Date of submission: 28 / 03 / 2025
Batch: A1 Roll No.: 16010123012
Div: A
Student Name: Aaryan Sharma
Experiment No: 6
Staff In-charge:

TITLE: Demonstrate the use of Cookies and Sessions in PHP.

AIM: To Understand the Difference between cookies & sessions.

To Understand how to create & destroy session /Cookies variables.

Expected Outcome of Experiment:
The expected outcomes aim to understand the differences between cookies and session with respect to storage, persistence, data accessibility, scope and security.

Books/ Journals/ Websites referred:

Steve Prettyman ,"Learn PHP 8 Using MySQL, JavaScript, CSS3, and HTML5", Apress 2nd / 2020 edition.

- **Problem Statement:** Design and implement an application to demonstrate HTML form integration with PHP script for experiment 1 to

  1. Create, modify with timer and delete cookies for a user.

  a. A cookie is created with the **setcookie( )** function.

  Syntax: **setcookie(name, value, expire, path, domain, secure, httponly);**

  bool setcookie (string $name [, string $value [, int $expire = 0 [, string $path [, string $domain [, bool $secure = false [, bool $httponly = false]]]]]])

  e.g. setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/");

  b. Retrieve the value of the cookie "user" (using the global variable **$_COOKIE**) and also use the **isset()** function to find out if the cookie is set.

c. modify the user value= other user with **setcookie()** function

d. delete the Cookie with the setcookie() function with an expiration date in the past

e.g. setcookie ("CookieName", "", time() -3600);

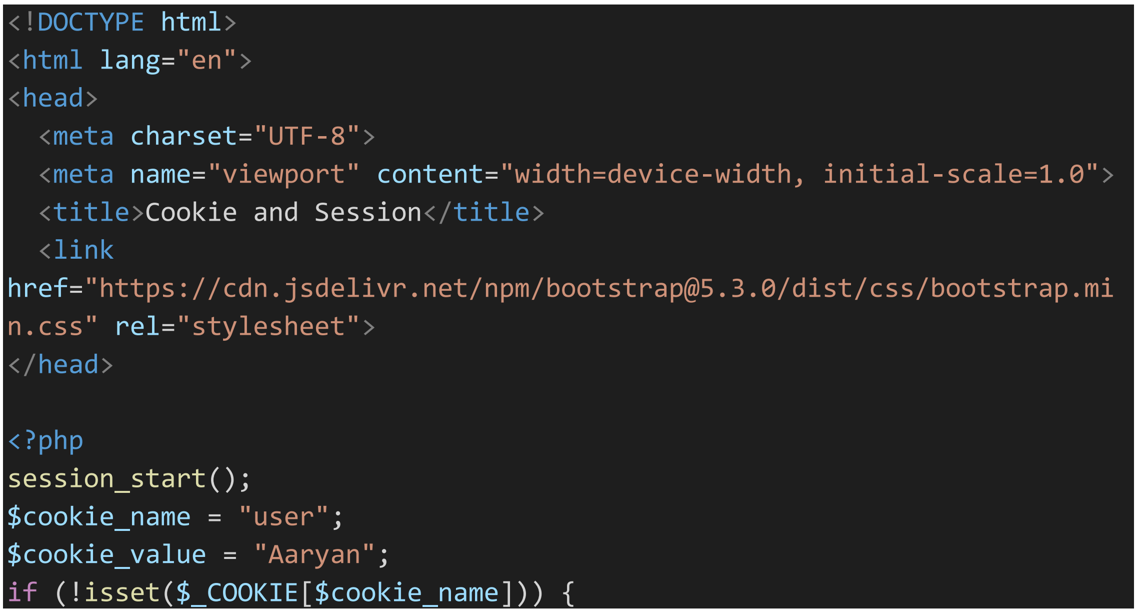
- **2.** Handle PHP session (which is used to store and pass information from one page to another temporarily (until user closes the website))

  a. Start PHP Session, get session variable values, modify session variable, destroy PHP session.

