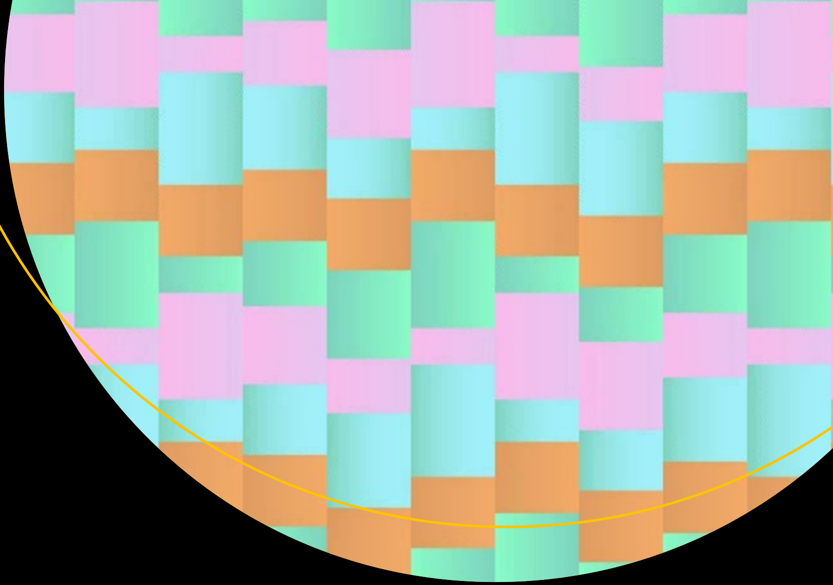


PHP Arrays

Single, Multi-dimensional, Associative and Object Arrays in PHP 7

Steve Prettyman

Apress®

PHP Arrays

Single, Multi-dimensional, Associative and Object Arrays in PHP 7

Steve Prettyman

Apress®

PHP Arrays: Single, Multi-dimensional, Associative and Object Arrays in PHP 7

Steve Prettyman
Key West, Florida, USA

ISBN-13 (pbk): 978-1-4842-2555-4 ISBN-13 (electronic): 978-1-4842-2556-1
DOI 10.1007/978-1-4842-2556-1

Library of Congress Control Number: 2016961720

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
Technical Reviewer: Tri Phan
Editorial Board: Steve Anglin, Pramila Balan, Laura Berendson, Aaron Black, Louise Corrigan, Jonathan Gennick, Robert Hutchinson, Celestin Suresh John, Nikhil Karkal, James Markham, Susan McDermott, Matthew Moodie, Natalie Pao, Gwenan Spearing
Coordinating Editor: Mark Powers
Copy Editor: Brendan Frost
Compositor: SPi Global
Indexer: SPi Global
Artist: SPi Global
Cover image: Designed by Freepik

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 materials referenced by the author in this text are available to readers at www.apress.com. For detailed information about how to locate 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

This book is dedicated to my wife, Beverly. Thank you for over 20 years of love and support; without you, this book would not be possible.

Contents at a Glance

About the Author xi

About the Technical Reviewer xiii

Introduction xv

■Chapter 1: PHP 7 Basics 1

■Chapter 2: Simple Arrays 27

■Chapter 3: Multidimensional Arrays 41

■Chapter 4: Associative and Object Arrays 57

■Chapter 5: PHP Functions—Changing, Splitting, Slicing, and Sorting Arrays.......... 75

