

Csci 4131

PHP Continued

Lecture 27, December 5th

Fall 2018

Dr. Dan Challou

Logistics

- Homework 7 out – Due (with Bonus) next Monday 12/10 2nd at 11:55 pm (or, Friday 11/14 without bonus)
- No Homework 7 Assignments accepted after Friday 11/14 at 2pm
- HW 8 Out – and Due Thursday 12/20 at 5pm.
- No Homework 8 Assignments accepted after 12/20 at 5pm
- Exam 2 – Next Wednesday December 12th , Here (Anderson Hall 370) from 2:30pm – 3:45pm. Topics covered after Exam 1 will be emphasized

Reading, PHP

- Do the **cseit** tutorial to make sure your account is configured to properly run PHP scripts:
 - <https://cseit.umn.edu/knowledge-help/php-tutorial>

Text Book Readings

- PHP
 - Sebesta – Chapter 9 – short section on Sessions
 - Sebesta – Chapter 13.1 – 13.5

Online References

- W3 Schools (www.w3schools.com) for
 - SQL (you knew that already)
 - PHP + MYSQL
 - PHP/ XML
 - PHP Sessions
 - And using PHP with a MySQL database

More Online References

- Tutorial on mysql at www.mysqltutorial.org (PHP PDO objects, my examples will be with PHP mysqli interface, which you should use – also has node.js / MySQL tutorial)
- Reference: www.mysql.org
- PHP + MySQL
- Reference: <http://php.net>

XML/PhP (PHP & AJAX)

- XML and PHP
http://www.w3schools.com/php/php_xml_simplexml_read.asp
- http://www.w3schools.com/php/php_xml_parser_expat.asp
- http://www.w3schools.com/php/php_xml_dom.asp
- http://www.w3schools.com/php/php_ref_xml.asp
- AJAX, PHP (shows how to call PhP script using AJAX)
https://www.w3schools.com/php/php_ajax_php.asp

PHP Sessions

- http://www.w3schools.com/php/php_sessions.asp

Last Time

- Review Basic PHP Example from Last Class
- Constructs, Tools, and Tips for Debugging PHP
- PHP / SQL / MySQL
- Reading XML with PHP

Questions?

Today

- PHP Sessions
- Topics in Internet Security

Review Exercises

Exercise 1 Review:

- Write the PHP Script that
 - A) calculates the tip from the values entered by the use on the form
 - B) Returns the HTML page with the answer to the client
 - Formula to calculate tip is:
 - $\text{Meal Cost} * (\text{Percent_tip} * 0.01)$

The name of the fields on the form that accept the values from the user:

Meal Cost is named `mcost`

and

Percent_tip is named `ptip`

Form sends those name/value pairs to a phpscript named “tipcalc.php” when submitted

The PHP script “tipcalc.php” will

- 1. get the values from the `$_POST` array**
- 2. Calculate the tip**
- 3. And return the HTML page with the**
 - i. Meal Cost**
 - ii. Percent Tip and**
 - iii. Tip**

Exercise 2 Review: Write a PHP program to print out the names and locations of each restaurant on separate lines (turn in at end of class)

Recall our Car example :

```
<?php
$xml=simplexml_load_file("cars.xml") or die("Error: Cannot create object");
foreach($xml->children() as $cars) {
    echo $cars->make . " , ";
    echo $cars->model . " , ";
    echo $cars->year . " , ";
    echo $cars->price . "<br>";
}
?>
```

The XML with the name and location of each restaurant is in the file **restaurants.xml**, whose contents are below:

```
<?xml version="1.0" encoding="utf-8"?>
<restaurants>
  <rinfo>
    <name>Everyday Italian Bistro</name>
    <loc>1542 University Avenue</loc>
  </rinfo>
  <rinfo>
    <name>The Little Cafe</name>
    <loc>1642 Hennepin Avenue</loc>
  </rinfo>
</restaurants>
```

Sessions in PHP

- When you work with an application, you open it, do some changes, and then you close it.
- This is much like a Session with a PHP application running on the server side of a client server application (the client is your browser).
- The computer knows who you are.
- It knows when you start the application and when you end.
- However there is one problem: the web server does not know who you are or what you do, because the HTTP address doesn't maintain state.

