

Learn PHP 7

Object-Oriented Modular Programming
using HTML5, CSS3, JavaScript, XML,
JSON, and MySQL



Steve Prettyman

Apress®

Learn PHP 7: Object-Oriented Modular Programming using HTML5, CSS3, JavaScript, XML, JSON, and MySQL

Steve Prettyman
Stone Mountain, Georgia USA

ISBN-13 (pbk): 978-1-4842-1729-0
DOI 10.1007/978-1-4842-1730-6

ISBN-13 (electronic): 978-1-4842-1730-6

Library of Congress Control Number: 2015960461

Copyright © 2016 by Steve Prettyman

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director: Welmoed Spahr

Lead Editor: Steve Anglin

Editorial Board: Steve Anglin, Louise Corrigan, Jonathan Gennick, Robert Hutchinson, Michelle Lowman,

James Markham, Susan McDermott, Matthew Moodie, Jeffrey Pepper, Douglas Pundick,

Ben Renow-Clarke, Gwenan Spearing

Coordinating Editor: Mark Powers

Copy Editor: Kezia Endsley

Compositor: SPi Global

Indexer: SPi Global

Artist: SPi Global

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a Delaware corporation.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this text is available to readers at www.apress.com. For additional information about how to locate and download your book's source code, go to www.apress.com/source-code/. Readers can also access source code at SpringerLink in the Supplementary Material section for each chapter.

Printed on acid-free paper

Contents at a Glance

About the Author	xi
Acknowledgments	xiii
Introduction	xv
■ Chapter 1: An Introduction to PHP 7.....	1
■ Chapter 2: Interfaces, Platforms, Containers, and Three-Tier Programming	39
■ Chapter 3: Modular Programming.....	77
■ Chapter 4: Secured User Interfaces.....	109
■ Chapter 5: Handling and Logging Exceptions.....	153
■ Chapter 6: Data Objects.....	187
■ Chapter 7: Authentication.....	223
■ Chapter 8: Multifunctional Interfaces.....	249
Index.....	289

Contents

About the Author	xi
Acknowledgments	xiii
Introduction	xv
■ Chapter 1: An Introduction to PHP 7.....	1
Chapter Objectives/Student Learning Outcomes	1
PHP 5.5+, PHP 7+, and PHP.NET	1
PHP 5.6+ and PHP 7+.....	8
Do It	8
PHP, JavaScript, CSS, HTML, and Apache Web Server.....	9
Do It	13
PHP, Apache, and MySQL	14
Do It	18
Putting it All Together—PHP, Apache, and MySQL.....	18
EasyPHP.....	18
XAMPP	22
Microsoft Internet Information Server	24
Do It	25
Testing Your Environment.....	25
Testing Your Administration Environment.....	25
Do It	27
Testing Your PHP Environment.....	28
EasyPHP’s Code Classroom	30
Do It	30

Alias Directories	30
Do It	33
Notepad++, Editors, and Code Testers.....	33
Notepad++	34
Other Editors.....	34
Do It.....	35
Chapter Terms	35
Chapter Questions and Projects	35
■ Chapter 2: Interfaces, Platforms, Containers, and Three-Tier Programming	39
Chapter Objectives/Student Learning Outcomes	39
PHP Platforms and Containers	39
PHP PC Applications	40
PHP Smart Phone Applications	40
PHP Facebook and Other Social Applications.....	40
Do It	41
PHP, AJAX, and CSS—Web Applications.....	47
PHP, AJAX, and CSS—Smart Phone Web Applications	52
PHP Three-Tier Architecture	57
Do It	58
Interface Tier	59
Do It	60
Business Rules Tier	61
Do It	62
Data Tier	63
Do It	64
Putting It All Together.....	64
Case Study.....	65
Do It	69

MVC and Dependency Injection.....	70
Chapter Terms	70
Chapter Questions and Projects	71
■ Chapter 3: Modular Programming.....	77
Chapter Objectives/Student Learning Outcomes	77
PHP Libraries, Extensions, Classes, and Objects.....	77
PHP Extensions	78
Classes and Objects	79
Creating a PHP Class.....	79
Do It	84
Return Method.....	84
Do It	86
Set Methods	87
Do It	93
Get Methods	93
Do It	96
Constructor Method.....	96
Do It	101
Chapter Terms	101
Chapter Questions and Projects	102
■ Chapter 4: Secured User Interfaces.....	109
Chapter Objectives/Student Learning Outcomes	109
Secured User Interaction.....	110
HTML5 Form Validation	110
Do It	113
JavaScript Validation.....	114
Do It	120

