# PHP Programming for the Web, EL ELNG X400

1 Semester units in EECS

## Course Description

PHP was created by Rasmus Lerdorf in 1995 to track visitors to his online resume. (Consequently he called it "Personal Home Page.") Over the next year, he combined PHP with a form interpretation (FI) project he had also written, which resulted in PHP2 (also known as PHP-FI).

Using this package as a base, developers worldwide joined Lerdorf in adding increased database support, cross-platform compatibility, and standardized syntax. The result was PHP 3, released in 1998. To better reflect its new, professional image, the PHP acronym was changed to mean "PHP: Hypertext Preprocessor."

PHP was rebuilt by Zend Technologies and released in the summer of 2000 as PHP 4. It retained backward-compatibility but improved infrastructure -- greatly expanding the language's power and usefulness.

The PHP5 is now maintained by the PHP Group. PHP has come to the forefront of Web scripting technology, rivaling the capabilities of ASP, CGI, and ColdFusion.

## Learning Outcomes

By successfully completing this course, you will be able to:

* Write PHP scripts
* Use PHP to manage data
* Create Web applications with PHP
* Utilize advanced PHP capabilities and programming techniques
* Understand what PHP can do with other Web and server technologies

## Course Materials and Technical Requirements

### Required Texts

* Ullman, L. (2008). *PHP 6 and MySQL 5 for Dynamic Web Sites: Visual QuickPro Guide*. San Francisco: Peachpit Press.
* You will also be asked to read sections of the *PHP Manual*, please download it in one of several languages and formats from http://php.net/download-docs.php

### Recommended Reading

### Ullman, L. (2009). *PHP for the World Wide Web, 3rd Ed.* San Francisco: Peachpit Press.

### Technical Requirements

Each student should have the following available for the course:

* Hosted Web site with 10MB of disk space
* PHP (Version 4 and above)
* MySQL (Version 4.0 and above)
* SQLYog MySQL GUI Tool (free edition)

In general, most Web hosts offer workable sites. There are even free PHP Web hosting options available. PHPwebhosting is pretty good and costs only $10 a month.

Some of you may want to go ahead and set up your own Windows or Linux server, which isn’t that difficult or expensive.

There are numerous Web sties and books that render help in installing MySQL, Apache, and PHP. The fastest and easiest way to do this is by downloading XAMPP for free from Apache Friends. It is a great tools that does all three installations for you at the click of a button. You can then run your PHP scripts without worrying too much.

Alternatively, as far as the database goes, if you buy a hosting account you will get a database with it that you can work on. If you make your own server, you can set up your own database and take it from there. As long as the computer you are using has MySQL (or PostgreSQL or some other database application), you will be fine.

If you are having trouble getting a Web site hosted or getting a PHP script to execute, first go to the discussion forums and read the posting under FAWs and Module 1. If these don’t answer your questions, send me a message.

This course is built on a Learning Management system (LMS) called Canvas and you will need to meet these [computer specifications to participate within this online platform.](http://guides.instructure.com/s/2204/m/4214/l/82542-what-are-the-basic-computer-specifications-for-canvas" \o "Confirm your computer meets Canvas specifications)

#### Optional

Canvas allows you to record audio or video files of yourself and upload them in the course. Although doing so is not required for any of the activities, using these features will enhance your engagement in the course. If you would like to use these features, you will need to have a webcam and a microphone installed on your computer.

## Learning Activities

This course consists of five modules of study outlined at the end of this syllabus. Each module contains some combination of learning activities including a reading assignment, discussion assignments and programming assignments. In addition, there is a comprehensive final project and two self-study (ungraded) quizzes to help you gauge your progress. Throughout the course, I also link you to helpful tutorials and sources of code. Watch for those links when new topics are defined, or click on Resources to view all the links for the course.

### Programming Assignments

### Each of the five modules in this course has an associated programming assignment to be completed and turned in. The instructions for the assignment can be found on the module's overview page. You will submit your assignments within the Learning Management System in each module. Be sure to keep copies of your work, as assignments sometimes go astray.

