INFO1-CE9224: Introduction to PHP Programming

Instructor
Dave Hauenstein
dah16@nyu.edu
cell: 732-754-3870

Term
Summer 2012

Office Hours
By appointment only

Class Sessions
10 Sessions
Wednesdays, 6:30pm - 9:30pm
May 30 - August 15
(*No class June 6 or July 4)*

Course Description

PHP is an increasingly popular open-source scripting language that can be used to rapidly program dynamic, database-driven websites. This fast-paced introduction covers installation and setup of a development environment, language basics, validating and processing user input, generating dynamic output, and more. Participants with experience developing Web content study the syntax and conventions of PHP, learn to make websites interact with databases, understand production of content management systems, and gain experience with file handling and graphics. Learn to integrate PHP and SQL with markup languages, such as HTML and XML, and complete a programming project that uses these techniques.

Prerequisite: Thorough familiarity with HTML; previous programming experience is recommended.

Reading and Materials

Required

Beginning PHP 5.3 (Wrox Programmer to Programmer)
By: Matt Doyle
Published by Wrox
ISBN-10: 0470413964
ISBN-13: 978-0470413968

Recommended / Supplemental (Optional)

We will touch upon object oriented software design at the very end of the course. This book will help you take your skills a bit further, and will explain how to build object oriented systems using well-known design patterns. *This book is advanced.*

PHP Objects, Patterns and Practice (Expert's Voice in Open Source)
By: Matt Zandstra
Published by Apress
ISBN-10: 143022925X
ISBN-13: 978-1430229254

Recommended / Supplemental (Optional) cont...

Sometimes designing object oriented architectures can be a mental hurdle after writing procedural code for a while. This book will help your mind make the paradigm shift necessary to start thinking in an object oriented way. *This book is advanced.*

The Object-Oriented Thought Process (3rd Edition)
By: Matt Weisfeld
Published by Addison-Wesley Professional
ISBN-10: 0672330164
ISBN-13: 978-0672330162

This book will aid in your knowledge of becoming a better MySQL database administrator. It will cover far more advanced topics that we've touched upon in class, such as *transactions, stored procedures,* and *triggers*.

MySQL (4th Edition)
By: Paul DuBois
Published by Addison-Wesley Professional
ISBN-10: 0672329387
ISBN-13: 978-0672329388

If you would like me to make other recommendations or review a book before you make a purchase, please let me know and I'll be happy to do so.

Homework

Homework assignments are listed in the weekly class schedule. Each assignment is due the Tuesday night following the Thursday class session in which it is assigned. Please submit by email to dah16@nyu.edu and include your name along with "Homework Assignment XX" in the subject line (where XX is the assignment number — see weekly class schedule).

Each assignment is worth a total of 10 points. Points will be deducted for programming errors that disrupt or terminate your scripts (1 point each), cause non-fatal warnings or notices (.5 points each), or for failing to deliver part or all of the assignment. Points also will be deducted for late assignments (1 point for each full week past the deadline). A complete assignment, delivered on time, with no errors receives a full 10 points.

Grading Policy and Scale

- Homework Assignments: 40%
- Final Project: 50%
- Attendance and Class Participation: 10%

Grade	Percentage
A	93 - 100
A-	90 - 92
B+	87 - 89
B+	83 - 86
B-	80 - 82
C+	77 - 79
C	73 - 76
C-	70 - 72
D+	67 - 69
D	60 - 66
F	<= 59

Note: "A+" and "D-" are not officially recognized as grades for the continuing education programs and cannot be assigned.

Homework Naming Conventions

One file; file name:
first initial + last name + dash + hw + assignment #
dhauenstein-hw1.php

Many files; folder name:
first initial + last name + dash + hw + assignment #
dhauenstein-hw1/
  script1.php
  etc...

Other Policies

Course Updates and Announcements: Please check Blackboard regularly for timely information about forthcoming class sessions, tests, and projects. For information on activating your NYUHome account and accessing Blackboard, see this page http://www.nyu.edu/its/blackboard.

Backups: Files on the class server or on the computers in the classroom may be overwritten or become corrupt between sessions. You are responsible for saving digital backups of your work in some other form. I recommend bringing a flash drive to class with you so you can save your work to it at the end of each session.