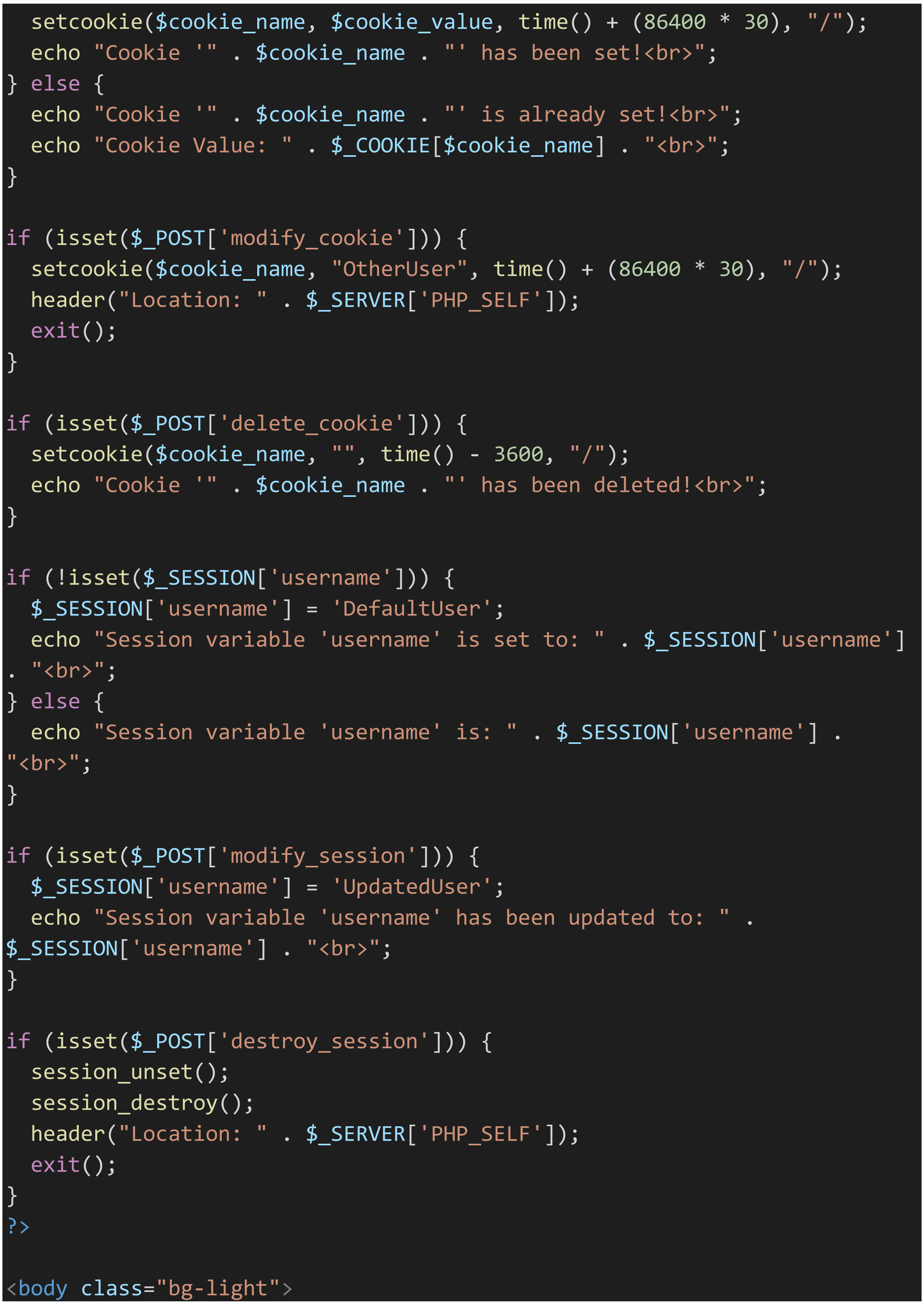
  session_start() function : A session is started with the session_start() function. Session variables are set with the PHP global variable: $_SESSION. Get session variable values, modify session variable with $_SESSION.

