

2

Setting Up a Development Server

If you wish to develop Internet applications but don't have your own development server, you will have to upload every modification you make to a server somewhere else on the Web before you can test it.

Even on a fast broadband connection, this can still represent a significant slowdown in development time. On a local computer, however, testing can be as easy as saving an update (usually just a matter of clicking once on an icon) and then hitting the Refresh button in your browser.

Another advantage of a development server is that you don't have to worry about embarrassing errors or security problems while you're writing and testing, whereas you need to be aware of what people may see or do with your application when it's on a public website. It's best to iron everything out while you're still on a home or small office system, presumably protected by firewalls and other safeguards.

Once you have your own development server, you'll wonder how you ever managed without one, and it's easy to set one up. Just follow the steps in the following sections, using the appropriate instructions for a PC, a Mac, or a Linux system.

In this chapter, we cover just the server side of the web experience, as described in Chapter 1. But to test the results of your work—particularly when we start using JavaScript and CSS later in this book—you should also have an instance of every major web browser running on some system convenient to you. Whenever possible, the list of browsers should include at least Internet Explorer, Mozilla Firefox, Opera, Safari, and Google Chrome.

If you plan to ensure your sites look good on mobile devices too, then you should also try to arrange access to a wide range of Apple iOS and Google Android phones and tablets.

What Is a WAMP, MAMP, or LAMP?

WAMP, MAMP, and LAMP are abbreviations for “Windows, Apache, MySQL, and PHP,” “Mac, Apache, MySQL, and PHP,” and “Linux, Apache, MySQL, and PHP.”

These abbreviations describe a fully functioning setup used for developing dynamic Internet web pages.

WAMPs, MAMPs, and LAMPs come in the form of a package that binds the bundled programs together so that you don't have to install and set them up separately. This means you can simply download and install a single program, and follow a few easy prompts, to get your web development server up and running in the quickest time with a minimum hassle.

During installation, several default settings are created for you. The security configurations of such an installation will not be as tight as on a production web server, because it is optimized for local use. For these reasons, you should never install such a setup as a production server.

But for developing and testing websites and applications, one of these installations should be entirely sufficient.

If you choose not to go the W/L/MAMP route for building your own development system, you should know that downloading and integrating the various parts yourself can be very time-consuming and may require a lot of research in order to configure everything fully. But if you already have all the components installed and integrated with each other, they should work with the examples in this book.

Installing a WAMP on Windows

There are several available WAMP servers, each offering slightly different configurations, but probably the best is Zend Server Free Edition, because it's free and is from the developers of PHP itself. You can download it at tinyurl.com/zendfree as shown in Figure 2-1. If there is a problem with that short URL, the full URL of the download page is <http://zend.com/en/products/server/free-edition>.

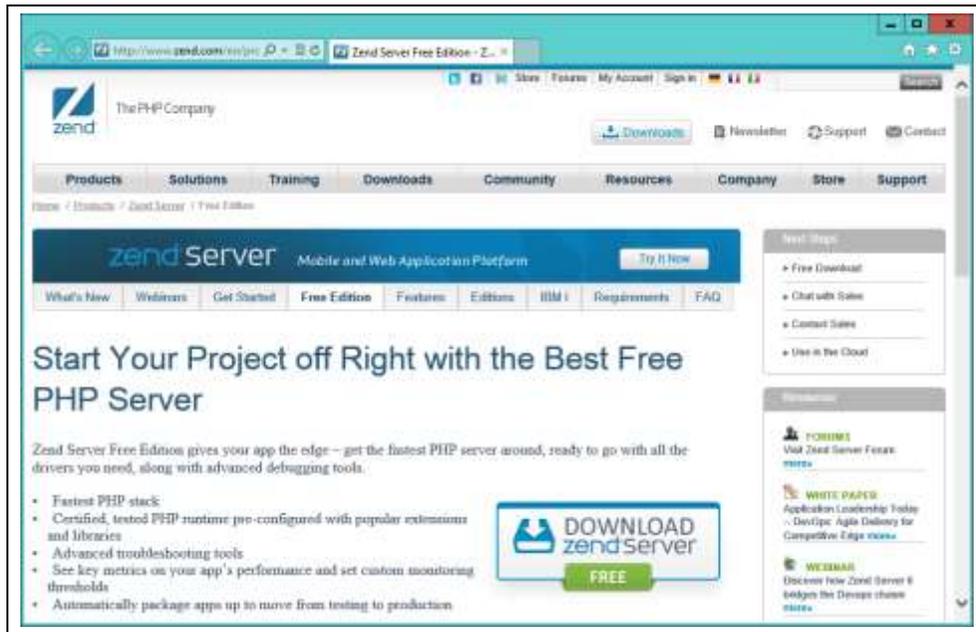


Figure 2-1. You can download the Free Edition from the Zend Website

I recommend you always download the latest stable release (in this instance it's 6.0.1/PHP 5.4 for Windows), which will be listed first in the download section of the web page, which should display the correct installer for your computer out of Linux, Windows OS X and IBM i.

It's possible that during the lifetime of this edition some of the screens and options shown in the following walkthrough may change. If so just use your common sense to proceed in as similar manner to the sequence of actions described.

Once downloaded, run the installer to bring up the window in Figure 2-2.



Figure 2-2. The main installation window of the installer

Click 'Next' and accept the license agreement that follows to move onto the 'Setup Type' screen (see Figure 2-3), select the 'Custom' option so that the MySQL server can also be installed.



Figure 2-3. Choose the 'Custom' install option

When the Custom Setup window appears scroll down the list of options to the bottom and ensure that MySQL Server is checked, as shown in Figure 2-4, then click 'Next'.

