

Web Dev Assignment

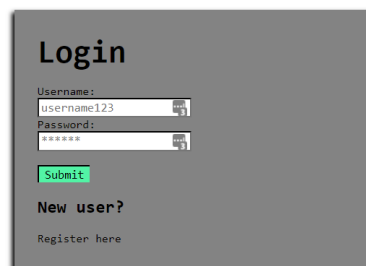
C20381946 – Matthew Tweedy

'HTML/PHP' Files:

index.php

Screenshot:

Welcome to Tweedy's Library



The screenshot shows a login form with the title "Login". It contains two input fields: "Username:" with the value "username123" and "Password:" with masked characters "*****". Below the password field is a green "Submit" button. Underneath the button, it says "New user?" followed by a link "Register here".

Description:

This is the first page the user is shown. A small box is shown in the center prompting the user to enter their login details into a small form. When the user hits the “submit” button, it posts the username and password entered to the login.php file which will handle the login. If they do not have an account, they can click the “Register here” hyperlink which will bring them to a new page (regPage.php) to register.

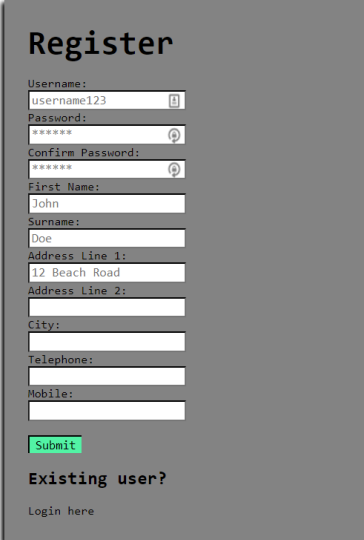
Code:

```
<!DOCTYPE html>
<?php include_once("../scripts/test_connection.php") ?>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="../css/styles.css">
    <title>Library - Login</title>
</head>
<body>
    <header>
        <h1>Welcome to Tweedy's Library</h1>
    </header>
    <div class="form-center">
        <form method="POST" action="../scripts/login.php">
            <h1>Login</h1><br>
            <?php
                if (isset($_GET['error'])) { ?>
                    <p class="error"><?php echo $_GET['error'];?></p>
                <?php }
            ?>
            <label>Username:</label><br>
            <input type="text" name="Username" placeholder="username123"/><br>
            <label>Password:</label><br>
            <input type="password" name="Password"
placeholder="*****"/><br><br>
            <input type="submit" value="Submit"/>
        </form>
        <h2>New user?</h2>
        <a href="regPage.php">Register here</a>
    </div>
    <footer>
        <p>© 2021 Matthew Tweedy</p>
    </footer>
</body>
</html>
```

regPage.php

Screenshot:

Welcome to Tweedy's Library



The screenshot shows a registration form with the following fields and elements:

- Register** (Section Header)
- Username:** Input field containing "username123" with a copy icon.
- Password:** Input field with masked characters "*****" and a toggle icon.
- Confirm Password:** Input field with masked characters "*****" and a toggle icon.
- First Name:** Input field containing "John".
- Surname:** Input field containing "Doe".
- Address Line 1:** Input field containing "12 Beach Road".
- Address Line 2:** Empty input field.
- City:** Empty input field.
- Telephone:** Empty input field.
- Mobile:** Empty input field.
- Submit** (Green button)
- Existing user?** (Section Header)
- [Login here](#) (Text link)

©2021 Matthew Tweedy

Description:

This page displays a form for the user to enter multiple details into which are stored in a table of users for the application. When the “submit” button is pressed, all info entered is taken and sent to the register.php file to handle the registration process and all validation of the user inputs. There is a hyperlink under the register form with a hyperlink back to index.php if the user wishes to return there.

Code:

```

<!DOCTYPE html>
<?php include_once("../scripts/test_connection.php") ?>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="../css/styles.css">
    <title>Library - Register</title>
</head>
<body>
    <header>
        <h1>Welcome to Tweedy's Library</h1>
    </header>
    <div class="form-center">
        <form method="POST" action="../scripts/register.php">
            <h1>Register</h1><br>
            <?php
                if (isset($_GET['error'])) { ?>
                    <p class="error"><?php echo $_GET['error'];?></p>
                <?php }
            ?>
            <label>Username:</label><br>
            <input type="text" name="Username" placeholder="username123"
required/><br>
            <label>Password:</label><br>
            <input type="password" name="Password" placeholder="*****"
required/><br>
            <label>Confirm Password:</label><br>
            <input type="password" name="PwordConfirm" placeholder="*****"
required/><br>
            <label>First Name:</label><br>
            <input type="text" name="FirstName" placeholder="John"
required/><br>
            <label>Surname:</label><br>
            <input type="text" name="Surname" placeholder="Doe" required/><br>
            <label>Address Line 1:</label><br>
            <input type="text" name="AddressLine1" placeholder="12 Beach Road"
required/><br>
            <label>Address Line 2:</label><br>
            <input type="text" name="AddressLine2" required/><br>
            <label>City:</label><br>
            <input type="text" name="City"required/><br>
            <label>Telephone:</label><br>
            <input type="text" name="Telephone" placeholder="" required/><br>
            <label>Mobile:</label><br>
            <input type="text" name="Mobile" placeholder="" required/><br><br>
            <input type="submit" value="Submit"/>

```

```

    </form>
    <h2>Existing user?</h2>
    <a href="index.php">Login here</a>
  </div>
  <footer>
    <p>&#169;2021 Matthew Tweedy</p>
  </footer>
</body>
</html>