■Chapter 6: PHP Functions—Comparing and Merging Arrays 101

■Chapter 7: PHP Functions—Searching, Traversing, and Displaying Arrays 123

■Appendix A: Case Study—Playing Checkers 143

Index 159

Contents

About the Author xi

About the Technical Reviewer xiii

Introduction xv

■**Chapter 1: PHP 7 Basics 1**

1.1 Installation 1

1.2 EasyPhp 2

1.2.1 Installing EasyPhp 3

1.2.2 Resolving Problems 3

1.2.3 Missing C# Library 3

1.2.4 Port Conflicts 3

1.2.5 Missing Files 4

1.2.6 Can't Install Files in Program Files Directory 4

1.2.7 Apache Delays and Hang-ups 4

1.3 Testing Your Environment 5

1.3.1 Resolving Problems 5

1.4 Alias Directories 6

1.5 How It All Works 6

1.6 Editors 7

1.7 The Basic Syntax 8

1.8 Conditional Statements 12

1.9 Switch Statement 17

1.10 Functions 17

1.11 Try/Catch Blocks 20

1.12 Arrays 21
1.13 For, While, Foreach Loops 22
1.14 Classes, Properties, Objects 22
Chapter 2: Simple Arrays 27
2.1 What Are Arrays? Why Do We Need to Use Them? 27
2.1.1 Why Do We Have a Choice? 30
2.2 Other Ways to Define Arrays 34
2.3 Html Arrays 36
2.3.1 Deleting 37
2.4 Updating & Inserting 38
Chapter 3: Multidimensional Arrays 41
3.1 Html Arrays 45
3.2 Deleting 53
3.2.1 Updating & Inserting 54
Chapter 4: Associative and Object Arrays 57
4.1 Deleting 64
4.1.1 Updating & Inserting 64
4.2 Object Arrays 68
Chapter 5: PHP Functions—Changing, Splitting, Slicing, and Sorting Arrays 75
5.1 Changing Array Contents 75
5.2 Splitting and Slicing Arrays 85
5.3 Sorting Arrays 88
Chapter 6: PHP Functions—Comparing and Merging Arrays 101
6.1 Comparing Arrays 101
6.2 Merging Arrays 112

■Chapter 7: PHP Functions—Searching, Traversing, and Displaying Arrays 123

7.1 Searching Arrays 123

7.2 Traversing Arrays 128

7.3 Displaying Array Contents 133

■Appendix A: Case Study—Playing Checkers 143

Index 159

About the Author

Steve Prettyman earned his bachelor 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. The last, almost 20 years, he has been an instructor and professor at Chattahoochee Technical College, Kennesaw State University, and Southern Polytechnic State University. He is currently the Computer Science Department Chair for Florida Keys Community College, Key West, Florida. His primary teaching responsibilities include programming, web design, and web application development.

About the Technical Reviewer

Tri Phan is the founder of the Programming Learning Channel on YouTube. He has over seven years of experience in the software industry. Specifically, he has worked in many outsourcing companies and has written many applications of many fields in different programming languages such as PHP, Java, and C#. In addition, he has over six years of experience in teaching at international and technological centers such as Aptech, NIIT, and Kent College.

Introduction

PHP Arrays: Single, Multidimensional, Associative, and Object Arrays in PHP 7 is intended for use as a supplemental 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 beginning knowledge of general PHP programming concepts is expected but no actual programming experience or education is assumed.

All code examples in this book are compatible with PHP 7. The newest 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 as shown within. The reader may, and should, research additional advanced array techniques after understanding the material presented in this book.

Special Note—Teachers

This book is provided as a supplementary guide to introductory textbooks on PHP 7. The intent of this book is to provide additional examples and explanation of the power and use of arrays in the PHP language. PHP arrays provide many capabilities that arrays in other languages do not provide.

Teaching tools, including test banks, course outline, and PowerPoint slides are available as part of the source code download available at the Apress website.

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 the following location. Please download code examples from the website. Copying code from the book may cause errors due to format requirements for publishing.

www.apress.com/9781484225554

CHAPTER 1

PHP 7 Basics

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

Create a simple error-free PHP program

Understand the use and value of conditional statements

Understand the use and value of for, while, and foreach loops

Understand the use and value of functions

Understand the use and value of arrays

Understand the basic structure of an object-oriented PHP program

1.1 Installation

The PHP environment can be installed on almost any operating system. This allows the developer the ability to easily create a development and testing environment. Complete testing can and should be completed before the code is installed in a live environment. The developer should determine the major PHP version used in the live environment and replicate this same version in the test environment.

PHP 7 includes many new tools and has removed some tools from previous versions. Therefore, it is imperative that both the live environment and the testing environment be the same. The testing environment can also be used to test minor release changes on existing code before the live environment is upgraded to the new release.

Although PHP can be installed by itself, novice and less experienced programmers should use one of the many installation tools available to install PHP with Apache Server, MySQL, PhpAdmin, and other related applications. These packages greatly simplify the process and are free and open source. Installing PHP separately requires a more in-depth knowledge of what versions of tools are compatible and changes required to the configuration files to link these tools together.

Apache, which is open source and free, is the most common server to use with PHP. However, PHP can be used with other servers, including Microsoft's Web Server. It is beyond the scope of this book to look at other servers. However, you can find installation information on the Internet.