### Be sure to label your assignments by typing your name, the course name ("PHP" will do), and the module number in the title. DO NOT compress the files with .zip, .sit, .tar, or similar packages. Also, in order to test your exercises, I will need to run them on my server. If any special installation instructions are required -- such as creating tables or databases, adjusting the permissions on files or folders, and the like -- please include them. This will be more of an issue beginning in Module 3.

### Self-Study Quizzes

### There are two ungraded quizzes at the end of Modules 2 and 5. I want to give you chance to check for yourself how you are doing. You may take the quizzes as many times as you would like.

### Discussions

### There are two types of discussion forums in this course:

### Discussion Forum: The Discussion Assignments forum provides a vehicle for classroom interactivity based on questions created specifically for each module. All posts are "public," that is, available for all class participants to read and comment on.

### When a module has a discussion assignment, please respond to the questions in a manner that reflects critical thinking. Please feel free, as well, to positively critique and offer leads and suggestions to comments and questions that other students have made.

### Each of us brings a unique perspective to this class based upon our life experiences and previous studies. But because of the continuous, open-enrollment nature of this course, at times it might be hard to sustain an actual conversation in the discussion threads with your fellow students. So instead, let's look to the discussion threads as a place to connect our multiple perspectives and construct an evolving knowledge base. The process will enrich your own studies, and the words you leave behind will help students who join the class even after you've gone on to new pursuits.

### Student Lounge: Questions and Comments Forum is our place for informal discussion, a place to create new topic threads and share common issues or experiences, class-related or not. Please use this forum to post questions about the course readings, answers to help other students, or to test-launch your own original ideas so that all students in the course may benefit from the exchange. You will discover that your fellow students come from a diverse set of professional backgrounds. Part of the richness of the course will occur when students apply what they learn to their own situations.

### Final Project

### You will create a PHP Web application that incorporates what you've learned throughout the course. The application must do at least three of the following:

### registers users and allows them to log in

### dynamically generates pages based upon HTML templates

### uses cookies, sessions, and/or authentication

### involves image, Shockwave, or PDF creation

### handles file uploads

### logically incorporates regular expressions, user-defined functions, or include()

### uses object-oriented programming

### utilizes databases, directories, or files

### Possible, real-world applications are (you are not limited to any of the following):

### shopping cart

### message forum (similar to the discussion forums)

### online email system (send and receive using IMAP)

### search engine

### Whois check

### schedulable calendar

### PHP-based chat rooms

### HTML design quality will not be a factor in the project grade, but you must demonstrate: solid programming skills; adequate documentation within the code; efficiency; security; understanding of related technologies (databases, LDAP, and so on); and long-range thinking (room for the site to grow and be easily modified).

### Submit the URL and attached scripts in the last module. Be sure to also post the URL to the Discussion Assignments forum as a reply to "Final Project" thread.

## Communication and Office Hours

You can always get in touch with me during the course. You can access course email by clicking on the Inbox link on the Corner Help toolbar (see also [Canvas Overview Video](http://guides.instructure.com/s/2204/m/4210/l/141852-canvas-overview-video)). You can expect me to respond to email within 48 hours of receiving messages unless I have notified the class otherwise (e.g., because of vacation or other reasons).

Please note: all course communication between students/instructor must occur within the course.

## Grading and Course Policies

Final grades will be assigned according to the following percentages:

* Programming Assignments, 35%
* Discussion Assignments, 30%
* Final Project, 35%

**You must pass the final project with a score of 70% or better to pass the course.**

### Grading Information

Final grades follow the UC Berkeley grading system:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Letter Grade** | A | A- | B+ | B | B- | C+ | C | C- | D+ | D | D- | F |
| **Percentage** | 100-94 | 93-90 | 89-86 | 85-83 | 82-80 | 79-76 | 75-73 | 72-70 | 69-66 | 65-63 | 62-60 | < 60 |

Table 1: UC Berkeley Grading Systems

To [view your final grade and request official transcripts](http://extension.berkeley.edu/), login to your student account and go to "My Enrollment History."