```

library.php

Screenshot:

Hello joecrotty
[Logout](#)

Tweedy's Library

[View All of Your Reserved Books](#)

All Books

Displaying 5 of 15 result(s)

ISBN	Book Title	Author	Edition	Year	Category	Available	
093-403992	Computers in Business	Alicia Oneill	3	1997	Biography	No	Reserve Book
23472-8729	Exploring Peru	Stephanie Birchington	4	2005	Travel	No	Reserve Book
237-34823	Business Strategy	Joe Peppard	2	2002	Business	Yes	Reserve Book
23u8-923849	A guide to nutrition	John Thorpe	2	1997	Health	No	Reserve Book
2983-3494	Cooking for children	Anabelle Sharpe	1	2003	Cookery	Yes	Reserve Book

[Previous](#) |
 [Next](#) |
 [Last](#)

©2021 Matthew Tweedy

Description:

The library page is the page displayed to the user directly after they have either logged in or registered. In the top right the current logged in user has their username displayed along with a button that links to logout.php, which would log the user out and then return them to index.php. In the center of the page there is a box containing a search box, a dropdown list to filter by category, a search button, and a link to reservedBooks.php, which is the page that displays all books currently reserved by that user.

Under the search box is the main div containing all records of every book stored in the library, however using pagination, only a max of 5 records are displayed per page, with navigation links at the bottom to allow the user to go from page to page to see all the books. Beside each book record is a button to reserve the book, which links to reserve.php and passes the ISBN of the book, along with its current reservation state.

Code:

```
<!DOCTYPE html>
<?php include_once("../scripts/test_connection.php") ?>
<?php include_once("../scripts/categories.php") ?>
<?php echo "Hello ", $_SESSION['Username'], "<br>"; ?>
<a class="logout" href="../scripts/logout.php">Logout</a>

<?php

if (isset($_SESSION['Username'])) {
?>

    <html lang="en">

        <head>
            <meta charset="UTF-8">
            <meta http-equiv="X-UA-Compatible" content="IE=edge">
            <meta name="viewport" content="width=device-width, initial-scale=1.0">
            <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
            <link rel="stylesheet" href="../css/styles.css">
            <title>Library</title>
        </head>

        <body>
            <header>
                <h1>Tweedy's Library</h1>
            </header>
            <div class="search-box">
                <form action="searchResult.php" method="GET" class="search-form">
                    <input type="search" placeholder="Search book name or
author..." name="search">
```

```

        <label id="filter">Filter</label>
        <select name="categories" id="categories">
            <option value="NULL">Select</option>
            <?php
                while ($row = mysqli_fetch_assoc($result)) {
                    echo "<option value='". $row['CategoryDesc']. "'>";
                    echo ($row['CategoryDesc']);
                    echo "</option>";
                }
            ?>
        </select>
        <button type="submit"><i class="fa fa-
search"></i></button><br><br>
        <a href="reservedBooks.php" id='reserved_books'>View All of
Your Reserved Books</a>
    </form>
</div>
<div class="main">
    <?php
        // calculating offset and limit
        if (isset($_GET['page_num']) && $_GET['page_num'] != "") {
            $page_num = $_GET['page_num'];
        } else {
            $page_num = 1;
        }
        $limit_records = 5;
        $offset = ($page_num - 1) * $limit_records;
        $previous = $page_num - 1;
        $next = $page_num + 1;

        $num_result = mysqli_query($conn, "SELECT COUNT(*) AS
total_records FROM books");
        $sql = "SELECT books.ISBN, books.BookTitle, books.Author,
books.Edition, books.Year, category.CategoryDesc, books.Reserved
FROM books
INNER JOIN category ON books.Category=category.CategoryID
LIMIT $limit_records OFFSET $offset";
        // echo $sql;
        $total_records = mysqli_fetch_array($num_result);
        $total_records = $total_records['total_records'];
        $total_pages = ceil($total_records / $limit_records);

        $result = mysqli_query(
            $conn,
            $sql
        );

        if ($result->num_rows > 0) {

```

```

        echo "<h1>All Books</h1>";
        echo "<h3>Displaying $result->num_rows of $total_records
result(s)</h3>";
        if (isset($_GET['error'])) { ?>
            <p class="error"><?php echo $_GET['error']; ?></p>
        <?php }
        echo "<table class='book-table'>";
        echo "<th>ISBN</th>
            <th>Book Title</th>
            <th>Author</th>
            <th>Edition</th>
            <th>Year</th>
            <th>Category</th>
            <th>Available</th>";
        while ($row = mysqli_fetch_array($result)) {
            $isbn = $row['ISBN'];
            $rflag = $row['Reserved'];
            echo "<tr><td>";
            echo ($isbn);
            echo "</td><td>";
            echo ($row['BookTitle']);
            echo "</td><td>";
            echo ($row['Author']);
            echo "</td><td>";
            echo ($row['Edition']);
            echo "</td><td>";
            echo ($row['Year']);
            echo "</td><td>";
            echo ($row['CategoryDesc']);
            echo "</td>";
            if ($rflag === "Y") {
                echo "<td class='yFlag'>";
                echo "No";
                echo "</td>";
            } else {
                echo "<td class='nFlag'>";
                echo "Yes";
                echo "</td>";
            }
            echo "<td class='reserve'>";
            echo "<a
href='../scripts/reserve.php?isbn=$isbn&rflag=$rflag' class='submit_button'>";
            echo "Reserve Book";
            echo "</a>";
            // }
        }
        echo "</table>\n";
    } else {