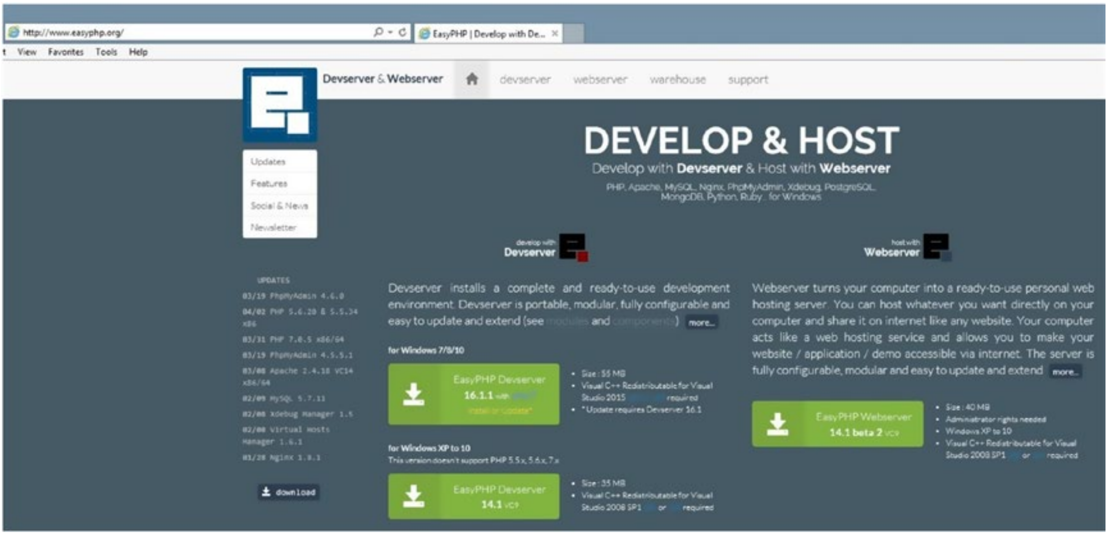
Electronic supplementary material The online version of this chapter (doi:10.1007/978-1-4842-2556-1_1) contains supplementary material, which is available to authorized users.

S. Prettyman, *PHP Arrays*, DOI 10.1007/978-1-4842-2556-1_1

MySQL, which is also open source and free, is the most common database used with PHP. PHP has the ability to use other databases, including Oracle and SQL Server among others. In many cases, the coding used to manipulate MySQL databases is very similar to the code used to manipulate other databases. php.net includes some basic information on drivers and coding for non-MySQL databases. You can also find additional information on the database's websites. PhpAdmin is a free tool to easily create and update MySQL database information. As stated before, most packages include a version of this software.

LAMP (Linux, Apache, MySQL, PHP), MAMP (Mac, Apache, MySQL, PHP), and WAMP (Windows, Apache, MySQL, PHP) package versions are readily available on the web. There are many organizations that currently create similar packages. We will briefly look at one of them. However, you will find that they all work in a similar way.

1.2 EasyPhp


http://www.easyphp.org/
EasyPHP | Develop with De...
t View Favorites Tools Help
Devserver & Webserver
devserver
webserver
warehouse
support
Updates
Features
Social & News
Newsletter
DEVELOP & HOST
Develop with Devserver & Host with Webserver
PHP, Apache, MySQL, Nginx, PhpMyAdmin, Xdebug, PostgreSQL, MongoDB, Python, Ruby... for Windows
develop with Devserver
host with Webserver
more...
EasyPHP Devserver 16.1.1
Install or Update*
EasyPHP Devserver 14.1 VC9
EasyPHP Webserver 14.1 beta 2 VC9
download

UPDATES
03/19 PhpMyAdmin 4.6.0
04/02 PHP 5.6.20 & 5.5.34 x86
03/31 PHP 7.0.5 x86/64
03/19 PhpMyAdmin 4.5.5.1
03/08 Apache 2.4.18 VC14 x86/64
02/09 MySQL 5.7.11
02/08 Xdebug Manager 1.5
02/08 Virtual Hosts Manager 1.6.1
01/28 Nginx 1.8.1

Devserver installs a complete and ready-to-use development environment. Devserver is portable, modular, fully configurable and easy to update and extend (see modules and components)

for Windows 7/8/10

- Size : 55 MB
- Visual C++ Redistributable for Visual Studio 2015 required
- *Update requires Devserver 16.1

for Windows XP to 10
This version doesn't support PHP 5.5.x, 5.6.x, 7.x

- Size : 35 MB
- Visual C++ Redistributable for Visual Studio 2008 SP1 or required

Webserver turns your computer into a ready-to-use personal web hosting server. You can host whatever you want directly on your computer and share it on internet like any website. Your computer acts like a web hosting service and allows you to make your website / application / demo accessible via internet. The server is fully configurable, modular and easy to update and extend

- Size : 40 MB
- Administrator rights needed
- Windows XP to 10
- Visual C++ Redistributable for Visual Studio 2008 SP1 or required

Figure 1-1. *EasyPhp*

The EasyPhp Development Server (available at easyphp.net) is a WAMP package which includes many additional tools, including the following:

Python

Ruby

Perl

Nginx

You can install this package on any storage device, including a usb key, memory stick, external hard drive, or your internal hard drive. This package provides easy configurations, along with direct access to your applications and files. In addition to configuration files, error logs, access logs, and application logs are provided. Additional features can be added to the base installation. Some of these are provided on the EasyPhp website.