  b. To remove all global session variables and destroy the session, use **session_unset()** and **session_destroy():**

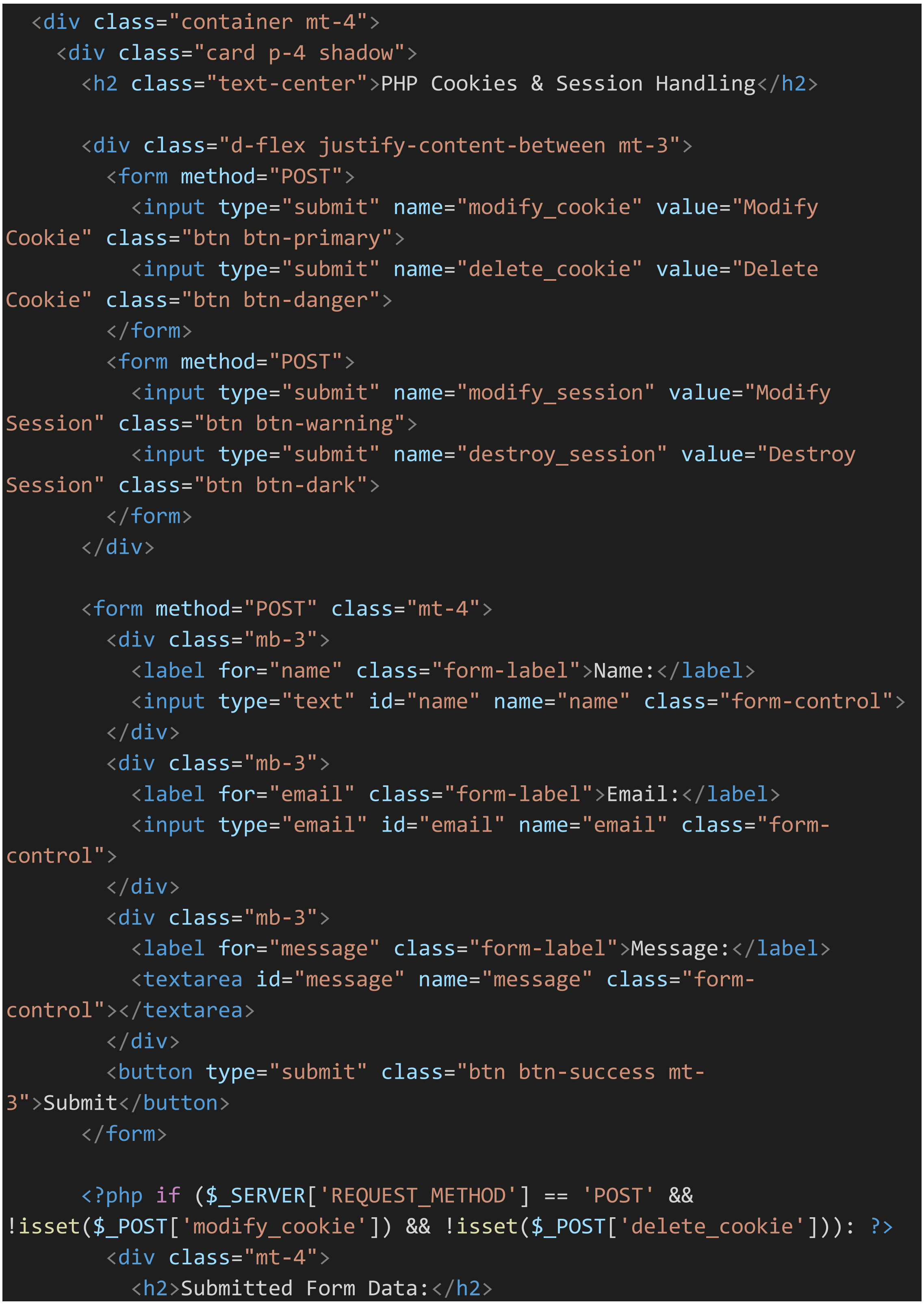
Implementation and screenshots of output