Syllabus: This syllabus is tentative and subject to change at the discretion of the instructor.

INFO1-CE9224: Weekly Course Schedule

Class 1

- Introductions
- Course Overview
- Web Technology Primer
    - Markup Languages
        - XHTML & HTML5
        - XML, XSLT, XPath
    - Presentation Layer
        - CSS, JS, AJAX
    - Protocols
        - HTTP, FTP, SFTP, SSH, TCP/IP
    - More on the HTTP Protocol - Request and Response
    - Defining *client* and *server*
    - What is MySQL?
    - What is Apache?
- Exploring the lab environment
    - Ubuntu Linux and why it was the right choice
    - Choosing a text editor - Sublime Text in the Lab
    - Web directory structure - where your PHP scripts will be placed
    - Introduction to FTP (file transfer protocol) and SSH (Secure Shell)
    - First PHP Script
- Getting Set Up on Windows and Mac
    - Basics for running PHP, Apache, and MySQL in the Windows environment
    - Basics for running PHP, Apache, and MySQL in the Mac OS X environment
- PHP Language Basics
    - PHP - “PHP: Hypertext Preprocessor”
    - Language basics
        - Variables
        - Data Types
            - *Scalar / primitive*: Boolean, string, integer, float
            - *Compound / composite*: array, object
            - *Resource* and *null*
        - Loose typing and automatic type conversion (vs. strongly typed)
        - Type detection: gettype()
        - Explicit type casting: (int), (float), (bool), (string), (array), (object)
        - Type testing: is_int, is_float, is_bool, is_string, is_array, is_object

Homework

- **Read** *Beginning PHP 5.3*:
    - Chapter 1 - Introducing PHP
    - Chapter 2 - Your First PHP Script
    - Chapter 3 - PHP Language Basics (up to pg. 40)

Class 2

- PHP Languages Basics, Part II
    - Operators and Expressions
        - Arithmetic Operators
        - Assignment Operators
        - Comparison Operators
        - Incrementing/Decrementing Operators
        - String Operators
        - Operator Precedence
    - Logical Operators
    - Constants (*scalar values only*)
    - Introduction to functions
    - Built in string functions

Lab Assignments

- Experiment with variables
- Experiment with built in string functions

Homework

- **Read** *Beginning PHP 5.3*:
    - Chapter 3 - PHP Language Basics (pgs. 40-50)
    - Chapter 5 - Strings

Class 3

- PHP Languages Basics, Part III
    - Arrays basics
        - Indexed and Associative
        - Superglobals: $_SERVER, $_GET, $_POST, $_FILES, $_COOKIE, $_SESSION, $_REQUEST, $_ENV
    - Flow Control Structures
        - Conditional Statements (if ... else)
        - Branches (switch)
        - Comparison Operators
        - Assignment (=) vs. equality (==) vs. identity (===)
        - Logical Operators
        - Ternary (and shorthand ternary) Operator (?:)
    - Truthy vs. Falsey

Lab Assignments

- Experimenting with arrays
- Form Handling Lab

Homework

- **Assignment #1**
  - Create an “About Me” form that meets the following criteria:
    - Form collections “First Name”, “Last Name”, “Age”, “Email Address”, “Short Bio”.
    - Form submits to PHP script using the HTTP POST method.
    - PHP Script prints the submitted form data back out to the page.
    - Email the script to dah16@nyu.edu
    - ***Note***: Chapter 9 of *Beginning PHP 5.3* will help with this assignment.
    - ***Note***: Future lab/homework assignments will be based of of this
- **Read** *Beginning PHP 5.3*:
  - Chapter 6 - Arrays (up to pg. 107)
  - Chapter 4 - Decisions and Loops (up to pg. 58)

Class 4

- PHP Languages Basics, Part IV
  - Loops (while, do...while, for, foreach)
  - Multidimensional arrays (nested arrays)
  - Functions
    - Arguments/parameters
    - Return statements
    - Built-in functions: Include directives
    - Built-in functions: arrays (implode, explode, sorting, etc...)
    - Custom functions
      - Type hinting (objects and arrays only)
      - Variable scope
      - Globals and $GLOBALS
      - Static variables

Lab Assignments

- Experimenting with custom functions
- Building a navigation menu from nested associative arrays