1.2.1 Installing EasyPhp

The first time you attempt to install the development sever, you should accept the default settings provided by the developers. If you run into problems during installation, review the “Resolving Problems” section of this chapter.

■ **Warning** Pay attention to what buttons you are clicking when downloading the software. You might install much more than just EasyPhp.

1.2.2 Resolving Problems

The following are some of the most common installation problems and resolutions to these problems. If you encounter a problem not shown, or the resolution shown does not work in your environment, copy and paste any error code you receive in a search engine (such as Google) to determine how others have solved the problem.

1.2.3 Missing C# Library

PHP 7 and earlier versions of PHP require the Microsoft Visual Studio C# library. If you have Windows 8/10, this library is probably already installed. Also, if you have a recent version of Microsoft Visual Studio, it is also probably already installed. If you receive an error indicating that C# is missing or the wrong version, paste the message into a search engine on the Internet. Search for a response from Microsoft for directions to fix the error. The response should include a link to download the missing files and installation instructions.

1.2.4 Port Conflicts

If you already have a service using port 80, the default port for HTML traffic between your PC and the outside world, you will receive an error message from Apache when it attempts to run. You can fix this problem in multiple ways.

A. If you don’t mind shutting down other services using the port while you are developing, you can follow the next directions. Once you are done using Apache and PHP, you can turn the services back on or just reboot your PC and the services will turn back on.

1. Go to the Microsoft Windows 7/8/10 Task Manager (press Ctrl+Alt+Delete at the same time).

2. Select the Services tab.

3. Look for any of the following services in Windows 7/8/10. If you find one running, right-click it and turn it off. Then try restarting Apache again. If that does not work, turn that one back on and try another one. (The names may be slightly different depending on the version of Windows.)

SQL Server Reporter, Web Deployment Agent, BranchCache, Sync Share Service, WAS (IIS Administrator), and W3SVC

B. If you need your other services running or you do not have the administrative privileges to turn off services on port 80, you can change the default listening port location for Apache.

Go to your system tray (bottom-right corner of your screen). Find the EasyPHP icon by scrolling over the icons. A description of each should appear. If you don't see the icon, click the up arrow in the system tray to see more icons. Right-click the EasyPHP icon. Select Configuration and then select Apache. This will open the Apache configuration file (httpd.conf) into Notepad (or your default text editor). First save a copy of this file somewhere in case you make an error. This will allow you to recover from any major mistakes that occur.

Search for "Listen 127.0.0.1:80" within the file. Change the occurrence of 80 to 8080 or to 81; on that line only. This will allow the Apache server to listen to one of the ports that are not commonly used. Resave the file (make sure you are resaving the original file to the original location).

■ **Note** Make sure when you're using Notepad or any other text editor that you use Save As, and then select All Files for the file type. Also make sure to include the .conf file extension. If you do not change the file type to all files, your file will be saved as httpd.conf.txt. If that happens, the server will not see the file. You can easily fix the problem by reopening the file and saving it in the proper method.

You can then restart Apache by going back to your system tray to find the EasyPHP icon. Double-click the icon; a message box will appear that will give you the status Apache and MySQL. You will probably see red for the Apache status. Click the Apache button. Within a few moments, it should turn green. This will indicate that the server is now running. Do the same for MySQL.

1.2.5 Missing Files

If you receive an error message related to this, somehow your files have become corrupted before installation. Return to the EasyPHP web site and download the files again. Also, if you somehow mess up the Apache configuration file, go back and reinstall the product again.

1.2.6 Can't Install Files in Program Files Directory

This indicates that you or something else has a high security restriction on that directory. Rerun the installation and change the location of your installation to another directory. Just remember when we reference the program files directory later in this book that you should instead look at the directory in which your files were installed.