```



```

        echo "<h3>Displaying $result->num_rows of $total_records
result(s)</h3>";
        echo "0 results";
    }
    ?>
    <ul class="pagination">
        <?php if ($page_num > 1) {
            echo "<a href='?page_num=1' class='logout'>First Page</a>
|";

        } ?>

        <li <?php if ($page_num <= 1) {
            echo "class='disabled'";
        } ?>>
        <a <?php if ($page_num > 1) {
            echo "href='?page_num=$previous' class='logout'";
        } ?>>Previous</a> |

    </li>

        <li <?php if ($page_num >= $total_pages) {
            echo "class='disabled'";
        } ?>>
        <a <?php if ($page_num < $total_pages) {
            echo "href='?page_num=$next' class='logout'";
        } ?>>Next</a>

    </li>

        <?php if ($page_num < $total_pages) {
            echo "| <li><a href='?page_num=$total_pages'
class='logout'>Last</a></li>";
        }
        $_SESSION['page_num'] = $page_num;
    ?>
    </ul>
</div>
<br>
<br>
<footer>
    <p>&#169;2021 Matthew Tweedy</p>
</footer>
</body>

</html>
<?php
} else {
    header("location: ../html/index.php?error=*you must be logged in to see
this*");
    die();
}

```

```
}  
?>
```

reservedBooks.php

Screenshots:

(with reserved books)

Hello joecrotty
[Logout](#)

Tweedy's Library

[View All Books](#)

Your Reserved Books

Displaying 2 result(s)

ISBN	Book Title	Author	Edition Year		Category	
093-403992	Computers in Business	Alicia Oneill	3	1997	Biography	Remove reservation
23472-8729	Exploring Peru	Stephanie Birchington	4	2005	Travel	Remove reservation

(with no reserved books)

Hello joecrotty
[Logout](#)

Tweedy's Library

[View All Books](#)

Your Reserved Books

Displaying 0 result(s)

You currently have 0 books reserved.
Reserve some [here](#)

©2021 Matthew Tweedy

Description:

This page displays to the user all books they currently have reserved, along with the button next to them that links to the removeRes.php script and passes it the ISBN of the book, allowing the user to remove the reservation for the book. If there are none, it informs the user of this and provides them with another link back to library.php so they can browse the books and reserve the books they want (provided they are not reserved already).

Code:

```
<!DOCTYPE html>
<?php include_once("../scripts/test_connection.php") ?>
```

```

<?php include_once("../scripts/categories.php") ?>
<?php echo "Hello ", $_SESSION['Username'], "<br>"; ?>
<a class="logout" href="../scripts/logout.php">Logout</a>

<?php

if (isset($_SESSION['Username'])) {
    ?>

    <html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
        <link rel="stylesheet" href="../css/styles.css">
        <title>Library</title>
    </head>
    <body>
        <header>
            <h1>Tweedy's Library</h1>
        </header>
        <div class="search-box">
            <form action="searchResult.php" method="GET" class="search-form">
                <a href="library.php" id='reserved_books'>View All Books</a>
            </form>
        </div>
        <div class="main">
            <?php
                $un = $_SESSION['Username'];

                // calculating offset and limit
                if (isset($_GET['page_num']) && $_GET['page_num'] != "") {
                    $page_num = $_GET['page_num'];
                } else {
                    $page_num = 1;
                }
                $limit_records = 5;
                $offset = ($page_num - 1) * $limit_records;
                $previous = $page_num - 1;
                $next = $page_num + 1;

                $sql = "SELECT
                        books.ISBN,
                        books.BookTitle,
                        books.Author,

```

```

        books.Edition,
        books.Year,
        category.CategoryDesc,
        books.Reserved
FROM
    books
INNER JOIN
    category ON books.Category = category.CategoryID
WHERE
    ISBN IN(
        SELECT
            ISBN
        FROM
            reservations
        JOIN users WHERE reservations.Username =
'$un')";

$result = mysqli_query(
    $conn,
    $sql
);

if ($result->num_rows > 0) {
    // output data of each row into a table
    echo "<h1>Your Reserved Books</h1>";
    echo "<h3>Displaying $result->num_rows
result(s)</h3><br>";
    echo "<table class='book-table'>";
    echo "<tr><th>ISBN</th>
        <th>Book Title</th>
        <th>Author</th>
        <th>Edition</th>
        <th>Year</th>
        <th>Category</th>";

    while ($row = $result->fetch_assoc()) {
        $isbn = $row['ISBN'];
        echo "<tr><td>";
        echo ($row['ISBN']);
        echo "</td><td>";
        echo ($row['BookTitle']);
        echo "</td><td>";
        echo ($row['Author']);
        echo "</td><td>";
        echo ($row['Edition']);
        echo "</td><td>";
        echo ($row['Year']);
        echo "</td><td>";
        echo ($row['CategoryDesc']);
    }
}

```

```

        echo "</td>";

        echo "<td class='reserve'>";
        echo "<a href='../scripts/removeRes.php?isbn=$isbn'
class='submit_button'>";
        echo "Remove reservation";
        echo "</a></td>";
    }
    echo "</table>\n";
} else {
    echo "<h1>Your Reserved Books</h1>";
    echo "<h3>Displaying $result->num_rows
result(s)</h3><br>";
    echo "You currently have 0 books reserved.<br>";
    echo "Reserve some <a href='library.php'
class='logout'>here</a>";
}
?>
</div>
<footer>
    <p>&#169;2021 Matthew Tweedy</p>
</footer>
</body>
</html>
<?php
} else {
    header("location: ../html/index.php?error=*you must be logged in to see
this*");
    die();
}
?>