When I grade your writing assignments, I'll be looking at content, organization, and mechanics. Please keep the following criteria in mind:

|  | **Poor** | **Needs Improvement** | **Meets Expectations** | **Exceptional** |
| --- | --- | --- | --- | --- |
| **Content** | Poor writing style with little or no specific details, no evidence of having studied the material, and/or off topic. | Adequately written; some points elaborated but with minimal use of concepts from the material. | Well written, most points elaborated with clear and detailed information that supports thoughts and ideas and uses concepts from the material. | Well written, fully elaborates points. Clear and detailed information supports thoughts and ideas and shows full acquisition of concepts from the material. |
| **Organization and Mechanics** | Little or no structure present. Grammatical errors interfere with comprehension. | Organization present but awkward. Some grammatical errors present. | Good organization with few statements out of place. Minor grammatical errors. | Clearly organized and remains focused. Few or no grammatical errors. |

Table 2: Criteria for Writing Assignments

In the Discussion Assignments forum, I'll also be looking for evidence of participation:

|  | **Poor** | **Needs Improvement** | **Meets Expectations** | **Exceptional** |
| --- | --- | --- | --- | --- |
| **Participation** | Minimal posts in number or length. Posts show little or no reflection on the topics or previous posts. | Posts address the topic but consist mostly of a rote repetition of the study materials. Little or no reflection on previous posts. | Posts address the topic with reflection. Many responses build on previous posts. | Posts show a genuine interest in contributing to the overall life of the forum. |

Table 3: Evidence of Participation for the Discussion Assignments Forum

## DSP Accommodations

If you are a student with special needs and haven't already contacted the [Disabled Student Services](http://extension.berkeley.edu/static/studentservices/career/#disabled) (DSS), please contact the office right away. Be sure to review our detailed DSP accommodations instructions.

## Academic Integrity, Research, and Proper Citation

As an online student, you are encouraged to reach out to your fellow students in the online classroom to discuss materials and ask each other questions, but there are limits to this collaboration. Reviewing lecture and reading materials and studying for exams can be enjoyable and enriching things to do with fellow students. This is recommended. However, unless otherwise instructed, homework assignments are to be completed independently, and materials submitted as homework should be the result of your own independent work.

As a UC Berkeley student you are bound by the [Academic Integrity, Research and Proper Citation policies](http://extension.berkeley.edu/upload/academic_integrity.pdf) outlined in the [UC Berkeley Extension Code of Student Conduct Policy Statement](http://extension.berkeley.edu/upload/studentconduct.pdf) dated July 11, 2013 that clearly defines what constitutes cheating, as well as plagiarism and other forms of academic misconduct.

You must review all sections of the Academic Integrity Pledge and Course Policies Module within your Canvas course and complete the following item prior to gaining access to course content:

Take the Pledge to Academic Integrity

## Course Evaluation and Course End Date

### Course Evaluation

### You are an important part of our community and we value your opinion! Before your course End Date, please take a few minutes to fill out a survey about your experience in this course so that we can continue to improve the online learning environment. Your instructor also values your constructive feedback. Course Evaluations are a valuable way for instructors to gain insight on their online teaching practices.

### You can find a link to the Course Evaluation in the final module of your Course. Please note that your identity remains confidential during the evaluation process, and evaluation results aren't shared with instructors until after final grades are due. For questions or technical problems with the evaluation process, please email UC Berkeley Extension’s Center for Instructional Excellence at [teach-extension@berkeley.edu](mailto:teach-extension@berkeley.edu).

### Thank you in advance for sharing your thoughts with us!

### Course End Date

Your access to the online classroom will expire on the course End Date, which is indicated in the initial e-mail you received when you enrolled.

## Pacing Yourself through the Course

With this course, you have the freedom to design your study time to meet your schedule. However, this style of learning also requires dedication and commitment to ensure that you get the homework and assignments done in a timely fashion. Use these tips to stay on track and get the most from this class.

