



Building Websites with PHP-Nuke

Douglas Paterson



Chapter 4 "Managing the Site"

In this package, you will find:

A Biography of the author of the book

A preview chapter from the book, Chapter 4 "Managing the Site"

A synopsis of the book's content

Information on where to buy this book

About the Author

Douglas Paterson is a full time development editor and part-time author for Packt Publishing. He is a doctor of Mathematics and has over five years experience working on programming books across a range of subjects.

He lives in Birmingham, England with his girlfriend, and his unusually hairy dog Zak. He can be reached at doug@packtpub.com.

For More Information: www.packtpub.com/phpnuke/book

4

Managing the Site

In this chapter, we will begin to acquaint ourselves with PHP-Nuke's administration area, which allows you to manage your site from the comfort of your web browser. We'll look first at the website configuration settings. These options control many global properties of the site.

Then we will look at block and module administration. The tools available here let you change the functionality of your site with ease, and customize the site's features to your liking.

Your Site, Your Database

The database that we created when we installed PHP-Nuke in Chapter 2 is PHP-Nuke's storage repository.

That may sound like a rather trivial remark; we know PHP-Nuke is a database-driven web content management system. However, it is worth understanding the nature of what PHP-Nuke stores. PHP-Nuke stores not only information about registered users of the site, and such things as your news stories, features about you, your company, or your club, your photos and other images, but also stores *all* the information about your site and the content it holds.

In its database, PHP-Nuke stores such things as the name of your site, the site URL, the site logo, how many stories are displayed on the front page, whether users can comment anonymously on stories, the footer text displayed at the bottom of the page, how many people have read the stories, the voting information about stories, and also what layout and choice of colors are used to display the site. There are many, many more things PHP-Nuke squirrels away into its database, but the point in general is that your site is determined by the contents of its database.

This may sound rather overwhelming, particularly if you are new to databases—but this is precisely where the real power of PHP-Nuke lies. You don't have to be a MySQL master or know anything about the finer points of database theory; in fact, you generally won't be touching the database yourself. PHP-Nuke has a powerful web-based administration tool that lets you control and maintain your site. Through it you are effectively managing the database but this is happening behind the scenes and it is not something that you need to overly concern yourself with.

For More Information: www.packtpub.com/phpnuke/book

Visiting the Administration Area

With PHP-Nuke's awesome administration tool, you manage your site through your browser, controlling almost every aspect of its behavior, as well as adding and maintaining the content that is displayed. This doesn't mean that anyone can mess with your site; access to the administration area is restricted. You, the super user, as head of administrators have supreme power and can even appoint other people to act as limited administrators, with specific abilities to moderate and approve content for certain parts of the site.

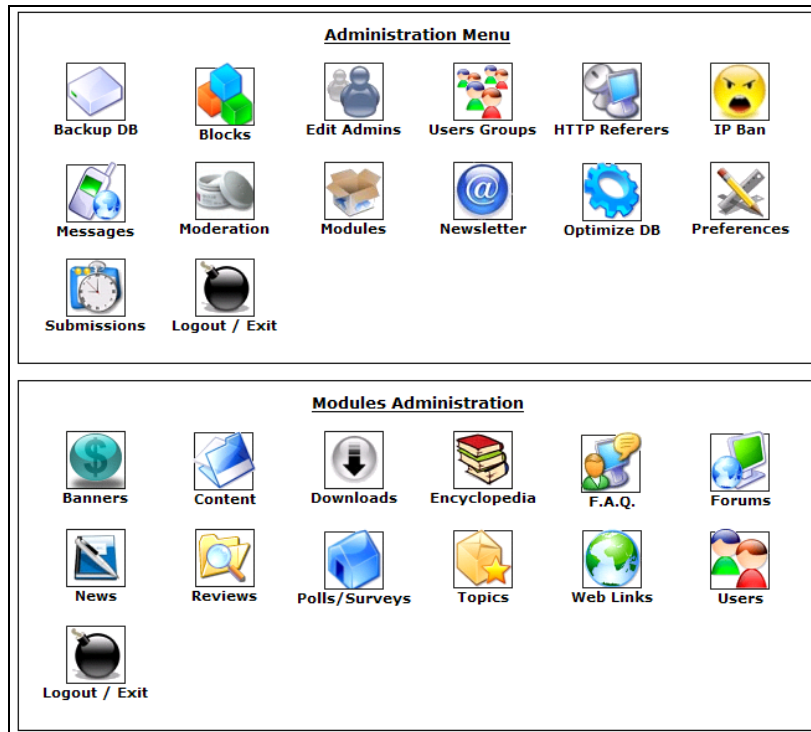
PHP-Nuke's administration area can sometimes feel too comprehensive and often be overwhelming, occasionally counterintuitive in its behavior. This is the jungle we will beat our way through in the next few chapters, and in fact, it's where you will spend most of your PHP-Nuke life (the administration area, not these chapters!).

The first thing to do is to log in to the administrator account. Enter the following URL into your browser:

`http://localhost/nuke/admin.php`

If you are not already logged in, you will be prompted for the administrator username and password created in the previous chapter. Enter these and click the Login button to proceed.

Once you log in, you will be in the administration area and are confronted with two monstrous administration menus in the center of the screen:



This is the central hub of the administration interface. Each of the icons you see on screen is a link to specific parts of the administration area, responsible for the control and management of particular features. If you scroll down the page, you will find some panels with information about the current home module, how many users are online, and some details of recently published stories, although at the moment, there is not much to see in any of these displays since we have no content or users!

The top menu of the administration interface, the Administration Menu, has icons for general 'system' management functions. These control the 'core' operations of PHP-Nuke, such as:

- Block and module management
- Database backup and optimization
- Banner management, users and user groups, and newsletters
- Site configuration
- Logging out of the administrator account

The lower menu, Modules Administration, has icons that take you through to the administration areas of individual modules. There is also another logout link.

Note that if you are using a version of PHP-Nuke earlier than 7.5, there is only one large menu, the Administration Menu, and this contains all the above icons mixed in together.

The two-menu split emphasizes the division of labor for managing a PHP-Nuke.

- The top menu has tasks for maintaining and configuring the site.
- The bottom menu has tasks for maintaining and configuring individual modules.

First experiences of the administration menu are often perplexing—you click on one of the images and the page reloads, but nothing seems to have happened. The menus are still there in the middle of your page, grinning at you. (Particularly the rather gruesome looking IP Ban icon; you may begin to believe its eyes follow you around the room.)

What you're actually after is displayed below the menus. By default, the administration menus are always displayed at the top of the page in the administration area; the action you're trying to do is contained in the panels underneath, and you will generally have to scroll down to get at what you want. Possibly, if your screen resolution is sufficiently high and your browser window sufficiently sized, then you won't get this problem, but for most of us, we will find ourselves wondering if anything has happened.

The advantage of these ever-present menus is that if you suddenly find yourself needing to switch to another task, you simply scroll back up to the top of the page and click on the desired icon.

If you want to return to the administration homepage at any point, you can either enter the URL of the administration homepage (`http://localhost/nuke/admin.php`) or if you glance to the left-hand side of your page, you will see the Administration block.



The top link in this block, Administration, returns you to the administration homepage. This block is ever present if you are logged in as the administrator. Administrator movements are not necessarily restricted to the 'back end' (the administration interface) of the site. You can also visit the 'front end' of the site and view the site as your visitors see it. However, for the administrator, extra, context-sensitive links appear on various items that provide you with a fast track to the administration area, and let you control that item. We'll see more of these links as we look in detail at the default modules over the next few chapters.

Also, there are special blocks that are only visible to the administrator, the Administration block being one of them.

