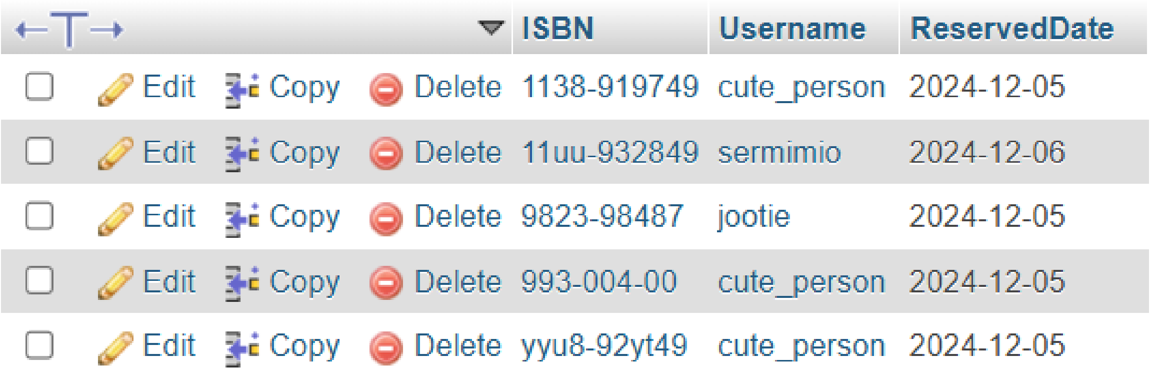
| ←T→ | | | | Username | Password | FirstName | Surname | AddressLine | AddressLine2 | City | Telephone | Mobile |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| □ | Edit | Copy | Delete | alanjmckenna | t1234s | Alan | McKenna | 38 Cranley Roa | Fairview | Dublin | 9998377 | 856625567 |
| □ | Edit | Copy | Delete | catlover | 123456 | Paulina | Tiiukawina | 13-18 Grangegorman Lower | Swuit Dublin Accommodation | Dublin | 894234251 | 894234231 |
| □ | Edit | Copy | Delete | cute_person | 14527K | John | Walsh | Bishop street | 12 | Moscow | 2147483647 | 1498267407 |
| □ | Edit | Copy | Delete | hhyuy | $2y$10 | Paulina | Tiiukawina | 13-18 Grangegorman Lower | Swuit Dublin Accommodation | Dublin | 894314276 | 894314276 |
| □ | Edit | Copy | Delete | joecrotty | kj7899 | Joseph | Crotty | Apt 5 Clyde Ro | Donnybrook | Dublin | 8887889 | 876654456 |
| □ | Edit | Copy | Delete | jootie | 23145J | Jennie | Pink | High street | 12-14 | Amsterdam | 892415692 | 894314274 |
| □ | Edit | Copy | Delete | koko | 123456 | Paulina | Tiiukawina | 13-18 Grangegorman Lower | Swuit Dublin Accommodation | Dublin | 2147483647 | 2147483647 |
| □ | Edit | Copy | Delete | roro | 123456 | Paulina | Tiiukawina | 13-18 Grangegorman Lower | Swuit Dublin Accommodation | Dublin | 2147483647 | 894314276 |
| □ | Edit | Copy | Delete | sermimio | 235460 | Polina | Tiukavkina | 13-18 Grangegorman Lower | Swuit Dublin Accommodation | Dublin | 894314276 | 894314276 |
| □ | Edit | Copy | Delete | tommy100 | 123456 | tom | behan | 14 hyde road | dalkey | dublin | 9983747 | 876738782 |

Reserved books structure:



| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| □ | 1 | **ISBN** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change | Drop | More |
| □ | 2 | **Username** | varchar(255) | utf8mb4_general_ci | | No | *None* | | | Change | Drop | More |
| □ | 3 | **ReservedDate** | date | | | No | *None* | | | Change | Drop | More |

Reserved books records:



| ←T→ | | | ▼ | ISBN | Username | ReservedDate |
|---|---|---|---|---|---|---|
| □ | Edit | Copy | Delete | 1138-919749 | cute_person | 2024-12-05 |
| □ | Edit | Copy | Delete | 11uu-932849 | sermimio | 2024-12-06 |
| □ | Edit | Copy | Delete | 9823-98487 | jootie | 2024-12-05 |
| □ | Edit | Copy | Delete | 993-004-00 | cute_person | 2024-12-05 |
| □ | Edit | Copy | Delete | yyu8-92yt49 | cute_person | 2024-12-05 |