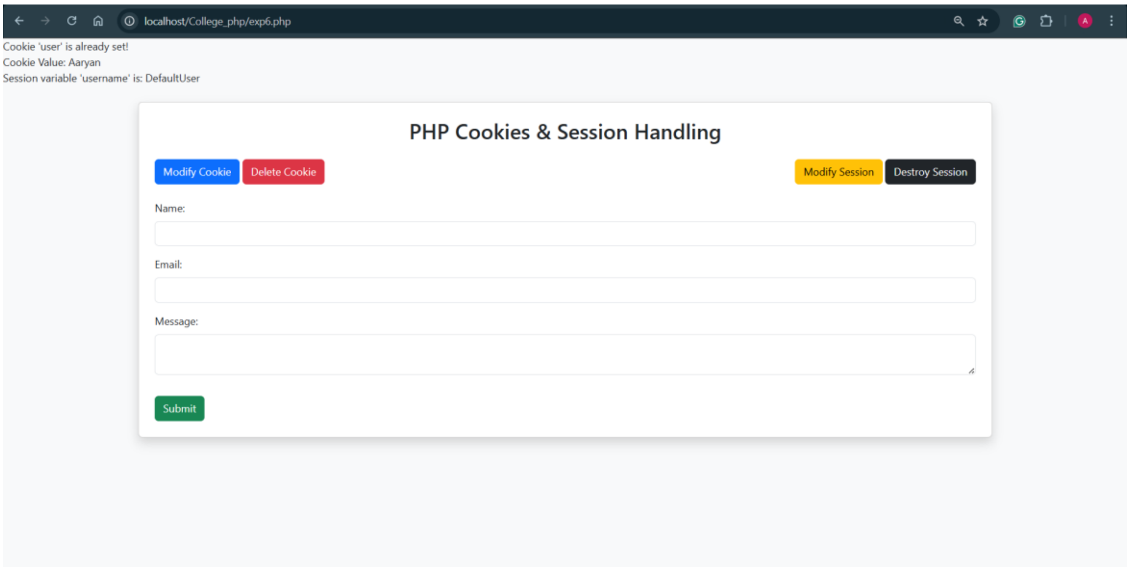
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Cookie and Session</title>
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.mi
n.css" rel="stylesheet">
</head>

<?php
session_start();
$cookie_name = "user";
$cookie_value = "Aaryan";
if (!isset($_COOKIE[$cookie_name])) {
```



```
  setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/");
  echo "Cookie '" . $cookie_name . "' has been set!<br>";
} else {
  echo "Cookie '" . $cookie_name . "' is already set!<br>";
  echo "Cookie Value: " . $_COOKIE[$cookie_name] . "<br>";
}

if (isset($_POST['modify_cookie'])) {
  setcookie($cookie_name, "OtherUser", time() + (86400 * 30), "/");
  header("Location: " . $_SERVER['PHP_SELF']);
  exit();
}

if (isset($_POST['delete_cookie'])) {
  setcookie($cookie_name, "", time() - 3600, "/");
  echo "Cookie '" . $cookie_name . "' has been deleted!<br>";
}

if (!isset($_SESSION['username'])) {
  $_SESSION['username'] = 'DefaultUser';
  echo "Session variable 'username' is set to: " . $_SESSION['username']
. "<br>";
} else {
  echo "Session variable 'username' is: " . $_SESSION['username'] .
"<br>";
}

if (isset($_POST['modify_session'])) {
  $_SESSION['username'] = 'UpdatedUser';
  echo "Session variable 'username' has been updated to: " .
$_SESSION['username'] . "<br>";
}

if (isset($_POST['destroy_session'])) {
  session_unset();
  session_destroy();
  header("Location: " . $_SERVER['PHP_SELF']);
  exit();
}
?>

<body class="bg-light">
```



```
  <div class="container mt-4">
    <div class="card p-4 shadow">
      <h2 class="text-center">PHP Cookies & Session Handling</h2>