You can replace the graphical administration menus by a more manageable text menu, but for now we will be working in the more familiar graphical environment, at least until we know our way round.

Site Preferences

Our first job will be to change some global settings of our site; we do so by clicking on the Preferences option:



When you do this, the page will reload and the Administration Menu will reappear, and then you should scroll down to the Web Site Configuration menu. This is a long list of options; the top part is shown in the following figure:

Web Site Configuration

General Site Info

Site Name:

Site URL:

Site Logo: [must be in /images/ directory. Valid only for AvantGo module]

Site Slogan:

Site Start Date:

Administrator Email:

Number of Items in Top Page:

Stories Number in Home:

Stories in Old Articles Box:

Activate Ultramode? Yes No

Allow Anonymous to Post? Yes No

Default Theme for your site:

Select the Language for your Site:

Locale Time Format:

At the foot of the list is a Save Changes button (not seen in the screenshot as it is too far below). This button has to be clicked for any changes to persist.

The list of Web Site Configuration options is divided into a number of panels, grouping together options for particular tasks:

- General Site Info
- Multilingual Options
- Banners Options
- Footer Messages
- Backend Configuration
- Mail New Stories to Admin
- Comments Moderation
- Comments Option
- Graphics Options
- Miscellaneous Options
- Users Options
- Censure Options

We won't look at all of these panels now; instead we will look at them as we need them. For example, when covering story management in Chapter 6, we'll explore the Comments Moderation, Comments Option, Censure Options, Backend Configuration, and the Mail New Stories to Admin panels.

Managing the Site

We shall now look at some of the options in General Site Info; these control some basic options for our site.

First up is the Site Name option. This is the name of your site, and is usually displayed in the title-bar at the top of the browser. It also used in any text referring to your site, such as email messages automatically sent out by the site (for example, the confirmation message sent to a user who has created an account on your site).

Let's stamp the identity of this site, by changing the Site Name value to the Dinosaur Portal:

General Site Info	
Site Name:	<input type="text" value="the Dinosaur Portal"/>

Now scroll to the bottom of the list of preferences where you see the Save Changes button. Click this to update your site. After the page reloads, you should see that the title bar in your browser has changed from PHP-Nuke Powered Site to our new site name—the Dinosaur Portal.

When the page reloads, you are still on the Web Site Configuration menu page. This is good in case you need to make any further changes, or if you got something wrong with the last change you made. There are some parts of the PHP-Nuke administration interface where clicking a Save or Ok button does not keep you in the same part of the administration interface but returns you to the administration homepage. This kind of thing can make you lose your bearings early on.

Although we only made one change before clicking the Save Changes button, you can, of course, make as many changes to the preferences as you like before clicking the button.

The Site URL is important too. This field holds the URL of your site homepage (without the `index.php` bit). If you specify the wrong Site URL or, more likely, forget to change it from `http://phpnuke.org`, then the consequences are not drastic; visitors will not suddenly find themselves transported to another site when they click a link on your site. However, the Site URL is used in emails sent to newly registered users with a link to confirm their registration. With the wrong Site URL here, people will go to the wrong site to register (and fail!). We will remind you of this when we discuss emails sent out by the system.

Let's change the Site URL before we forget. Since our site is at `http://localhost/nuke`, enter that into the Site URL field, and then scroll down and click the Save Changes button.

The Site Slogan, Site Start Date, and Administrator Email are straightforward to change. The Administrator Email account is the email account that will be used to send out user registration confirmations. The Site Slogan value is used in the META DESCRIPTION tag in the page header of your page:

```
<META NAME="DESCRIPTION" CONTENT="Your slogan here">
```

This tag is used by some search engines to create the description of your page in its listing. (That is *when* you are visited by search engines, which is still a long way off!)

By default, the value of the META DESCRIPTION tag is fixed for all pages in PHP-Nuke, and takes the value of your Site Slogan field.

The Site Logo specifies an image used by some modules to 'stamp' their pages. This value does not control any site logo image that may appear in the site banner at the top of your page.

Another interesting option is the Default Theme for your site. This gives you a drop-down box with a list of the currently installed themes. Select NukeNews from the list, scroll down, and click Save Changes. When the page reloads, it looks rather different:



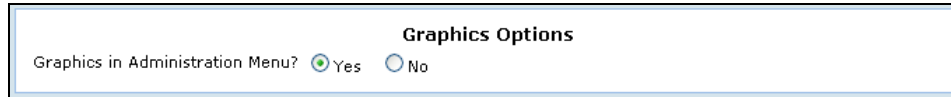
Not bad for two clicks of a mouse. We just changed the site's default theme and immediately the new theme has been applied, and we now have a very different looking site. It still 'works' the same, but it looks very different. One of the most obvious changes is the icons in the Administration Menu. There are some standard images in PHP-Nuke that can be overridden by images from the theme. The icons in the Administration Menu are one set of images that can be overridden like this.

Every visitor sees the theme that is specified as the default theme. Registered users have the option to choose their own theme, to personalize the site to their liking.

Now let's select the DeepBlue theme from the list of themes and click the Save Changes button. In the next few chapters we're going to see a lot of screenshots from the PHP-Nuke administration interface and the front end, and they're all going to be taken with the DeepBlue theme. If you're not using this theme as you follow along, things could look different. The DeepBlue theme is the default theme.

Turning off the Graphical Icons

For future reference, if you get sick of the Administration Menu icons (perhaps the terrifying IP Ban icon is finally getting to you), the Graphics Options panel is where you can turn off the graphical administration menu:



Graphics Options

Graphics in Administration Menu? Yes No

We will leave it set to Yes for now as we explore the administration interface. When you feel more confident, you can return here and set it to No to replace the graphical menu by a text menu.

The Cookie Crumbles

That's enough of the Web Site Configuration menu for now. Don't worry; we will come back to it over the next few chapters. Your next task is to close your browser.

Now open a new browser window, and navigate to your site's homepage (<http://localhost/nuke/>). You will notice that you are still logged in as the administrator—you can see the Administration block in the left-hand side column.

You may find this rather strange—you didn't enter a username or a password or go through the `admin.php` page, so how did it know? The answer is a cookie. PHP-Nuke issues cookies to visitors, which contain a number of user preferences, including their login details. This means that when the visitor returns to the site they are identified, and dealt with accordingly. This explains why you are logged back in as an administrator without having taken any action.

An annoying side-effect is that if you wanted to view the site as a visitor and administrator at the same time, you would have to log out and log in again before viewing. Should you find yourself doing this often, an obvious solution is to use two different types of browsers—say Mozilla Firefox and Internet Explorer (cookies are distinct on the two applications)—so one can be your administration browser and the other can be your visitor browser.

Backing Up the Site Database

It's not an exciting activity, but backing up your database is essential for the continuous running of your site. Almost everything about your PHP-Nuke site is stored in the database, and if there's some problem with the database, it will surely translate into a big problem on your site. Also, creating a backup of the database is not only to prevent a catastrophe; if you plan to develop your site locally (as we are doing) before uploading to your web host, then you will want a backup of the local database with you. This copy can be directly uploaded to the web host database server so that your site is ready to go without having to go through the entire configuration setup again.

You can create a backup of the database from within PHP-Nuke's administration area. Doing this creates a copy of the database in the same format as the `nuke.sql` file that you used to construct the original PHP-Nuke database. This file should then be stored safely, ready to be used in case of a database emergency.