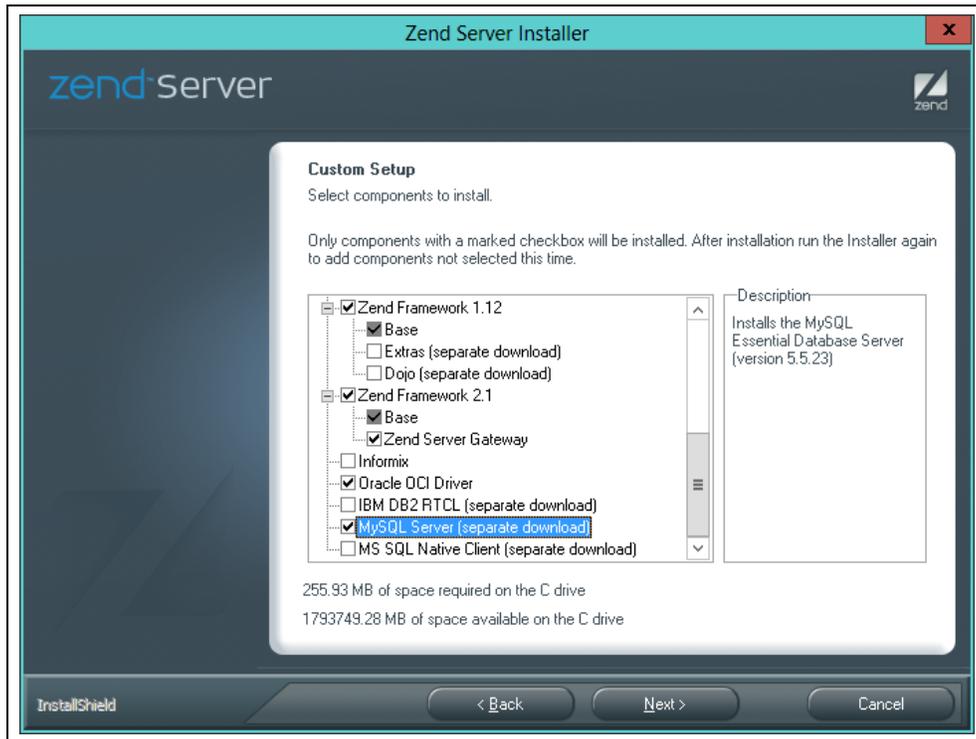


Figure 2-4. Check MySQL Server before continuing

On the following screen (see Figure 2-5), even if you already have an IIS web server installed, I recommend that you choose to install the Apache web server because the examples in this book are for Apache, then click 'Next'.



Figure 2-5. Install the Apache web server

Now accept the default values of 80 for the Web Server Port, and 10081 for the Zend Server Interface Port (see Figure 2-6) and click 'Next'.

If either of the ports offered state that they are occupied (generally this will be because you have another web server running), and don't allow you to use the defaults, then try a value of 8080 (or 8000) for the Web Server Port, and 10082 for the Zend Server Interface Port, and remember these values later when accessing either web pages or the Zend server. For example, instead of visiting *localhost/index.htm* in your web browser you would use *localhost:8080/index.htm*.



Figure 2-6. Accept the default values offered for the Ports

Once the ports have been assigned you will reach the screen in Figure 2-7, and you should click 'Install' to start the installation.



Figure 2-7. Now you are ready to click 'Install' to proceed

During installation some extra files may be downloaded so it may take a few minutes for the programs to get set up. But when they are ready you will be notified and offered to start using the software by clicking 'Finish'. When you do so your default browser will be opened up with the page shown in Figure 2-8 where, to continue, you must check the box to agree with the terms.

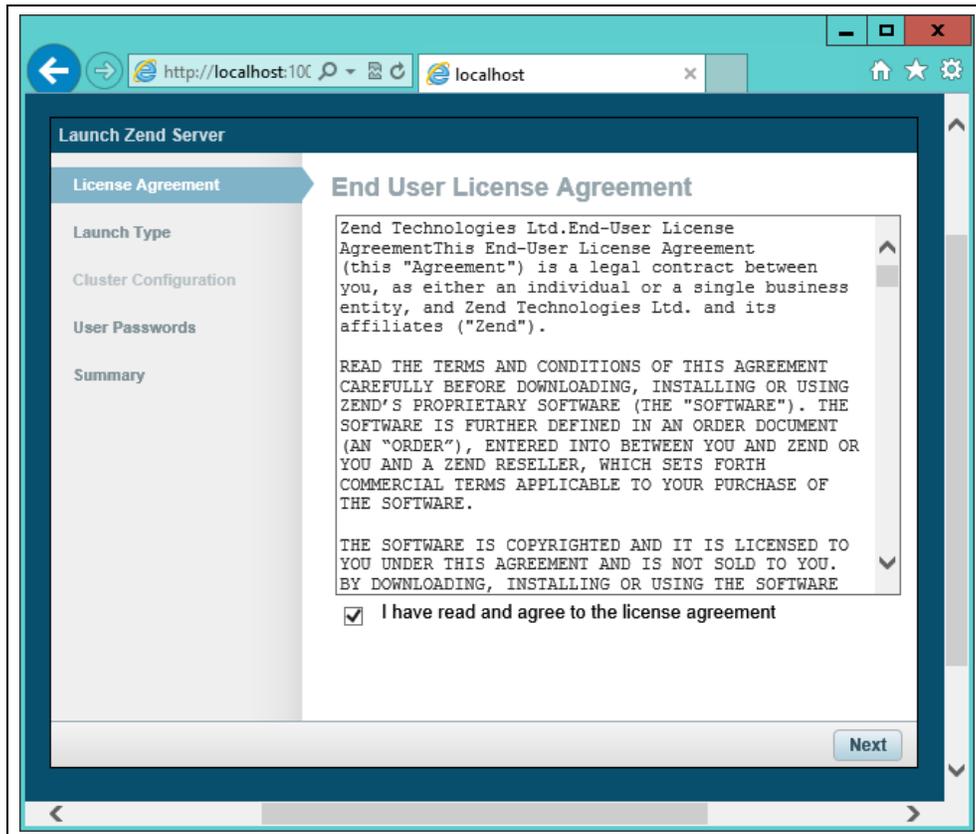


Figure 2-8. You must agree to the terms in order to use the server

Next you are asked how you will be using the server. I recommend you select the 'Development' option for the purposes of working through the exercises in this book (see Figure 2-9).