1.2.7 Apache Delays and Hang-ups

In Windows 8/10 you may experience problems with Apache working slowly or hanging up. To correct this problem go to your system tray (bottom-right corner of your screen). Find the EasyPHP icon by scrolling over the icons. A description of each should appear. If you don't see the icon, click the up arrow in the system tray to see more icons. Right-click the EasyPHP icon. Select Configuration and then select Apache.

This will open the Apache configuration file (httpd.conf) into Notepad (or your default text editor). First save a copy of this file somewhere in case you make an error. This will allow you to recover from any major mistakes that occur.

Then add the following two lines to the bottom of the file.

```
AcceptFilter http none

AcceptFilter https none
```

Resave the file (make sure you are resaving the original file to the original location).

1.3 Testing Your Environment

Before spending a lot of time coding, you need to ensure that your environment has been set up properly. The following suggestions will provide some clues to indicate if the environment is working.

1. Testing Your Administration Environment

   First we need to test the server and see if our administration pages will display. Open your favorite browser and enter the following: http://127.0.0.1/home/

   If the environment is working, you will see an administrative page for Apache Server. Otherwise, check to see if the server is actually running in your computer's services tab.

2. Testing the Development Environment

   Open a text editor (such as Notepad, Notepad++, or Textedit) and enter the following code:

   ```
   <?php
       print "Hello World";
   ?>
   ```

Using the *Save As* selection on the File menu, change the File Type to *All Files* or to *php*. Enter the file name test.php and save it to the following location.

```
C:\Program Files (x86)\EasyPHP-DevServer-16.1VC11\data\localweb\projects
```

Of course, you should change the version name (or program file name) to the correct version (location) that you are using on your machine.

If you saved your files correctly, you can attempt to run your program by entering the following in your browser.

```
http://127.0.0.1/projects/test.php
```

1.3.1 Resolving Problems

If "Hello World" does not display in your browser when you run the test program, review the following error resolutions. If you do not see your error, or the suggested resolution does not work, copy and paste your error message into a browser and try some of the suggestions from other users on the web.

Nothing is displayed, error 404:

1. Make sure you typed the address exactly as shown.
2. Your server might be hung up. Stop and restart it.
3. Make sure you placed your file in the correct location.
4. Make sure you saved your file as a .php file and not as .txt. Try *Save As* again and renaming the file (make sure file type is either All Files or php).
5. Check for typos in your actual program code. Did you remember the semicolon (;)?

   Fix any and resave. You might need to stop and start the server if it does not see the changes for some reason. You can go look at the PHP log files to see any errors that might exist in your code.

6. Go to the Apache log files to look for errors. If you cannot correct them, copy the errors and paste them in a search engine to see what others have found as solutions.

The actual program code is displayed not the results of executing the code:

1. Make sure you saved your file as a .php file and not as .txt. Try Save As again and renaming the file (make sure file type is either All Files or php).
2. Your Apache server or PHP might not be started or is hung. Stop and start Apache again.
3. Did you forget or have a typo in the *<?php* or *?>* lines?
4. Go to the Apache log files to find the errors. If you cannot correct them, copy the errors and paste them in a search engine to see what others have found as solutions.

1.4 Alias Directories

Apache allows you to create additional directories to host your PHP programs. Alias directories must include an alias name (such as php1) and the actual physical location of the file(s) (such as c://myfiles/php). The alias name is used by the server to determine the actual location of the file.

```
www.nothingmuch.com/php1/test.php
```

The user entering this address might assume that the test.php program exists in a folder with a name of php1. However, this might not be the case. If php1 is actually an alias directory the browser will redirect to a different location (such as c://myfiles). This feature allows some sense of security because the user does not know which files are kept where. It also allows flexibility as files could be moved (from c://myfiles) to another location (such as c://oldfiles) without the user knowing. The person moving the files can update the alias information in the server to a new location, and keep the same alias name (such as php1).

1.5 How It All Works

When a url is entered into a browser, the request is sent to a web server (Apache). If the file contains only html, JavaScript, and/or css, then the file requested is sent directly to the requesting browser. The requesting browser will then interpret the html, css, and JavaScript code. The results of the interpretation are then displayed in the browser.

If the file contains additional code, such as PHP, the server (Apache) will determine how to handle the additional code. PHP programs run within the Apache server (not the browser). The Apache server uses the file ending (.php) to determine if there is PHP code within a file. It will use the opening (<?php) and closing (?>) tags that surround PHP code to determine what needs to be sent to the PHP environment to be interpreted and executed. The PHP environment will return the results of the execution of the program, such as “Hello World” in the test example, back to the Apache server. The server will then return any output (along with any html, css, and/or JavaScript code) back to the browser.