```

searchResult.php

Screenshot:

Hello joecrotty
[Logout](#)

Tweedy's Library

Filter

Select ▼

Q

[View Your Reserved Books](#) | [View All Books](#)

Select

Health

Business

Biography

Technology

Travel

Self-Help

Cookery

Fiction

Results for "computer"

ISBN	Book Title	Author	Edition	Year	Category	Available	
093-403992	Computers in Business	Alicia Oneill	3	1997	Biography	Yes	Reserve Book
82n8-308	computers for idiots	Susan O'Neill	5	1998	Technology	No	Reserve Book

[Previous](#) | [Next](#)

©2021 Matthew Tweedy

Description:

This page contains the same search box as library.php, however it is the page displayed to the user when they have inputted text into the search box, filtered by category or both on the library.php page. The same pagination is used here to display a max of 5 results per page. Under the search box there are two links, one to reservedBooks.php, and the other back to library.php so that the user can navigate about the site without becoming stuck.

Code:

```

<!DOCTYPE html>
<?php include_once("../scripts/test_connection.php") ?>
<?php include_once("../scripts/categories.php") ?>
<?php echo "Hello ", $_SESSION['Username'], "<br>"; ?>
<a class="logout" href="../scripts/logout.php">Logout</a>

<?php

if (isset($_SESSION['Username'])) {
?>

    <html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
        <link rel="stylesheet" href="../css/styles.css">
        <title>Library</title>
    </head>

    <body>
        <header>
            <h1>Tweedy's Library</h1>
        </header>
        <div class="search-box">
            <form action="searchResult.php" method="GET" class="search-form">
                <input type="search" placeholder="Search book name or
author..." name="search">
                <label id="filter">Filter</label>
                <select name="categories" id="categories">
                    <option value="NULL">Select</option>
                    <?php
                    while ($row = mysqli_fetch_assoc($result)) {
                        echo "<option value='" . $row['CategoryDesc'] . "'>";
                        echo ($row['CategoryDesc']);
                        echo "</option>";
                    }
                    ?>
                </select>
                <button type="submit"><i class="fa fa-
search"></i></button><br><br>
                <a href="reservedBooks.php" id='reserved_books'>View Your
Reserved Books</a> |
                <a href="library.php" id='reserved_books'>View All Books</a>
            </form>

```



```

</div>
<div class="main">
    <?php
        // getting user input for search and cleaning it
        if (isset($_GET['search'])) $search = $_GET['search'];
        function validate_input($user_input) {
            $user_input = trim($user_input);
            $user_input = htmlspecialchars($user_input);
            $user_input = htmlentities($user_input);
            $user_input = stripslashes($user_input);

            return $user_input;
        }
        $search = validate_input($search);

        if (isset($_GET['categories'])) $categories = $_GET['categories'];
        if (strlen($search) < 3) {
            $error = "*invalid search*";
        }
        // calculating offset and limit
        if (isset($_GET['page_num']) && $_GET['page_num'] != "") {
            $page_num = $_GET['page_num'];
        } else {
            $page_num = 1;
        }
        $limit_records = 5;
        $offset = ($page_num - 1) * $limit_records;
        $previous = $page_num - 1;
        $next = $page_num + 1;

        $num_result = mysqli_query($conn, "SELECT COUNT(*) AS
total_search_results FROM books WHERE books.BookTitle LIKE '%$search%' or
books.Author LIKE '%$search%'");
        if ($categories == "NULL") {
            $sql = "SELECT
                        books.ISBN,
                        books.BookTitle,
                        books.Author,
                        books.Edition,
                        books.Year,
                        category.CategoryDesc,
                        books.Reserved
                    FROM
                        books
                    INNER JOIN category ON books.Category =
category.CategoryID
                    WHERE

```

```

                                books.Author LIKE '%$search%' OR books.BookTitle
LIKE '%$search%'
                                LIMIT 5 OFFSET 0;";
    } else if (empty($search)) {
        $sql = "SELECT
                                books.ISBN,
                                books.BookTitle,
                                books.Author,
                                books.Edition,
                                books.Year,
                                category.CategoryDesc,
                                books.Reserved
                                FROM
                                    books
                                INNER JOIN category ON books.Category =
category.CategoryID
                                WHERE
                                    category.categoryDesc = '$categories'
                                LIMIT 5 OFFSET 0;";
    } else {
        $sql = "SELECT
                                books.ISBN,
                                books.BookTitle,
                                books.Author,
                                books.Edition,
                                books.Year,
                                category.CategoryDesc,
                                books.Reserved
                                FROM
                                    books
                                INNER JOIN category ON books.Category =
category.CategoryID
                                WHERE
                                    category.categoryDesc = '$categories' AND
books.Author LIKE '%$search%' OR books.BookTitle LIKE '%$search%'
                                LIMIT 5 OFFSET 0; ";
    }
    $total_records = mysqli_fetch_array($num_result);
    $total_records = $total_records['total_search_results'];
    $total_pages = ceil($total_records / $limit_records);

    $result = mysqli_query(
        $conn,
        $sql
    );

    if ($result->num_rows > 0) {
        echo "<h1>Results for \"\$search\"</h1>";
    }