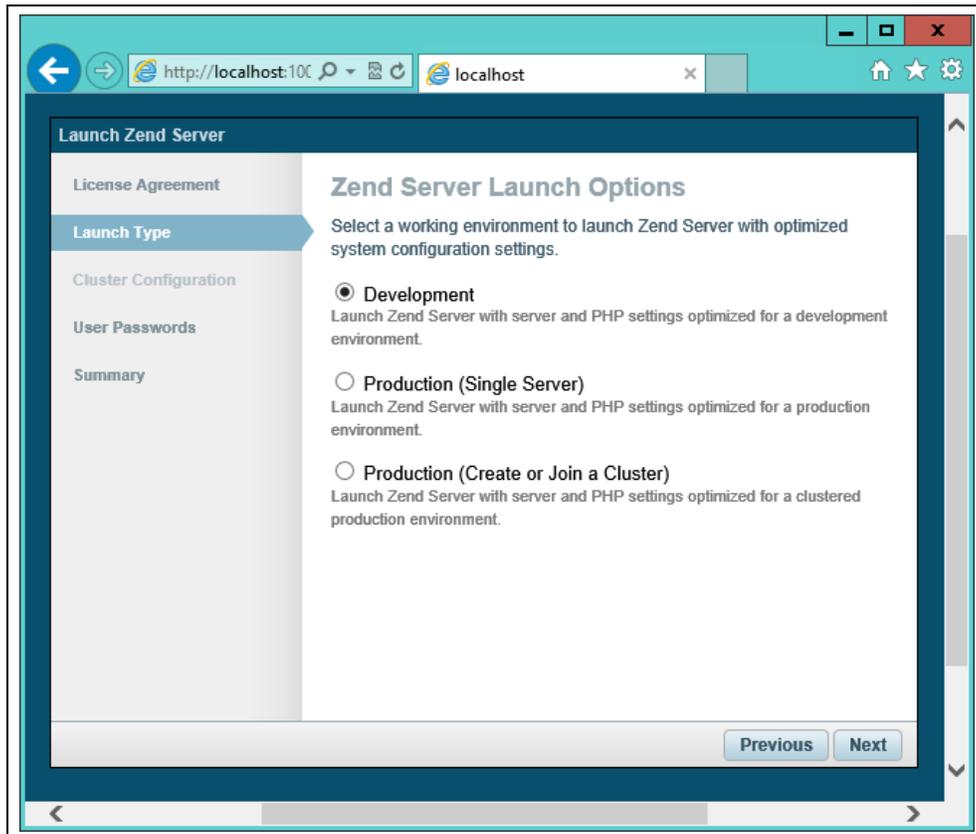


Figure 2-9. Select the Development option

Now you are ready to set a password for the user 'admin' (see Figure 2-10). You do not need to enter a password for the user 'developer'. Make sure you choose a password you will remember and click 'Next' to proceed to the screen shown in Figure 2-11, where you can now click 'Submit' to finish installation. Note that the 'Create or Join a Cluster' option may not appear on the OS X version of the installer.

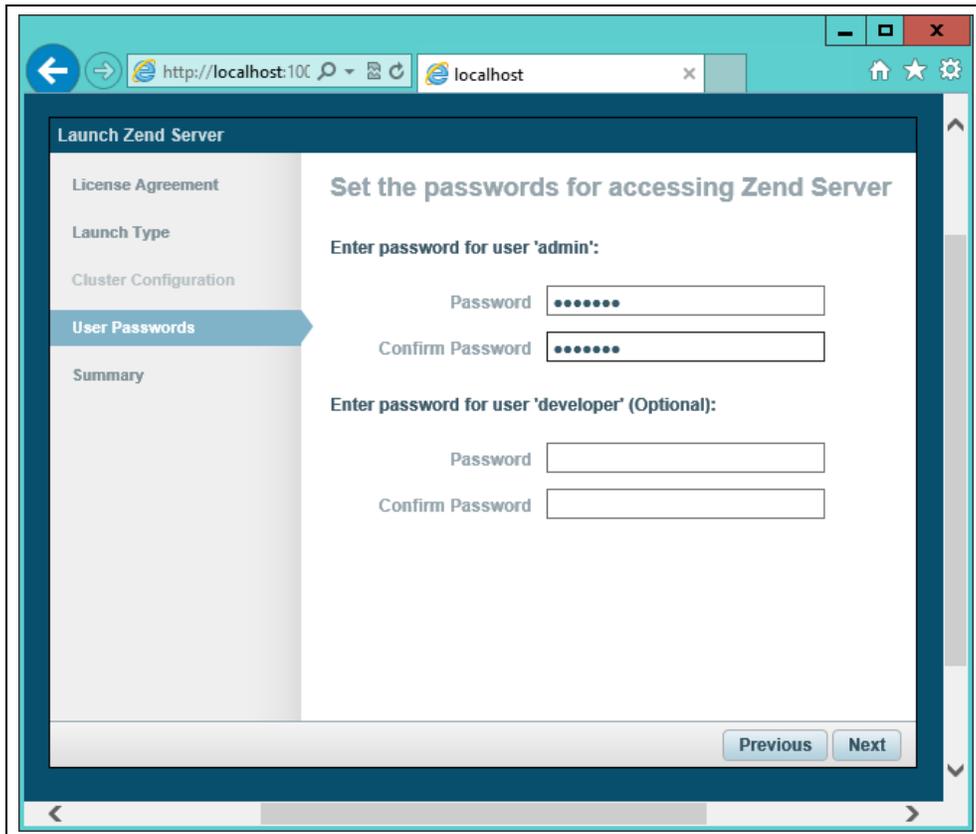


Figure 2-10. Choose your password and enter it twice

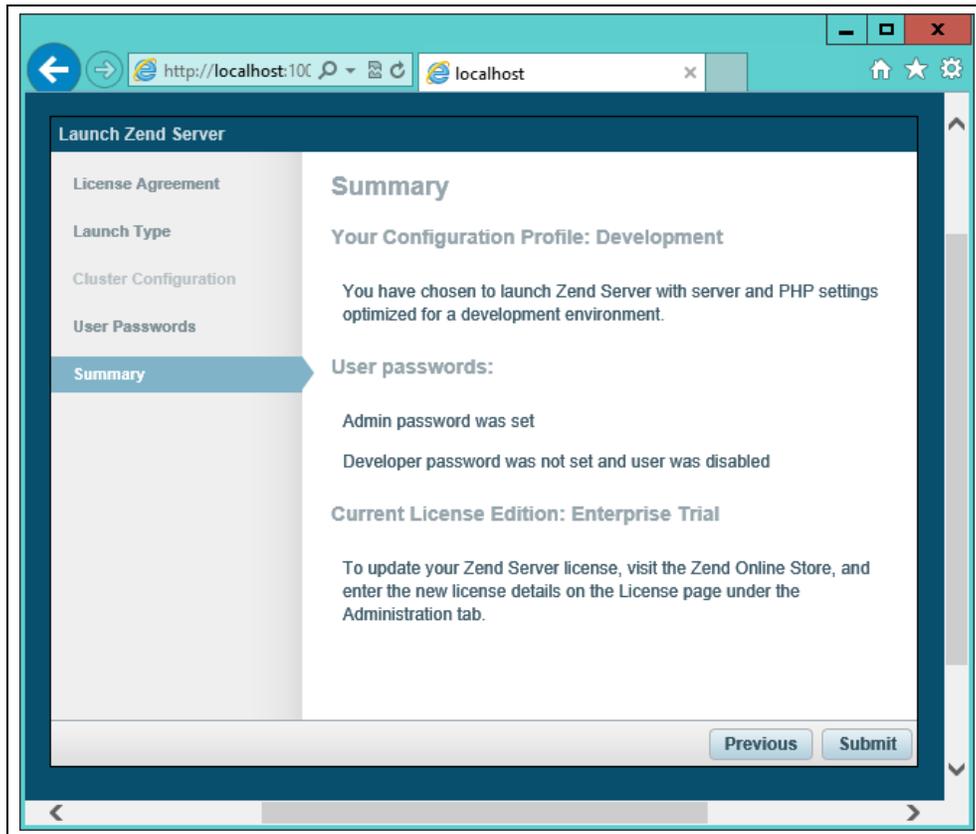


Figure 2-11. Click 'Submit' to complete setup

After a short wait your browser will show the Dashboard screen in Figure 2-12, which is the place where you can administer the server.



Figure 2-12. The Zend Server administration screen

You can return to this screen at any time by entering <http://localhost:10081> into your browser. Or, if you entered a value other than 10081 for the Zend Server Interface Port (or 10088 on a Mac), then you can get to this screen by using that value after the colon instead.

Testing the Installation

The first thing to do at this point is verify that everything is working correctly. To do this, you are going to try to display the default web page, which will have been saved in the server's document root folder (see Figure 2-13). Enter either of the following two URLs into the address bar of your browser:

```
| localhost  
| 127.0.0.1
```

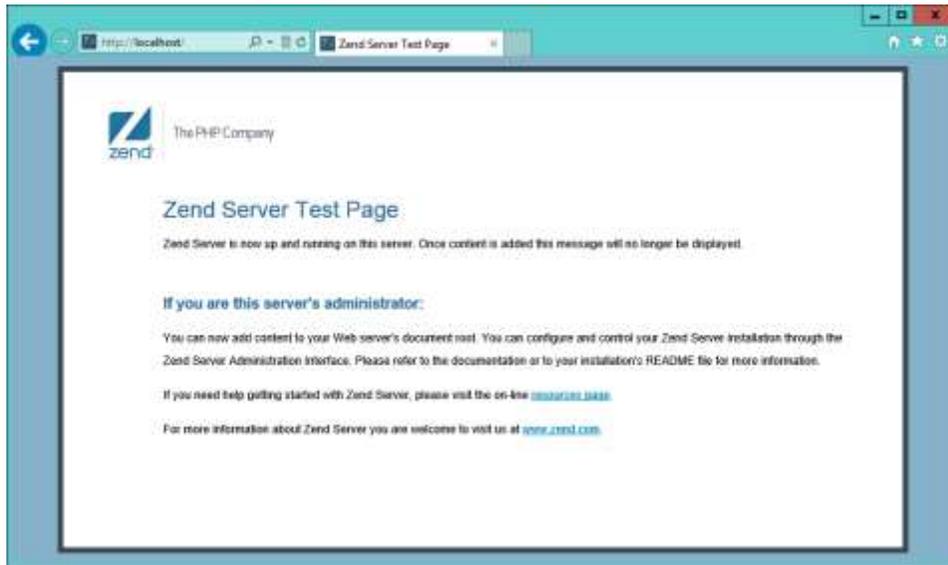


Figure 2-13. How the home page should look by default

The word *localhost* is used in URLs to specify the local computer, which will also respond to the IP address of *127.0.0.1*, so you can use either method of calling up the document root of your web server.

If you chose a server port other than 80 during installation (such as 8080) then you must place a colon followed by that value after either of the preceding URLs. For example: *localhost:8080*. You will also have to do the same for all example files in this book. For example, instead of the URL *localhost/example.php* you should enter the following one: *localhost:8080/example.php* (or whatever value you chose).

The document root is the directory that contains the main web documents for a domain. This is the one that is entered when a basic URL without a path is typed into a browser, such as *http://yahoo.com*, or for your local server, *http://localhost*.

By default, Zend Server uses one of the following locations for this directory (the former for 32-bit computers and the latter for 64-bit):

```
| C:/Program Files/Zend/Apache2/htdocs
| C:/Program Files (x86)/Zend/Apache2/htdocs
```

If you are not sure whether your computer is 32-bit or 64-bit, try to navigate to the first directory and, if it exists, you have a 32-bit machine. If not, open up the second directory because you have a 64-bit computer.

To ensure that you have everything correctly configured, you should now create the obligatory “Hello World” file. So create a small HTML file along the following lines using Windows Notepad or any other program or text editor, but not a rich word processor such as Microsoft Word (unless you save as Plain Text):

```
| <html>
```

```
<head>
  <title>A quick test</title>
</head>
<body>
  Hello World!
</body>
</html>
```

Once you have typed this in, save the file into the document root directory previously discussed, using the file name *test.htm*—if you are using Notepad make sure that the “Save as type” box is changed from “Text Documents (*.txt)” to “All Files (*.*)”. Or, if you prefer, you can save the file using the *.html* file extension.