      <div class="d-flex justify-content-between mt-3">
        <form method="POST">
          <input type="submit" name="modify_cookie" value="Modify
Cookie" class="btn btn-primary">
          <input type="submit" name="delete_cookie" value="Delete
Cookie" class="btn btn-danger">
        </form>
        <form method="POST">
          <input type="submit" name="modify_session" value="Modify
Session" class="btn btn-warning">
          <input type="submit" name="destroy_session" value="Destroy
Session" class="btn btn-dark">
        </form>
      </div>

      <form method="POST" class="mt-4">
        <div class="mb-3">
          <label for="name" class="form-label">Name:</label>
          <input type="text" id="name" name="name" class="form-control">
        </div>
        <div class="mb-3">
          <label for="email" class="form-label">Email:</label>
          <input type="email" id="email" name="email" class="form-
control">
        </div>
        <div class="mb-3">
          <label for="message" class="form-label">Message:</label>
          <textarea id="message" name="message" class="form-
control"></textarea>
        </div>
        <button type="submit" class="btn btn-success mt-
3">Submit</button>
      </form>

      <?php if ($_SERVER['REQUEST_METHOD'] == 'POST' &&
!isset($_POST['modify_cookie']) && !isset($_POST['delete_cookie'])): ?>
        <div class="mt-4">
          <h2>Submitted Form Data:</h2>
          <p><strong>Name:</strong> <?= isset($_POST['name']) ? htmlspecialchars($_POST['name']) : 'N/A' ?></p>
          <p><strong>Email:</strong> <?= isset($_POST['email']) ? htmlspecialchars($_POST['email']) : 'N/A' ?></p>
          <p><strong>Message:</strong> <?= isset($_POST['message']) ? htmlspecialchars($_POST['message']) : 'N/A' ?></p>
        </div>
      <?php endif; ?>
    </div>
  </div>

  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```



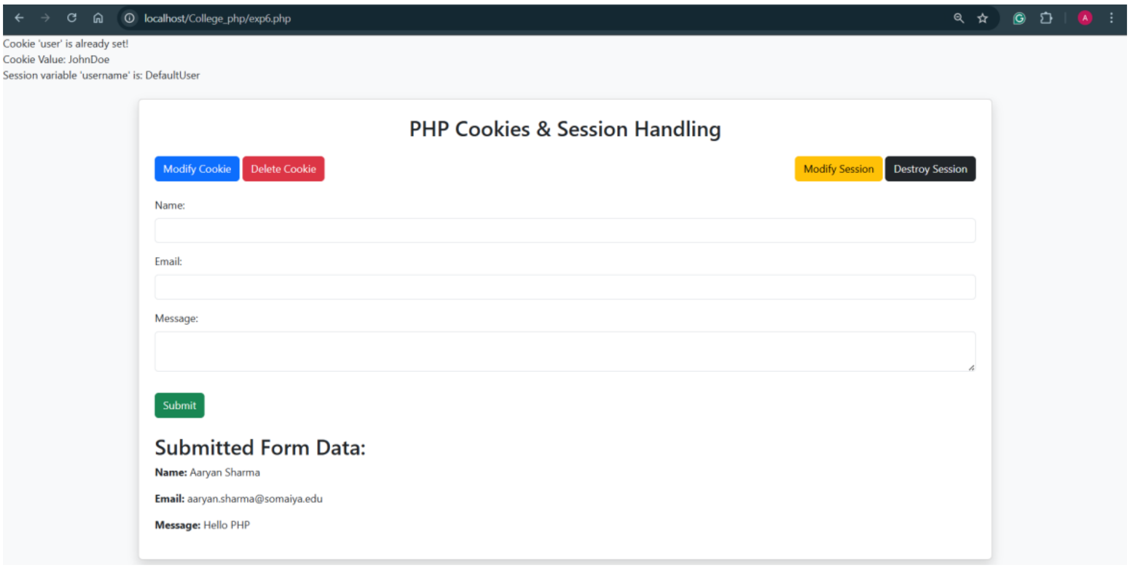
```
Cookie 'user' is already set!
Cookie Value: Aaryan
Session variable 'username' is: DefaultUser
```