```

```

        if (isset($_GET['error'])) { ?>
            <p class="error"><?php echo $_GET['error']; ?></p>
    <?php }
    echo "<table class='book-table'>";
    echo "<th>ISBN</th>
        <th>Book Title</th>
        <th>Author</th>
        <th>Edition</th>
        <th>Year</th>
        <th>Category</th>
        <th>Available</th>";
    while ($row = mysqli_fetch_array($result)) {
        $isbn = $row['ISBN'];
        $rflag = $row['Reserved'];
        echo "<tr><td>";
        echo ($isbn);
        echo "</td><td>";
        echo ($row['BookTitle']);
        echo "</td><td>";
        echo ($row['Author']);
        echo "</td><td>";
        echo ($row['Edition']);
        echo "</td><td>";
        echo ($row['Year']);
        echo "</td><td>";
        echo ($row['CategoryDesc']);
        echo "</td>";
        if ($rflag === "Y") {
            echo "<td class='yFlag'>";
            echo "No";
            echo "</td>";
        } else {
            echo "<td class='nFlag'>";
            echo "Yes";
            echo "</td>";
        }
        echo "<td class='reserve'>";
        echo "<a
href='../scripts/reserve.php?isbn=$isbn&rflag=$rflag' class='submit_button'>";
        echo "Reserve Book";
        echo "</a>";
        // }
    }
    echo "</table>\n";
} else {
    echo "<h1>Results for \"\$search\"</h1>";
    echo "<h3>Displaying $result->num_rows of $total_records
result(s)</h3>";

```

```

        echo "0 results";
    }
    ?>
    <ul class="pagination">
        <?php if ($page_num > 1) {
            echo "<a href='?page_num=1' class='logout'>First Page</a>
|";

        } ?>

        <li <?php if ($page_num <= 1) {
            echo "class='disabled'";
        } ?>>
        <a <?php if ($page_num > 1) {
            echo "href='?page_num=$previous' class='logout'";
        } ?>>Previous</a> |
    </li>

    <li <?php if ($page_num >= $total_pages) {
        echo "class='disabled'";
    } ?>>
    <a <?php if ($page_num < $total_pages) {
        echo "href='?page_num=$next' class='logout'";
    } ?>>Next</a>
    </li>

    <?php if ($page_num < $total_pages) {
        echo "| <li><a href='?page_num=$total_pages'
class='logout'>Last</a></li>";
    }
    $_SESSION['page_num'] = $page_num;
    ?>
    </ul>
</div>
<footer>
    <p>&#169;2021 Matthew Tweedy</p>
</footer>
</body>
</html>
<?php
} else {
    header("location: ../html/index.php?error=*you must be logged in to see
this*");
    die();
}
?>

```

Script/PHP Files:

login.php

Description:

This php script handles the verification of the user's input for their username and password. The username and password are received from the form on index.php where the user has attempted to log in. These are passed through a function to strip and trailing characters from the end of it as well as normalise any special characters and convert them to HTML entities to prevent any code injection or general mis-input. If the username and/or password are left empty, the user is returned to index.php with an error displayed conveying what went wrong. A select query is made to the database checking to see if there are any users with the username and password matching what the user has entered. If there is exactly one with matching credentials (usernames are primary keys so they must be unique) then a \$_SESSION variable is created and stores the username of the user. The user is now sent to library.php to browse the books. If the username or password doesn't match, it will tell the user one of them is incorrect.

Code:

```
<?php
session_start();
require_once "test_connection.php";

if (isset($_POST['Username']) && isset($_POST['Password'])) {
    $un = $_POST['Username'];
    $p = $_POST['Password'];

    function validate_input($user_input) {
        $user_input = trim($user_input);
        $user_input = htmlspecialchars($user_input);
        $user_input = htmlentities($user_input);
        $user_input = stripslashes($user_input);

        return $user_input;
    }

    $un = validate_input($un);
    $p = validate_input($p);

    if (empty($un)) {
        $_SESSION['error'] = $error;
        header("Location: ../html/index.php?error=*you must enter a username*");
        die();
    } else if (empty($p)) {
        $_SESSION['error'] = $error;
```

```

    header("Location: ../html/index.php?error=*you must enter a password*");
    die();
}

$sql = "SELECT * FROM users WHERE Username='$un' AND Password='$p'";
$result = $conn->query($sql);

if (mysqli_num_rows($result) === 1) {
    $row = mysqli_fetch_assoc($result);
    if ($row['Username'] === $un && $row['Password'] === $p) {
        $_SESSION['Username'] = $row['Username'];
        header("location: ../html/library.php");
        die();
    }
} else {
    header("location: ../html/index.php?error=*username or password is incorrect*");
    die();
}
?>

```

logout.php

Description:

This file simply unsets all \$_SESSION variables stored and then destroys the current session. The user is returned to index.php, now logged out.

Code:

```

<?php
session_start();

session_unset();
session_destroy();

header("location: ../html/index.php");
?>

```

register.php

Description:

This script receives the variables posted from the form on regPage.php and stores them. They are then run through various functions to validate the input, removing any spaces or back slashes at the end of the strings, changing any symbols that could be used to inject code to normalized HTML entities. Checks are then performed on things such as the password and mobile entered to ensure they are the correct lengths, also making sure the

input for mobile number is strictly numerical. If these validations fail, the user is sent back to the register page with an error specifying what was wrong.

If the username inputted does not exist for another user already stored, then all the relevant data is inserted into a new record in the users table in the database and the user is directed to the library.php page.