Homework

- **Read** *Beginning PHP 5.3*:
  - Chapter 4 - Decisions and Loops (complete from pg. 58)
  - Chapter 6 - Arrays (complete from pg. 107)
  - Chapter 7 - Functions (up to pg. 158)

Class 5

- HTTP Protocol Review
    - A deeper look at the Request and Response
    - HTTP Methods (**GET**, **POST**, **PUT**, **DELETE**, HEAD, OPTIONS, TRACE)
    - HTTP Status Codes and descriptions
    - Monitoring HTTP requests and responses in the browser
    - HTTP: A stateless protocol
- HTTP and PHP
    - Sending headers with PHP
    - HTTP Redirects
    - Sending email
    - Data persistence
        - Cookies
        - Sessions

Lab Assignments

- Redirecting HTTP requests
- Forcing a download of Contact information (vCard)
- Sending email from a PHP script

Homework

- **Assignment #2** (changes are in red)
    - Modify your “About Me” form in the following way:
        - Form collections “First Name”, “Last Name”, “Age”, “Email Address”, “Short Bio”.
        - Form submits to PHP script using the HTTP POST method.
        - The PHP script that handles the form should do the following:
            - Send an email to the address from the form
            - The subject should be: “Notification: Your bio was just updated”
            - The body of the email should say:
                - “Your bio was updated on {date} at {time}” -- followed by two new lines
                - Then each field and it’s value should be printed in the email
            - After the email is sent, the page should redirect the user to a generic “Thank You” page.
        - Email the script to dah16@nyu.edu
        - ***Note***: Chapter 9 of *Beginning PHP 5.3* will help with this assignment.
        - ***Note***: Future lab/homework assignments will be based of of this
    - **Read** *Beginning PHP 5.3*:
        - Chapter 10 - Preserving State with Query Strings, Cookies, and Sessions
        - Chapter 16 - PHP and the Outside World (complete from pg. 488)

Class 6

- Server Side Validation
    - Using isset() to check if a variable is set
    - Using empty to check if a variable is “empty”
    - “Escaping” input data
    - Complex checking
        - Date validation
        - Regular expressions Part 1

Lab Exercises

- Experimenting with regular expressions
- Redisplaying the form, pre-filled and with error messages

Homework

- **Read** *Beginning PHP 5.3*
    - Chapter 16 - PHP and the Outside World (up to pg. 488)
    - Chapter 18 - String Matching with Regular Expressions

Class 7

- Regular expressions Part 2
- Reading from / writing to files
- Uploading files

Lab Exercises

- Creating a file uploader

Homework

- **Assignment #3** (changes are in red)
  - Modify your "About Me" form in the following way:
    - Form collections "First Name", "Last Name", "Age", "Email Address", "Short Bio".
    - Form submits to PHP script using the HTTP POST method.
    - The PHP script that handles the form should do the following:
      - Validate all fields:
        - "First Name", "Last Name", and "Short Bio" should be strings and should not be empty.
        - "Age" should be an int that is between 13 and 100
        - "Email Address" should be a valid email address
      - If all fields aren't valid, redisplay the form, pre-filled with error messages next to the appropriate field
      - If everything is correct, continue with the following:
      - Send an email to the address from the form
      - The subject should be: "Notification: Your bio was just updated"
      - The body of the email should say:
        - "Your bio was updated on {date} at {time}" -- followed by two new lines
        - Then each field and it's value should be printed in the email
      - After the email is sent, the page should redirect the user to a generic "Thank You" page.
    - Email the script to dah16@nyu.edu
    - ***Note***: Chapter 9 of *Beginning PHP 5.3* will help with this assignment.
    - ***Note***: Future lab/homework assignments will be based of of this
- **Read** *Beginning PHP 5.3*:
  - Chapter 11 - Working with Files and Directories

Class 8

- Relational Databases
    - Servers, DBMS
    - Tables, rows, attributes, constraints
    - Normalization
    - Primary and foreign keys
- SQL (Structured Query Language)
    - Querying for data
    - Inserting and updating rows
    - Simple joins

Lab Exercises

- Experiment with SQL using phpMyAdmin and MySQL

Homework

- **Assignment #4**
    - Instructions are available online:
    - http://davehauenstein.com/nyu/INFO1-CE9224-2012-Summer/hw/class8/hw_assignment.pdf
- **Read** *Beginning PHP 5.3*
    - Chapter 12 - Introducing Databases and SQL