Creating a database backup is a one-click process. You simply click the Backup DB icon in the Administration Menu:

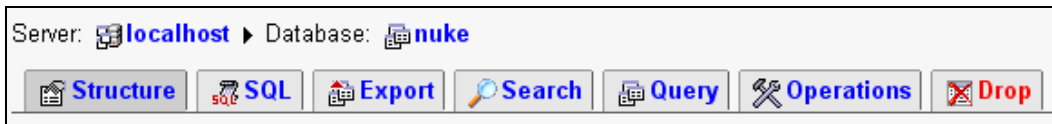


After a moment, a Save dialog box will open in your browser. The database backup is a text file having the default filename of Save.

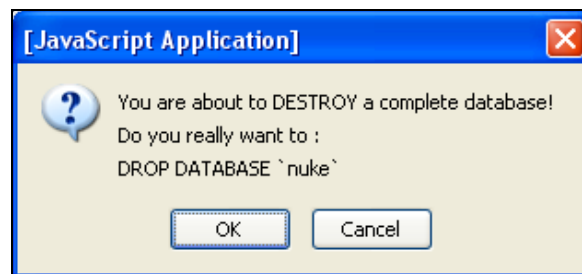
You can give the file a name in the format <tablename>_backup_<date>.sql, and then open it in a text editor to check what it contains.

Once you have the backup of the database, restoring the database is similar to the process of creating the original PHP-Nuke database that we saw in Chapter 2. The only difference is the choice of file to upload, and the fact that you will have to remove the existing data in the database before 'applying' the backup.

You can easily remove all the data, in fact the entire database, from phpMyAdmin. Simply open it up in your browser, and select your PHP-Nuke database from the drop-down list on the left-hand side of databases. Once it is selected, and all the tables are displayed, click the Drop tab on the right-hand side:



An alert message will appear:



The use of uppercase for DESTROY signals that you are about to perform a drastic operation. Clicking OK will remove the database from your database server, but this is not a problem since you did make a backup copy of it.

To recreate the database, simply retrace the steps outlined in Chapter 2 to create the database (you will have to define the database user and role again), and then to create the tables and fill them with the data from the backup, click the SQL tab, click the Browse button, and navigate to your

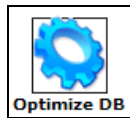
database backup file. Selecting this and clicking Go will restore your database. Removing the database will, of course, mean your PHP-Nuke site is temporarily broken, and any visitors will see one of the data access error screens that we saw in Chapter 2.

Note that you can zip up the database backup and upload the ZIP file to phpMyAdmin rather than the plain text file. phpMyAdmin will automatically extract the text file and process it. However, it is possible that when you are attempting this on a web hosting provider, the server may not be able to work with the ZIP file in this way; it should become obvious that this is a problem, since you will get a page full of error messages, and you will have to upload the file in the standard text format.

Be careful about the size of the file you upload when working with a hosting provider; it is also likely that there will be some restriction on the size of file that you can upload to the server—typically 2MB.

You should get in the habit of backing up the database once your site is up and running. How often you back up will depend on how busy your site is, and how often new content is added or changes are made, but a daily backup is wise for any active site.

There is also an Optimize DB (Database) option in the PHP-Nuke Administration Menu:



As the name suggests, clicking this icon will have PHP-Nuke go through all the tables in the database in an attempt to 'optimize' them. In fact, PHP-Nuke issues a MySQL OPTIMIZE TABLE statement against every table in the database, and reports any space saved in the database by doing this. You can read more about exactly what the OPTIMIZE TABLE statement does, and when you may want to use it, in the MySQL manual at:

<http://dev.mysql.com/doc/mysql/en/optimize-table.html>

Managing Blocks

Blocks are a key part of the layout of a PHP-Nuke website, and you have a great deal of ability to customize the way in which they are displayed. We've already seen the Modules block; this is the main navigational control in PHP-Nuke. Here are some of the other blocks shown on the default homepage to a visitor:

Languages: This block is placed on the left of the page. This block allows you to choose the PHP-Nuke interface language.

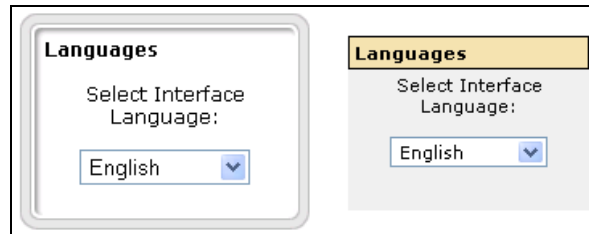
Survey: This block is placed on the right-hand side of the page. This allows visitors to take part in straw polls on some pressing matter of the day. The default poll is the question we're all asking at this point in the book.

If you are logged in as an administrator, you will also see blocks such as the Administration block that we saw earlier, and the Waiting Content block in the left-hand column.

This block shows a summary of any pending user-submitted content that needs to be moderated and approved before publishing. We'll find out more about this 'workflow' in the next few chapters.

You will notice that all the blocks on the page look the same as this, and the overall columnar layout is a hallmark of most PHP-Nuke sites.

You will also notice that each block consists of two parts—one part is the **title** (Languages or Survey for example), and the other part, the body of the block, is the **content**. A block does not concern itself with how it is displayed; this is taken care of by the PHP-Nuke theme. The pictures below show the Languages block when it is rendered in two different themes; 3D-Fantasy and Karate in these cases:



Although the blocks look different, you can see that the title-content split persists.

Types of Blocks

A block's job is to produce its content. This content may come from one of three places, and the origin of the content determines the type of block:

- HTML stored in the database
- Dynamic content created by PHP files
- RSS feeds from other sites

Let's talk about the third option for a moment. RSS is a standard format for sharing web content or summaries of web content, together with data about the content such as a link to the full version of the content. Typically, the content changes often, such as news stories, latest postings to a personal weblog, or assorted new additions to a site. RSS stands for Really Simple Syndication; you can read more about its definitions (the specification) and its history at <http://blogs.law.harvard.edu/tech/rss>. The site <http://www.whatissrss.com/> has more information about RSS. You can also refer to *RSS and Atom* by Heinz Wittenbrink from Packt Publishing (ISBN 1-904811-57-4).

The information is delivered via an RSS feed, and using such a mechanism allows a site to make its content available to others without having to notify them when new material is published. In other words, the receiver of the feed can have up-to-date information about the content available on the feeding site.

The process of making these headlines available is known as 'syndicating content', and is becoming an ever more popular way of distributing information without actually doing anything; anyone interested will consume the news feed.

PHP-Nuke's RSS blocks allow you to include RSS feeds on your site, so that you can suck up the headlines of stories from relevant sites and display them on your own site, creating a network of content for visitors to your site.

PHP-Nuke also provides facilities for syndicating news stories on your PHP-Nuke site via RSS; we'll look at that in Chapter 6.

Block Positioning

To determine where a block is displayed on the page, the block is given a position (left, right, or center), and a number called its weight. Blocks with a lower weight are displayed first; we'll see that changing the weight of a block moves it up and down the page.

Block Visibility

Blocks have a property that determines who is able to see the block. The administration blocks, for example, are only visible to the administrator and not to any other user. A block that contains the details of a logged-in visitor should only be displayed to that visitor—for an unregistered visitor this block would be meaningless. Also, it is often wise to not show advertising blocks to paying subscribers of your site.

The Blocks Administration Area

To set about our work with blocks, click on the Blocks icon in the Administration Menu:



This brings up the Blocks Administration area. From here you can control all the blocks on your site. You can change the position of a block, the order in which it appears, its visibility to users, or even activate or deactivate the block.