PHP Cookies & Session Handling

Name:

Email:

Message:



Cookie 'user' is already set!
Cookie Value: JohnDoe
Session variable 'username' is: DefaultUser

PHP Cookies & Session Handling

Name:

Email:

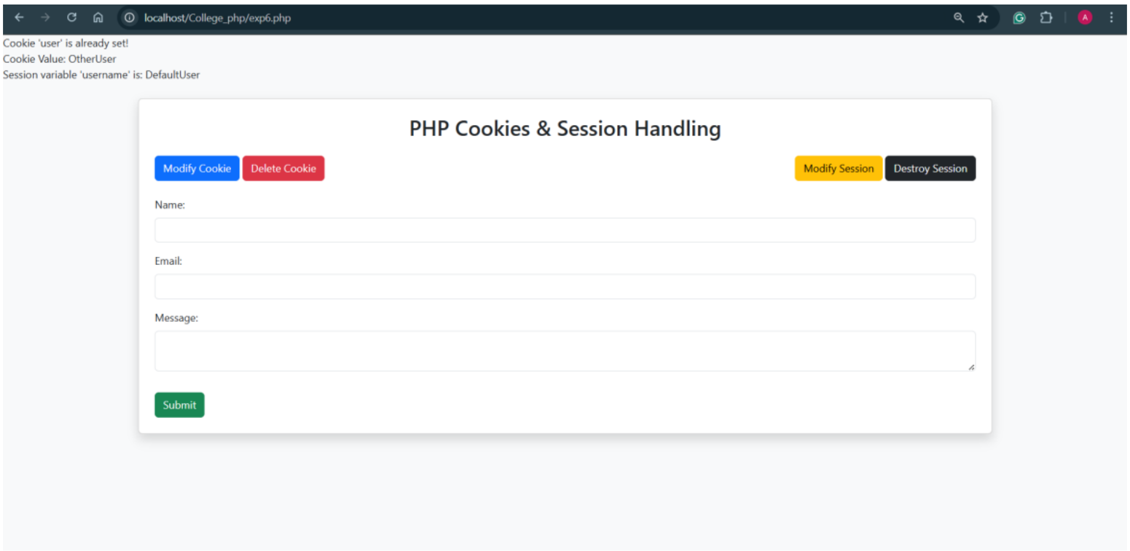
Message:

Submitted Form Data:

Name: Aaryan Sharma

Email: aaryan.sharma@somaiya.edu

Message: Hello PHP



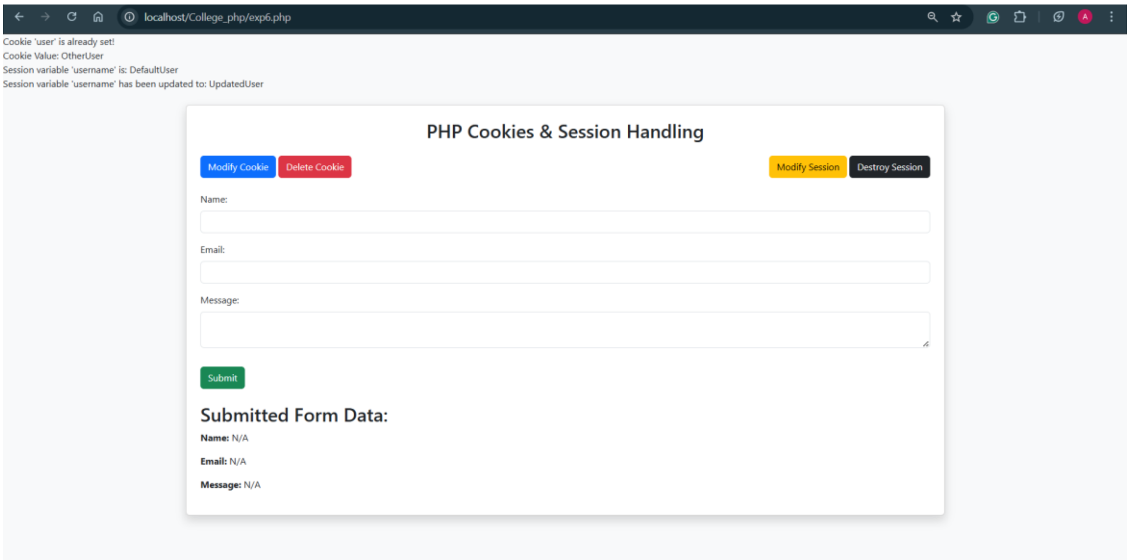
Cookie 'user' is already set!
Cookie Value: OtherUser
Session variable 'username' is: DefaultUser