* **Get started as soon as possible**: Students who submit their first assignment within the first month are more likely to complete the course than those who delay. Please do not wait to submit everything toward the end of the class. Assume that I will need a week to grade each assignment and provide feedback that will help you complete subsequent assignments.
* **Create a planning calendar**: Plan your homework and submittal dates, and stick with them. Students who submit assignments regularly are more likely to complete the course than those who do not. Take the time now at the beginning of the course to plan your study time by using the course Calendar tool.
* **Ask questions**: You are always encouraged to ask questions. Use course mail to ask me questions, ask for feedback, or just to request encouragement. I'm always pleased to be of assistance. Use the Student Lounge forum for general questions about the course or assignments that other students might need answered as well.
* **Submit complete assignments**: Submit only completed assignments. If you are unsure about a question, e-mail me and ask me first before submitting the assignment. I'm always pleased to be of assistance. Any incomplete assignments will be returned to you to complete and resubmit.

## Canvas Tech Support and UC Berkeley Extension Student Services

### Canvas Tech Support

The learning management system (LMS) used in this course is Canvas, which has convenient mobile apps for phones and tablets. Part of the orientation materials in your course will help you make sure that your computer is at par with Canvas specifications. Anytime you are in Canvas you can report problems, get support, and search Canvas user guides from the Help link on the top menu bar. Other options:

* Canvas Support 24/7 Hotline:  855-308-2758
* Email: [msupport@instructure.com](mailto:msupport@instructure.com)

### UC Berkeley Extension FAQs and Student Services

Start at the [Student Services webpage](http://extension.berkeley.edu/static/studentservices/) to find help with issues such as the following:

* Course registration
* Exam proctoring services
* Refunds, withdrawals, and transfers
* Grade options
* Requests for transcripts or official receipts

If you need further help, UC Berkeley’s [Extension Contact page](http://extension.berkeley.edu/static/online/#contact) lists Program Coordinator’s email and phone numbers for each academic department.

## Course Outline

You'll find complete instructions for your assignments within the course modules.

**Module 1: Welcome to PHP**

Commentary

1.1 What PHP Is and How It Works

1.2 Writing and Using PHP Scripts

1.3 PHP Variable Types, Part I

1.4 PHP Variable Types, Part II

1.5 Retrieving Data from an HTML Form

1.6 Summary

Programming Assignment 1: Web-based Calculator

Discussion Assignments

Module 1: Introductions

Module 1: Getting Started

**Module 2: The Heart of PH**

Read PHP manual: Section II, Parts 9-12; Section IV, Part LXXIV

Commentary

2.1 Conditionals

2.2 Types of Operators

2.3 Precedence

2.4 Loops

2.5 Introduction to Functions

2.6 Functions and Values

2.7 Variable Scope

2.9 Pattern Examples

2.10 Summary

Self Study Quiz 1

Programming Assignment 2: PHP Variable Name Validator

Discussion Assignment:

Module 2: Regular Expressions

**Module 3: Data Storage**

Read text: PHP 6 and MySQL 5 for Dynamic Web Sites: Chapters 4, 5 and 6

Commentary

3.1 Files

3.2 Directories

3.3 File Uploads

3.4 Databases

3.5 An Example Database

3.6 Database Abstraction

3.7 Summary

Programming Assignment 3: Web-based Registration

Discussion Assignment:

Data Storage

**Module 4: Creating Web Application**

Read text: PHP 6 and MySQL 5 for Dynamic Web Sites: Chapters 3, 11, and 12

Commentary

4.1 Creating Templates Using Included Files

4.2 Persistence -- Cookies and Sessions

4.3 Authentication

4.4 A PHP Grab Bag: Date and Time, HTTP Headers, and E-mail

4.5 Summary

Programming Assignment 4: Enhanced Web-based registration

Discussion Assignment:

Creating Web Applications

**Module 5: Taking PHP to the Next Level**

Read text: PHP 6 and MySQL 5 for Dynamic Web Sites: Chapter 7

Commentary

5.1 Debugging

5.2 Security

5.3 Introduction to Objects and Classes (OOP)

5.4 Defining the Class

5.5 Creating an Object

5.6 PHP's Extended Capabilities

5.7 Third-Party Applications

5.8 Summary

Self-Study Quiz 2

Programming Assignment 5: Error Management

Discussion Assignment:

Strengths and Weaknesses

**Final Project**