The top part of the Blocks menu is show here:

Title	Position	Weight	Type	Status	Visible to	Functions
Modules	◁ Left	1 ↓	FILE	🔧	All Visitors	📄 ⚙️ 🗑️ 🔍
Administration	◁ Left	2 ↑ ↓	SYSTEM	🔧	Administrators Only	📄 ⚙️ 🗑️ 🔍
Who's Online	◁ Left	3 ↑ ↓	FILE	🔧	All Visitors	📄 ⚙️ 🗑️ 🔍
Search	◁ Left	4 ↑ ↓	FILE	🔧	All Visitors	📄 ⚙️ 🗑️ 🔍
Languages	◁ Left	5 ↑ ↓	FILE	🔧	All Visitors	📄 ⚙️ 🗑️ 🔍
Random Headlines	◁ Left	6 ↑	FILE	🔧	All Visitors	📄 ⚙️ 🗑️ 🔍



If the icons themselves aren't clear enough, hovering your mouse cursor over the graphic will display a tooltip explaining the function of that graphic.

Let's have a run-through of the columns in the table of blocks.

We have already seen the Title and the Position attributes of the block; the Weight column determines a logical ordering of the blocks within their 'column'. The order in which the blocks appear can be adjusted with the up and down arrows in the Weight column. In this way, you can reorder the blocks as you please, moving them up and down your page.

Managing the Site

The Type column specifies the type of block: FILE, HTML, RSS/RDF, or SYSTEM. We discussed the first three types of block earlier; SYSTEM blocks are added by PHP-Nuke itself.

If the block Status is shown as , then the block is active and is displayed on the site. If this icon  is shown in the Status column, then the block is inactive. If the block is inactive, the next block (namely the block with the next lowest weight) will be displayed in its place—there is no gap left where an inactive block should be. Thus blocks can be removed from the page without leaving unsightly holes in the layout.

The Visible to column determines which group of visitors can see the block; the groups are, as we know, All Visitors, Registered Users Only, Administrators Only, or Anonymous Users Only.

The last column, Functions, is where the fun starts. Each block has four icons in the Functions column:



From left to right these icons allow you to:

- Edit the details of the block
- Activate or deactivate the block (depending on whether it is currently active)
- Delete the block
- Preview or 'show' the block





















The preview icon is only enabled when the block is currently deactivated; clicking this icon gives you a view of what the block will look like when activated on your site. When the icon is not enabled, it will be grayed out.

If you forget which icon is which, then hover your mouse cursor over it, and the action of the icon will be displayed—Edit, Activate (or Deactivate), Delete, or Show.

We've talked about how we can reorder and reorganize the blocks, so let's actually have a go at it. Our first task will be to swap the order in which the Who's Online and Languages blocks appear on the page. The inactive Search module is sandwiched between them, but that isn't going to concern us at the moment.

Let's see how we can change the position of the Languages block:

1. Click on the up arrow in the Weight column of the Languages block in the block table. As your mouse cursor hovers over the arrow, you should see the helpful Block UP text popup, to let you know what clicking that image will do:

Who's Online	◁ Left	3	↑ ↓	FILE		All Visitors	   
Search	◁ Left	4	↑ ↓	FILE		All Visitors	   
Languages	◁ Left	5	↑ ↓	FILE		All Visitors	   
Random Headlines	◁ Left	6	↑ ↓	FILE		All Visitors	   

2. When the page reloads, the Languages and Search block have swapped weights in the blocks table, as seen in the picture below. Since the Search block is inactive, it is not displayed, so at this point you cannot see any change to the look of your page.

Who's Online	◁ Left	3	↑ ↓	FILE		All Visitors	
Languages	◁ Left	4	↑ ↓	FILE		All Visitors	
Search	◁ Left	5	↑ ↓	FILE		All Visitors	

3. Click on the up arrow for Languages, and when the page reloads, our work is done.

Languages	◁ Left	3	↑ ↓	FILE		All Visitors	
Who's Online	◁ Left	4	↑ ↓	FILE		All Visitors	
Search	◁ Left	5	↑ ↓	FILE		All Visitors	

If you glance to the left-hand side of your page, you will see that the Languages and Who's Online blocks have switched positions. The picture below shows a 'before and after' shot of these two blocks on the page (the before shot is on the left):

Who's Online

There are currently, 1 guest(s) and 0 member(s) that are online.

You are Anonymous user. You can register for free by clicking [here](#)

Languages

Select Interface Language:

English

Languages

Select Interface Language:

English

Who's Online

There are currently, 1 guest(s) and 0 member(s) that are online.

You are Anonymous user. You can register for free by clicking [here](#)

That's moving blocks up. You go through a similar process for moving blocks down.

Occasionally, things don't quite go to plan when moving blocks down, and you may find that you have two blocks with the same weight. To resolve this problem, there is a way to 'rebalance' the weights. At the bottom of the blocks table you will find a Fix Block's Weight Conflicts link. Clicking this link, as the name suggests, corrects the list of block weights, ensuring that each block has a distinct weight.

Time For Action—Changing Block Position

Now we've made the Languages block move up the page, let's move it to the other side of the page.

1. To see the properties of a block, we click its Edit icon in the Functions column. Click the Edit icon of the Languages block now. You should see the Edit Block panel appear:

Block: Languages (Block File)

Title:

Filename: (Select a custom Block to be included.
All other fields will be ignored)

Position:

Activate? Yes No

Expiration: days

After Expiration:

Who can View This?

Visible to Subscribers? Yes No

2. Select Right from the Position drop-down box:

Position:

Activate?

Expiration:

After Expiration:

After Expiration:

3. Click the Save Block button at the bottom of the panel.
4. When the page reloads, the Languages block has moved way down the list of blocks in the table, and also it has vanished from the left-hand column of the page. You will now find the Languages block along with other blocks whose Position is Right:

Languages	Right ▾	3	↑ ↓	FILE		All Visitors	
Survey	Right ▾	4	↑ ↓	FILE		All Visitors	

5. Now open a new browser window and go to your site's homepage. The Languages block is now on the right-hand side column, at the top of that column at the moment in fact. (The blocks above it in the list are either inactive or are displayed to registered visitors only.)



The process to change the block position was straightforward. This combined with the ability to reorder the blocks on the page means that you have great control over where blocks go on your page. In addition to Left and Right, there are two other positions for blocks:

- Center Up: In this position the block appears *before* the main module output.
- Center Down: In this position the block appears *after* the main module output.

Blocks in these last two positions are only shown on the site homepage. On the homepage of your site, blocks will be displayed in any of the four positions. In general, when you are looking at any of the modules, only the left-hand block column is displayed (Downloads, Feedback, and Web Links are exceptions to this).

An interesting thing to note about the repositioning of a block is the new weight it is given.

- When you change the position of a block, its weight *remains the same*.
- Any blocks of your target position whose weight is equal to or more than your block's weight will have their weight increased by one; they will all shuffle down to make room for your block.
- Any block in the same position as your block's original position whose weight is more than your block's weight will have their weight's decreased by one; they will all shuffle upwards to fill in the gap left by the moving block.
- If the weight of your block is higher than the maximum weight of any block in the target position, then your block will be given a new weight that is one more than that highest weight, so that again there is a nice sequence of consecutive weight values.

In short, when you change the position of blocks, PHP-Nuke will make some adjustments to all the other weights to retain sequences of consecutive weight values, which keeps things tidy.

Adding Blocks

The interface for adding new blocks to PHP-Nuke is the Add a New Block panel, found underneath the list of blocks in the Blocks Administration area.

Add a New Block

Title:

RSS/RDF file URL: Custom [Setup]
(Select Custom and write the URL or just select a Site from the list to grab news headlines)

Filename: None (Select a custom Block to be included. All other fields will be ignored)

Content:

If you fill the URL the content you write will not be displayed!

