

Introduction to PHP

PHP is a scripting language used primarily on Web servers running the Apache HTTP server software. While uses for PHP are many, the core functionality is to provide a language and framework for the development of Web applications. As of November 2004, the reported usage of PHP was at nearly 18 million domains. Combined with Linux, the Apache HTTP server, and the MySQL database system, PHP provides functionality to the great portion of the servers on the World Wide Web.

History

PHP/FI

The name PHP is a recursive acronym meaning "PHP: Hypertext Preprocessor." The roots of PHP are in the Perl language, most notably in a set of Perl scripts created by Rasmus Lerdorf in 1995 for use in developing his own personal Web site. The toolset he created was called PHP/FI, or "Personal Home Page/Forms Interpreter," and allowed users to easily manage submitted forms and perform logging and tracking functions. The initial version of PHP was an interpreted scripting language that could be embedded into HTML pages and a parser that processed the embedded scripts on each page request. By adhering to the standard HTTP request-response cycle the scripts could be easily developed, debugged, deployed, and maintained.

PHP 3

The popularity of the PHP module and the enthusiasm of the Open Source community lead to PHP being rewritten as an Apache module with robust capabilities and enhancements. The first version of PHP written as an Apache module and resembling the language we use today was PHP 3. The PHP 3 project was created by Andi Gutmans and Zeev Suraski in 1997. The most important change to the PHP effort in version 3 was an architecture that allowed developers to easily contribute to and extend PHP.

PHP 3 continued the notion of using embedded scripts, or *scriptlets*, in an HTML document. By passing documents with a `.php` extension through the PHP preprocessor before allowing Apache to serve them, each embedded scriptlet could be parsed and executed and any relevant output substituted in its place. Simple tasks such as variable substitution or more complex actions such as database connectivity were suddenly available to users with a knowledge of HTML and access to the PHP documentation. Because of the interpreted nature of PHP, compiling scripts was not a requirement. Much like the way that browsers render HTML each time a page is viewed, the PHP module for Apache would render server-side scripts. The change-and-refresh development cycle was familiar to HTML developers, and PHP quickly developed a following among programmers and markup specialists alike. At its peak, it is estimated that PHP 3 was installed on 10 percent of the Web servers on the Internet.

History (continued)**PHP 4**

Each version of PHP seems to not only add functionality but also optimize the existing code base. Much like the development of PHP 3, the next version, PHP 4, was in large part rewritten to take into account new features and optimizations.

The central goals in the PHP 4 project were to optimize the code base to allow for more efficiency in large and complex applications and to facilitate modularity and portability in both the underlying language implementation and the interpreted scripting language used by Web developers. An important addition to PHP 4 was limited support for object-oriented programming (OOP) using classes. Though incomplete in comparison to other languages, the support for OOP in PHP 4 allowed developers with little experience in other languages to begin to understand the importance and power of OOP.

PHP 5

Object-oriented programming is a central feature in the newest version of PHP, which is PHP 5. Though still in its early releases, PHP 5 has been praised as a mature language for Web programming, comparable in many ways to Java and the Microsoft C# language. Though different from both in many ways, PHP 5 has already proven to be an excellent object-oriented language for Web development whose benefits in many ways outshine those of the aforementioned languages.

Open Source

The licensing of PHP is undoubtedly one of the core attractions. PHP is open source and free software, which means that there is no cost for installation or usage and that any skilled developer can contribute to the project. Because the source code from which PHP is compiled is open, a community of peers comprised of people from all over the world can review the source, helping to identify inefficiencies, stability issues, and security risks. Along with locating these problems, any developer may help repair the code base.

Cross-Platform

PHP is a cross-platform language, meaning that both the engine and the code can be used on nearly every major operating system. The majority of the code written in PHP is portable from Unix to Windows to Mac OS X, for example. The cross-platform nature makes it easy for developers working on different operating systems and server configurations to share their code, creating a rich code base available by way of open-source projects and community Web sites.

Database Interaction

Another great strength of PHP is the ease with which a developer can create scripts that interact with databases. Database-driven Web sites are in high demand and the robust database libraries available for PHP allow developers to create sites quickly and efficiently, giving them the luxury of concentrating not on database connectivity implementation, but on the ways in which databases can be used with their sites.

Development Tools

Aside from a Web server with a build of PHP, the only additional tool you need to develop PHP applications is a text editor. There are no special integrated development environments (IDEs) or compilers required, though there are many available. Anyone can open a standard text file and begin writing PHP code. The low overhead and accessible learning curve are big attractions for developers looking for the path of least resistance in generating dynamic, server-side content.

Install Apache on Windows

In order to use PHP, you must have Web server software installed and configured to use PHP. The most popular Web server software is the Apache HTTPD server. Check the Apache Web site, www.apache.org, for the most recent stable release.

The Apache HTTPD Web server is extremely reliable and robust. Apache is the most popular Web server on the Internet and is used on small, low-traffic sites as well as on extremely large, load-balanced commercial sites.

The Apache Web server for Windows installation process is fairly simple. Like most Windows installers, the Apache Web server installer is an executable "wizard" installer that takes you through the license acceptance and installation process. The Apache Web server can be installed on computers running any version of Windows, from Windows 95 through 98, 2000, Me, and XP.

The first screen of the installation wizard gives an overview of the software to be installed. The second screen is a standard End User License Agreement (EULA) stating the rights and restrictions for the software. You must consent to the license in order to proceed to the next step of the installation process.

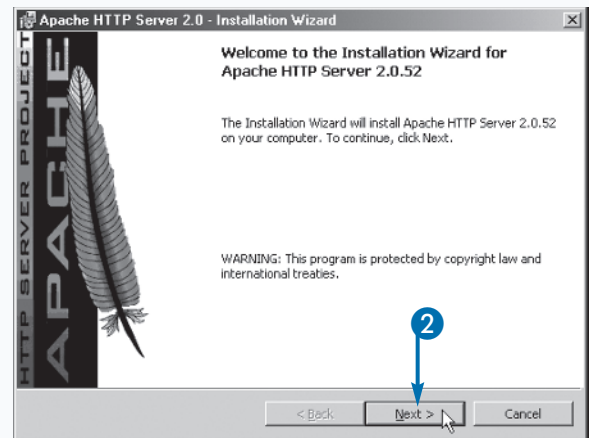
After agreeing to the EULA, you are presented with a screen asking you to specify your Network Domain, Server Name, and the e-mail address of the administrator for the computer. The administrator will most likely be yourself, and you can use your own e-mail address. For development on a local computer, this is not as important as when hosting a production server. This dialog box will also present the option to install Apache for all users or only for your account. Generally it is best to install for all users, especially if you are the sole user of the computer.

You can choose the type of installation you want to perform during the install process, allowing for a custom configuration. You can always modify the configuration of Apache after installation, if need be.

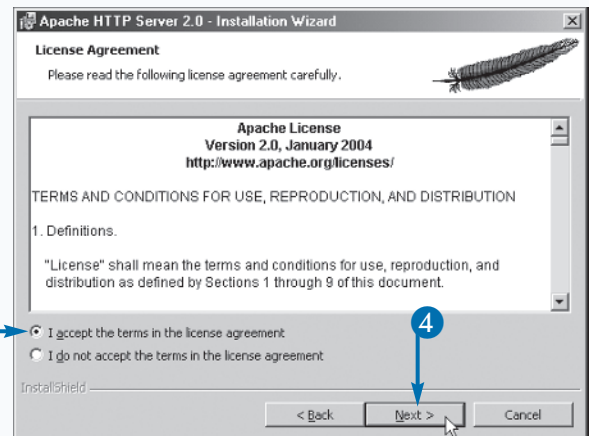
Install Apache on Windows

- 1 Begin the installation process by double-clicking the icon for the Apache Web server installation program.
- 2 Click Next to continue on to the first step of the installer.

The Installation Wizard appears.



- 3 When presented with the license agreement, read the license and select the I accept option to accept the terms.
- 4 Click Next to continue.
- 5 View the contents of the Read This First window and then click Next to continue.



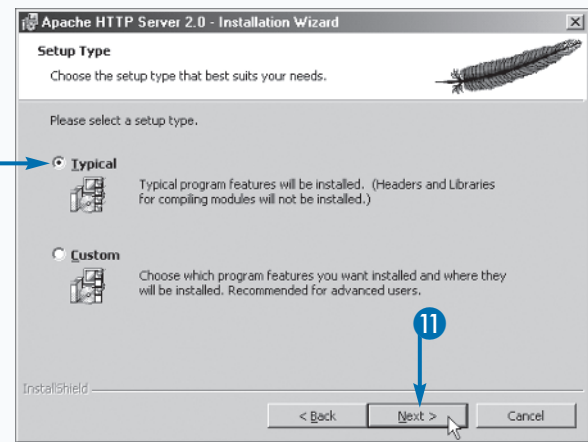
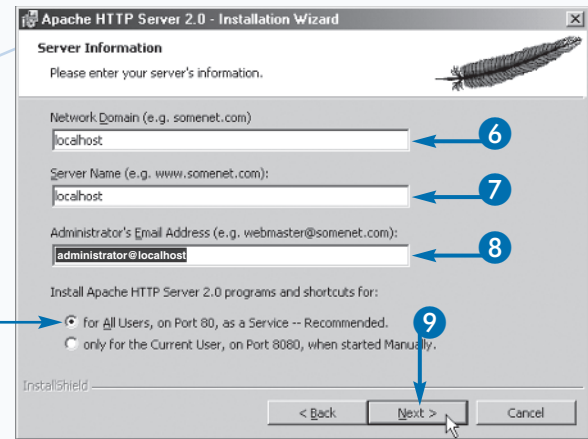
- 6 In the Server Information window, type your network domain.

Note: You can use the default, localhost, if no other is available.

- 7 Type the name of your server here.
- 8 Type an administrator e-mail address, such as administrator@localhost.
- Leave the recommended installation option selected.
- 9 Click Next to continue.

- 10 When the Setup Type window appears, select the default Typical installation.

- 11 Click Next to continue.



Extra

When installing Apache on Windows, you can choose one of two standard installation options. The Typical option is the recommended option and installs the most standard extensions and configuration options. The Custom option allows advanced system administrators and developers to select which modules they want to install.

Because Apache is an open-source project, there are no telephone hotlines or commercial support Web sites to which you can go for customer or technical support. There are a number of community sites on the Web whose members can provide assistance, but discretion should be used when communicating potentially sensitive information or taking advice. Most members of the open-source community are extremely helpful and kind, but all advice should be checked against manuals and documentation before committing changes to your server software.

continued →

Install Apache on Windows (continued)

The default destination for the installation is C:\Program Files\Apache Group\Apache2. You can specify another destination during the installation process, but it is recommended that you accept the default value for ease of maintenance and future updates to the Apache software. When browsing the Apache installation directory, you will find documentation as well as several standard modules that are used to add functionality to Apache.

The installation wizard displays a screen stating the default installation location of the Apache Web server software. You can choose to leave the default location as your target location, or change the location to another directory on your computer. Typically, it is best to install in the default location so that there is consistency across similar machines on which Apache is installed. You can change the installation directory by clicking Browse in the dialog box.

After choosing the location in which you want to install the Apache Web server, the wizard presents a final confirmation

screen that allows you to go back and change any settings of which you are unsure. When you are confident in your choices for installation parameters, you can click Install to begin the automated installation process. If you decide against installing the Apache Web server at this point, you can click Cancel to abandon the process.

After the installation process is complete the wizard displays a screen noting that a successful installation operation has occurred. You can click Finish to end the wizard and resume normal computing.