Code:

```
<?php
session_start();
require_once "test_connection.php";

if (isset($_POST['Username']) && isset($_POST['Password']) &&
isset($_POST['FirstName']) && isset($_POST['Surname']) &&
isset($_POST['AddressLine1']) && isset($_POST['AddressLine2']) &&
isset($_POST['City']) && isset($_POST['Telephone']) &&
isset($_POST['Mobile'])) {
    $un = $_POST['Username'];
    $p = $_POST['Password'];
    $pcfrm = $_POST['PwordConfirm'];
    $fn = $_POST['FirstName'];
    $sn = $_POST['Surname'];
    $al1 = $_POST['AddressLine1'];
    $al2 = $_POST['AddressLine2'];
    $cty = $_POST['City'];
    $tp = $_POST['Telephone'];
    $mb1 = $_POST['Mobile'];

    function validate_input($user_input) {
        $user_input = trim($user_input);
        $user_input = htmlspecialchars($user_input);
        $user_input = htmlentities($user_input);
        $user_input = stripslashes($user_input);

        return $user_input;
    }

    function validate_password_match($user_input, $validate) {
        if ($user_input === $validate) {
            return $user_input;
        } else {
            header("location: ../html/regPage.php?error=*passwords do not
match*");
            die();
        }
    }

    function valdiate_is_num ($user_input, $value) {
```

```

        if (!is_numeric($user_input)) {
            header("location: ../html/regPage.php?error=*$value must only
contain numbers*");
            die();
        }
    }

function validate_correct_length($user_input, $length, $value) {
    $num_len = strlen((string)$user_input);
    if ($num_len === $length) {
        return $user_input;
    } else {
        header("location: ../html/regPage.php?error=*$value must have
length of $length*");
        die();
    }
}

// validating input
$un = validate_input($un);
$p = validate_input($p);
$pcfrm = validate_input($pcfrm);
$fn = validate_input($fn);
$sn = validate_input($sn);
$a11 = validate_input($a11);
$a12 = validate_input($a12);
$cty = validate_input($cty);
$tp = validate_input($tp);
$mbl = validate_input($mbl);

// checking password and confirm password are a match
$p = validate_password_match($p, $pcfrm);

// checking password and mobile are correct lengths
$p = validate_correct_length($p, 6, "password");
$mbl = validate_correct_length($mbl, 10, "mobile");

// checking telephone and mobile only contain numbers
$tp = validate_is_num($tp, "telephone");
$mbl = validate_is_num($mbl, "mobile");

// checking username doesn't already exist
$sql = "SELECT * FROM users WHERE Username='$un'";
$result = $conn->query($sql);
if (mysqli_num_rows($result) === 1) {
    $row = mysqli_fetch_assoc($result);
    if ($row['Username'] === $un) {

```



```

        header("location: ../html/regPage.php?error=*username is already
taken*");
    }
}

$sql = "INSERT INTO users (Username, Password, FirstName, Surname,
AddressLine1, AddressLine2, City, Telephone, Mobile) VALUES ('$un', '$p',
'$fn', '$sn', '$al1', '$al2', '$cty', '$tp', '$mbl')";
if ($conn->query($sql) === TRUE) {
    $_SESSION['Username'] = $un;
    header("location: ../html/library.php");
    die();
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
}
?>

```

reserve.php

Description:

This script is called when the user clicks the “Reserve Book” button in either library.php or searchResult.php. It receives the ISBN and current reservation state from the \$_GET method as the “Reserve Book” button includes these in the URL as it redirects to the script.

If the book currently has a “Y” value for the reserved flag, the user is sent back to library.php with an error telling them that book has already been reserved. If the book does not have this flag, an INSERT statement adds the ISBN of the book, the user’s username and the current date into the reservations table. If this is successfully inserted, an UPDATE statement is used to change the reserved flag of the book from “N” to “Y”, to indicate it has been reserved and to prevent other users from double reserving it. The user is then returned to the same page they were on previously (library.php).

Code:

```

<?php
session_start();
require_once "test_connection.php";

$isbn = $_GET['isbn'];
$un = $_SESSION['Username'];
$rflag = $_GET['rflag'];
$page_num = $_SESSION['page_num'];

if ($rflag === 'Y') {
    header("Location: ../html/library.php?error=*book is already reserved*");
}

```

```

} else {
    $sql = "INSERT INTO reservations (ISBN, Username, ReservedDate) VALUES
('$isbn', '$un', now())";
    if ($conn->query($sql) === TRUE) {
        echo "New record created successfully";
        echo "<pre>\n$sql\n</pre>\n";

        $sql = "UPDATE books SET Reserved = 'Y' WHERE ISBN = '$isbn'";
        if ($conn->query($sql) === TRUE) {
            echo "Record updated successfully";
            echo "<pre>\n$sql\n</pre>\n";
            header("location: ../html/library.php?page_num=$page_num");
            return;
        } else {
            echo "Error: " . $sql . "<br>" . $conn->error;
        }
        return;
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
}
?>

```

removeRes.php

Description:

This script does the opposite of reserve.php. It is called when the user clicks the “Remove Reservation” button on reservedBooks.php and is passed the ISBN of the book the user wishes to remove from their reservation list.

A DELETE statement is then used to remove the record of the book from the reservations table. If this successfully runs, the UPDATE query then changes the reserved flag from “Y” to “N” to allow other users to reserve it. The user is then returned to reservedBooks.php.