PHP Filtering.....	120
Do It	123
Additional HTML Input Security	123
HTML5 Select List Box and Radio Buttons.....	124
Do It	128
Validating Input with an XML File	128
Dependency Injection	135
Do It	147
Chapter Terms	147
Chapter Questions and Projects	148
■ Chapter 5: Handling and Logging Exceptions	153
Chapter Objectives/Student Learning Outcomes	153
Handling Exceptions.....	153
Do It	160
Exception and Error Handling vs. If/Else Conditions	160
Do It	167
Logging Exceptions	167
Do It	173
Reading Log and Text Files.....	174
Do It	182
Chapter Terms	182
Chapter Questions and Projects	183
■ Chapter 6: Data Objects	187
Chapter Objectives/Student Learning Outcomes	187
The Data Class	187
JSON Data	197
MySQL Data.....	197
Do It	199

Backup and Recovery.....	199
JSON Backup and Recovery.....	211
MySQL Backup and Recovery	211
Do It	214
Connecting the Data Tier	214
Do It	219
Chapter Terms	220
Chapter Questions and Projects	220
■ Chapter 7: Authentication.....	223
Chapter Objectives/Student Learning Outcomes	223
Verification and Sessions	223
JSON Data	232
MySQL Data.....	232
Do It	233
Registration.....	233
JSON Data	236
MySQL Data.....	237
Logging In.....	237
JSON Data	242
MySQL Data.....	242
Change Password.....	243
JSON Data	246
MySQL Data.....	246
Do It	246
Chapter Terms	247
Chapter Questions and Projects	247

Chapter 8: Multifunctional Interfaces.....	249
Chapter Objectives/Student Learning Outcomes	249
The Complete Application.....	249
Data Handling Using JavaScript.....	249
Do It	263
Updating, Deleting, and Inserting in the Interface Tier	263
Do It	270
Updating, Deleting, and Inserting in the Business Rules Tier	270
Do It	276
Final Touches.....	276
Do It	284
ABC Canine Shelter Reservation System Logical Design	285
Limitations.....	285
Chapter Terms	287
Chapter Questions and Projects	287
Index.....	289

About the Author

Steve Prettyman earned his Bachelor's of Arts Degree in education from Oglethorpe University in 1979. He quickly began his teaching career as a high school mathematics instructor while continuing his education by earning a Master's Degree in business information systems from Georgia State University (1985). Since then, Steve has spent over 30 years in the IT industry. He has been an instructor at Chattahoochee Technical College, Kennesaw State University, and Southern Polytechnic State University for almost 20 years. His primary teaching responsibilities include programming, web design, and web application development.

Acknowledgments

Thank you to everyone who has helped put this book together. Special thanks to the Introduction to PHP classes that have been the true testers and debuggers for this journey.

Special acknowledgement to all the open source developers and providers of free tutorials and training to every Internet user who wants to learn more about programming, especially w3schools and The New Boston.

Introduction

Learn PHP 7: Object-Oriented Modular Programming using HTML5, CSS3, JavaScript, XML, JSON, and MySQL is intended for use as a beginning level programming book. It is not the goal of this book to cover advanced techniques in the current versions of the PHP programming language. Some knowledge of general programming concepts is expected but no actual programming experience or education is assumed.

All code examples in this book are compatible with PHP 7. Most examples are compatible with PHP 5.6. The newest (as of the publication date) methods (functions) available in PHP have been used to provide the reader with the most current coding techniques. The examples use core methods provided in the PHP language. PHP includes many additional methods to accomplish similar tasks. The reader may, and should, research additional ways of improving security, performance, and other techniques. It is the goal of this book to prompt users to always consider using the most secure and efficient methods of program development. The code in this book provides some examples of using these techniques. The user should remember that *no program is 100% secure*. The programmer can only strive to make an application as secure as possible. It takes a team of developers, network personnel, security administrators, data center personnel, and others working together to provide the safest environment.

A Different Approach

There are quite a number of PHP books on the market today. What makes this book any different than any other?

- This book uses the concept of “learning by doing,” which shows the reader how to develop applications with conditional statements, loops, arrays, and methods. Over 70 PHP methods (functions) are introduced and demonstrated in coding examples.
- From the very first examples, the reader is introduced to object-oriented programming techniques. Many other books only briefly cover OO programming (if at all) in the final chapters.
- Object-oriented set methods are used to verify and filter user input. Many other books simply show a set method accepting data and storing it.
- A major objective of the book is to convince the reader to create all programs as secure and efficient as possible. The newest password encryption techniques (`password_hash`) are demonstrated.
- The try and catch methods are introduced to capture exceptions and some errors. The newest versions of PHP have been created to handle exceptions and errors using this approach. Many other books use die or other techniques to shut down a program.
- Multi-tier program design is introduced in the early chapters. This allows the reader to discover what logic and coding should take place in each tier. Many PHP books do not even cover this topic.

