COMP30680
Web Application Development

PHP – Cookies and sessions

David Coyle
d.coyle@ucd.ie

PHP Cookies

A cookie is a small file that the server embeds on the user's computer.

Each time the same computer requests a page with a browser, it will send the cookie too.

With PHP, you can both create and retrieve cookie values.

A cookie is created with the setcookie() function.

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

Create/Retrieve a Cookie

The following example creates a cookie named "user" with the value "John Doe".

The cookie will expire after 30 days (86400 * 30).
The "/" means that the cookie is available in entire website (otherwise, select the directory you prefer).

```php
<?php
$cookie_name = "user";
$cookie_value = "John Doe";
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/"); //
86400 = 1 day
?>
```

Note: The setcookie() function must appear BEFORE the <html> tag.

See **create_cookie.php**

Create/Retrieve a Cookie

You can then retrieve the value of the cookie "user" (using the global variable $_COOKIE).

You can also use the isset() function to find out if the cookie is set.

```php
<html>
<body>

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

Modify a Cookie

To modify a cookie, just set (again) the cookie using the setcookie() function:

```php
<?php
$cookie_name = "user";
$cookie_value = "Alex Porter";
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/");
?>
<html>
<body>

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

</body>
</html>
```

Delete a Cookie

To delete a cookie, use the setcookie() function with an expiration date in the past:

```php
<?php
// set the expiration date to one hour ago
setcookie("user", "", time() - 3600);
?>
<html>
<body>

<?php
echo "Cookie 'user' is deleted.";
?>

</body>
</html>
```

Are cookies enabled?

This small script checks whether cookies are enabled.

First, it tries to create a test cookie with the setcookie() function, then count the $_COOKIE array variable.

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

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

</body>
</html>
```

PHP Sessions

A problem: when using the Internet web servers do not know who you are or what you do, because the HTTP address doesn't maintain state.

Session variables solve this problem by storing user information to be used across multiple pages (e.g. username, favorite color, etc).

By default, session variables last until the user closes the browser.

Session variables hold information about one single user, and are available to all pages in one application.

A session is a way to store information (in variables) to be used across multiple pages.

Unlike a cookie, the information is not stored on the users computer.

Start a PHP Session

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

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

<?php
// Set session variables
$_SESSION["favcolor"] = "green";
$_SESSION["favanimal"] = "cat";
echo "Session variables are set.";
?>

</body>
</html>
```

Note: The session_start() function must be the very first thing in your document. Before any HTML tags.

See **demo_session1.php**

Get PHP Session Variable Values

Another page called "**demo_session2.php**" can then access the session information set on the first page ("demo_session1.php").

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

<?php
// Echo session variables that were set on previous page
echo "Favorite color is " . $_SESSION["favcolor"] . ".<br>";
echo "Favorite animal is " . $_SESSION["favanimal"] . ".";
?>

</body>
</html>
```

Note: session variables are not passed individually to each new page, instead they are retrieved from the session we open at the beginning of each page.

Note also: all session variable values are stored in the global $_SESSION variable

Modify a PHP Session Variable

To change a session variable, just overwrite it:

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

<?php
// to change a session variable, just overwrite it
$_SESSION["favcolor"] = "yellow";
print_r($_SESSION);
?>

</body>
</html>
```

Destroy a PHP Session

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

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

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

// destroy the session
session_destroy();
?>

</body>
</html>
```

Questions, Suggestions?

Materials and further reading:

w3Schools advanced PHP tutorials:
http://www.w3schools.com/php/php_includes.asp

Next:

Wrap up.