Code:

```

<?php
session_start();
require_once "test_connection.php";

$isbn = $_GET['isbn'];
$un = $_SESSION['Username'];

$sql = "DELETE FROM reservations WHERE ISBN = '$isbn'";
if ($conn->query($sql) === TRUE) {
    echo "Record deleted successfully";
    echo "<pre>\n$sql\n</pre>\n";
}

```

```

$sql = "UPDATE books SET Reserved = 'N' WHERE ISBN = '$isbn'";
if ($conn->query($sql) === TRUE) {
    echo "Record updated successfully";
    echo "<pre>\n$sql\n</pre>\n";
    header("location: ../html/reservedBooks.php");
    return;
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
return;
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
?>

```

categories.php

Description:

This script queries the database for everything stored inside the category table for use in displaying the categories in the dropdown list to filter searches.

Code:

```

<?php
session_start();

$sql = "SELECT * FROM category";
$result = $conn->query($sql);
?>

```

test_connection.php

Description:

This script stores the information needed to connect to the database into separate variables, and then creates a new connection to it. If the connection fails, it will display the resulting connection error.

Code:

```

<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "WebD_assignment";

```

```
// create connection
$conn = mysqli_connect($servername,$username,$password,$dbname);

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

CSS File:

style.css

Description:

This file is used to alter the appearance and layout of the html website. I made the choice of keeping the layout simple and easy to navigate, with it being stylized quite similarly to early 2000's websites.

Code:

```
body{ height:100vh; margin:8px; }

header{ min-height: 100px;}
footer{ min-height: 50px;}

body {
    background-color: #ffffff;
    font-family: monospace;
    display: flex;
    flex-direction: column;
}

header {
    text-align: center;
    border-bottom: 3px solid #000;
}

footer {
    text-align: center;
    margin-top:auto;
    border-top: 3px solid #000;
}

h1 {
    font-size: 6.8ch;
}
```

```
.error {
  color: #ffffff;
}

.logout {
  color: rgb(17, 243, 168);
  text-decoration: none;
}

a:hover {
  color: #000;
}

.form-center {
  max-width: 350px;
  width: 40%;
  min-width: 250px;
  margin: 10vh auto;
  background-color: rgb(131, 131, 131);
  padding: 2%;
  box-shadow: -3px 3px 5px #000;
}

.form-center h1 {
  margin: auto;
  font-size: 5ch;
}

.form-center input {
  border-top: solid 2px #000;
}

.form-center input::placeholder {
  font-family: monospace;
}

.form-center input[type=submit] {
  /* border-top: none; */
  border-left: solid 2px #000;
  border-right: none;
  border-bottom: none;
  font-family: monospace;
  background-color: rgb(17, 243, 168);
}

.form-center input[type=submit]:active {
  border-right: solid 2px #000;
  border-bottom: solid 2px #000;
```

```
border-top: none;
border-left: none;
font-family: monospace;
background-color: rgb(17, 243, 168);
}

.form-center a , .search-box a{
  color: #000;
  text-decoration: none;
}

#reserved_books {
  font-size: 2.2ch;
}

.form-center a:hover , .search-box a:hover {
  color: rgb(17, 243, 168);
}

.search-box {
  /* width: 21vw; */
  width: 50%;
  margin: 4vh auto;
  margin-bottom: 2.4vh;
  background-color: rgb(131, 131, 131);
  padding: 4%;
  box-shadow: -3px 3px 5px #000;
}

.search-form form {
  width: 400px;
  margin: auto;
  margin-top: 250px;
}

.search-form input {
  padding: 4px 10px;
  border: 0;
  font-size: 16px;
}

.search-form input[type=search] {
  border-right: none;
  border-bottom: none;
  border-top: solid 2px #000;
  border-left: solid 2px #000;
  font-family: monospace;
  font-size: 17px;
```

```
width: 75%;
/* width: 54%; */
min-width: 15%;
}

@media only screen and (max-width: 2148px) {
  .search-form input[type=search] {
    width: 70%;
  }
}

@media only screen and (max-width: 1920px) {
  .search-form input[type=search] {
    width: 65%;
  }
}

@media only screen and (max-width: 1400px) {
  .search-form input[type=search] {
    width: 55%;
  }
}

@media only screen and (max-width: 1010) {
  .search-form input[type=search] {
    width: 45%;
  }
}

.search-form #categories {
  font-family: monospace;
  width: 12%;
  min-width: 100px;
  padding: 4px;
  border: 0;
  font-size: 16px;
}

.search-form button {
  font-size: 17px;
  padding: 4px 10px;
  cursor: pointer;
  background: #ddd;
}

.search-form button:hover {
  background: rgb(17, 243, 168);
}
```

```
#filter {
    font-size: 17px;
    padding: 4px 10px;
    background: #ddd;
}

.main {
    background-color: rgb(131, 131, 131);
    padding: 2%;
    width: 90%;
    margin: auto;
    box-shadow: -3px 3px 5px #000;
}

.book-table {
    width: 100%;
}

.book-table th {
    font-size: 3.4ch;
}

.book-table td{
    /* border: solid 2px black; */
    border-top: solid 2px #000;
    border-left: solid 2px #000;
    border-right: none;
    border-bottom: none;
    background-color: #ffffff;
    padding: 5px;
    padding-top: 40px;
    font-size: 3ch;
}

.book-table .yFlag {
    background-color: #f33333;
}

.book-table .nFlag {
    background-color: #59f333;
}

.book-table .reserve {
    border: none;
    background-color: rgb(131, 131, 131);
    margin: auto;
}
```



