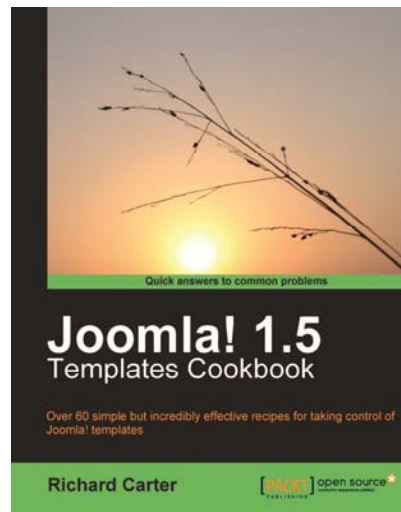


Joomla! 1.5 Templates Cookbook

Richard Carter



Chapter No.3 **"Theming the Details"**

In this package, you will find:

A Biography of the author of the book

A preview chapter from the book, Chapter NO.3 "Theming the Details"

A synopsis of the book's content

Information on where to buy this book

About the Author

Richard Carter started as a freelance web designer in Leicestershire, England. He founded his current business, Peacock Carter, in 2006. Currently, he is the senior frontend web developer and a managing director of Peacock Carter Ltd—a web design agency based in the North East of England with specialist knowledge in open source software for business and organizations across the world.

He has worked for clients including Directgov, NHS Choices, and BusinessLink.gov.uk. He tweets from twitter.com/RichardCarter and blogs at earlgreyandbattenburg.co.uk.

Richard is the author of *MediaWiki Skins Design* and *Magento 1.3 Theme Design*, and has also reviewed *MediaWiki 1.1 Beginner's Guide*.

I'd like to thank Michael, my business partner, for keeping the business running while I was busy writing and researching the book. A brief "hello" and thanks is also due to Alex, EJ, and Joy.

For More Information:

www.PacktPub.com/joomla-1-5-templates-cookbook/book

Joomla! 1.5 Templates Cookbook

With the widespread empowering of website owners being able to manage their own website's content, there are a huge number of content management systems available. Joomla! is one of the most popular of these content management systems with a large user base and active community who are constantly working to improve Joomla! for new and future users.

With the popularity of Joomla! and the relative lack of customized templates, there is much that can be done to change the appearance of your Joomla! website—from customizing the administration panel to creating print-friendly views of your website content and integrating popular applications such as Twitter and Facebook into your Joomla! website.

What This Book Covers

Chapter 1, Joomla! Theming Basics, covers changing the basics of your Joomla! templates, from the color scheme using template variations to your site's logo, and using the administration panel to edit your template's HTML and CSS.

Chapter 2, Custom Joomla! Templates, goes through more detailed templating tasks, from using CSS resets with your template to providing a guide to jdoc statements in your Joomla! templates.

Chapter 3, Theming the Details, covers styling search forms at both module and component view to pagination and beyond.

Chapter 4, Custom Page Templates, contains guides on creating custom error and site offline pages, and a walkthrough to module chrome.

Chapter 5, Styling Joomla! for Print, covers everything to do with print stylesheets, from typography to layout and troubleshooting common bugs in print CSS.

Chapter 6, Joomla! Admin Templates, contains guides to customizing Joomla!'s administration panel, from installing a new admin template to changing the admin panel's colors.

For More Information:

www.PacktPub.com/joomla-1-5-templates-cookbook/book

Chapter 7, Social Media and Joomla!, covers guides to integrating social media content such as videos from YouTube and content from Twitter into your Joomla! templates.

Chapter 8, Styling Joomla! for Mobiles, provides guides to styling your Joomla! website for mobile devices such as the iPhone, from creating handheld devices stylesheets in CSS to adding iPhone icons for your website.

Chapter 9, Joomla! and JavaScript, includes tips on minimizing page load time when using JavaScript in your Joomla! template and integrating Lightbox and other JavaScript features into your template.

Chapter 10, Miscellaneous Joomla! Templating, includes a miscellany of content for Joomla! template designers, from using conditional comments only for Internet Explorer CSS fixes to fixing Firefox/Mozilla bugs and more.

Appendix, Joomla! Output Overrides, covers the use of template overrides in Joomla!.