- The majority of the examples in the book are used to develop one main application (ABC Canine Shelter Reservation System). As the book progresses, the application is built from the beginning, in stages, showing the reader that application development should be broken into stages. Only after each stage is completed and tested, can the next stage begin. This approach works hand in hand with multi-tier design. Additional programming exercises and a term project are provided to enhance the understanding of development.
- The creation of user, change, and error logs are introduced. This allows the reader to gain an understanding of how to provide backup and recovery ability to keep an application functioning properly when security breaches or exceptions occur.
- The introduction of data objects and the data tier demonstrates to the reader the importance of creating an application that provides the ability to change data storage techniques and data storage location without requiring a major rewrite of the application. XML, JSON, and MySQL examples are provided.
- A natural relationship between PHP, HTML5, CSS3, and JavaScript is demonstrated throughout the book. This relationship is one of the major strengths of PHP.
- Throughout the book, web links are provided to point the user to additional resources to help understand the material or to dig deeper into the subject matter. Updates to link locations are provided on the book's web site.

Special Note for Teachers

The design of the content of this book provides flexibility in teaching styles and approaches. Each college and university approaches the initial education of programming concepts in different ways. This book provides three different types of programming exercises, which allow teachers to pick and choose what would work best in their environment. “Do It” exercises are provided in each chapter to allow the student to gain hands-on experience with techniques shown by modifying existing examples to produce the desired results. These exercises provide a level of confidence before the student attempts to program exercises at the end of the chapters. In addition, a Term Project is provided that builds an application that uses the same types of algorithms and programming techniques shown in the book.

Teaching tools, including test banks, course outline, and PowerPoint slides are available for use from the book's web site and from apress.com.

Code Examples, Images, and Links

Every effort has been made to catch any errors in code (and grammar). Please let us know if/when you discover problems in this book. Please send all corrections to Steve Prettyman (steve_prettyman@hotmail.com).

All code examples, images, and links are available for download from apress.com and the following location. You can download code examples from either web site. Copying code from the book may cause errors due to format requirements for publishing.

Book's web site: www.littleoceanwaves.com/securephp/

Chapter Overview

Chapter 1: An Introduction to PHP 7

After completing this chapter, the student will be able to:

- Understand the difference between LAMP, WAMP, and MAMP
- Successfully install a version of LAMP, WAMP, or MAMP
- Search the Internet for troubleshooting problems
- Explain the difference between a programming language and a scripting language
- Create an error-free simple PHP program

Chapter 2: Interfaces, Platforms, Containers, Three Tier Programming

After completing this chapter, the student will be able to:

- Give examples of platforms or containers that can host PHP programs
- Create a simple, dynamic web application using PHP
- Explain three-tier design and determine what is contained in each tier
- Design a three-tier application
- Explain each step of the program development life cycle (PDLC)
- Define and explain MVC and dependency injection

Chapter 3: Modular Programming

After completing this chapter, the student will be able to:

- Create an error-free simple objected-oriented (OO) modular PHP program
- Create a PHP class and make an instance of the class (object)
- Create an OO PHP encapsulated program, including get and set methods
- Create PHP methods (functions) that accept parameters and return information
- Create PHP public and private properties (variables)
- Import existing PHP code from another file or library into a program
- Validate information received using ternary (conditional) operators

Chapter 4: Secure User Interfaces

After completing this chapter, the student will be able to:

- Explain why user input must be validated in the interface and business rules tiers
- Explain why user input must be filtered in the business rules tier
- Use HTML5 code to validate user input
- Use JavaScript code to validate user input
- Use PHP if statements (conditional statements) to validate and filter input
- Use foreach loops to dynamically create an HTML select box from an XML file
- Use simple arrays for filtering and validation
- Pass simple arrays into methods (functions)
- Understand how to use dependency injection to control code version changes

Chapter 5: Handling and Logging Exceptions

After completing this chapter, the student will be able to:

- Explain the difference between errors and exceptions
- Create a PHP program that can handle general exceptions
- Create a PHP program that can create, raise, and handle user exceptions
- Explain and use a switch and/or embedded if/else statement
- Create a PHP program that uses the while loop and/or the for loop
- Create a program that reads/updates a text file using a two-dimensional array
- Create a PHP program that logs exceptions and e-mails support personnel

Chapter 6: Data Objects

After completing this chapter, the student will be able to:

- Create a data class that inserts, updates, and deletes XML or JSON data
- Explain how to create a data class that updates MySQL Data using a SQL Script
- Create a PHP program that creates a change backup log
- Create a PHP program that can recover data from a previous backup
- Apply changes to create up-to-date valid information
- Use dependency injection to attach a data class to another class in the BR tier
- Create a three-tier PHP application

Chapter 7: Authentication

After completing this chapter, the student will be able to:

- Define sessions and explain how sessions are used for authentication
- Create a PHP program that authenticates user logon
- Create a PHP program that register users
- Create a PHP program that will allow users to change passwords
- Create a PHP program that logs invalid login attempts

Chapter 8: Multifunctional Interfaces

After completing this chapter, the student will be able to:

- Create a complete PHP application that deletes, updates, and inserts data
- Create a professional look to a completed application using CSS
- Use JavaScript to accept and manipulate data from another program
- Secure all programs within an application requiring user IDs/passwords
- Populate HTML objects with values from a JSON object