Position: Left

Activate? Yes No

Expiration: days

After Expiration: Deactivate

Refresh Time: 1 Hour (Only for Headlines)

Who can View This? All Visitors

Visible to Subscribers? Yes No

There are lots of options on this panel; some of the options apply to all types of block, and some only to certain types of block. This can be rather confusing to start with and is something that can make block management tricky for the beginner.

There is also no option to specify what kind of block you are creating (HTML, RSS/RDF, FILE); this is because the type of block you create is determined for you by PHP-Nuke based on what you put into the fields of the Add a New Block panel.

Options for All Blocks

Here are the options that apply to all blocks, followed by a description of their purpose:

Option	Description
Title	This is the title of the block; this appears in the list of blocks in the Blocks Administration menu, and usually identifies the block on the page when it is displayed. You have to choose the value for Title. A block cannot be created without a title. It is possible to create two different blocks with the same title.

Option	Description
Position	Here you select from Left, Right, Center Up, or Center Down. This determines where the block appears on the page when displayed.
Activate?	Determines whether the block should be activated now. Once activated, a block can be deactivated from its Functions column. The default setting for this option is Yes—the block will be activated once it is created.
Expiration	Allows a time period—specified as a number of days—after which the block will be deactivated. The default value is 0, and this means the block will remain indefinitely. Note that if you set an expiration date, you cannot modify this value by editing the block properties.
After Expiration	Determines what action should be taken after the block expires, if an expiration time period has been set. This can be one of two values; Deactivate or Delete. Deactivate means the block will be deactivated and no longer visible once it expires; this option can be set on any type of a block. The Delete value is only applicable to HTML or RSS/RDF blocks, and will actually remove their details from the database.
Who can View This?	Determines the type of visitor that can view this block. The value is chosen from the categories we saw at the end of the previous chapter; Anonymous Users Only, Registered Users Only, Administrators Only, or All Visitors. The default value is All Visitors. Thus any freshly created block is visible to any visitor to the site.
Visible to Subscribers?	Determines if the block can be seen by subscribers. If it is set to No, then subscribers will not be able to see the block, regardless of the Who can View This? setting. This is commonly used to hide advertisement blocks from subscribers who will have already paid a fee for their subscription. The default value is Yes.

We will cover the options that apply to specific types of blocks as we come across them.

Time For Action—Adding a Static Block

Our first attempt at creating a block will be a simple HTML block; for this we simply use a lump of HTML. This will be stored in PHP-Nuke's database, and then retrieved and displayed when this block is output on the page. The HTML block cannot take advantage of any 'server-side' PHP processing, so the output is always fixed. The block will always look the same, whenever you look at it, whoever you are. The block is truly 'static'.

First of all, let's make sure we're in the Blocks Administration area; click on the Blocks icon in the Administration Menu just to make sure, and scroll down to the Add a New Block panel.

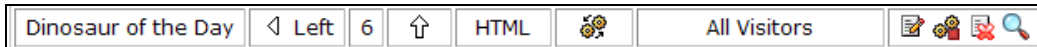
Our block will be a 'Dinosaur of the Day' block, displaying an image of today's dinosaur, and its name. We're going to position the block on the left-hand side of the page, and make it available to every visitor, and it shall remain on the site indefinitely.

1. Enter Dinosaur of the Day into the Title field.
2. Ignore the next two fields (RSS/RDF file URL and Filename); these have nothing to do with our HTML block.

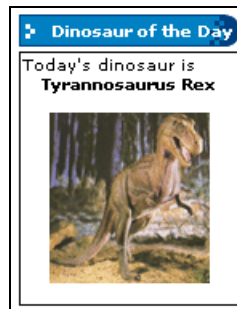
3. Enter the following text into the Content field:

```
Today's dinosaur is  
<br><center><b>Tyrannosaurus Rex</b></center><br>  
<center></center><br>
```
4. Leave the Position field set to Left, and leave Activate? set to Yes.
5. Leave all the other fields as they are—the block is to remain indefinitely so we leave Expiration set to 0. Hence the After Expiration field is redundant and Refresh Time does not apply to this type of block. The block is to be visible to everyone, so leave All Visitors selected in Who can View This?
6. Click the Create Block button.

When the page reloads, you will see your new block added to the list of blocks:



And if you glance over the left of your page, you will see the block displayed:



Immediately, you can see the kind of limitations static HTML blocks have—if we want a different dinosaur displayed on a different day, we would have to edit the block text itself. We could change the image, but still the name of the dinosaur remains.

Be careful when creating HTML blocks—if you enter anything into the RSS/RDF file URL field, whatever you type into the Content field will be ignored, and you will not create an HTML block but an RSS/RDF block. There is a warning about this at the foot of the Content field, so pay attention!

Adding Other People's News with RSS/RDF Blocks

OK, we've added our static HTML to the site. The next thing we'll do is genuinely impressive—we will add a news feed from an external site to our page. This is accomplished with the RSS/RDF block.

As we mentioned earlier, RSS feeds are a method for syndicating content. (Note that RDF is essentially a variation of RSS.) The data from an RSS feed is just a text file, in XML format, although that is something that doesn't concern us for now.

XML stands for Extensible Markup Language, a general-purpose markup language for carrying data between different sources and platforms. You can read more about XML at <http://www.w3schools.com/xml/default.asp>.

The first thing we'll need is a good news source. We'll grab the information about the latest adventures of the cartoon pair, Beaver and his dinosaur pal Steve from www.beaverandsteve.com. This news feed is found at:

<http://www.beaverandsteve.com/rss.xml>

Let's get back to the Blocks Administration area by clicking on the Blocks icon in the Administration Menu, and scroll down to the Add a New Block panel.

There are two ways we can store the URL of a target news feed in PHP-Nuke; we can either enter the URL directly into the RSS/RDF file URL field in this panel, or else we can create a new **headline** site. A headline site is just a named site with a URL for its news feed.

The advantage of creating a new headline site is that the URL is stored independent of the block; if you delete the block, then you can easily create a new block with the same news feed, without having to type (or maybe find!) the URL of the news feed again. Using a headline site also gives you the block title for free, as we will see now.

Time For Action—Creating a New Headline Site

1. From the Add a New Block panel, click on the Setup link on the right-hand side of the panel:

Add a New Block

Title:

RSS/RDF file URL: Custom
(Select Custom and write the URL or just select a Site from the list to grab news headlines)

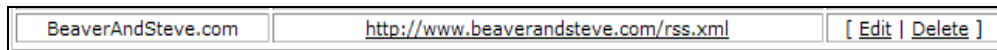
2. The Headlines Administration panel is displayed. This is a list of the currently defined headline sites; there are some twenty or so sites already defined, some of which are related to PHP-Nuke (PHP-Nuke, NukeCops, and NukeResources for example) while others are about general open-source news or just general news. The top of this list is shown here:

Headlines Administration		
Site Name	URL	Functions
AbsoluteGames	http://files.gameaholic.com/aqfa.rdf	[Edit Delete]
BSDToday	http://www.bsdtoday.com/backend/bt.rdf	[Edit Delete]
BrunchingShuttlecocks	http://www.brunching.com/brunching.rdf	[Edit Delete]
DailyDaemonNews	http://daily.daemonnews.org/ddn.rdf.php3	[Edit Delete]
DigitalTheatre	http://www.dtheatre.com/backend.php3?xml=yes	[Edit Delete]
DotKDE	http://dot.kde.org/rdf	[Edit Delete]

3. If you scroll to the bottom of this list, you will see the Add Headline panel. This is where we define our headline site. Enter BeaverAndSteve into the Site Name, and into the RSS/RDF file URL field enter our news feed URL:



4. Click the Add button, and when the page reloads, you will see our new news feed added to the bottom of the list of headlines:



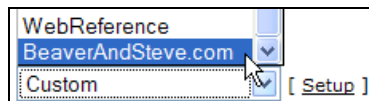
You can use the Edit or Delete links to amend the details of the news feed link. Clicking on the link itself will open the news feed in a new browser window. This is useful for checking if you've actually got the URL correct—if you specify an incorrect URL for an RSS/RDF block, then your page output can be severely disrupted, so it's wise to check that you're actually pointing at the news feed.