Class 9

- MySQL and PHP
    - Connecting to a MySQL Database using PHP
    - Querying the database using PHP
        - Retrieving data and looping over results
        - Inserting / Updating database rows
        - Escaping and Sanitizing data
    - Importance of validation and error handling

Lab Exercises

- Creating a registration form
- Logging in / out registered users
- Retrieving a list of blog articles
- Viewing a blog article detail page

Homework

- **Read** *Beginning PHP 5.3*:
    - Chapter 13 - Retrieving Data from MySQL with PHP
    - Chapter 14 - Manipulating MySQL Data with PHP

Class 10

- Introduction to Object Oriented Programming
- PHP Classes and Objects
    - Scope (public, private, protected)
    - Properties
    - Methods
- Introduction to MVC design pattern

Lab Exercises

- Creating a User class and integrating with Class 9 lab

Homework

- **Read** *Beginning PHP 5.3*:
    - Chapter 8 - Objects

INFO1-CE9224: Final Project

The Due Date is our final day of class: TBA

The project is worth 40% of your grade.

Each project can be designed as a single PHP page that displays different information in response to different inputs or as a multi-page mini Web site, with each separate page dedicated to a specific function. The overall architecture and organization of the application is entirely up to you, as long as it meets the minimum criteria described below.

Please do not use any pre-packaged content management frameworks such as Drupal or Wordpress to build your project. You may however use PHP-based template systems, such as Twig or Smarty, if you like. You may also use HTML page designs that are not your own, as your project will be evaluated purely on its back-end functionality and the quality of your PHP code—not on its graphic design merits or the quality of any front-end code (HTML, CSS, JavaScript, etc.).

Using a MySQL database is a requirement for all projects. Please type out the MySQL *create* and *insert* (if any) statements and send them to me in a file called db.sql along with the rest of your project.

Please choose from one of the following projects below:

Option 1: Photo Gallery Application

Build a Web-based application with two distinct pages (gallery page, image upload form) using the following guidelines:

Image Upload Form: A form that allows for the uploading of a single (or multiple, if you prefer) image (JPEG, GIF, or PNG) to the server. It should also have a field for a caption that is stored along with the image that will be displayed on the gallery page. *Optional*: Collects any additional information you might like to save and display about the uploaded images.

Image Upload Processing Instructions

- Saves the image to a folder for later display on the site.
- Saves information about the image (title and/or caption and any additional information you may choose to collect in your form) for later display.
- Saves the information using a MySQL database table(s).
- Automatically records the date the image was uploaded, without requiring the user to enter it in the form.

Gallery Page: (pagination is ok, if you prefer) that displays thumbnail images of uploaded photographs or artwork (JPEGs, GIFs, and/or PNGs), along with a caption and the date the image was uploaded by the user.
Optional: Clicking on a thumbnail image on the gallery page displays the larger, full-size image originally uploaded by the user.

Option 2: Mini Twitter-esque Messenging System

Build a Web-based application with two distinct pages (sign-up, send message) using the following guidelines:

Sign-Up: This page will take in user input and save it as a subscriber. The following fields must be present: first name, last name, and email address. Upon submission, display a “thank you for signing up” message.

Sign-Up Processing Instructions: Store the date the user signed up by generating the data automatically in PHP. *Optional*: Sanitize and validate the data that is going to be stored in the database.

Send Message: This page will have a form that will send emails out to the list of subscribers. The following fields must be present: from address, brief message.

Send Message Processing Instructions: The following steps should be taken:

- Reads the latest list of subscribers from the database.
- Sends the message entered by the sender to each of those subscribers, one by one.
- The from address and body should be what the user who filled the form out specified.

Option 3: Custom Project

Build a Web-based application of your own design that is similar in scope and function to one of the above two projects. At a minimum the application must:

- Accept some input/collect some information from one or more users.
- Store some of the collected information somewhere, in some fashion, on the server—and/or, if appropriate, in a cookie on the user's computer.
- Store some of the data in a MySQL database.
- Read the stored information and output it somewhere—either onto an HTML page for display or by sending it somewhere via email, for example.

If you choose the custom option, please submit a brief proposal for your project no later than our ninth class session.