You can now call this page up in your browser by entering one of the following URLs (according to the extension you used) in its address bar (see Figure 2-14):

```
http://localhost/test.htm
http://localhost/test.html
```

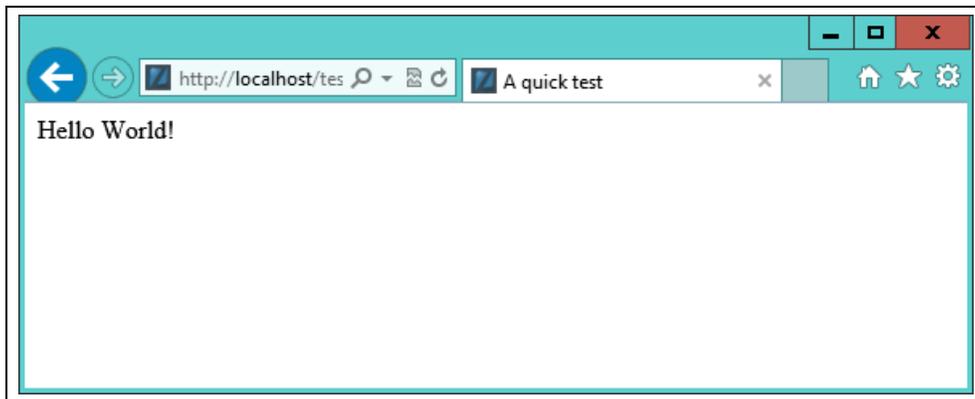


Figure 2-14. Your first web page

You should now have had a trouble-free installation, resulting in a fully working WAMP. But if you encountered any difficulties, check out the comprehensive Documentation at <http://tinyurl.com/zenddocs>, which should sort out your problem.

Throughout this book, whenever there's a long URL to type in, I use the tinyurl.com web address shortening service to save time and reduce typos. For example tinyurl.com/zendfree and tinyurl.com/zenddocs are much shorter than the URLs they lead to:

<http://www.zend.com/en/products/server/free-edition>
<http://files.zend.com/help/Zend-Server-6/zend-server.htm>

Alternative WAMPs

When software is updated, it sometimes works differently than you'd expected, and bugs can even be introduced. So if you encounter difficulties that you cannot resolve, you may prefer to choose one of the various other solutions available on the Web instead.

You will still be able to make use of all the examples in this book, but you'll have to follow the instructions supplied with each WAMP, which may not be as easy to follow as the preceding guide.

Here's a selection of the best in my opinion:

- EasyPHP: <http://www.easyphp.org/>
- XAMPP: <http://apachefriends.org/en/xampp.html>
- WAMPServer: <http://wampserver.com/en/>
- Glossword WAMP: <http://glossword.biz/glosswordwamp/>

Installing a MAMP on Mac OS X

Zend Server Free Edition is also available on OS X, and you can download it from tinyurl.com/zendfree as shown in Figure 2-15. If there is a problem with that short URL, try entering <http://zend.com/en/products/server/free-edition>.

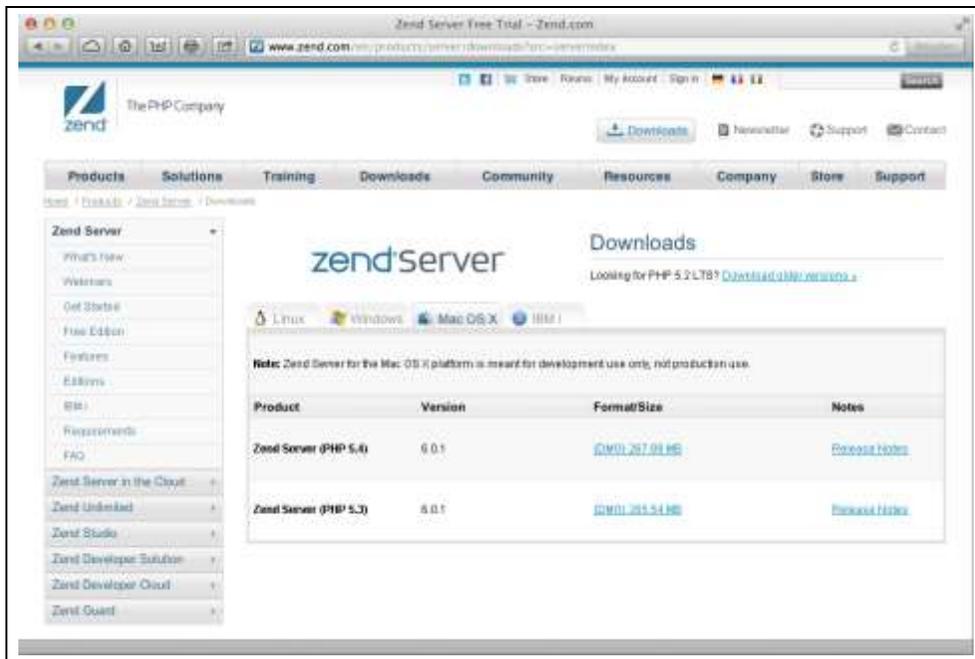


Figure 2-15. You can download the server from the Zend Website

I recommend you always download the latest stable release (in this instance it's 6.01/PHP 5.4 for OS X), which will be listed first in the download section of the web page, which should display the correct installer for your computer out of Linux, Windows, OS X, and IBM i. You may be asked to log-in before you download, but you can also click a link to get the file without logging in or registering, although you'll miss out on product update emails and other news.

Once downloaded, double click the *.dmg* file, wait for the download to verify, and then the window in Figure 2-16 will appear.



Figure 2-16. Double click Zend Server to install it

Here you can double click the 'README' file for instructions, or double click 'Zend Server' to open up the installation window shown in Figure 2-17.



Figure 2-17. The Zend Server Installer

Now click 'Continue', read the instructions that are then displayed, and then click 'Continue' again to reach the screen shown in Figure 2-18, where you can decide where to put the installed software, the default being *Macintosh HD*. Click 'Install' when you are ready, and enter your password if prompted for it.



Figure 2-18. Choosing the destination for installation

During installation you may be asked whether you wish to install additional software. If so I recommend accepting everything offered to you by clicking the 'Install' button. Upon completion you can click 'Close' to close the installer.

Once installed, to complete setup locate the 'ZendServer' program in your Applications folder and double click it. This will bring up a page in your default web browser similar to that shown in Figure 2-8. Now follow the prompts you are given (shown in Figures 2-8 through 2-11), in which you must accept the license agreement and choose a password, before being taken to the main dashboard, as shown in Figure 2-12.

Configuring MySQL

Unfortunately the Installer doesn't set up the commands needed to be able to start, stop and restart the MySQL server, so you're going to have to do this manually by opening the Terminal and entering the following command:

```
| sudo nano /usr/local/zend/bin/zendctl.sh
```

After entering your password you will now be in the Nano text editor so move the cursor down a few lines using the down cursor key, and where you see the line that reads `MySQL_EN="false"`, change the word `false` to `true`.

Now scroll down some more until you find these two lines:

```
| case $1 in
|     "start")
```

Then below it you'll see an indented line that reads:

```
|         $0 start-apache %
```

Just after this line insert a new one that reads as follows:

```
|         $0 start-MySQL %
```

This will allow MySQL to start, but now you need to scroll down a little more until you get to the section that starts with:

```
|     "stop")
```

Then below it you'll see an indented line that reads:

```
| $0 stop-apache %
```

Just after this line insert a new one that reads as follows:

```
| $0 stop-MySQL %
```

This will allow MySQL to be stopped. Now you can press Ctrl-X to exit from edit mode, press the Y key when prompted to save the changes, and then press Return to save the edited file.

Ensuring MySQL starts on booting

Unfortunately there's another edit you have to make so that MySQL will start when your Mac does, and that's to issue the following commands from the Terminal (supplying your password in the relevant place if prompted for it):

```
| cd /Library/StartupItems/ZendServer_init/  
| sudo rm zendctl.sh  
| sudo ln -s /usr/local/zend/bin/zendctl.sh ./
```

Your Mac is now configured but MySQL has not yet been started, so now you must issue the following command (along with password if prompted) and you should be all set to go:

```
| sudo /Library/StartupItems/ZendServer_init/zendctl.sh restart
```

Testing the Installation

You can now test the installation by entering either of the following URLs into your web browser to call up the screen shown in Figure 2-13:

```
| localhost:10088  
| 127.0.0.1:10088
```

The word *localhost* specifies the local computer (which will also respond to the IP address of *127.0.0.1*). And the reason for having to enter *:10088* is because many Mac computers will already have a web server running, so this avoids any clash.

You must therefore remember to place *:10088* after every *localhost* for all examples in this book. So, for example, if the file name *test.php* is being accessed, you would call it up from the browser using the URL *localhost:10088/test.php*.

If you are sure that there isn't another web server running on your Mac you can edit the configuration file at the following URL (ensuring you have permission to do so), changing the command (at around line 40) that reads `Listen 10088` to `Listen 80`:

```
/usr/local/zend/apache2/conf/httpd.conf
```

You will then need to restart the server by opening the Terminal utility and issuing the following command (along with your password if prompted), and you will then no-longer need to add the *:10088* to local URLs:

```
sudo /usr/local/zend/bin/zendctl.sh restart
```

The page that gets displayed in the browser when you go to *http://localhost* or *http://localhost:10088* is the file *index.html* in the server's document root (the directory that contains the main web documents for a domain). This is the directory that is entered

when a basic URL without a path is typed into a browser, such as <http://yahoo.com>, or in the case of your local web server, <http://localhost>, etc...

By default, Zend Server on OS X uses the following as its document root folder:

```
| /usr/local/zend/apache2/htdocs
```

To ensure that you have everything correctly configured, you should now load a test file. So create a small HTML file along the following lines using Windows TextEdit or any other program or text editor (such as the popular TextWrangler), but not a rich word processor like Microsoft Word (unless you save as Plain Text):

```
<html>
  <head>
    <title>A quick test</title>
  </head>
  <body>
    Hello World!
  </body>
</html>
```

Once you have typed this in, save the file into the document root directory using the file name *test.htm*. Or, if you prefer, use the *.html* file extension. You can now call this page up in your browser by entering one of the following URLs (according to the extension you saved with) in its address bar (see Figure 2-14):

```
| http://localhost:10088/test.htm
| http://localhost:10088/test.html
```

You should now have had a trouble-free installation, resulting in a fully working MAMP. But if you encountered any difficulties, check out the comprehensive Documentation at <http://tinyurl.com/zenddocs>, which should sort out your problem.

Throughout this book, whenever there's a long URL to type in, I use the tinyurl.com web address shortening service to save time and reduce typos. For example tinyurl.com/zendfree and tinyurl.com/zenddocs are much shorter than the URLs they lead to:

```
http://www.zend.com/en/products/server/free-edition
http://files.zend.com/help/Zend-Server-6/zend-server.htm
```

Installing a LAMP on Linux

This book is aimed mostly at PC and Mac users but its contents will work equally well on a Linux computer. However there are dozens of popular flavors of Linux and each of them may require installing a LAMP in a slightly different way, so I can't cover them all in this book.

However, many Linux versions come pre-installed with a web server and MySQL, and the chances are that you may already be all set to go. To find out try entering the following into a browser and see whether you get a default document root web page:

```
| http://localhost
```

If this works you probably have the Apache server installed and may well also have MySQL up and running too—check with your system administrator to be sure, though.

If you don't yet have a web server installed, however, there's a version of Zend Server Free Edition available that you can download at <http://tinyurl.com/zendfree>.

All the instructions and help you need are detailed on the download page. Follow them closely or use the provided scripts and you should be able to work through all the examples in this book.

Working Remotely

If you have access to a web server already configured with PHP and MySQL, you can always use that for your web development. But unless you have a high-speed connection, it is not always your best option. Developing locally allows you to test modifications with little or no upload delay.

Accessing MySQL remotely may not be easy either. You may have to Telnet or SSH into your server to manually create databases and set permissions from the command line. Your web hosting company will advise you on how best to do this and provide you with any password they have set for your MySQL access (as well as, of course, for getting into the server in the first place).

Logging In

I recommend that, at minimum, Windows users should install a program such as PuTTY, available at <http://putty.org>, for Telnet and SSH access (remember that SSH is much more secure than Telnet).