Curiously, there is no link to the Headlines Administration panel in the Administration Menu by default. You will have to come through the Blocks Administration area by clicking the Setup link, or bookmark the Headlines Administration page in your browser.

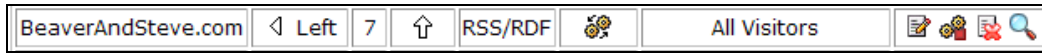
Time For Action—Adding the RSS/RDF Block

Now our headline site has been defined, let's get back to creating the RSS/RDF block. Since we are currently in the Headlines Administration area you will have to click on the Blocks icon in the Administration Menu to get back to the Blocks menu, and then scroll down to our Add a New Block panel and follow these steps:

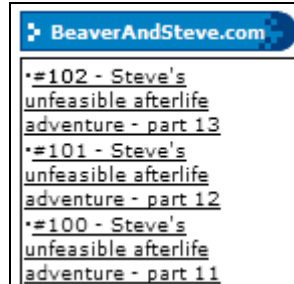
1. There is a drop-down box to the right of the RSS/RDF file URL field currently holding the value Custom. This contains a list of all our defined headlines. From this box, select BeaverAndSteve.com (it's right at the bottom!):



2. Let's leave the block on the Left, and set Activate? to Yes, and leave Refresh Time set to 1 Hour. We will also set this block to be viewable by All Visitors.
3. Click the Create Block button.
4. When the page reloads, you will see your newly created block in the list of blocks. Note that the block is of type RSS/RDF, and the Title has automatically been assigned from the name that we gave to the headline site:



Glance to the left of your page, and you will see the block in action. This is what the block displayed at the time of writing—when you see the block's output, the news will be different:



If you hover the mouse cursor over any of these links, you will see that these are links to the actual news stories on www.beaverandsteve.com, and clicking one will open that story in a new window. There is a [read more...](#) link at the foot of an RSS/RDF block that takes you to the homepage of the source site. We did not add any information about the URL for this homepage ourselves—this information is contained within the news feed itself, and was read and processed by PHP-Nuke for us.

Note that PHP-Nuke does not suck in the news feed from the source site every time the block is displayed. The operations of acquiring and processing the RSS feed every time the block was displayed would adversely affect the performance of the web server running PHP-Nuke, and so PHP-Nuke caches the feed in its database to use subsequently.

However, the Refresh Time setting of the block determines when PHP-Nuke should obtain a new version of the feed to keep it 'fresh'. When PHP-Nuke needs to display the RSS feed in the block, it will check the time it last stored the feed. If the time elapsed exceeds the Refresh Time value, PHP-Nuke will get a new version of the feed, and cache that. The Refresh Time setting doesn't have any effect on the other types of blocks.

Adding a File Block

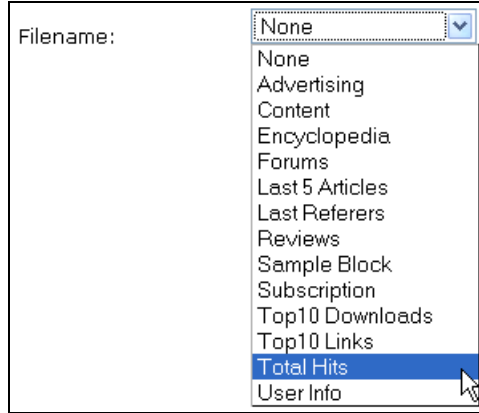
A file block is a PHP script that is stored in the `blocks` folder of the PHP-Nuke installation. At this point, we're not going to create a new block of this type; we'll see that in Chapter 10. There are a number of file blocks already in the `blocks` folder, provided by PHP-Nuke. We will choose one of those.

Time For Action—Adding the Total Hits Block

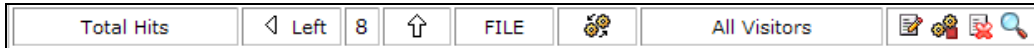
The Total Hits block simply displays the number of times the site has been accessed. PHP-Nuke maintains a counter that is updated whenever any page is requested from the site. This is the number that is displayed by the Total Hits block.

From the Administration Menu, click on the Blocks icon to get back to the blocks menu, and then scroll down to our Add a New Block panel.

1. Select Total Hits from the Filename drop-down list:



2. Let's leave the block on the Left, and set Activate? to Yes. We will also set this block to be viewable by All Visitors.
3. Click the Create Block button.
4. When the page reloads, you will see your newly created block in the list of blocks. Note it is of type FILE, and the Title has been provided from the filename:



5. A glance over to the left of your page, and you will see the Total Hits block is in place:



What Just Happened?

We just added a file block, and it was very simple. PHP-Nuke helps by providing a list of the available file blocks in the Filename drop-down list, and all you have to do is to select the filename from the list.

The names in the list correspond to PHP files in the blocks folder, and PHP-Nuke checks this list of files every time you come to the Blocks Administration area, and the drop-down list is populated. To show up in this list, the name of the file in the blocks folder must be of the form block-NAME.php, where NAME is the name of the block that shows up in the drop-down list. The block does not take its title from this part of the filename, you still have to add that yourself in the panel. If you have a quick look in the blocks folder in your PHP-Nuke installation, you will see that all the files are named like this.

Installing a new file block is simply a case of copying the required PHP file into the blocks folder. There are some example blocks in the code download for this book, and you can experiment with them. If you have placed a file into the blocks folder but it is not showing in the Filename drop-down list, then the file is likely to have its name in the wrong format for PHP-Nuke to pick up.

Managing Modules

Modules are like limbs to the body of your PHP-Nuke site; they kind of stick out and do stuff for you. However, PHP-Nuke allows your system to have an almost indefinite number of limbs, er, modules, and also allows you to add or remove them with ease. In this section, that's exactly what we're going to look at.

First of all, return to the Administration Menu and click on the Modules icon:



This brings you to the Modules Administration area. This area is for administration of all modules, not the properties of individual modules. Individual modules such as Downloads, FAQ, and so on have their own sections in the administration area.

Visiting the Module Administration area like this also 'refreshes' PHP-Nuke's module list. If you have installed any new modules, then PHP-Nuke will detect them and add them to its list.

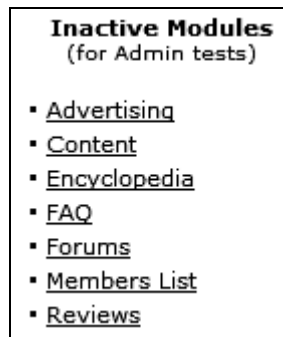
In the Module Administration area you will see some helpful text, describing some general instructions for module administration, and below that is a table from which you can actually control your modules:

Title	Custom Title	Status	Visible to	Group	Functions
Advertising	Advertising		All Visitors	None	
AvantGo	AvantGo		All Visitors	None	
Content	Content		All Visitors	None	
Downloads	Downloads		All Visitors	None	
Encyclopedia	Encyclopedia		All Visitors	None	
FAQ	FAQ		All Visitors	None	
Feedback	Feedback		All Visitors	None	
Forums	Forums		All Visitors	None	
Journal	Journal		All Visitors	None	
Members_List	Members List		Registered Users Only	None	
News	News		All Visitors	None	
Private_Messages	Private Messages		All Visitors	None	
Recommend_Us	Recommend Us		All Visitors	None	

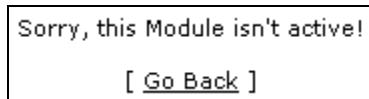
Managing the Site

This table is similar to the one we saw for managing blocks in the Blocks Administration area. Each module has a Title and Custom Title; the Title actually refers to a folder in your PHP-Nuke installation, and to avoid spaces in the filenames, underscores are used, and the Custom Title provides a friendlier name for the module with the use of spaces instead of underscores.