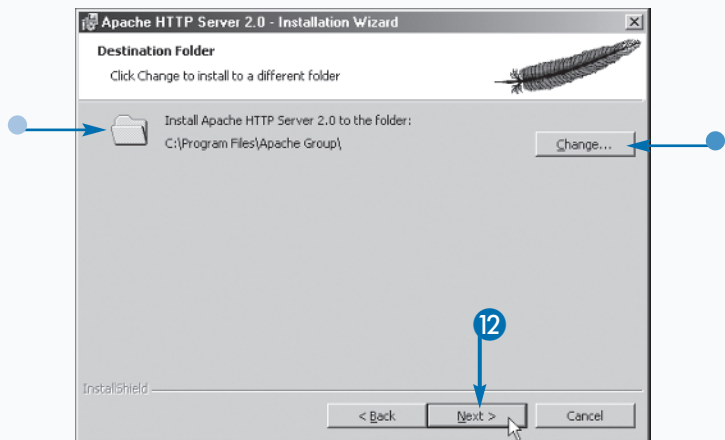
To confirm that the Apache Web server has been installed and that it is running properly, you can use your Web browser to view a page. By default, Apache installs a welcome page in the main Web directory from which it serves files. You can view this page using the local *loopback address* `http://127.0.0.1/`. The loopback address is a special address each computer uses to identify itself to local applications. If the Apache installation process was successful, the default Apache welcome page appears.

Install Apache on Windows (continued)

The Destination Folder window appears.

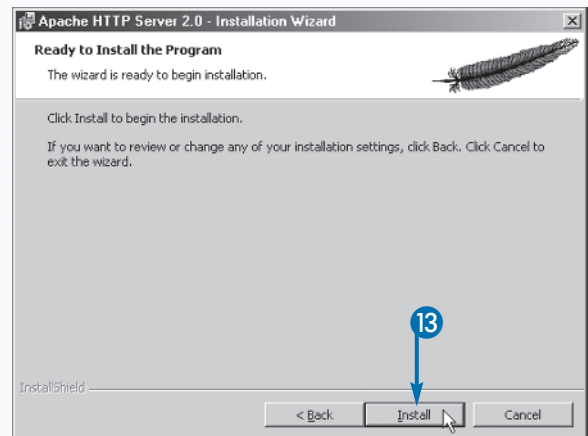
- It is recommended that you use the default folder for the installation directory.
- Alternately, you can click Change and browse to another folder.

12 Click Next to continue.



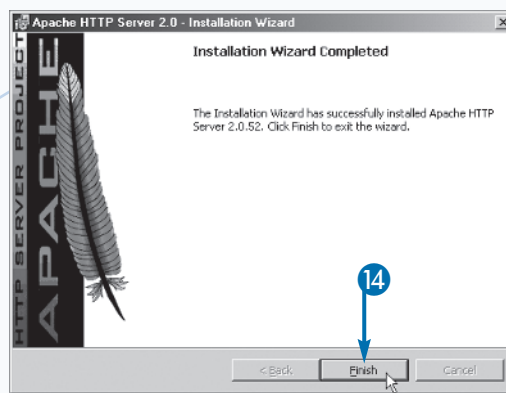
The Ready to Install the Program window appears.

13 View the information and click Install to continue.



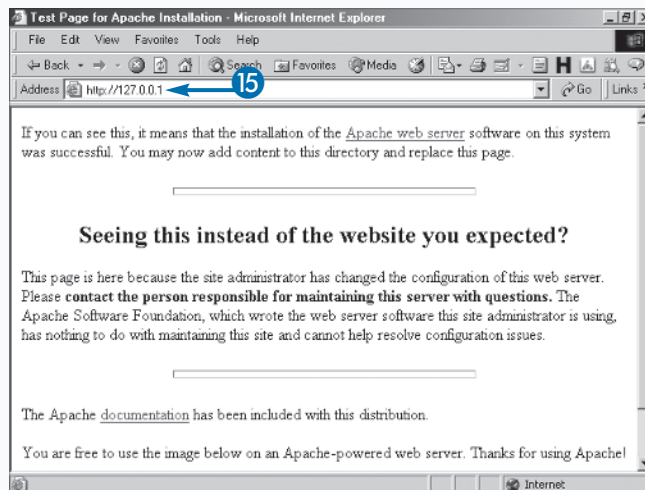
The installer sets up the Apache files.

- 14 Click Finish to conclude the installation process.



- 15 You can test the installation by opening a Web browser and going to the location: `http://127.0.0.1/`.

If your browser does not load the page, try the installation again.



Extra

You can start and stop Apache using the command prompt. In the command prompt window, change to the directory in which Apache was installed, usually `C:\Program Files\Apache Group\Apache2`. Type `apache` to start the Web server. You can use the same command to stop Apache by passing arguments. To stop Apache, type `apache -k shutdown`.

When installing Apache Web server on a computer running a Windows operating system, a folder containing shortcut commands for the Apache Web server is typically added to the Programs folder on the Start menu. This folder allows quick access to common Apache tools and commands.

The installation program installs a README file in the Apache installation folder that contains information specific to the release of Apache you installed. The README file is an important part of the application package because it provides succinct notes about changes in the software as well as any special configuration advice or compatibility notes. While not required, it is a good idea to read this file.

Install Internet Information Services on Windows

The Microsoft Internet Information Server (IIS) is Web server software that allows you to host multiple Web sites on the Internet or on a network such as a corporate Intranet using a standard Windows computer. With the IIS and accompanying tools, you can host and manage even the busiest Web sites. Though PHP is used more often with the Apache Web server than with Internet Information Services, IIS can be used with PHP quite well in both development and production.

You can install IIS from the Windows XP CD-ROM with which you installed Windows. First, use the Start menu to begin adding the program. Click Start, Control Panel, and then Add or Remove Programs. From the Add or Remove

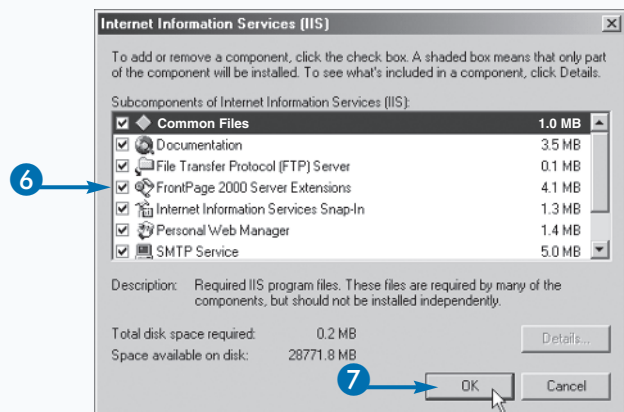
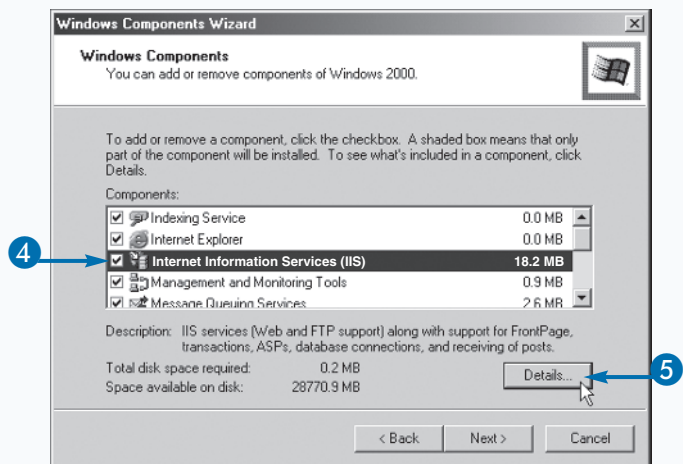
Programs menu, click Add/Remove Windows Components. Find the option for Internet Information Services (IIS), select it, and then click Details.

In the Internet Information Services (IIS) window that appears, check the boxes for SMTP Service and World Wide Web Service, and then click OK. In the Windows Component selection, make sure that the Internet Information Services (IIS) box is checked and then click OK. The components you selected will be installed.

After IIS is installed, it is a good idea to restart your computer. The Internet Information Services automatically starts each time you restart your computer. When IIS is installed and running, you can install PHP.

Install Internet Information Services on Windows

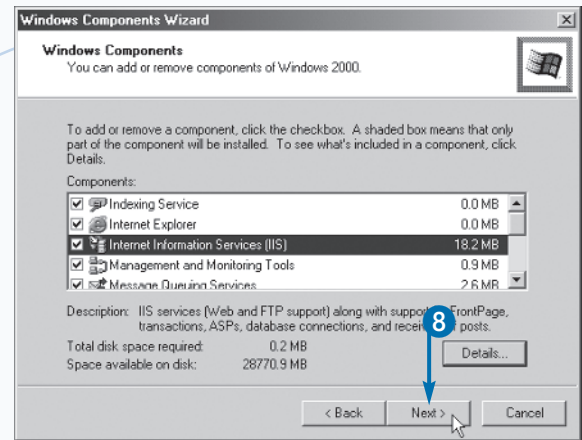
- 1 Insert the Windows CD-ROM.
- 2 In the Control Panel, select Add/Remove Programs.
- 3 Click Add/Remove Windows Components to display the Windows Components Wizard.
- 4 Click Internet Information Services (IIS) (☐ changes to ☒).
- 5 Click Details to display the IIS window.
- 6 Click beside each IIS component you want to install (☐ changes to ☒).
- 7 Click OK to confirm your configuration.



- 8 In the Windows Components Wizard, click Next to install the IIS package you have configured.

- 9 When the completion dialog box appears, click Finish.

The Internet Information Services is installed.



Extra

Before attempting to install IIS, you should log on to the computer with an Administrator account, because a standard user account is restricted from making machine-wide changes such as installing a Web server. This restriction is in large part in order to enforce security on the machine, but also helps ensure that applications are not accidentally installed or uninstalled. Logging on with an Administrator account ensures that you have proper permissions for executing installers. Take care when acting as an Administrator, because damage can be done to settings or questionable software can be installed.

After IIS is installed, the Internet Information Services window provides tools that allow you to administer your IIS Web sites. You can select the Administrative Tools option in the Control Panel to reveal the Internet Information Services window.

Add a Virtual Directory to Internet Information Services

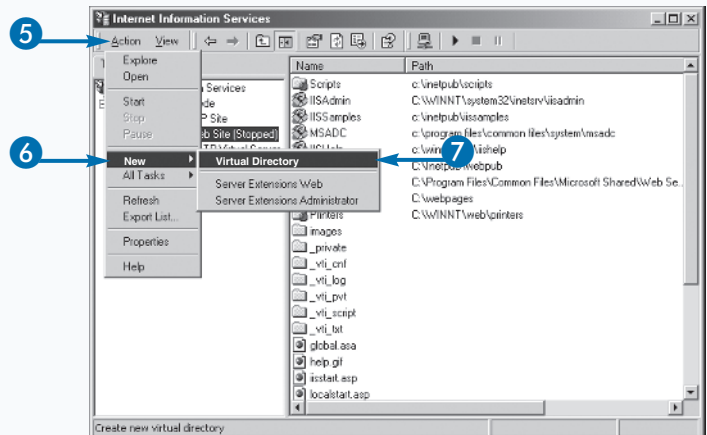
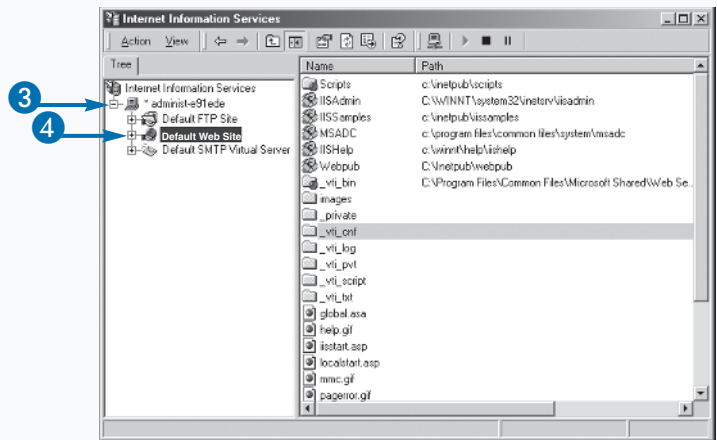
The default installation of Internet Information Services (IIS) creates a number of directories that can be used to serve Web pages, including PHP scripts. These directories are stored in the `C:\inetpub\wwwroot` folder. Only pages stored in this directory, or a subdirectory, can be served by IIS. This is a security feature that allows a computer to act as a Web server without compromising all files on the server. You can create a virtual directory to serve pages that are not contained within the document root as if they are. You can use the Virtual Directory Creation Wizard to create and properly configure a virtual directory from which PHP pages can be served. You must give the virtual directory an alias, which is a name by which users will access pages in the directory.