On a Mac, you already have SSH available. Just select the *Applications* folder, followed by *Utilities*, and then launch Terminal. In the terminal window, log in to a server using SSH as follows:

```
| ssh mylogin@server.com
```

where *server.com* is the name of the server you wish to log into and *mylogin* is the username you will log in under. You will then be prompted for the correct password for that username and, if you enter it correctly, you will be logged in.

Using FTP

To transfer files to and from your web server, you will need an FTP program. If you go searching the Web for a good one, you'll find so many that it could take you quite a while to come across one with all the right features for you.

Nowadays I always recommend FireFTP, because of these advantages:

- It is an add-on for the Firefox web browser, and will therefore work on any platform on which Firefox runs.
- Calling it up can be as simple as selecting a bookmark.
- It is one of the fastest and easiest to use FTP programs that I have encountered.

You may say "But I use only Microsoft Internet Explorer and FireFTP is not available for it," but I would counter that if you are going to

develop web pages, you need a copy of each of the main browsers installed on your PC anyway, as suggested at the start of this chapter.

To install FireFTP, visit <http://fireftp.mozdev.org> using Firefox and click on the Download FireFTP link. It's about half a megabyte in size and installs very quickly. Once it's installed, restart Firefox; you can then access FireFTP from the Tools menu (see Figure 2-19).

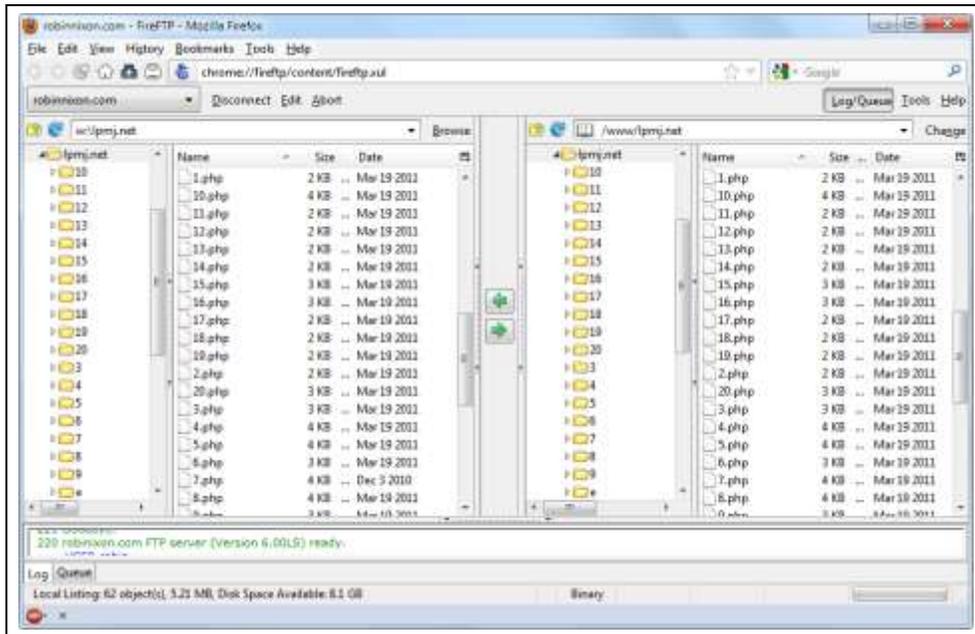


Figure 2-19. FireFTP offers full FTP access from within Firefox

Another excellent FTP program is the open source FileZilla, available from <http://filezilla-project.org>, for Windows, Linux and Mac OS X 10.5 or newer.

Of course, if you already have an FTP program, all the better—stick with what you know.

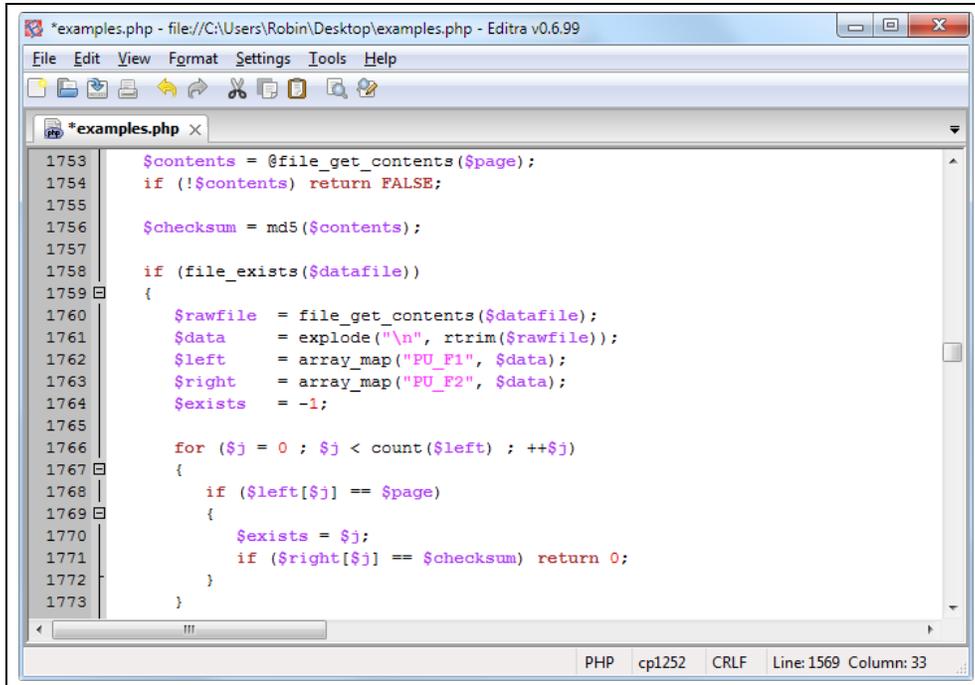
Using a Program Editor

Although a plain-text editor works for editing HTML, PHP, and JavaScript, there have been some tremendous improvements in dedicated program editors, which now incorporate very handy features such as colored syntax highlighting. Today's program editors are smart and can show you where you have syntax errors before you even run a program. Once you've used a modern editor, you'll wonder how you ever managed without one.

There are a number of good programs available, but I have settled on Editra, because it's free and available Mac, Windows and Linux/Unix. You can download a copy by visiting <http://editra.org> and selecting the Download link towards the top left of the page, where you can also find the documentation for it.

As you can see from Figure 2-20, Editra highlights the syntax appropriately using colors to help clarify what's going on. What's more, you can place the cursor next to brackets or

braces and Editra will highlight the matching pair so that you can check whether you have too many or too few. In fact, Editra does a lot more in addition, which you will discover and enjoy as you use it.



```
*examples.php - file://C:\Users\Robin\Desktop\examples.php - Editra v0.6.99
File Edit View Format Settings Tools Help
*examples.php X
1753 $contents = @file_get_contents($page);
1754 if (!$contents) return FALSE;
1755
1756 $checksum = md5($contents);
1757
1758 if (file_exists($datafile))
1759 {
1760     $rawfile = file_get_contents($datafile);
1761     $data = explode("\n", rtrim($rawfile));
1762     $left = array_map("PU_F1", $data);
1763     $right = array_map("PU_F2", $data);
1764     $exists = -1;
1765
1766     for ($j = 0 ; $j < count($left) ; ++$j)
1767     {
1768         if ($left[$j] == $page)
1769         {
1770             $exists = $j;
1771             if ($right[$j] == $checksum) return 0;
1772         }
1773     }
PHP cp1252 CRLF Line: 1569 Column: 33
```

Figure 2-20. Program editors are superior to plain-text editors

Again, if you have a different preferred program editor, use that—it's always a good idea to use programs you're already familiar with.

Using an IDE

As good as dedicated program editors can be for your programming productivity, their utility pales into insignificance when compared to *Integrated Developing Environments* (IDEs), which offer many additional features such as in-editor debugging and program testing, as well as function descriptions and much more.

Figure 2-21 shows the popular phpDesigner IDE with a PHP program loaded into the main frame, and the right-hand Code Explorer listing the various classes, functions, and variables that it uses.

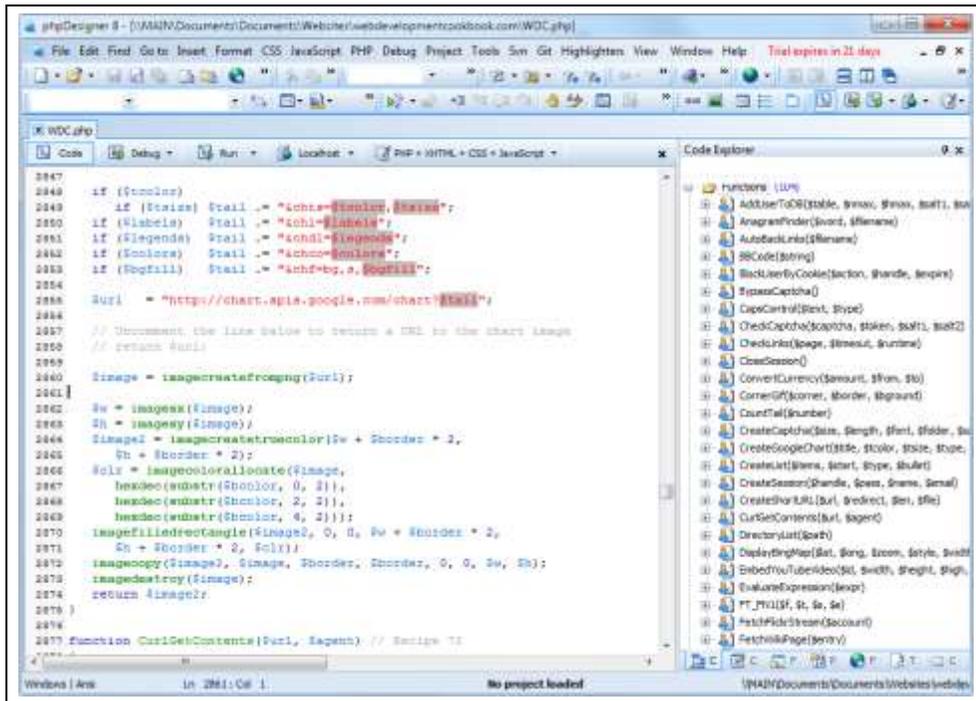


Figure 2-21. When using an IDE such as phpDesigner, PHP development becomes much quicker and easier

When developing with an IDE, you can set breakpoints and then run all (or portions) of your code, which will then stop at the breakpoints and provide you with information about the program’s current state.

As an aid to learning programming, the examples in this book can be entered into an IDE and run there and then, without the need to call up your web browser.

There are several IDEs available for different platforms, most of which are commercial, but there are some free ones too. Table 2-1 lists some of the most popular PHP IDEs, along with their download URLs.

Choosing an IDE can be a very personal thing, so if you intend to use one, I advise you to download a couple or more to try them out first—they all either have trial versions or are free to use, so it won’t cost you anything. Table 2-1. A selection of PHP IDEs

IDE	Download URL	From	Win	Mac	Lin
Eclipse PDT	http://eclipse.org/pdt/downloads/	Free	☑	☑	☑
Komodo IDE	http://activestate.com/Products/komodo_ide	\$245	☑	☑	☑
NetBeans	http://www.netbeans.org	Free	☑	☑	☑
phpDesigner	http://mpsoftware.dk	\$39	☑	☐	☐
PHPEclipse	http://phpeclipse.de	Free	☑	☑	☑
PhpED	http://nusphere.com	\$119	☑	☐	☑

IDE	Download URL	From	Win	Mac	Lin
PHPEdit	http://phpedit.com	\$119	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Zend Studio	http://zend.com/en/downloads	\$395	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

You should take the time to install a program editor or IDE you are comfortable with and you'll then be ready to type in and try out the examples in the coming chapters.

Armed with these tools, you are now ready to move onto Chapter 3, where we'll start exploring PHP in further depth and find out how to get HTML and PHP to work together, as well as how the PHP language itself is structured. But before moving on, I suggest you test your new knowledge with the following questions.

Questions

- 2-1. What is the difference between a WAMP, a MAMP and a LAMP?
- 2-2. What do the IP address 127.0.0.1 and the URL <http://localhost> have in common?
- 2-3. What is the purpose of an FTP program?
- 2-4. Name the main disadvantage of working on a remote web server.
- 2-5. Why is it better to use a program editor instead of a plain text editor?

See the "Chapter 2 Answers" section in Appendix A for the answers to these questions.