The Status column has an icon, similar to the blocks table, indicating whether the module is Active or Inactive. Active modules are visible on the site, and can be accessed by visitors. Inactive modules are generally not visible to the visitor, but can be accessed by the administrator. If you glance to the left-hand column of your page, at the bottom of the Modules block you will see a list of the inactive modules:



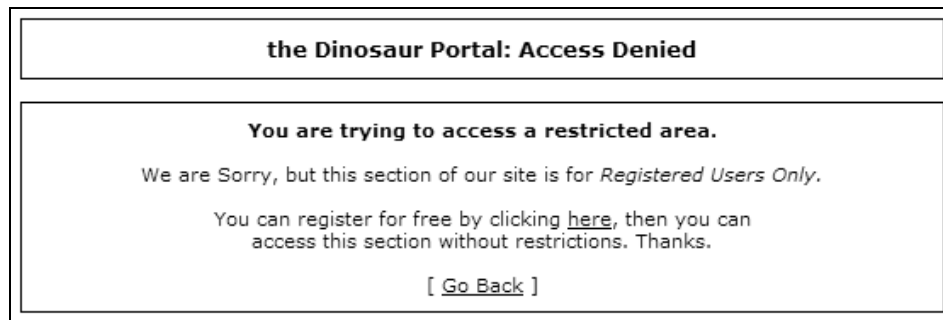
As an administrator, you can still visit and test these modules, either by clicking on these links or directly entering a URL. This list of inactive modules is only visible to the administrator, and should a sneaky visitor manage to find their way to an inactive module, they will be confronted with the following statement, and they are unable to use the module any further:



Making a module inactive does not 'remove' it from the site as you can see—it's still there, just sleeping. You can remove modules completely from the site, but this requires you to physically delete some files from your PHP-Nuke installation.


As with blocks, access to modules can be restricted to groups of users, as shown in the Visible to column. The group can be chosen from All Visitors, Registered Users Only, Administrators Only, and Subscribed Users.

Unlike the access restriction for blocks, whereby if a visitor is not permitted to see the block, the block itself will not be displayed, attempts to access a restricted module will display a message like this:



Module access can be restricted to registered users of your site, but you can also restrict access even further through the use of **user groups**. When users contribute to your site by posting a news story, commenting on a story, or recommending the site to a friend, they can earn points for themselves, as reward. When they have collected enough points, they become members of a particular user group.

Being a member of this group may be prestigious enough for some people, but it also means that you can allow these 'worthy' people access to parts of your site that are forbidden to those who haven't yet earned the privilege. The user group (and it can be only one group) that has exclusive access to a module is indicated by its Group column in the modules table. By default, this is set to None, meaning that no group has exclusive access. We'll look at creating user groups in the next chapter.

The Functions column has icons similar to the blocks table, and plays the same role as the Functions column in the blocks table. From here we can manage the properties of an individual module, in the same way that we edited the properties of blocks. Modules can be activated or deactivated, edited, or set as the default module—the 'home' module. You will notice that one of the modules (currently News) is highlighted in the table, and its Status has an Active (In Home) icon. This is the current default module, and is the module displayed when a visitor comes to the home page `index.php`. Thus this module will be the first thing that a new visitor sees. Clicking the Put in Home icon  from the Functions column of a module will allow you to set that module as the default module.

If you attempt to set an inactive module as the home module, PHP-Nuke will kindly activate it for you. Note that the module will remain active should you then choose another module as your home module.

Time For Action—Activating Modules

Before we go any further, let's take a moment to activate some modules. We'll want to look at some of these modules in the next few chapters, the Content, Encyclopedia, FAQ, Forums, Members_List, and Reviews modules, so we may as well activate them now. We'll only go through activating the Content module; the other modules can be similarly activated.

1. Ensure that you are in the Module Administration area, by clicking the Modules icon in the Administration Menu.

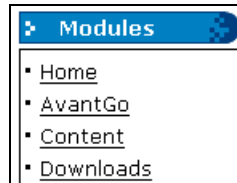
2. Scroll down the list of modules to find the Content module. It is currently marked as inactive in the list:

Content	Content		All Visitors	None	
Downloads	Downloads		All Visitors	None	

3. Click the Activate icon in the Functions column of the Content module. When the page reloads, you will see that the Content module is now marked as Active in the list:

Content	Content		All Visitors	None	
Downloads	Downloads		All Visitors	None	

4. The Content module is now showing in the Modules block:



That's all there is to activating an already installed module. Deactivating is similar; you simply click the deactivate icon in the Functions column. This icon is only shown when the module is active.

Editing Module Properties

The Top 10 module has a potentially confusing name. It isn't the list of top 10 singles or DVDs; it shows lists of the top 10 pieces of content on the site. Once you know what it shows, the title makes sense, but until then, it doesn't really. This kind of title won't help new visitors feel particularly comfortable navigating around your site. Let's start by getting that name changed to something that is more descriptive of its function.

In the Modules Administration table, find the module with the Title of Top, and click on the Edit icon in its Function column. You'll be presented with a list of options like this (remember to scroll down from the admin menu!):

Modules Edit

Change Module Name
(Top)

Custom Module Name:

Who can View This?

Visible in Modules Block? Yes No

[[Go Back](#)]

Here we can see that we can edit the Custom Module Name (its Custom Title in the modules table), which category of visitors is able to see the module, and also, if the module is to be Visible in the Modules block?.

Invisible Modules

If you were to set a module to be not visible in the Modules block, then it still remains active and usable. However, there is no link to the module in your main navigational device, the Modules block. You will have to direct visitors to the module by some other means. Modules like this are listed in the administrator's version of the Modules block, just above the list of inactive modules:

Invisible Modules
(Active but invisible link)

- *None*

These modules are called Invisible Modules. A rather colorful title, but it simply means that there is no explicit link to them in the list of active modules in the Modules block. Remember that an invisible module may still be accessible, and no extra security is placed upon the module. You might want to use this if you wanted a select group of people to preview a module before you unleashed it on the world, for example, and send them the URL to visit it directly. However, the 'no one will ever guess the name of that page' school of security is not a particularly advisable way of thinking.

Time For Action—Editing the Top 10

Now let's edit the properties of the Top 10 module. At the moment, you should still be on the Modules Edit page. If you're not, make your way there by clicking on the Modules icon in the Administration Menu, and in the table of modules, click the Edit icon in the Functions column of the module with the Title of Top 10.

Managing the Site

1. Change the Custom Module Name to Top 10 Bits of Content
2. Select All Visitors for Who Can View this?.
3. Leave Visible in Modules Block? set to Yes.
4. Click the Save Changes button.

After you click the button, the page will reload, and you will be back at the Modules Administration page. You will see the properties of the former Top 10 module have now changed in the modules table:

Top	Top 10 Bits of Content		All Visitors	None	  
-----	------------------------	---	--------------	------	---

And if you glance to the left of your screen to the Modules block, you will see the Top 10 entry has been replaced by our new title:

▪ Top 10 Bits of Content
--

That makes it so much clearer. Now everyone should understand exactly what that link is pointing to.

Adding New Modules





The standard installation of PHP-Nuke probably isn't going to do everything you want. Although that isn't part of its appeal, the fact that you can write your own modules and add them to your system is. That doesn't just mean you, but anybody with some PHP know-how. Besides, there is a vast library of PHP-Nuke modules available on the Internet.

Take due caution when downloading third-party modules, especially executable files, as they might not be thoroughly tested and it is possible that they contain spyware or viruses.

Generally, adding and installing third-party modules involves more work than we will do here, because there will often be database tables to be created, and files to be copied into different folders.

Time For Action—Installing a Simple Module

1. From the code download for this book, find the folder called MyModule in the ch04 folder.
2. Copy the MyModule folder into the modules folder of your PHP-Nuke installation.
3. Open your browser (make sure you are logged in as the super user) and click the Modules link in the Administration Menu.
4. Scroll down the list of modules to find the MyModule entry:

MyModule	MyModule		All Visitors	None	  
News	News	 	All Visitors	None	  

70

For More Information: www.packtpub.com/phpnuke/book
--

5. Click the **Activate** icon in the **Functions** column.
6. When the page reloads, **MyModule** will have appeared in the **Modules** block, and clicking its link will take you to the **MyModule** front page.

What Just Happened?

We just added a new module. The first stage was simple, copying the module files into the `modules` folder. A module is a set of files contained in a folder that goes in the `modules` folder, with PHP-Nuke picking up the name of the module from the name of the folder. For our module, it picks up the name `MyModule`.

Once the folder has been placed in the `modules` folder, the list of modules is refreshed by visiting the **Modules Administration** area. The list of modules is refreshed by PHP-Nuke going through all the folders in the `modules` folder to see if there are new folders added, and if so, adds them to the list of installed modules. By default, these newly found modules are not activated. After we clicked the **Activate** icon and the page reloaded, a glance over to the **Modules** block showed that the **MyModule** module has been registered as an inactive module:



This module now also appears in the list of modules in the **Module Administration** table, and from here, you can activate it or edit its properties like any other module.

Note that removing a module also works in the same way. To remove a module, simply remove its folder from the `modules` folder. When PHP-Nuke refreshes its list of modules, that module will be removed from the database.

In general, installing third-party modules involves more steps than we have seen here, particularly depending on the version of PHP-Nuke the module was originally written for. Usually, you will have to create some database tables for the module, since a module of any complexity will involve storing and retrieving data. (Our module has very little complexity!)

With third-party modules, there will often be instructions along with the module in the form of a `README` file or `INSTALL.txt` that provide specific guidelines on where to put various files, and the steps you must take to get the module up and running.

Summary

In this chapter we have had a lengthy tour of the PHP-Nuke administration interface, and accomplished some of the basic tasks of site management.

We looked first at the Web Site Configuration menu in the site Preferences. From here we set a number of global properties of the site, such as its name, its URL, and its description. It is important to set these early on when creating a new site, particularly since the URL of the site appears in emails generated by the system. There were a number of panels in the Web Site Configuration menu; we will encounter more of them as we move through the book.

Next we looked at creating a backup of the database. This is a very important task and should be performed regularly, since the PHP-Nuke database *is* your site. If something were to happen to it, you would lose all your site content, users, configuration settings, and so on, so taking regular backups will be important as soon as your site has any amount of traffic.

Block and module management were the next stops. We saw how to position and order blocks on the page, and how to set up the different types of blocks: static blocks of HTML stored in the database, RSS feeds from other sites, and PHP script blocks. The RSS block is particularly interesting, because it allows us to display details of content on other sites from the pages of our site.

Modules perform the main functionality of our site. We saw the Modules Administration area, from where we are able to activate or deactivate modules, or change some of their properties, such as which groups of users are able to access them.

Now it's time to move on to those whom your site is really for—the visitors.

Building Websites with PHP-Nuke

PHP-Nuke is a free tool to manage the content of dynamic websites, and allows you to create a dynamic, interactive website with minimum effort and programming knowledge.

As one of the most popular applications on the Internet, PHP-Nuke has grown into a complex, powerful tool with an extraordinary range of features, and a loyal community of supporters. Users can edit and manage their site through a web-based interface, without the need for any knowledge of web programming. PHP-Nuke is ideal for running a community-driven website, where visitors create accounts, contribute material, and interact with the site.

This book is packed with practical steps for you to learn how to build your own website with PHP-Nuke. From the basics of installing and configuring PHP-Nuke, you will learn how to manage your site, add content to it, and then customize its look.

What This Book Covers

The book begins with an overview of PHP-Nuke; what it is, and what it can do to help you build your own community-driven website. We shall learn all the features and functionality provided by PHP-Nuke by developing an example site, the Dinosaur Portal, as we progress through this book.

In *Chapter 2*, you will learn how to install and configure PHP-Nuke, apply patches, and also create the database. At the end of this chapter you will have a fully operational PHP-Nuke site, ready to go!

In *Chapter 3* you will begin to explore the many features of PHP-Nuke, create the super user (an all-powerful administrator of the site), and make your first modifications to the site. You will also learn about the ways in which PHP-Nuke restricts access to the site.

Chapter 4 introduces you to the administration area of PHP-Nuke, which allows you to manage your site from the comfort of your web browser. You'll see how to configure your site, back up the database, and manage blocks and modules.

Your site is created for visitors and *Chapter 5* shows you how to manage them. You will learn how to create users, explore the Your Account module, which is the user's private 'space', and set up other administrators to perform limited administrative tasks on the site.

Chapter 6 is where we really start adding content to the site. In this chapter, we have a thorough walkthrough of story management, from both the visitor and administrator viewpoint. A story is PHP-Nuke's most versatile type of content, and we explore how to add, edit, and manage stories, and the features that allow visitors to interact with the stories.

Chapter 7 covers some of the other standard modules for adding content in PHP-Nuke. The Content, FAQ, Encyclopedia, Web Links, Downloads, and Reviews modules are all covered in this chapter, from both the visitor and administrator viewpoint. Each of these modules handles a different type of content, and you will see the functionality they offer for visitors to interact with the content, and also how each module organizes its content.

For More Information: www.packtpub.com/phpnuke/book

PHP-Nuke has a fully featured discussion board module, the Forums module. In *Chapter 8*, you will take a tour of setting up a new forum, assigning moderators, and setting permissions to these forums.

Before *Chapter 9*, you've been working with the standard-looking PHP-Nuke site. In *Chapter 9*, it's time to unleash your creativity and create a new look for the site. This is done through customizing a PHP-Nuke theme. In this chapter, we race through a range of customizations, from changing the site logo, to adding and styling a navigation bar, creating new blocks, and changing the format of stories on the homepage, among others. This chapter features a lot of coding in HTML and CSS, in addition to some changes to PHP files, and there are clear instructions to help you through.

Chapter 10 shows you how to program PHP-Nuke to create new blocks and modules. You will see how PHP-Nuke handles page requests, data access, and language files. You will also see how code is organized in a module for both the visitor end and the administrator end.

Appendix A has a walkthrough of installing the XAMPP package, which provides a working installation of PHP, MySQL, and Apache, ready configured for you to test your PHP-Nuke site on.

Where to buy this book

You can buy *Building Websites with PHP-Nuke* from the Packt Publishing website:
<http://www.packtpub.com/phpnuke/book>.

Free shipping to the US, UK, Europe, Australia & New Zealand.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.



www.PacktPub.com

For More Information: www.packtpub.com/phpnuke/book