From the Start menu, navigate to the Programs directory, select Administrative Tools, and then click Internet Services Manager. Expand the server name. Right-click the Default Web Site entry, point to New, and click Virtual Directory. Type the name of the alias you want to use in the Virtual Directory Creation Wizard. Click Browse to locate the directory you want to use as the directory for your alias. On the last screen, click Finish to complete the wizard.

Virtual directories allow IIS to serve files from directories in locations other than the main Web root. For example, a folder that stores XML data for your site might be used as the location for a virtual directory with the alias 'xml.' This allows users to access the XML data using the path `/xml/` relative to the Web root.

Add a Virtual Directory to Internet Information Services

- 1 Use the Start menu to access the Control Panel, and then choose Administrative Tools.
- 2 Double-click Internet Services Manager to display the IIS window.
- 3 Click the plus sign (+) to display a list of Web sites currently running on the Web server.
- 4 Click the Web site to which you want to add a virtual directory.
- 5 Click Action.
- 6 Click New.
- 7 Click Virtual Directory.



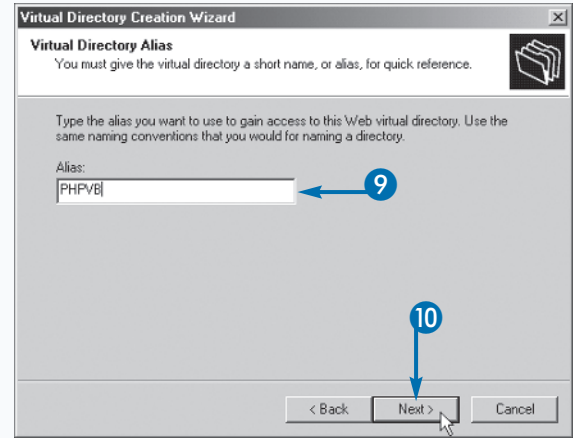
The Virtual Directory Creation Wizard appears.

- 8 Click Next to continue.




The Virtual Directory Alias window appears.

- 9 Type an alias for the new virtual directory.
- 10 Click Next to continue.



Extra

You can also use Windows Explorer to add a virtual directory to the Internet Information Services. The first step is to select the directory you want to use as the target of your virtual directory. Click the File menu and choose Properties. In the Properties dialog box, click the Web Sharing tab and select the "Share this folder" option. In the Edit Alias dialog box that appears next, specify the alias you want to use for your virtual directory. Select the access and application permissions to use for the virtual directory and click OK. Click OK in the Properties dialog box, and the virtual directory is added to IIS.

To remove a virtual directory from IIS, open the Internet Information Services window and select the virtual directory you want to remove. Click the Delete button () and choose Yes in the confirmation dialog box that appears. Removing a virtual directory immediately prevents users from accessing the content in the target directory.

continued →

Add a Virtual Directory to Internet Information Services (continued)

You must specify the location of the directory you want to add as a virtual directory. The access permissions for the directory must also be specified. The access permissions determine how the pages in the directory — whether plain HTML or dynamic pages like PHP — can be utilized.

The Read permission allows users viewing your site to access the Web pages with their Web browsers. This access permission must be set for a directory from which you want to serve Web content. Without the Read permission, no content in the directory will be accessible.

The Run scripts permission allows scripts to run in the directory. Execution of PHP and other dynamic content is dependent upon this access permission being set. By default, the Run scripts permission is turned on.

The Execute permission allows scripts and applications to be executed. This option should be set for any special executable applications installed on your Web server.

Add a Virtual Directory to Internet Information Services (continued)

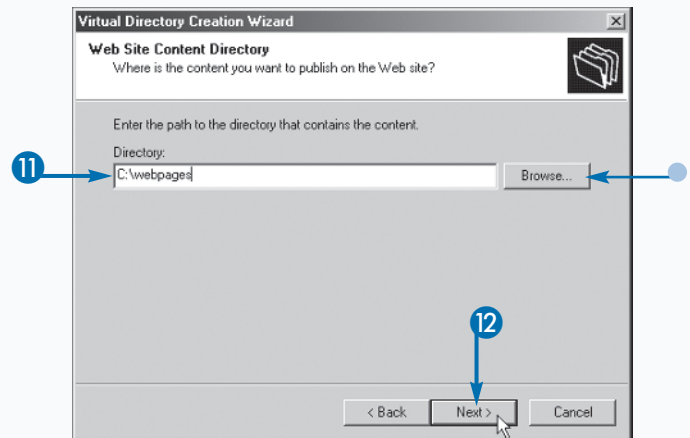
The Write permission allows files to be dynamically created in the directory targeted by your virtual directory. If you have scripts that need to create files dynamically, such as image-generation scripts, this permission must be turned on.

The Browse permission allows users to view a listing of files in the directory targeted by your virtual directory. When turned on, a user has access to all files and subdirectories in the directory. When a user specifies the path to the virtual directory but not a specific file, a listing of all contents of the directory is presented.

Because virtual directories function to provide access to files in various locations on your Web server, you should be careful to remove any sensitive content from the target directory before setting permissions allowing for the virtual directory to be used. Security through obscurity is a poor security policy and hoping that users will not find content in directories is not sufficient.

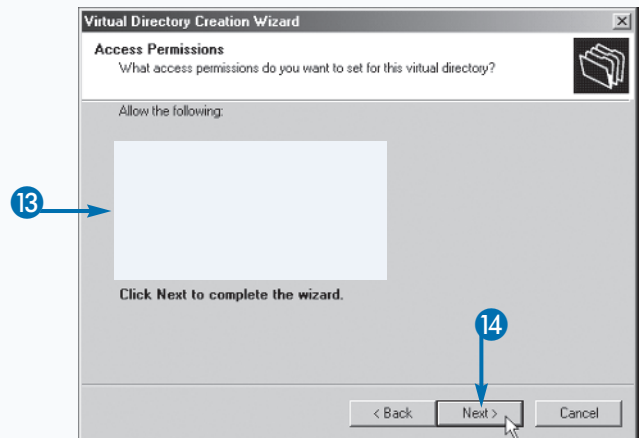
The Web Site Content Directory window appears.

- 11 Type the path to the directory you want to add as a virtual directory.
 - Alternately, you can click Browse to locate your target folder.
- 12 Click Next to continue.



The Access Permissions window appears.

- 13 Select the access permissions you want to allow (☐ changes to ☒).
- 14 Click Next to continue.

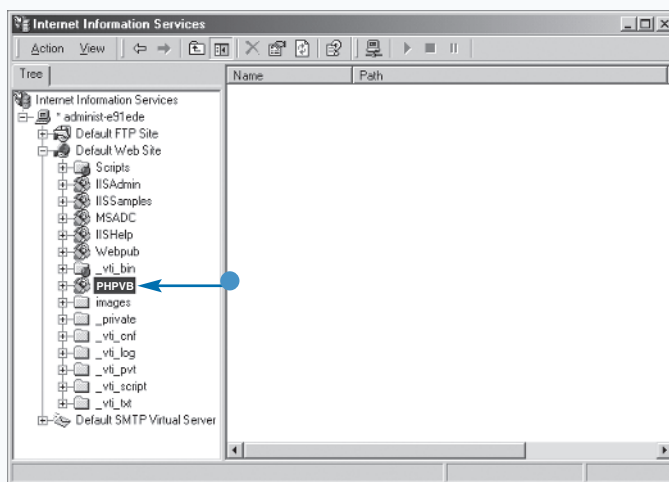


A message appears alerting you that you have successfully created a Virtual Directory.

- 15 Click Finish to close the installer.



- The virtual directory, with the alias you specified in Step 9, is added to IIS and appears in the directory listing.



Extra

You can change the execute permissions for a virtual directory in IIS after the directory has been created. To change the execute permissions, repeat Steps 1 and 2 from this task to open the Internet Information Services window. Select the virtual directory you want to change. Click the Properties button (🔗) to open the Properties dialog box. Select the option in the Execute Permissions area that you want to use. The Scripts and Executables option allows files in the directory to be accessed and executed. The Scripts only option allows only scripts to be run. The None option allows access to static files, such as HTML files and images, but no access to scripts or executables.

Set File and Folder Permissions on Windows

You can set file and folder permissions to allow or restrict access to files by certain users or groups of users. The Internet Information Services application serves files and executes scripts according to these permissions. The types of permissions that can be set for files and folders depend on the operating system and file system installed on the Web server. For example, setting permissions on a Linux server is much different than performing the same operations on a Windows XP server. When IIS is installed on your Windows server, a user account is created called IUSR_(computename), where (computename) is the name of the computer on which the user account is installed. For example, a computer called WEBDEV would have an account named IUSR_WEBDEV. This account is part of the Everyone group. If you deny the Everyone group access to resources such as files and

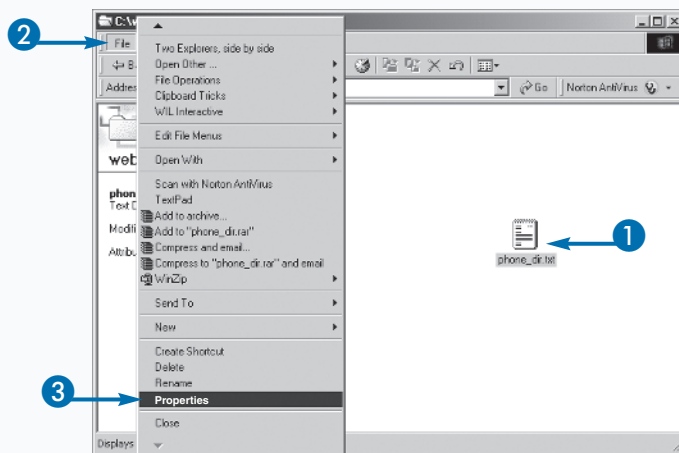
folders, you can use the IUSR account to allow users you specify to access information in the restricted resource.

You can set permissions for single users or groups to specify how either can access files and folders. Full Control allows users to modify, add, move and delete files. Additionally, these users can change permissions. The Modify option allows users to modify, add, move, and delete files. Read & Execute allows users to run applications and scripts, such as PHP scripts. Read allows users to display files, and Write allows users to modify files.

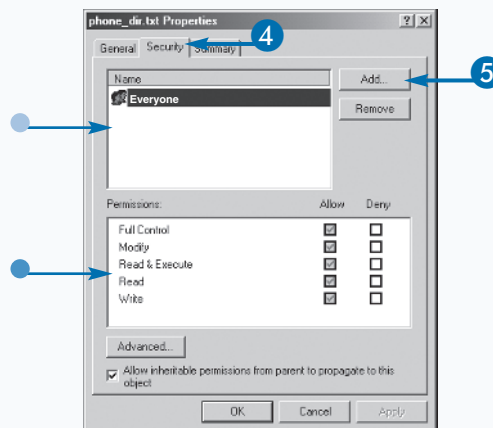
Windows file and folder permissions work with the access permissions set in Internet Information Services. Access permissions set for a folder in IIS apply to all users who access the folder. Windows folder permissions apply only to users and groups you specify. When Windows file and folder permissions and IIS access permissions differ for the same resource, the Web server uses the most restrictive permissions.

Set File and Folder Permissions on Windows

- 1 Click the file or folder for which you want to change permissions.
- 2 Click File.
- 3 Click Properties to display the Properties dialog box.

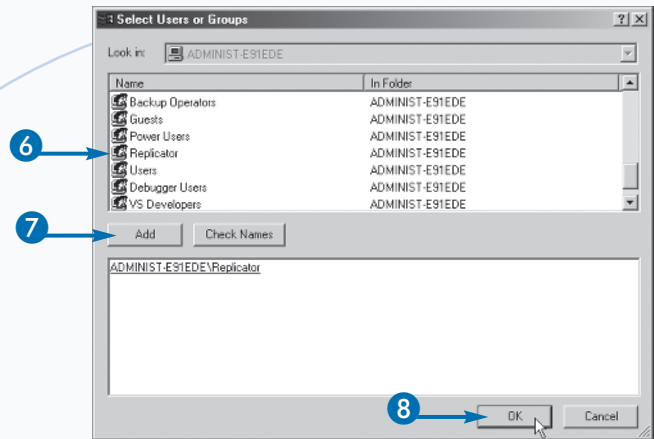


- 4 Click the Security tab.
- The users who have permission to the file or folder you chose in Step 1 appear here.
- Their current level of permission appears here.
- 5 Click Add.



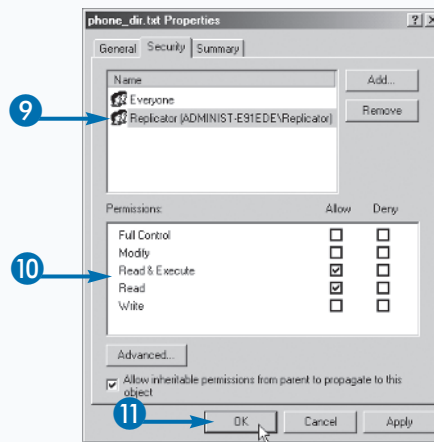
The Select Users or Groups dialog box appears.

- 6 Click a user or group to which you want to grant access to the file or folder.
- 7 Click Add.
- 8 Click OK.



- 9 In the Properties dialog box, click a user or group to view their permissions.
- 10 Click an option to allow or deny a given permission.
- 11 Click OK to confirm your changes.

Windows sets the permissions.



Extra

Take great care when setting file and folder permissions on a Web server. This is true on both Internet and intranet sites. To gain a full understanding of file and folder permissions and the implications of any changes you are considering, you should refer to the operating system and Web server documentation. Incorrectly setting permissions on a Web server can leave the server open to serious security problems as well as data fidelity problems such as corrupted or deleted data.

When moving PHP scripts and associated files between Web servers, you must be careful to ensure that the permissions on the final server match those on the initial server. The permissions are not transferred, and are often reset when files are uploaded through FTP or similar protocols.

If you want to revoke access to a file for a given user or group you can remove the user or group. Repeat Steps 1 to 4 of this task to display the Security tab of the Properties dialog box. Click the user or group that you no longer want to be able to access the file or folder and then click Remove.

Install PHP on Windows

In order to create and process PHP scripts on your Windows computer, you must install PHP. You can use the recent Windows versions of PHP on all modern Windows operating systems.

You can download a recent version of PHP for Windows from the PHP Web site at www.php.net.

The PHP installer is a standard Windows installation program that guides you through the installation process, including analyzing the current status of your system to determine the presence of required applications.

The computer on which you install PHP does not need to be connected to the Internet. You can use a standalone computer running Web server software such as the Apache HTTPD server or Internet Information Services to develop

your PHP pages. In fact, local development is often required as errors and vulnerabilities are never exposed to the outside world. Because of the portable, cross-platform nature of the majority of the PHP language, you will most likely be able to simply copy the files to your Web server for deployment.

Like most Windows installation programs, the PHP installer provides you with a copy of the license under which you can use the software. After accepting the license, you can choose the type of installation you want to perform. The Standard option is recommended for most users, but the Advanced option allows experienced users to specify options when performing the installation.

Install PHP on Windows

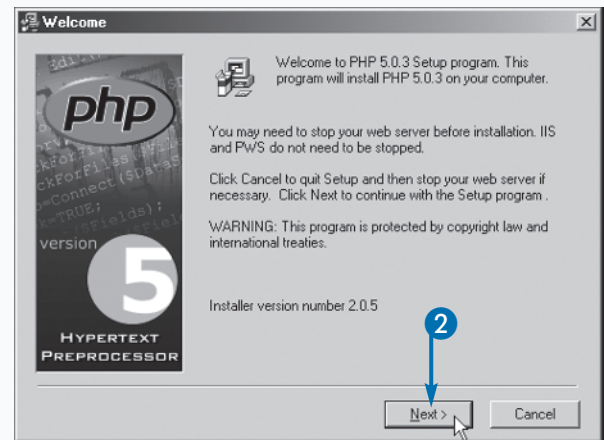
- 1 Double-click the icon for the PHP installation program to start installing the program.

A welcome dialog box appears.

- 2 Click Next to continue.
- 3 In the next window, read the license agreement and click I Agree to continue the installation.

The Installation Type window appears.

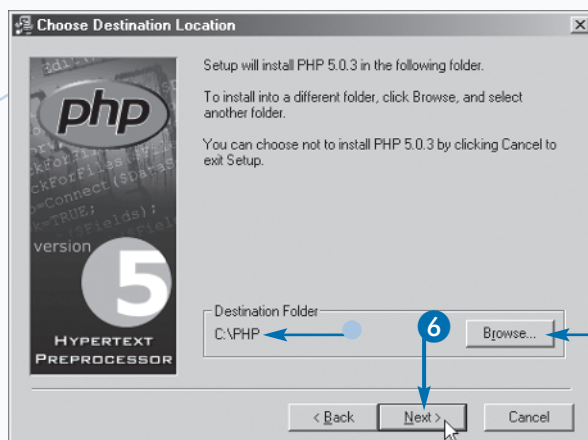
- 4 Select the type of installation you want to perform (☐ changes to ☒).
- 5 Click Next to continue.



The Choose Destination Location window appears.

- It is recommended that you use the default folder.
- Alternately, you can click Browse to locate another folder.

6 Click Next to continue.

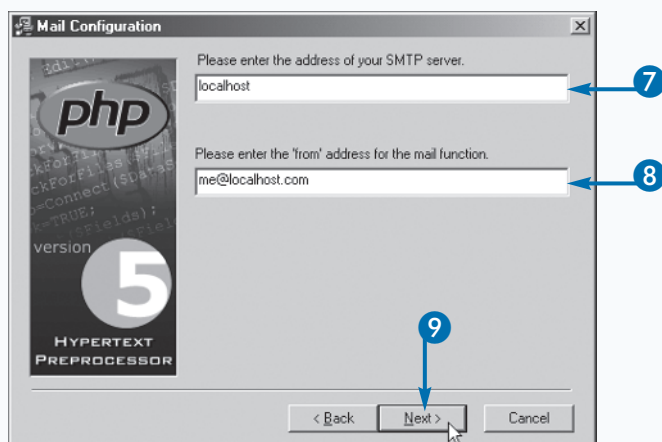


7 Type the name of the outgoing mail (SMTP) server.

Note: You can use the default, localhost, if no other is available.

8 Type the e-mail address you want displayed in outgoing mail messages.

9 Click Next to continue.



Extra

The most recent release of PHP is always available from the PHP Web site at www.php.net. Older versions of PHP are also available from the official site. Installing versions of PHP other than the newest release is helpful because many Web hosts do not immediately upgrade to new releases for fear of causing interruptions or introducing bugs into the sites they host for their clients. When working on a project that runs on an older version of PHP, you should develop the site using the same version.

While the differences in minor versions, such as PHP 4.3.2 and PHP 4.3.3, can be mild, it is best practice to develop your applications on a local development machine that matches the production server or servers in all respects. For PHP development, matching hardware is not important, though the local and production server should be roughly equivalent so that effects of load and heavy memory use can be taken into consideration. More important is that all software and permissions are matched. Ideally, the process of making a development version of your application available on the Internet should consist of nothing more than uploading files to your Web server.

continued →

Install PHP on Windows (continued)

The installer will prompt you for the location in which you want to install PHP. The default installation is `C:\PHP`. You can specify a different location, but the default option is recommended.

When installing PHP you can specify the name of the mail server the computer will use to send e-mail messages. This setting is important if you want to include the option of sending e-mail from your PHP scripts. You can specify any server running the Simple Mail Transfer Protocol (SMTP). Generally, the default value of "localhost" will suffice.

You should specify the e-mail address from which you want to send outgoing e-mail addresses when no additional mail

headers are used. Most often, this address is an alias for the system administrator or for the Web developer in charge of the server.

The installer application can automatically set certain configuration options for a given Web server. When prompted to choose the Web server with which you intend to use PHP, choose the version of IIS or Apache you have installed and configured.

After specifying all configuration options, you can finalize the installation by clicking Finish. When the installation is complete, you can configure PHP using the PHP configuration file as well as the Apache configuration file.

Install PHP on Windows (continued)

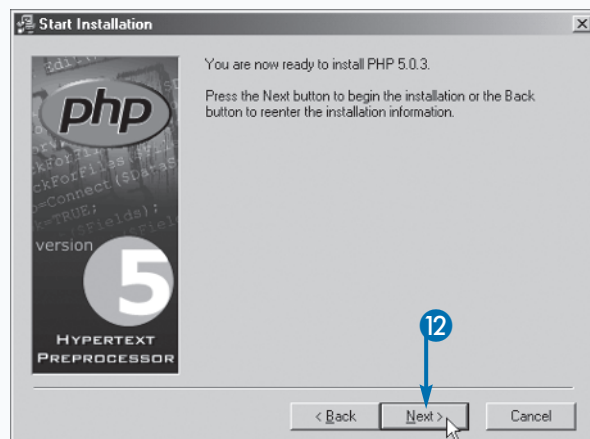
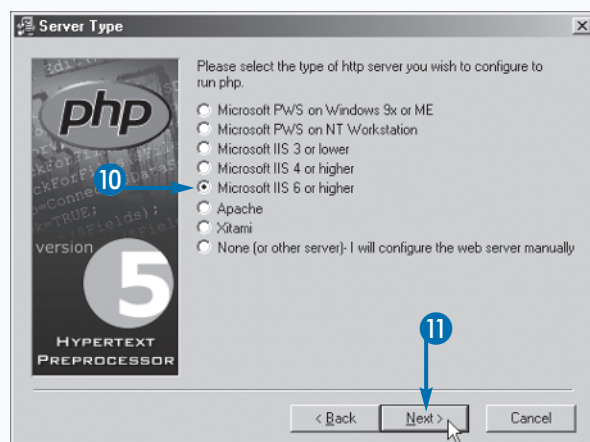
- 10 Select the type of Web server you want to be configured to use with PHP.

If your Web server does not appear in the list, you can select None to configure the server manually.

- 11 Click Next to continue.

A dialog box appears indicating that you are ready to install PHP.

- 12 Click Next to install PHP on your computer.



- 13 A dialog box appears telling you to configure Apache's httpd.conf file. Click OK to continue.

Note: Follow Steps 15 to 18 to configure Apache's httpd.conf file.

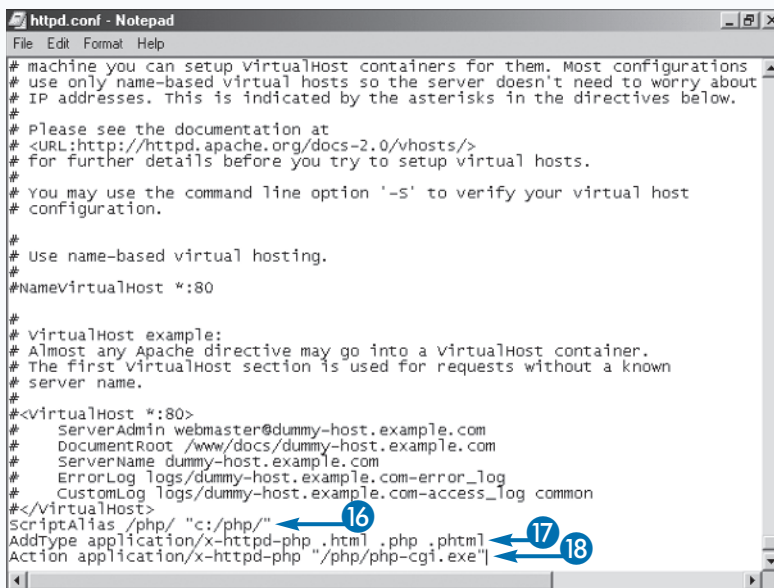
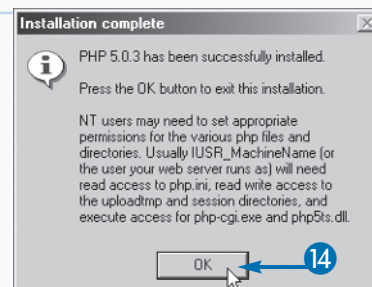
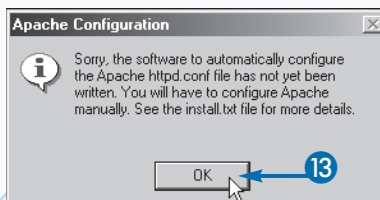
- 14 Click OK to close the next dialog box.

PHP is installed.

CONFIGURE APACHE'S HTTPD.CONF FILE

- 15 Open your httpd.conf file in a text editor. The httpd.conf is located in the conf directory of your Apache install directory.
- 16 Add a ScriptAlias directive mapping your PHP directory to the PHP interpreter at c:/php/.
- 17 Use an AddType directive to map the type application/x-httpd-php to the file extensions you want PHP to parse.
- 18 Add an Action directive mapping the application/x-httpd-php application type to the PHP executable.

Save the file and restart your computer to see the changes take effect.



Extra

PHP should be installed only on the hard drive of the Web server machine. You should not install PHP on a network drive or on removable media such as an external FireWire drive. While it is technically possible to perform such an installation, the performance of the Web server will often be less than ideal and may become unavailable.

When specifying the e-mail address you want to display on outgoing e-mail messages, you may want to use an e-mail address set up for only this purpose. This is especially true on servers that send a lot of mail as bounced messages and can quickly fill the space allocated to a typical e-mail address.

If the computer stops responding before the installation of PHP is complete, there may be a problem with the system hardware, network connection, or the PHP installer itself. In such cases, you should close all open programs and restart the computer before attempting to reinstall the program.

Install Apache on OS X

Mac OS X comes configured with proven versions of Apache and PHP installed by default. Both applications are configured with safety and efficiency in mind, and you will most likely want to update the version or enable additional options for each before beginning development.

Because Mac OS X is a Unix-based operating system, you can install software much the same as on other Unix and Linux systems, using the a shell session and the `configure`, `make`, `make install` sequence.

The first step in installing Apache is to download the most recent production version of Apache from the Apache Group Web site. Currently, the most recent major version is Apache 2. Unzip the archive containing the Apache source to a folder on your computer.

You will need to have administrative permissions to install software on the machine. If you are currently logged into the machine with a restricted user account, you can log off and back on as an administrator using the commands under the Apple menu.

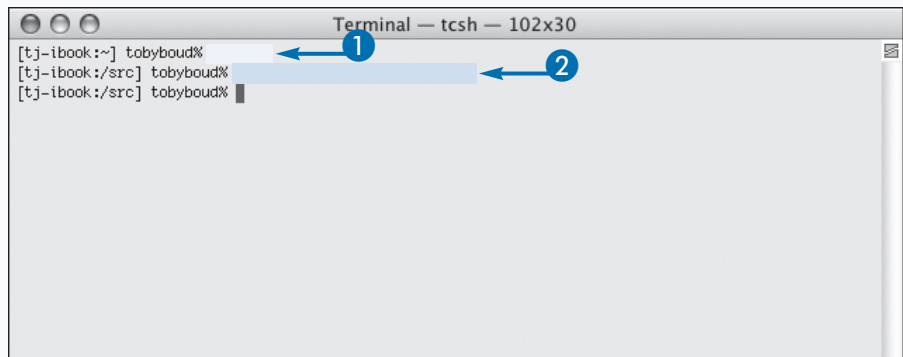
Open the Terminal application in your Applications/Utilities folder. Use the `cd` command to change to the directory in which you unzipped the Apache source.

You can configure Apache using the `configure` script in the source folder. To configure Apache 2 for PHP 5 support, you can use the `--enable-mods-shared=ALL` `configure` directive. After the `configure` command runs, you can type `make` and press Enter to make the installable application.

If `make` executes successfully, type `make install` to install Apache 2 to your system.

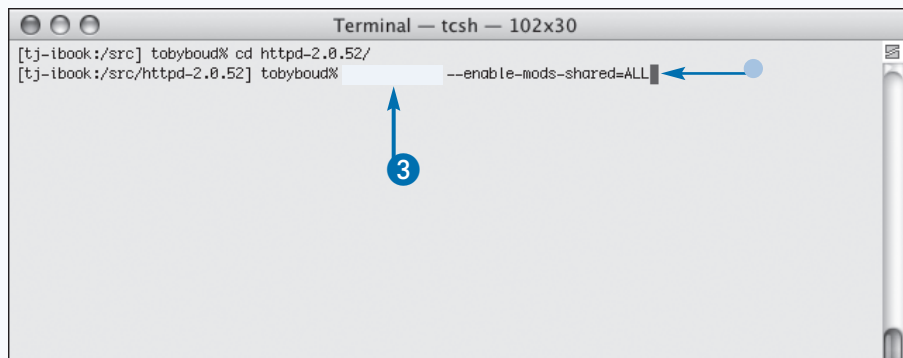
Install Apache on OS X

- 1 Change directories to the location of your install using the `cd` command.
- 2 Unarchive the Apache source archive using the `tar -xzf filename` command.
- 3 Set installation options using the `configure` script in the Apache source directory.
 - Pass any options, or flags, to the `configure` script according to your anticipated PHP use and the server on which it is being installed.



```
Terminal — tcsh — 102x30
[tj-ibook:~] tobyboud%
[tj-ibook:/src] tobyboud%
[tj-ibook:/src] tobyboud%
```

Diagram illustrating the first two steps of the installation process in a terminal window. Step 1 (arrow 1) shows the user at the home directory prompt. Step 2 (arrow 2) shows the user navigating to the `/src` directory.



```
Terminal — tcsh — 102x30
[tj-ibook:/src] tobyboud% cd httpd-2.0.52/
[tj-ibook:/src/httpd-2.0.52] tobyboud% --enable-mods-shared=ALL
```

Diagram illustrating the third step of the installation process in a terminal window. Step 3 (arrow 3) shows the user running the `configure` script with the `--enable-mods-shared=ALL` option.

- 4 Type `make` to make the installer.
- 5 After the installer is made, type `make install` to install Apache on your server.

```

Terminal — tcsh — 102x30
[tj-ibook:/src/httpd-2.0.52] tobyboud% --silent
[tj-ibook:/src/httpd-2.0.52] tobyboud% --silent
[tj-ibook:/src/httpd-2.0.52] tobyboud%

```

- 6 Change directories to the bin directory of the install location. Type `apachectl start` to start Apache.

```

Terminal — tcsh — 102x30
[tj-ibook:/src/httpd-2.0.52] tobyboud%
[tj-ibook:/usr/local/apache/bin] tobyboud%

```

Extra

You can choose to install either Apache 1.3 or Apache 2 on your OS X machine. Either will work with PHP 5 and Apache 1.3 comes preinstalled. The choice between Apache 1.3 and Apache 2 depends primarily on which version of Apache is running on your production server as well as any special constraints or parameters in your project specifications.

Generally, it is a good idea to install both versions of Apache on your development machine and switch versions according to your current development needs. You can switch versions by stopping the current version and starting the alternate version. To stop Apache, the `apachectl` command is used. Because each installed version has its own `apachectl` command, you should be careful to specify the full path to the `apachectl` version you want to use. For example, you might have Apache 1.3 installed in `/usr/local/apache`, in which case you could start the server using `/usr/local/apache/bin/apachectl start`.

For an Apache 2 install located at `/usr/local/apache2` you can use the `/usr/local/apache2/bin/apachectl start` command.

Install PHP on OS X

Mac OS X ships with PHP 4 installed alongside the Apache Web server. You can install PHP 5 using the typical Unix `configure`, `make`, `make install` compile process.

The first step in installing PHP is to download the most recent production version of PHP from the PHP Web site. At the time of this writing, the most recent version of PHP is version 5.0.3. Unzip the archive containing the PHP source to a folder on your computer.

You must have administrative permissions to install software on the machine. If you are currently logged on with a restricted user account, log out and back on as an administrator using the commands under the Apple menu.

Open the Terminal application in your Applications/Utilities folder. Use the `cd` command to change to the directory in which you unzipped the PHP source.

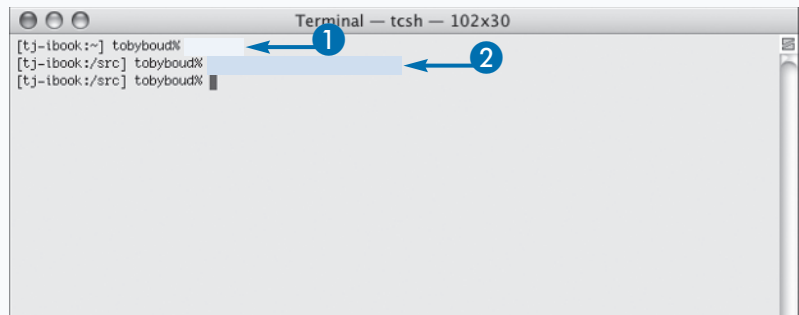
You can configure PHP with any number of configuration options. To configure PHP before compiling, you can use the `configure` script in the source folder. After the `configure` command runs, you can type `make` to make the installable application.

The `make` command compiles PHP for use on the server. When using the `make` command, you will see a number of messages scroll past in the Terminal window. If any errors are reported, take note and check your configuration options to ensure that libraries on which your configuration depends are installed properly.

If `make` executes successfully, type `make install` to install PHP on your system. You will need to restart Apache before using PHP.

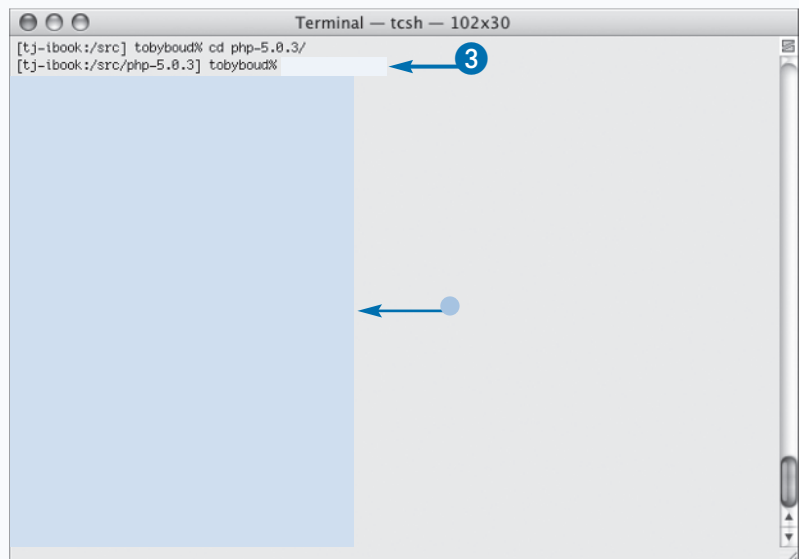
Install PHP on OS X

- 1 Change directories to the location of your install using the `cd` command.
- 2 Unarchive the PHP download archive using the `tar -xzf filename` command.
- 3 Set installation options using the `configure` script in the PHP source directory.
 - Pass any options, or flags, to the `configure` script according to your anticipated PHP use and the server on which it is being installed.



```
Terminal — tcsh — 102x30
[tj-ibook:~] tobyboud% 1
[tj-ibook:/src] tobyboud% 2
[tj-ibook:/src] tobyboud%
```

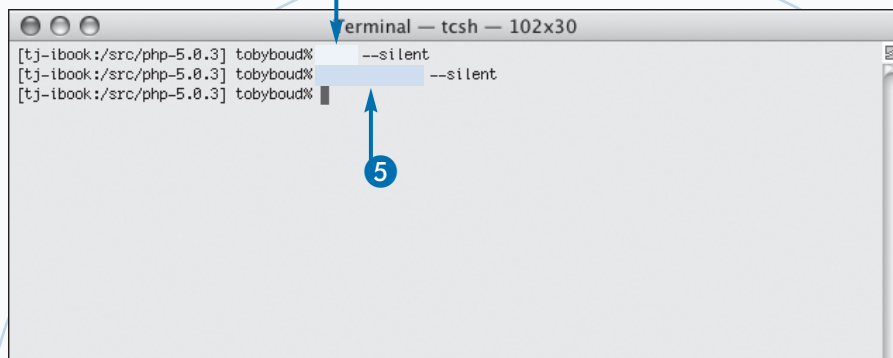
The image shows a Terminal window with a title bar 'Terminal — tcsh — 102x30'. The prompt is '[tj-ibook:~] tobyboud%'. A blue arrow labeled '1' points to the prompt. The next line shows the prompt has changed to '[tj-ibook:/src] tobyboud%'. A blue arrow labeled '2' points to this new prompt. The third line shows the prompt remains '[tj-ibook:/src] tobyboud%'.



```
Terminal — tcsh — 102x30
[tj-ibook:/src] tobyboud% cd php-5.0.3/
[tj-ibook:/src/php-5.0.3] tobyboud% 3
```

The image shows a Terminal window with a title bar 'Terminal — tcsh — 102x30'. The prompt is '[tj-ibook:/src] tobyboud%'. A blue arrow labeled '3' points to the prompt. The next line shows the prompt has changed to '[tj-ibook:/src/php-5.0.3] tobyboud%'. A blue arrow labeled '3' points to this new prompt. The third line shows the prompt remains '[tj-ibook:/src/php-5.0.3] tobyboud%'.

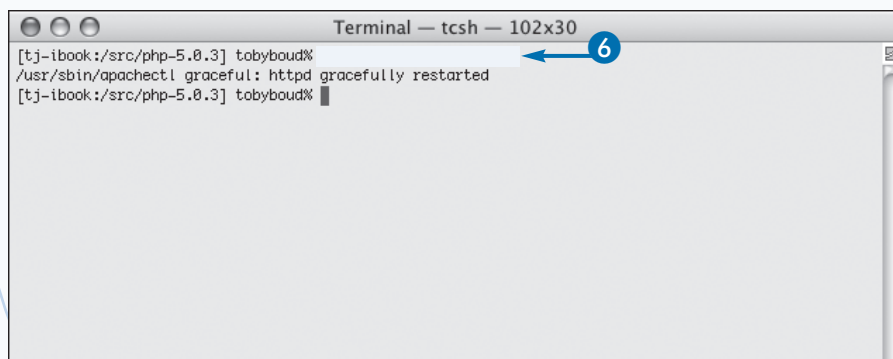
- 4 Type `make` to make the installer.
- 5 After the installer is made, type `make install` to install PHP on your server.



A terminal window titled "terminal — tcsh — 102x30" showing the execution of two commands. The first command is `make`, which is highlighted with a blue box and a blue arrow labeled "4". The second command is `make install`, which is also highlighted with a blue box and a blue arrow labeled "5". The output of both commands is `--silent`.

```
[tj-ibook:/src/php-5.0.3] tobyboud% make
[tj-ibook:/src/php-5.0.3] tobyboud% make install
[tj-ibook:/src/php-5.0.3] tobyboud%
```

- 6 Restart your Web server to load the PHP processor.



A terminal window titled "Terminal — tcsh — 102x30" showing the execution of the `apachectl graceful` command. The command is highlighted with a blue box and a blue arrow labeled "6". The output of the command is `httpd gracefully restarted`.

```
[tj-ibook:/src/php-5.0.3] tobyboud% apachectl graceful
/usr/sbin/apachectl graceful: httpd gracefully restarted
[tj-ibook:/src/php-5.0.3] tobyboud%
```

Extra

OS X comes with a version of PHP pre-installed. At the time of this writing, the version of PHP included on the OS X client install is PHP 4. You can install both versions of PHP as long as the location of the install is different. For example, you can install PHP 5 into the `/usr/local/php5` directory. Note that you can have both PHP 4 and PHP 5 installed on your machine, but only one may be run per instance of Apache at a given time. In fact, you can have as many versions of PHP installed on your computer as you want as long as there are no library conflicts. Often, a development machine will have at least two or three minor versions of the latest major version release to accommodate any number of projects that may be in development. You can specify which version of PHP is loaded by the Apache Web server using the Apache configuration file, usually found at `/usr/local/apache/conf/httpd.conf`.

Set File and Folder Permissions on OS X

You can use the Terminal to set file and folder permissions on OS X. You can also use the Get Info option in the Finder to view and modify permission information about files and folders.

Because OS X is a Unix-based operating system, you can set folder and file permissions using the `chmod`, `chown`, and `chgrp` commands. In order to change the permissions of a file or folder, you must have user or group ownership permissions. If your account does not have permissions to modify a file, you can use the `su` command to switch to a user account with adequate permissions. For example, you might switch to a user named 'admin' using the command `su admin`. When prompted for a password, you must type the password of the account to which you are switching. You can perform a quick action as another user using the `sudo` command. For example, to change ownership of

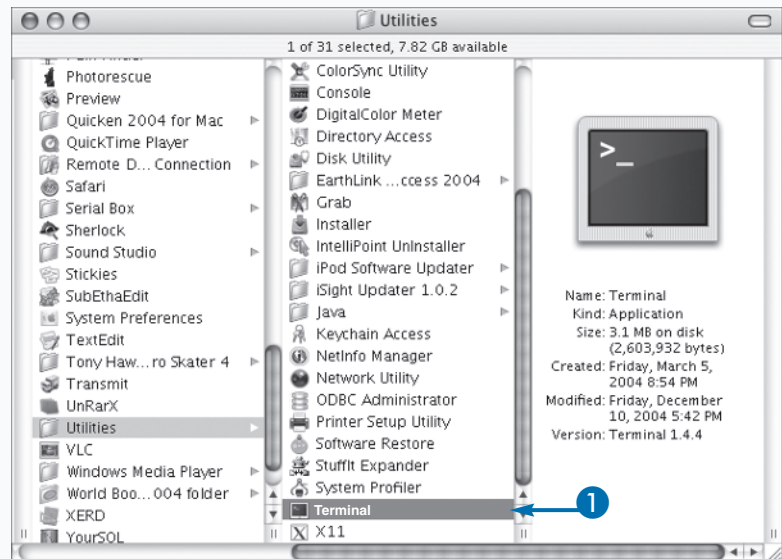
the file 'test.php' to the user 'toby' you might use a command such as `sudo chown toby test.php`.

Another method of altering file and folder permissions is to use the OS X Finder. You can use the Finder to change permissions by opening the Get Info palette for a given file or folder. Right-click the target file or folder in the Finder and choose Get Info from the menu. Alternately, you can press the Apple key and the I key in combination to open the Get Info palette.

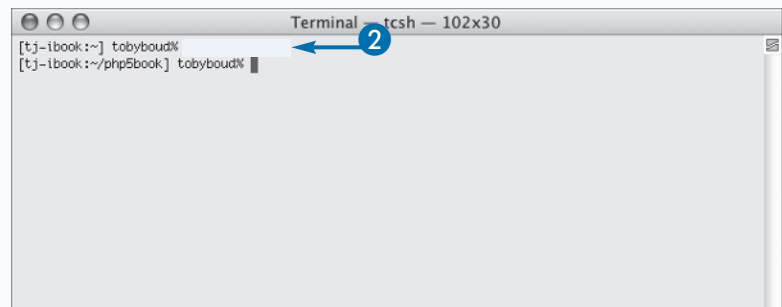
When the Get Info palette is open, you see a section called Ownership & Permissions. There are select boxes in this section. If you have permission to modify the ownership and permissions properties of the currently selected file, you can use the select boxes to change these properties. Otherwise, the menu is dimmed and unusable.

Set File and Folder Permissions on OS X

- 1 Open the Terminal application from the Applications/Utilities folder.



- 2 Change directories to the directory in which your file exists using the `cd` command.



- 3 Type `ls -la` to see the current permission information for your file.

```
Terminal — tcsh — 102x30
[tj-ibook:~] tobyboud% cd ~/php5book/
[tj-ibook:~/php5book] tobyboud% ls -la
total 0
drwxr-xr-x  3 tobyboud  tobyboud   102 27 Jan 21:52 .
drwxr-xr-x 45 tobyboud  tobyboud  1530 27 Jan 21:52 ..
-rw-r--r--  1 tobyboud  tobyboud    0 27 Jan 21:52 testfile.txt
[tj-ibook:~/php5book] tobyboud%
```

- 4 Change the permissions of the file using the `chmod` command.

```
Terminal — tcsh — 102x30
[tj-ibook:~/php5book] tobyboud% ls -la
[tj-ibook:~/php5book] tobyboud% ls -la
total 0
drwxr-xr-x  3 tobyboud  tobyboud   102 27 Jan 21:52 .
drwxr-xr-x 45 tobyboud  tobyboud  1530 27 Jan 21:52 ..
-rwxrwxrwx  1 tobyboud  tobyboud    0 27 Jan 21:52 testfile.txt
[tj-ibook:~/php5book] tobyboud%
```

Extra

You can recursively set ownership and permission privileges for a directory by specifying a special flag for the `chmod` command. The `-R` flag tells the `chmod` command to apply the changes not only to the directory in question but to move through the directory setting the same permissions on each file therein. Because this option is recursive, the permissions of items in subfolders will be set as well as those in the main folder.

The Details section of the Get Info palette gives detailed permissions information for a given file. You can set your own permissions level, the owner of the file, access level of the owner, the group ownership and accompanying group access, and the level of permissions applicable to other users.

Configure PHP

PHP configuration is managed by modifying the contents of a configuration file, usually named `php.ini`. The configuration file is a plain text file that controls how PHP processes scripts and is read by the Web server when PHP is started.

You can edit the `php.ini` file using a standard text editor, such as Notepad on a Windows system, and `vi` or `emacs` on a Unix or Linux system. The Mac OS X application `TextEdit` is a useful text editor when developing on your Apple.

The format of the `php.ini` file is a standard configuration style in which each new line represents a directive. Comments are delimited using a semicolon (;) and entire lines can be commented out by placing a semicolon at the beginning of each line.

Directives are set using the name of the directive, followed by an equals sign (=) with the directive value. Most often, possible values are listed in comments in the `php.ini` file.

To set a directive, simply change the value to a recognized value other than the current setting and save the file. The next request to PHP is processed according to the changed configuration.

The most common changes made to a new installation of PHP is to set the appropriate level of error reporting for the server, based on its use and availability. For local development servers, you generally want very verbose error reporting. You can change the error reporting values using the `error_reporting` directive. For example, you can turn on all error reporting using the lines:

```
error_reporting = E_ALL
```

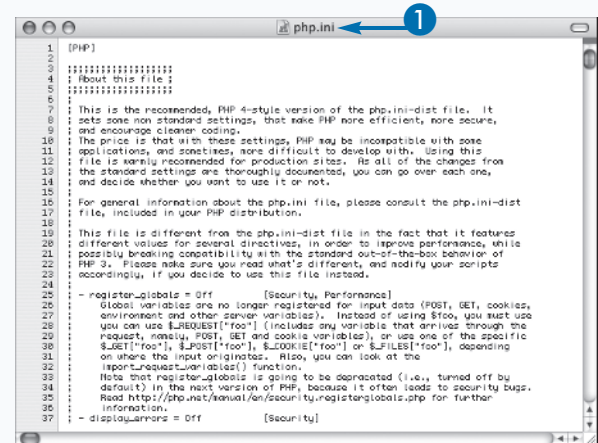
```
display_errors = On
```

Or, for a production server, where you want no errors displayed, you can set the error reporting level with the line:

```
display_errors = Off
```

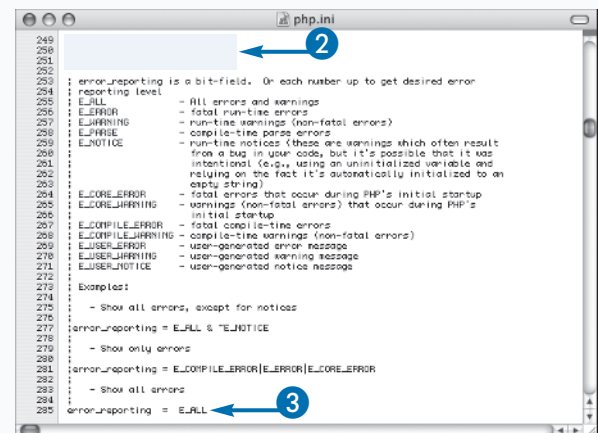
Configure PHP

- 1 Locate and open your `php.ini` file in a text editor.