PHP Cookies & Session Handling

Name:

Email:

Message:



```
Cookie 'user' is already set!
Cookie Value: OtherUser
Session variable 'username' is: DefaultUser
Session variable 'username' has been updated to: UpdatedUser
```

PHP Cookies & Session Handling

Name:

Email:

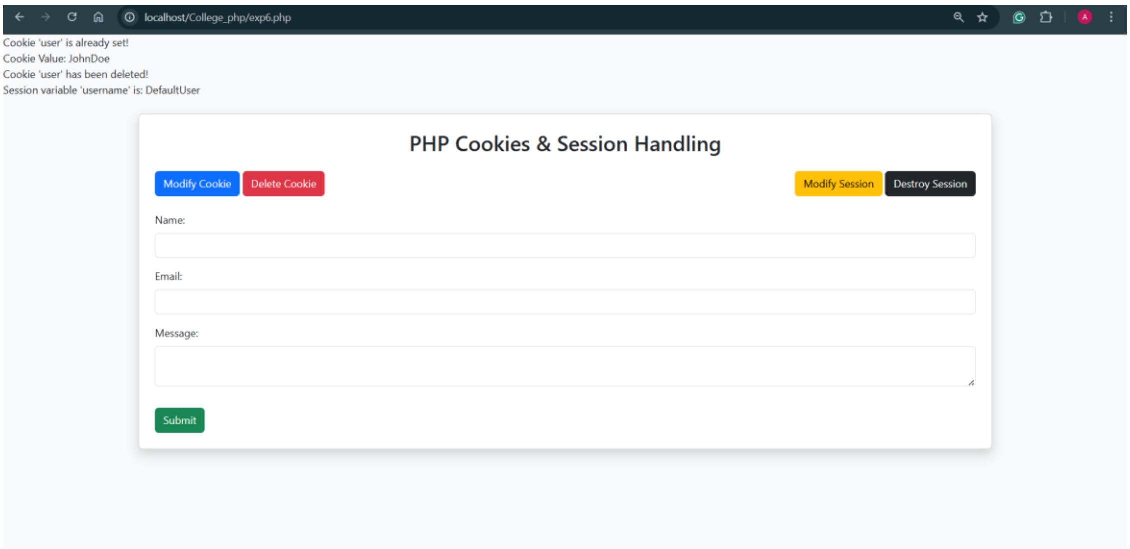
Message:

Submitted Form Data:

Name: N/A

Email: N/A

Message: N/A



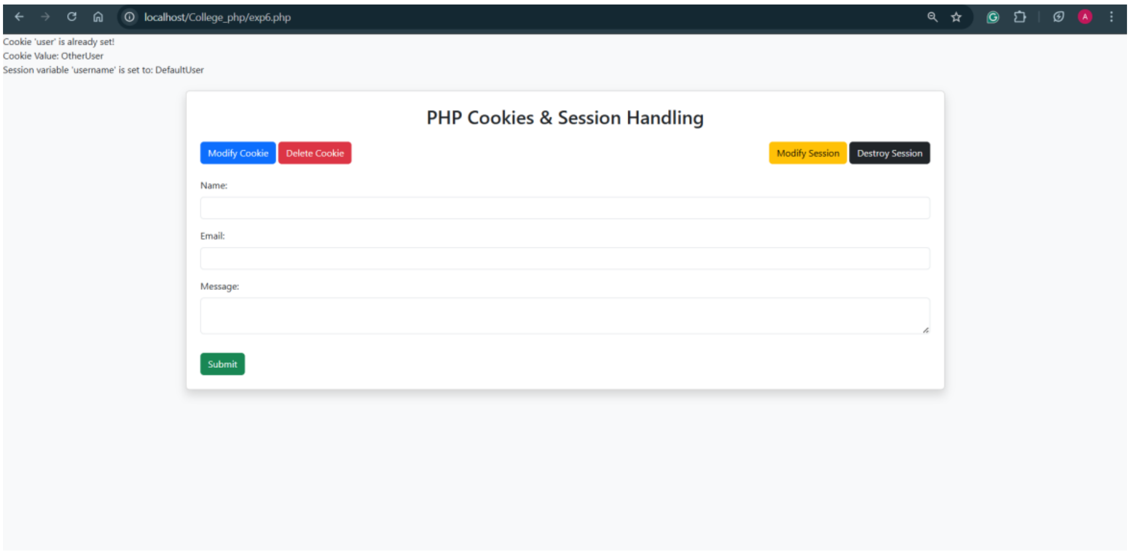
```
Cookie 'user' is already set!
Cookie Value: JohnDoe
Cookie 'user' has been deleted!
Session variable 'username' is: DefaultUser
```

PHP Cookies & Session Handling

Name:

Email:

Message:



```
Cookie 'user' is already set!
Cookie Value: OtherUser
Session variable 'username' is set to: DefaultUser
```

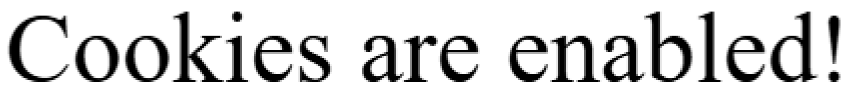
<u>Post Lab questions: Answer the following.</u>

1. Write the differences between cookies and sessions considering storage, persistence, data accessibility, scope and security as points of comparison.

Feature	Cookies	Sessions
Storage	Stored on the client-side (browser)	Stored on the server-side
Persistence	Can persist even after closing the browser (based on expiration)	Exists only until the browser is closed or session is destroyed
Data Accessibility	Accessible by both server and client (can be modified by users)	Only server-side can access and modify session data
Scope	Shared across multiple requests/pages as long as the cookie is valid	Available only within a specific session (i.e., until the user session ends)
Security	Less secure because users can view and edit cookies	More secure since session data is stored on the server and not exposed to users

2. How to check if cookies are enabled or not?

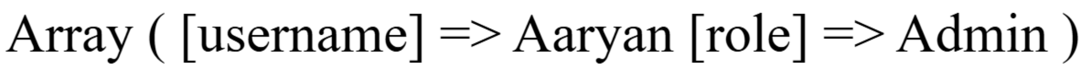
```php
<?php
setcookie("test_cookie", "test", time() + 3600, "/");

if (count($_COOKIE) > 0) {
  echo "Cookies are enabled!";
} else {
  echo "Cookies are disabled!";
}
```



Cookies are enabled!

3. How to get all session variable values? Explain with an example code.
   To retrieve all session variables, use the `$_SESSION` array:

```php
<?php
session_start();

$_SESSION["username"] = "Aaryan";
$_SESSION["role"] = "Admin";

print_r($_SESSION);
?>
```



```
Array ( [username] => Aaryan [role] => Admin )
```

Conclusion:

I have successfully learned and implemented the concepts of cookies and sessions in PHP. Through this experiment, I understood how cookies store client-side data for persistence, while sessions maintain user data securely on the server. I explored creating, modifying, and deleting cookies, as well as handling session variables efficiently. This knowledge will help in designing secure and optimized web applications with effective user data management.