K. J. Somaiya School of Engineering, Mumbai-77

(Somaiya Vidyavihar University)

Department of Computer Engineering

Date of submission: 31-03-2025	
Batch: E-2	**Roll No.: 16010123325**
Div:	
Student Name: Shreyans Tatiya	
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.

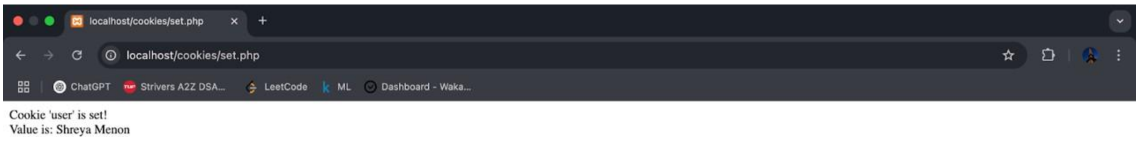
  session_start() function : A session is started with the session_start() function. Session variables are set with the PHP global variable: $_SESSION. Get session variable values, modify session variable with $_SESSION.

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

Implementation and screenshots of output

Cookies

```
<?php
$cookie_name = "user";
$cookie_value = "Shreyans Tatiya";

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

?>
<html>
<body>

<?php
if(!isset($_COOKIE[$cookie_name])) {
 echo "Cookie named '" . $cookie_name . "' is not set! Refresh the page.";
} else {
 echo "Cookie '" . $cookie_name . "' is set!<br>";
 echo "Value is: " . $_COOKIE[$cookie_name];
}
?>

</body>
</html>
```

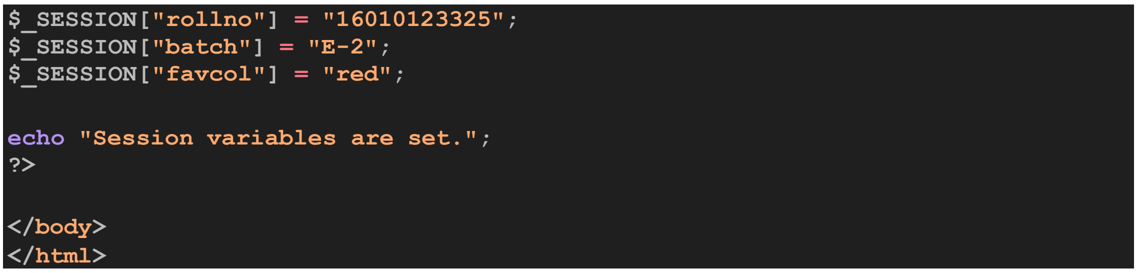


```
Cookie 'user' is set!
Value is: Shreya Menon
```

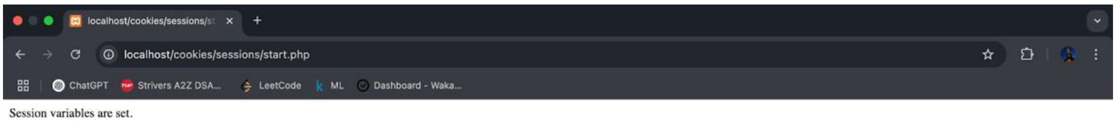
Sessions

start

```
<?php
// Start the session
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// Set session variables
$_SESSION["name"] = "Shreyans";
```

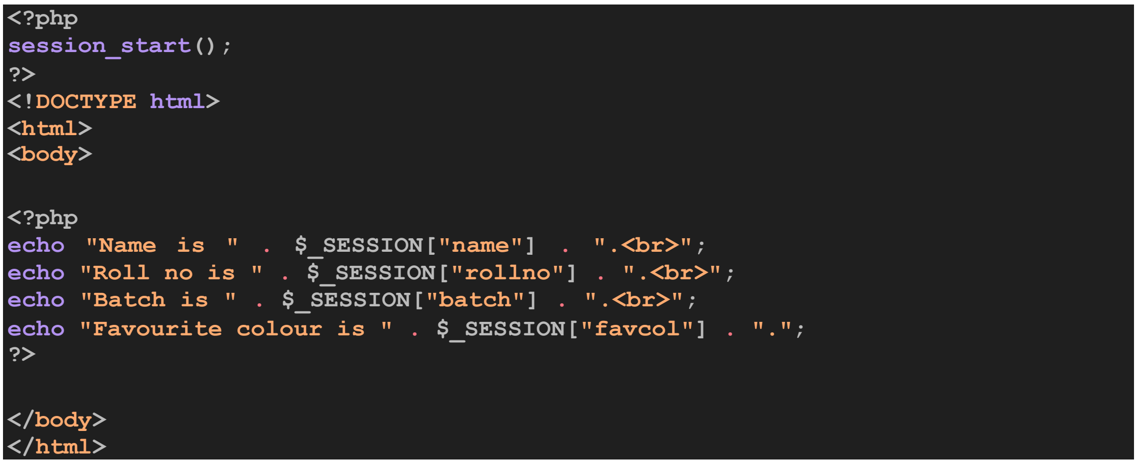


```
$_SESSION["rollno"] = "16010123325";
$_SESSION["batch"] = "E-2";
$_SESSION["favcol"] = "red";

echo "Session variables are set.";
?>

</body>
</html>
```

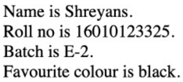

localhost/cookies/sessions/st
localhost/cookies/sessions/start.php
ChatGPT
Strivers A2Z DSA...
LeetCode
ML
Dashboard - Waka...

Session variables are set.

variables echo



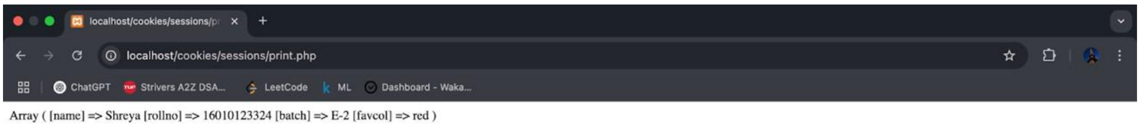
```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
echo  "Name is " . $_SESSION["name"] . ".<br>";
echo "Roll no is " . $_SESSION["rollno"] . ".<br>";
echo "Batch is " . $_SESSION["batch"] . ".<br>";
echo "Favourite colour is " . $_SESSION["favcol"] . ".";
?>

</body>
</html>
```



Name is Shreyans.
Roll no is 16010123325.
Batch is E-2.
Favourite colour is black.

print

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
print_r($_SESSION);
?>


</body>
</html>
```

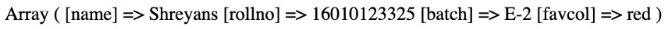

localhost/cookies/sessions/pr
localhost/cookies/sessions/print.php
ChatGPT
Strivers A2Z DSA...
LeetCode
ML
Dashboard - Waka...

```
Array ( [name] => Shreya [rollno] => 16010123324 [batch] => E-2 [favcol] => red )
```

Array ([name] => Shreyans [rollno] => 16010123325 [batch] => E-2 [favcol] => red)

modify

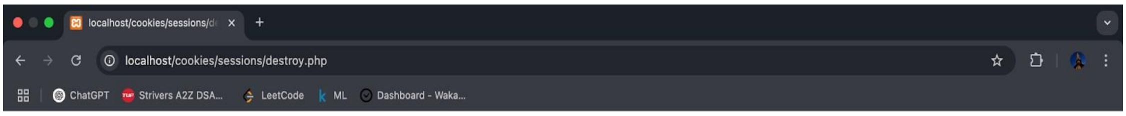
```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
$_SESSION["favcol"] = "blue";
print_r($_SESSION);
?>

</body>
</html>
```



```
Array ( [name] => Shreyans [rollno] => 16010123325 [batch] => E-2 [favcol] => red )
```

destroy

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// remove all session variables
session_unset();

// destroy the session
session_destroy();
?>

</body>
</html>
```


localhost/cookies/sessions/d
localhost/cookies/sessions/destroy.php
ChatGPT
Strivers A2Z DSA...
LeetCode
ML
Dashboard - Waka...

Post Lab questions: Answer the following.

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

Criteria	Cookies	Sessions
Storage	Stored on the user's browser	Stored on the server
Persistence	Can persist beyond browser sessions based on expiration time	Lasts only while the session is active (until the browser is closed or session expires)
Data Accessibility	Accessible by both client and server	Accessible only by the server
Scope	Available across multiple pages until expired or deleted	Available only during the user's session
Security	Less secure as data is stored on the client-side	More secure since data is stored on the server

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

   You can check if cookies are enabled in PHP using the following approach:

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

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

3. How to get all session variable values? Explain with an example code.

   In PHP, session variables are stored in the $_SESSION superglobal array. You can retrieve all session variables using print_r() or var_dump().

```
session_start();

// Set session variables
$_SESSION["username"] = "JohnDoe";
$_SESSION["email"] = "johndoe@example.com";
```

```
// Get all session variables
echo "<pre>";
print_r($_SESSION);
echo "</pre>";
```

Conclusion:

Cookies and sessions are both used to store user data, but they differ in storage location, persistence, security, and accessibility. Cookies are client-side and can persist, whereas sessions are server-side and more secure. Checking for cookies and retrieving session variables are crucial in web development for maintaining user data and preferences.