- 2 Find the Error handling and logging section.

- 3 Locate the `error_reporting` directive. Uncomment the line `error_reporting = E_ALL` if it is currently commented.



4 Locate the `display_errors` directive.

```

256 ; E_ERROR          - fatal run-time errors
257 ; E_WARNING        - run-time warnings (non-fatal errors)
258 ; E_PARSE          - compile-time parse errors
259 ; E_NOTICE         - run-time notices (these are warnings which often result
260 ;                  from a bug in your code, but it's possible that it was
261 ;                  intentional (e.g., using an uninitialized variable and
262 ;                  relying on the fact it's automatically initialized to an
263 ;                  empty string))
264 ; E_CORE_ERROR     - fatal errors that occur during PHP's initial startup
265 ; E_CORE_WARNING   - warnings (non-fatal errors) that occur during PHP's
266 ;                  initial startup
267 ; E_COMPILE_ERROR  - fatal compile-time errors
268 ; E_COMPILE_WARNING - compile-time warnings (non-fatal errors)
269 ; E_USER_ERROR     - user-generated error message
270 ; E_USER_WARNING   - user-generated warning message
271 ; E_USER_NOTICE    - user-generated notice message
272
273 ; Examples:
274 ;
275 ; - Show all errors, except for notices
276 ; error_reporting = E_ALL & ~E_NOTICE
277 ;
278 ; - Show only errors
279 ; error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
280 ;
281 ; - Show all errors
282 ; error_reporting = E_ALL
283 ;
284 ; Print out errors (as a part of the output). For production web sites,
285 ; you're strongly encouraged to turn this feature off, and use error logging
286 ; instead (see below). Keeping display_errors enabled on a production web site
287 ; may reveal security information and users, such as file paths on your Web
288 ; server, your database schema or other information.
289 ;
290 ; display_errors = 0
  
```

5 Change the value from `Off` to `On`.

6 Save and close the `php.ini` file.

```

256 ; E_ERROR          - fatal run-time errors
257 ; E_WARNING        - run-time warnings (non-fatal errors)
258 ; E_PARSE          - compile-time parse errors
259 ; E_NOTICE         - run-time notices (these are warnings which often result
260 ;                  from a bug in your code, but it's possible that it was
261 ;                  intentional (e.g., using an uninitialized variable and
262 ;                  relying on the fact it's automatically initialized to an
263 ;                  empty string))
264 ; E_CORE_ERROR     - fatal errors that occur during PHP's initial startup
265 ; E_CORE_WARNING   - warnings (non-fatal errors) that occur during PHP's
266 ;                  initial startup
267 ; E_COMPILE_ERROR  - fatal compile-time errors
268 ; E_COMPILE_WARNING - compile-time warnings (non-fatal errors)
269 ; E_USER_ERROR     - user-generated error message
270 ; E_USER_WARNING   - user-generated warning message
271 ; E_USER_NOTICE    - user-generated notice message
272
273 ; Examples:
274 ;
275 ; - Show all errors, except for notices
276 ; error_reporting = E_ALL & ~E_NOTICE
277 ;
278 ; - Show only errors
279 ; error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
280 ;
281 ; - Show all errors
282 ; error_reporting = E_ALL
283 ;
284 ; Print out errors (as a part of the output). For production web sites,
285 ; you're strongly encouraged to turn this feature off, and use error logging
286 ; instead (see below). Keeping display_errors enabled on a production web site
287 ; may reveal security information and users, such as file paths on your Web
288 ; server, your database schema or other information.
289 ;
290 ; display_errors = 1
  
```

Extra

On computers running a version of the Windows operating system, the `php.ini` file is typically located in the main Windows directory, such as `C:\WINNT` or `C:\WINDOWS`. On Unix systems, the location of the `php.ini` file can vary, but the file is often found in the `/usr/local/lib` directory. You can locate the file on a Unix system using the command-line `locate` tool. To locate all files whose path contains the string `php.ini` you can type `locate php.ini` and press Enter.

The `php.ini` configuration file can be used to manage which modules are enabled for your PHP installation. You can also use the configuration file to manage module-specific settings, such as the default host for MySQL connections, or session management options such as the location of session files and the name of the session identifier to be used in tracking user sessions.

You can read about all of the configuration options available in the `php.ini` file on the PHP Web site at www.php.net/manual/en/configuration.php.

Verify Installation

After the Web server software and a version of PHP are installed and the Web server is configured to process PHP files, you can verify the status of the installation and check the current configuration settings using a PHP script.

The `phpinfo()` function is a globally accessible function that is a useful utility when learning about the configuration of a given PHP installation. The `phpinfo()` function accepts no arguments and returns no result, acting instead as a utility function that gathers information about the current configuration of your Web server and displays it as an easy-to-understand HTML page.

You can use the `phpinfo()` function by creating a new PHP script in your Web server document root. You can open a

block of PHP code by typing the `<?php` opening PHP processing instruction delimiter. On a new line, call the `phpinfo()` function by typing `phpinfo();`.

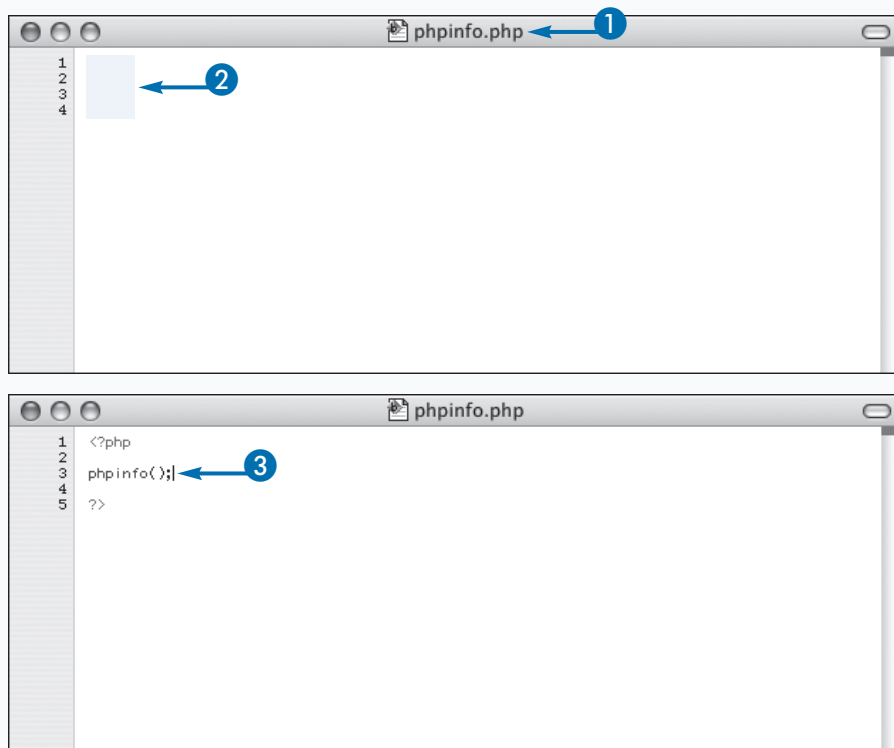
You can close the PHP processing block by typing the closing processing instruction delimiter, `?>`.

After you save the PHP script to your Web server, you can connect to the server using a standard Web browser. In the location bar, type the path to your server followed by the name of the `phpinfo()` script you created.

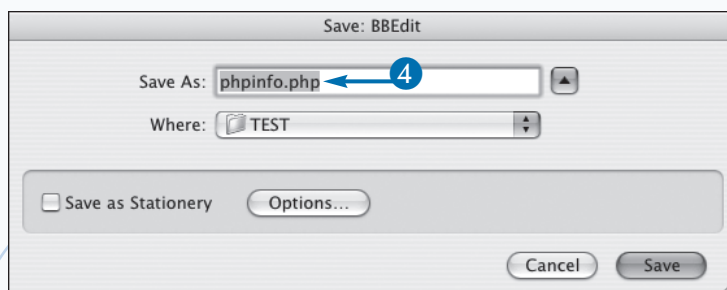
When the page loads, you see a screen presenting detailed information not only about PHP but also about your Web server software and configuration.

Verify Installation

- 1 Create a new PHP script using a text editor.
- 2 Create a PHP processing block using the PHP processing instruction delimiters `<?php` and `?>`.
- 3 Inside the PHP processing block, type `phpinfo();`.

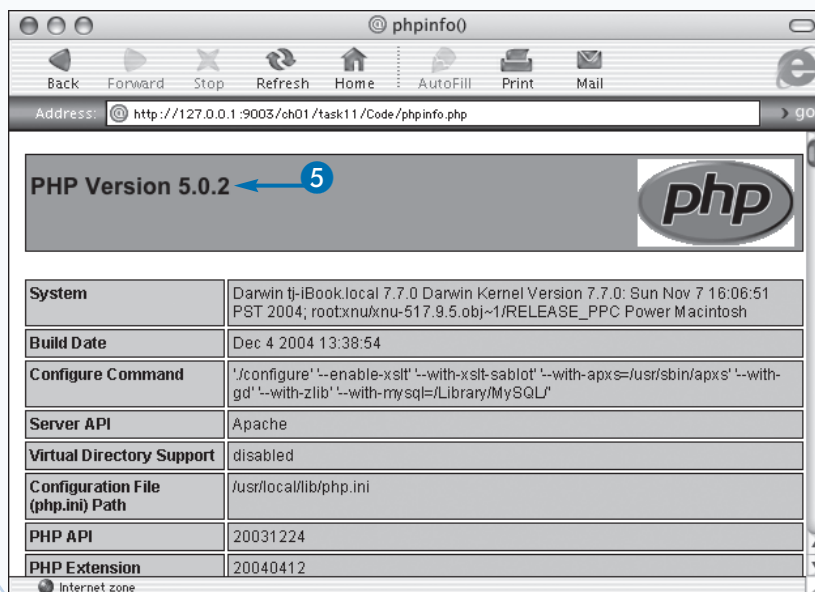


- 4 Save or upload the file to your Web server.



- 5 View the script in your Web browser.

Detailed configuration information about your installation of PHP is displayed.



Extra

If the Web server and PHP information is not displayed when you view the `phpinfo` script in your Web browser, the Web server may have a configuration error. You can review the topics in this book related to the installation and configuration of not only PHP but of your Web server to help determine the error. You can also review the manual and various README files that were included in the distributions of PHP and of your Web server software to determine if there are special considerations, options, or steps that you may have missed when installing and configuring your software.

It is considered a security risk to host a `phpinfo` script on your Web server for any longer than absolutely necessary. This is mostly due to the level of detailed information provided by the script. While "security through obscurity" is not in and of itself an acceptable security model, taking steps to obscure details from outsiders is a helpful part of a larger security model.

If you want to limit the information the `phpinfo` script displays, you can pass one of the following PHP constant values to the `phpinfo` function: `INFO_GENERAL`, `INFO_CREDITS`, `INFO_CONFIGURATION`, `INFO_ENVIRONMENT`, `INFO_MODULES`, `INFO_VARIABLES`, or `INFO_LICENSE`. These options are useful if you want to provide only a specific group of configuration settings to other developers without making available all of the information about your server.