```
.reserve a {
  border-top: solid 2px #000;
  border-left: solid 2px #000;
  border-right: none;
  border-bottom: none;
  font-size: 1.2ch;
  padding: 6px;
  color: #000;
  float: right;
  text-decoration: none;
  font-family: monospace;
  background-color: rgb(17, 243, 168);
  margin: auto;
  margin-bottom: 20%;
}

.reserve a:hover {
  background-color: rgb(22, 226, 158);
}

.reserve a:active {
  border-right: solid 2px #000;
  border-bottom: solid 2px #000;
  border-top: none;
  border-left: none;
}

.pagination {
  padding: 0;
  list-style-type: none;
  font-size: 16px;
}

.pagination li {
  display: inline;
}

#filter {
  font-size: 17px;
  padding: 4px 10px;
  background: #ddd;
}
```

Design Document:

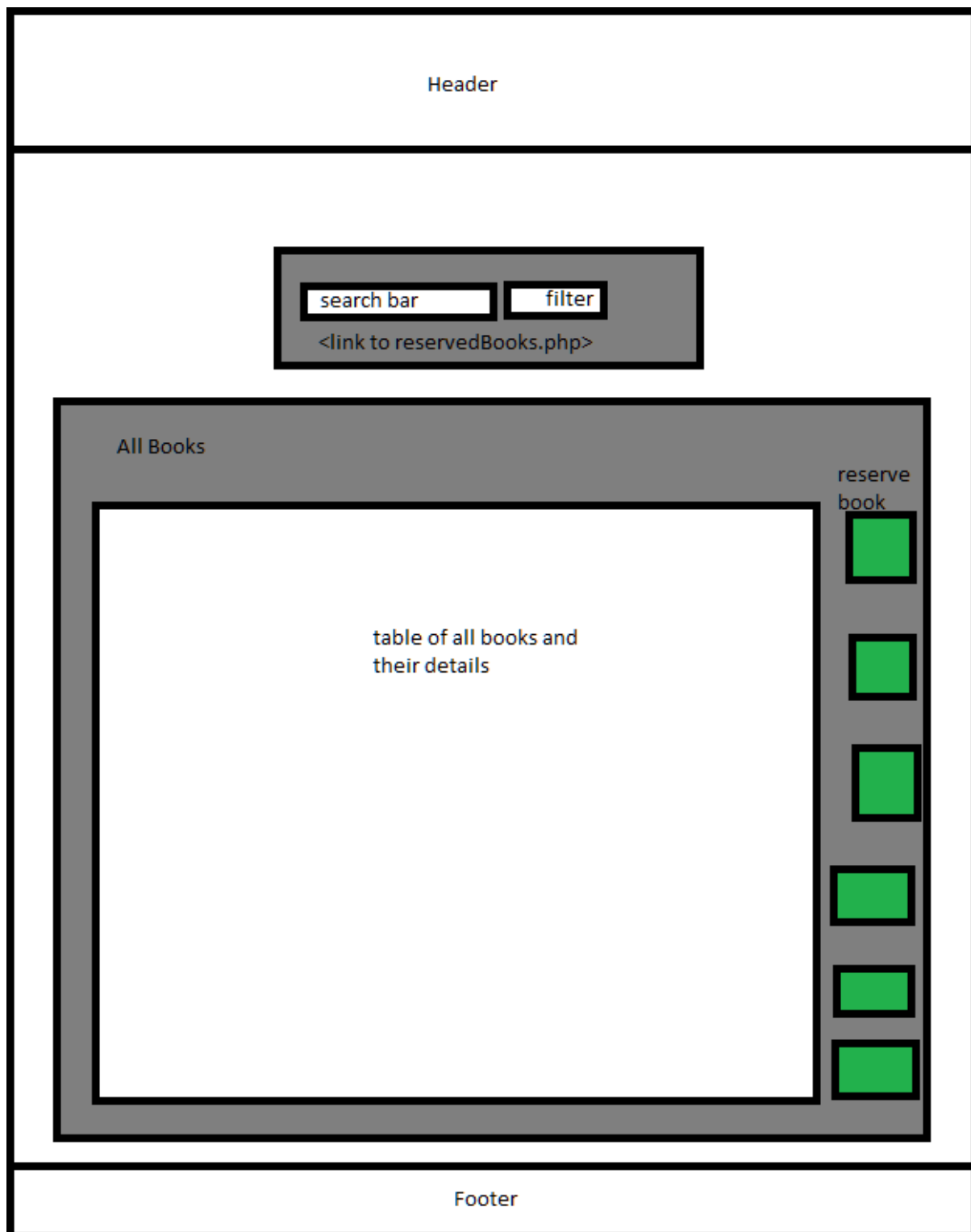
index.php:

Header
<div><div>Login</div><div><input type="text" value="username"/></div><div><input type="password" value="password"/></div><div><input type="submit" value="submit"/></div><div><link to regPage.php></div></div>
Footer

regPage.php:

Header
<div><div>Register</div><div><div>register form</div><div>details</div></div><div><div>submit</div><div><link to index.php></div></div></div>
Footer

library.php:



searchResult.php:

Header	
<div><div>search barfilter</div><div><link to reservedBooks.php></div></div>	
<div><div>Results for search</div><div>reserve</div><div><div>table of all books that meet search criteria and their details</div><div><div></div><div></div><div></div><div></div><div></div><div></div></div></div></div>	
Footer	

reservedBooks.php:

