

# PHP Cookies-Session

# Cookies

- A cookie is a small text file with a specific name and value used to store information about a user's browsing sessions, habits, preferences and browsing history .
- It is stored by the web browser on the client's machine.
- Cookies can be useful for applications such as *shopping carts* and *website customization* for each user.
- A disadvantage of using cookies is that may browsers and anti-viruses utilities block them or the user may delete them.

# Setting a Cookie in PHP

- The `setcookie()` function is used to create and set a cookie in PHP.
- The syntax of `setcookie()` function :

```
setcookie(name, value, expire, path, domain, secure);
```

- Only the name parameter is required. All other parameters are optional.
- The `setcookie()` function must appear BEFORE the `<html>` tag.

<b>name</b>	<b>The name of the cookie.</b>
value	The value of the cookie.
expires	The expiry date in UNIX timestamp format. The default value is 0.
path	Specify the path on the server for which the cookie will be available. If set to '/', the cookie will be available within the entire domain.
domain	Specify the domain for which the cookie is available to.
secure	This field, if present, indicates that the cookie should be sent only if a secure HTTPS connection exists.

# Example:

Cookies name

Cookies value

Expire date

```
<?php
// Setting a cookie
//This Cookies will expire after 30 days
setcookie("username", "Ahmad Ali", time() + 30 * 24 * 60 * 60);
?>
```

# Accessing Cookies Values

- The PHP `$_COOKIE` superglobal variable is used to retrieve a cookie value

Using `$_cookies`

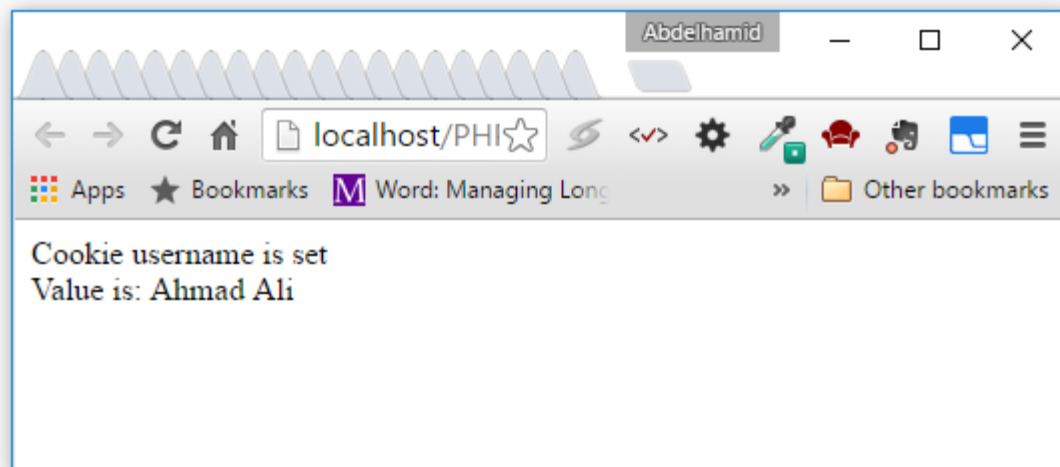
```
<?php
// Accessing cookie value
echo $_COOKIE["username"];
?>
```

# Example:

```
<?php
setcookie("username", "Ahmad Ali", time() + (60*60*24 * 30)); // for 30 days
?>
<html>
  <body>

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

  </body>
</html>
```





# Example:

```
<?php
setcookie("username", "Ahmad Ali", time() + (60*60*24 * 30)); // for 30 days
setcookie("Occupation", "Computer Engineer", time() + (60*60*24 * 30));
?>

<html>
  <body>

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

  </body>
</html>
```

# Removing Cookies

- The cookie is deleted by calling the same `setcookie()` function with the cookie name and any value and setting the expiration date in the past.

```
<?php
// Deleting a cookie
setcookie("username", "", time() - 3600);
?>
```

# Example:

```
<?php
setcookie("username", "", time() - 3600);
?>
<html>
  <body>
    <?php
    echo "Cookie 'username' is deleted.";
    ?>
  </body>
</html>
```

# Sessions

- A session is a way to store information to be used across multiple pages.
- A PHP session stores data temporarily on the server rather than user's computer.
- In a session based environment, every user is identified through a unique number called **session identifier** or SID.
- This unique session ID is used to link each user with their own information on the server.

# Starting a PHP Session

- The `session_start()` function is used to begin a new session
- It will create a new session and generate a unique session ID for the user.
- The session data are stored as key-value pairs in the `$_SESSION` superglobal variable.
- The stored data can be accessed during lifetime of a session using `$_SESSION` superglobal variable.

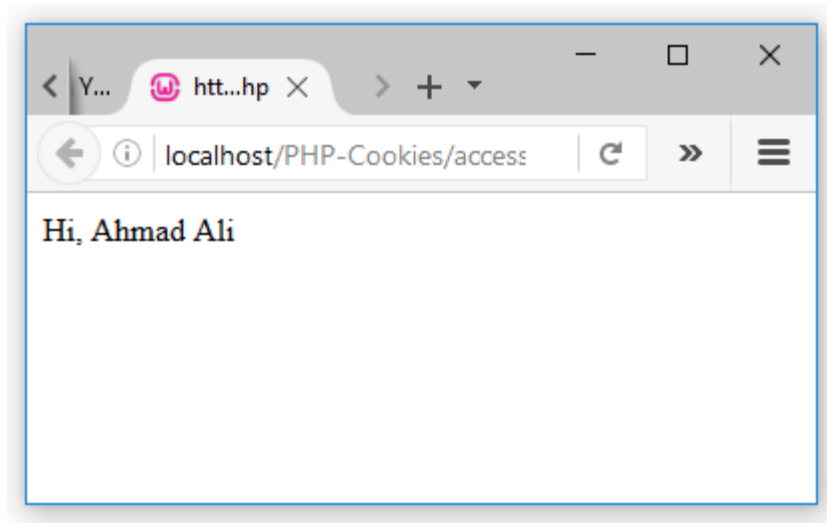
# Example

Creating sessions data

```
<?php
// Starting session
session_start();
// Storing session data
$_SESSION["firstname"] = "Ahmad";
$_SESSION["lastname"] = "Ali";
?>
```

Accessing sessions data

```
<?php
// Starting session
session_start();
// Accessing session data
echo 'Hi, ' . $_SESSION["firstname"] . ' ' . $_SESSION["lastname"];
?>
```



# Example:

```
<html>
  <head>
    <meta charset="UTF-8">
    <title></title>
  </head>
  <body>
    <?php
      // Starting session
      session_start();
      // Storing session data
      $_SESSION["student-name"] = "Ahmad Salem";
    ?>
    <h2> Using Session in different pages Example</h2>
    <input type="button" value="Page One"  onclick="location.href='page-1.php'" />
    <input type="button" value="Page Two" onclick="location.href='page-2.php'" />
    <input type="button" value="Page Three" onclick="location.href='page-3.php'" />
  </body>
</html>
```



```

<html>
<head>
  <meta charset="UTF-8">
  <title></title>
</head>
<body>
  <?php
    session_start();
    echo"<h2> Page One</h2>";
    echo"<h4> Student Name:" . $_SESSION["student-name"] . "</h4>";
  ?>
</body>
</html>

```

Page-1.php

```

<html>
<head>
  <meta charset="UTF-8">
  <title></title>
</head>
<body>
  <?php
    session_start();
    echo"<h2> Page Two</h2>";
    echo"<h4> Student Name:" . $_SESSION["student-name"] . "</h4>";
  ?>
</body>
</html>

```

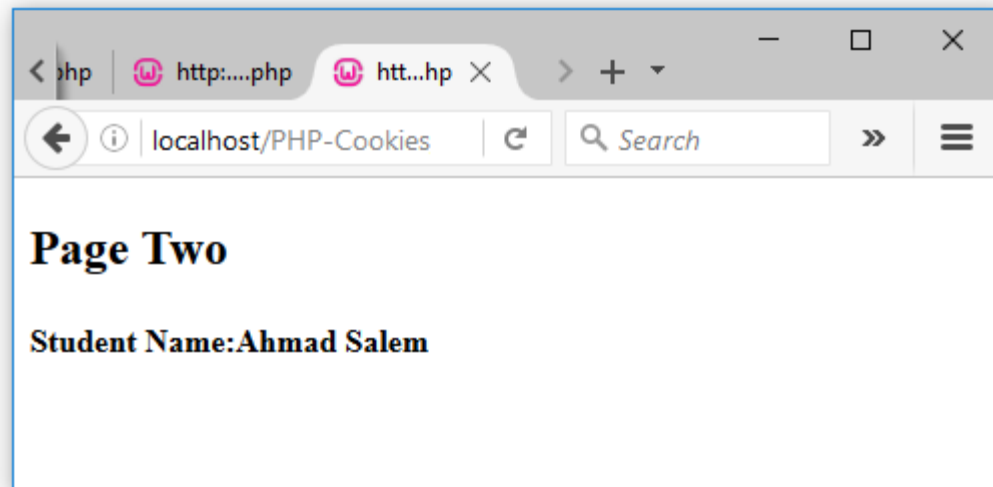
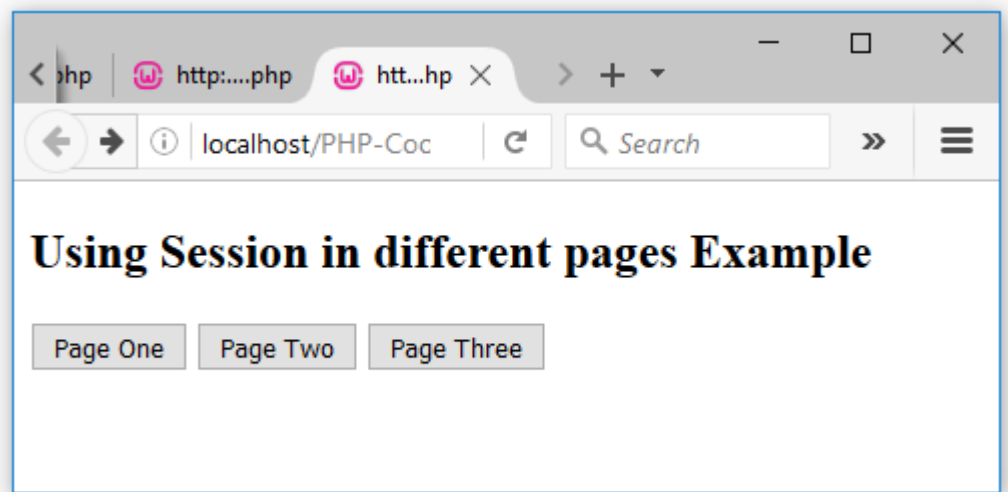
Page-2.php

```

<html>
<head>
  <meta charset="UTF-8">
  <title></title>
</head>
<body>
  <?php
    session_start();
    echo"<h2> Page Three</h2>";
    echo"<h4> Student Name:" . $_SESSION["student-name"] . "</h4>";
  ?>
</body>
</html>

```

Page-3.php



# Destroying a Session

- To remove certain session data, simply unset the corresponding key of the `$_SESSION` associative array.

Removing session data  
For lastname

```
<?php
// Starting session
session_start();
// Removing session data
if (isset($_SESSION["lastname"])) {
    unset($_SESSION["lastname"]);
}
?>
```

Removing all sessions data

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

- To destroy a session completely, simply call the `session_destroy()` function

```
<?php
// Starting session
session_start();
// Destroying session
session_destroy();
?>
```

Thanks!