Upload Pages to a Web Server

You can upload your scripts to a Web server different than the local server on which you develop. In most cases, the final production server on which you deploy your sites is hosted and maintained by a hosting company. You can use the File Transfer Protocol (FTP) to transfer your files to your external server.

Before you can transfer files to your remote Web server, you need to install an FTP client application. Due to the ubiquity of the File Transfer Protocol there are many available FTP client applications for every major operating system. When you choose an FTP client, install it according to the instructions provided by the software publisher.

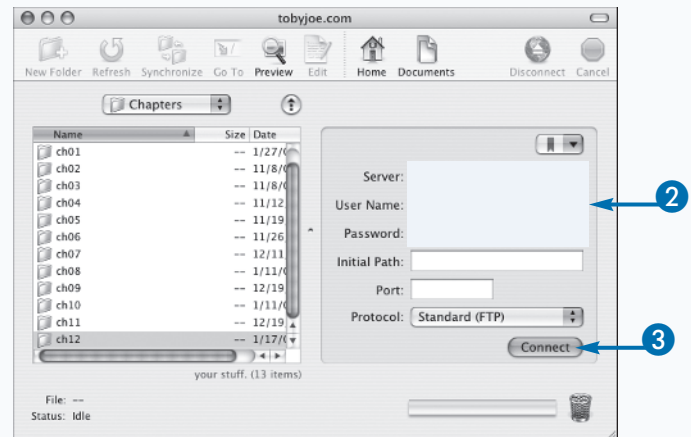
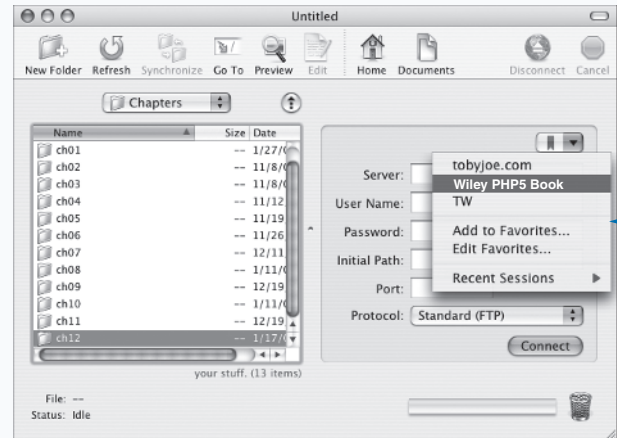
The first step in transferring files or folders to your remote server is to create a connection to the server. You can use an FTP program to transfer files from a machine running one operating system to a machine running a different operating system. For example, you can transfer files from

your Mac OS X computer to a Red Hat Linux Web server running your production site.

To create a connection with a remote FTP server, you must have certain information. The address of the server, either as a legitimate domain name or an IP address, allows your client application to locate the server on the Internet or intranet. You must also have a valid username and password for an account set up on the remote server in order to successfully establish a connection. Most FTP programs will allow you to save connection information for ease of reuse when connecting in the future. You should be aware that saving account information to a computer accessible by people other than yourself poses a significant security risk. If you use a computer accessible by others, be sure not to save connection information but instead type the address, username, and password each time a connection is necessary.

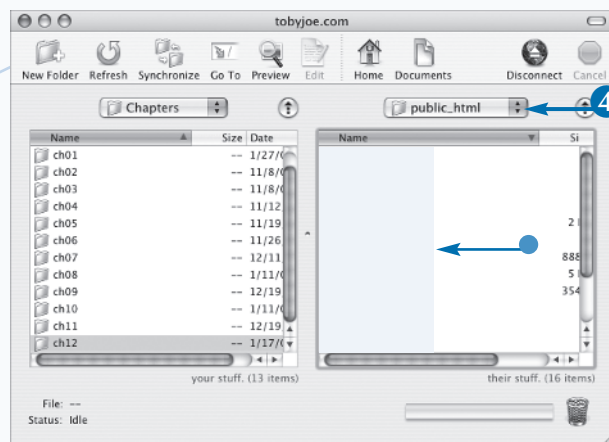
Upload Pages to a Web Server

- 1 Open your FTP client application.
 - Most FTP applications will open a display showing saved connections when first started.
- 2 Create a new connection by specifying the address of your server, your username, and your password.
- 3 Click Connect to open a connection.

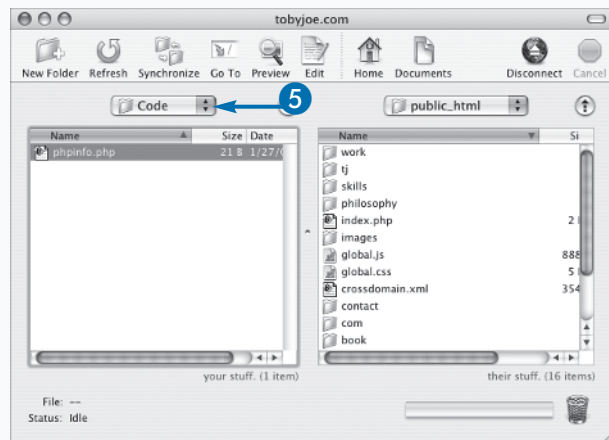


- When your connection is established, a listing of files on the remote server appears.

- 4 Navigate to the folder representing your Web server document root.



- 5 Using the local view of your files, navigate to the directory on your local machine in which your PHP scripts are saved.



Extra

Web hosting companies manage the hardware and software required to serve Web pages to people on the Internet. While it is possible to serve Web sites from machines on your own home Internet connection, for anything beyond the most trivial applications it is best to use a hosting company. Web hosts often provide nearly 100 percent uptime and help keep checks on security. You can also access support services if you need software installed or have questions about configuration options.

There are many available FTP applications for most popular operating systems. For Mac OS X, the Transmit client is a popular choice. For Windows, the CuteFTP program is a light and reliable FTP client. For Linux running the X Windows windowing system, you can use the Axy FTP program.

continued →

Upload Pages to a Web Server (continued)

After you establish a connection to your remote Web server, you can read, write, and delete information from the server. You can transfer entire directories, single files, or file archives. Most FTP clients also allow you to select any number of individual files and directories and upload them in one operation.

You must browse to the location on the remote server to which you want to transfer your files. When you initially connect to the FTP server, the working directory is set to the default directory associated with your account. If you upload files without changing to the proper location on the remote server, the files may not be visible to your Web server software. The directory to which you should upload your files changes on a server-by-server basis. You can ask the server administrator for the location of the Web server document root, which is the directory from which files are served to Web browsers. In some cases, your host may have a configuration that restricts the location from which

PHP files may be served. You can check with your host for any specific rules for the particular server.

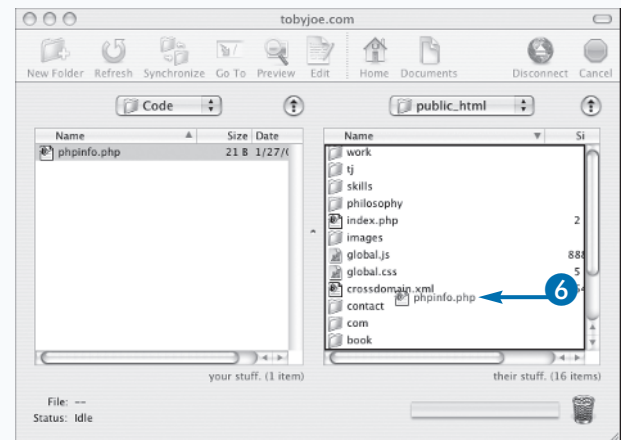
In some cases, FTP servers are configured to terminate a connection if no activity occurs for a given interval. This helps free unused resources for other users who may need to connect to the FTP server to manage their sites or files. If your session is dropped after an idle connection, you can simply reconnect.

A connection to an FTP server is temporary and volatile. That is, it is not a permanent and persistent connection. If you change files on your local development machine after uploading files to the remote server, you must connect and upload the new versions to the Web server to see changes take effect. Most FTP applications present a dialog box when an attempt to overwrite a file is made. You can choose to overwrite or cancel the upload. This safety mechanism helps prevent unwanted file overwrites.

Upload Pages to a Web Server (continued)

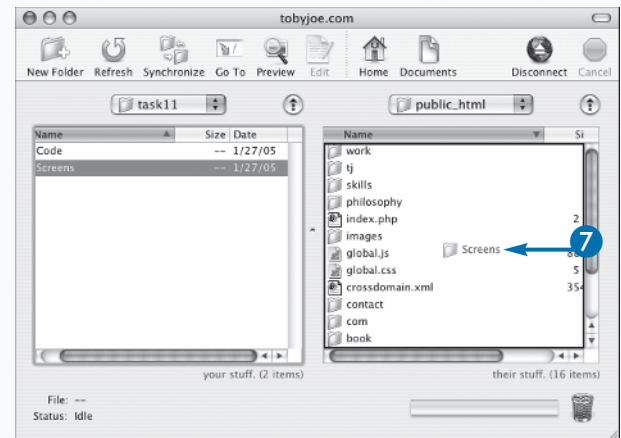
- 6 Drag a file from the local view to the remote view to upload it.

The progress is indicated in most FTP client applications.

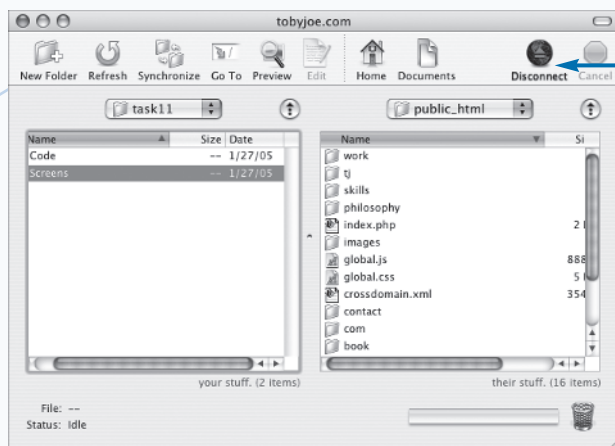


- 7 Upload a directory by clicking the directory in your local view and dragging it to the remote view.

The progress is indicated in most FTP client applications.

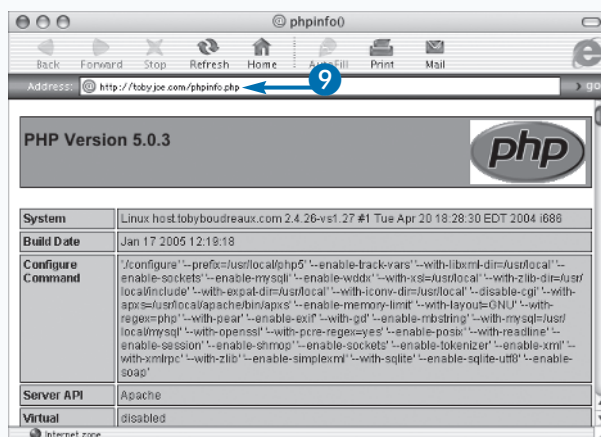


- 8 Disconnect from the FTP server either manually or by closing the FTP client application.



- 9 View your uploaded files in your Web server by typing the path to the newly uploaded file into a Web browser.

If the file is not displayed or if there is an error, check the location of the file on the remote server.



Extra

An important consideration when uploading files to a server is that any file in the document root of the server is most likely accessible to anyone with the ability to send HTTP requests to your server. Because the Web root is the directory from which files are meant to be served to your visitors, you should always keep sensitive information in a location outside of the Web root. PHP and similar languages can almost always access necessary files from locations outside of the Web root, but Web servers such as the Apache HTTPD server will never serve files protected in this manner.

PHP best practice where sensitive information is concerned is to keep files containing information such as database usernames and passwords in a special directory outside of the Web root. You can use the `require`, `require_once`, `include`, and `include_once` functions to access files located in any directory on your computer to which PHP and Apache have the appropriate permissions.

TYPE THIS:

```
$dbfile = $_SERVER['DOCUMENT_ROOT']
. '/../secure/db.php';
require_once($dbfile) or die('Error
accessing required files.');
```



RESULT:

The file located in a directory called 'secure' outside of the Web root is included into the script.