Sessions, continued

- Session variables solve this problem by storing user information to be used across multiple pages (e.g. *username*, error messages, etc).
- **By default, session variables last until the user closes the browser.**
- **Note – if you need storage that persists past one session, you should store the data in a database**

From W3 Schools

- **How does it work? How does it know it's me?**

Most sessions set a user-key on the user's computer that looks something like this: 765487cf34ert8dede5a562e4f3a7e12. Then, when a session is opened on another page, it scans the computer for a user-key. If there is a match, it accesses that session, if not, it starts a new session.

More on Sessions

- Session Variables are set with the PHP global variable **\$_SESSION**
- Sessions are accessed by each page run during a session by calling
session_start();
when the page is executed

A Session can be ended in a number of ways besides closing your browser:

unset(\$_SESSION['name']); // will delete just the name data
session_destroy(); // will delete ALL data associated with that user.

Example

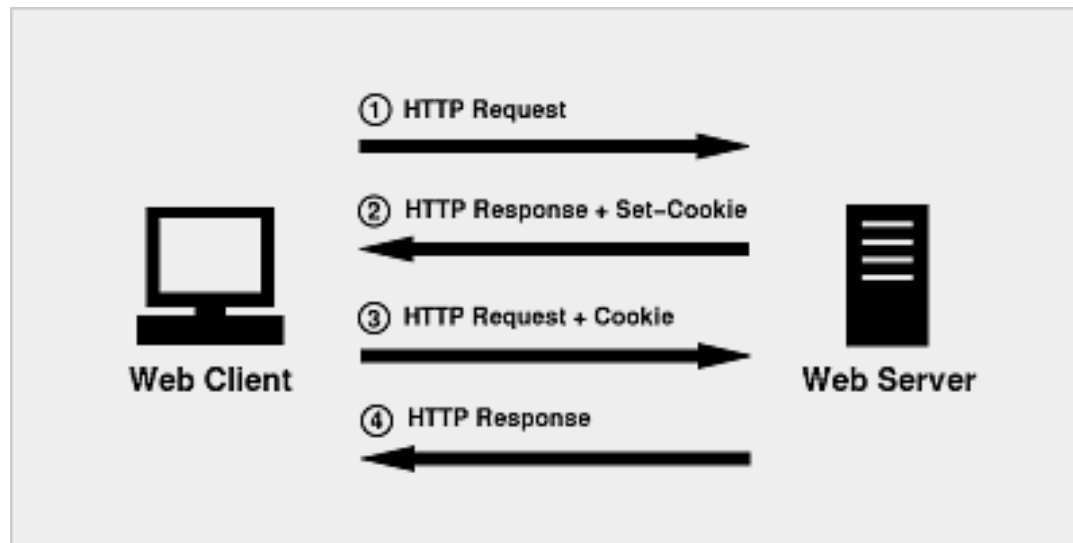
Two executables, executed in the same session of a browser. First page sets the `$_SESSION` array when it is executed

Second Page gets the values stored in the `$_SESSION` array

Example

- <http://www-users.cs.umn.edu/~chal0006/Examples/session1.php>
- <http://www-users.cs.umn.edu/~chal0006/Examples/session2.php>

Sessions, Security Issue – id used by both Client and Server



How to Combat?

- Make sure to check encrypted / hashed login of users before admitting them into the session
- Check to make sure that Client coming from same IP address each time (store `$_SERVER['REMOTE_ADDR']` in database after successful login, and check it each time a session is re-entered)
- Regenerate the session id using `session_regenerate_id()` after entering a session

And, Practice Form Security

- http://www.w3schools.com/php/php_form_validation.asp

Course Survey

- What worked well
 - Suggestions for improvement
- What didn't work
 - Suggestions for improvement
- Instructor: Challou
- Term: Fall
- Year: 18
- Department: Csci
- Course: 4131
- Section: 1

Next Class

- More Topics on Internet Security (Hashing, Asymmetric Encryption (Public/Private key))
- SQL Injection