For More Information:

www.PacktPub.com/joomla-1-5-templates-cookbook/book

3

Theming the Details

This chapter looks at customizing your Joomla! template further, including:

- ▶ Styling the search module
- ▶ Styling the search component
- ▶ Using template overrides in your Joomla! template
- ▶ Customizing the breadcrumb
- ▶ Styling pagination
- ▶ Linking back to the top of your page
- ▶ Adding a random background image to your Joomla! template

Introduction

Although we've seen how to alter much of our Joomla! website, there's still much we can do to improve and polish our Joomla! template to perfection.

Styling the search module

Joomla! is a powerful content management system that is capable of supporting websites with hundreds and even thousands of pages. When websites become this large, it's often important to provide your website's visitors with a search feature as a means of locating the information on your website that they are looking for. One option that Joomla! provides for your visitors to search your website is the **search module**, which is a block displayed within your template.

For More Information:

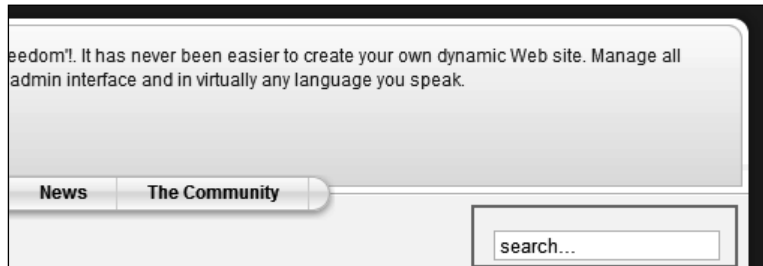
www.PacktPub.com/joomla-1-5-templates-cookbook/book

Getting ready

Identify the `class` or `id` assigned to your Joomla! template's search form, which is assigned by a `jdoc` include statement within your template's `index.php` file. In the *rhuk_milkyway* template—the one that we've been working with—the search feature is assigned to the `user4` block by default with this `jdoc` statement:

```
<jdoc:include type="modules" name="user4" />
```

It appears to the top-right of the template:



If we now look at the page's HTML source, the HTML generated by Joomla! for the search feature looks like this:

```
<div id="search">
  <form action="index.php" method="post">
    <div class="search">
      <input name="searchword" id="mod_search_searchword"
        maxlength="20" alt="Search" class="inputbox"
        type="text" size="20" value="search."
        onblur="if(this.value=='')this.value='search...';"
        onfocus="if(this.value=='search...') this.value='';" />
    </div>
    <input type="hidden" name="task" value="search" />
    <input type="hidden" name="option" value="com_search" />
    <input type="hidden" name="Itemid" value=1 />
  </form>
```

This means that we can apply CSS to `#search` to style our template's search box.

How to do it...

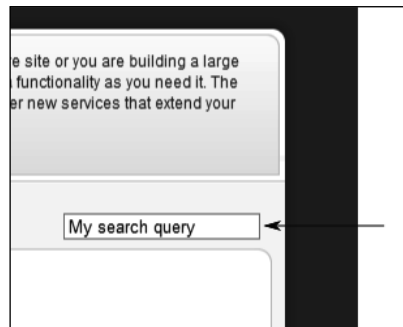
1. Open your template's primary stylesheet file, which is usually called `template.css`, and is located in the `templates\rhuk_milkyway\css\` directory of your Joomla! installation. The *rhuk_milkyway* template already defines the style for the form as follows:

```
#search {
  float: right;
  width: 320px;
  margin-top: -20px;
  margin-right: 30px;
  height: 40px;
  overflow: hidden;
  text-align: right;
}
```

- By adding CSS to change the search field's state when a visitor focuses within it, you can help improve your Joomla! template by orientating visitors to their whereabouts on the page:

```
#search input[type='text']:focus {
  border-color: #09C /* blue */
}
```

- Once you've uploaded the altered `template.css` file, you will now see a blue border surrounding the search field:



How it works...

By using the **CSS pseudo-class** `:focus`, the browser changes the attributes we define to make it clearer to our website's visitors that their input device (for example, keyboard) is focused on the search input field.



Internet Explorer versions 7 and below do not support the `:focus` pseudo-class. You can provide support in Internet Explorer for this feature of CSS with the use of JavaScript; see <http://james.padolsey.com/javascript/fixing-focus-in-internet-explorer/>.

For More Information:

www.PacktPub.com/joomla-1-5-templates-cookbook/book

See also

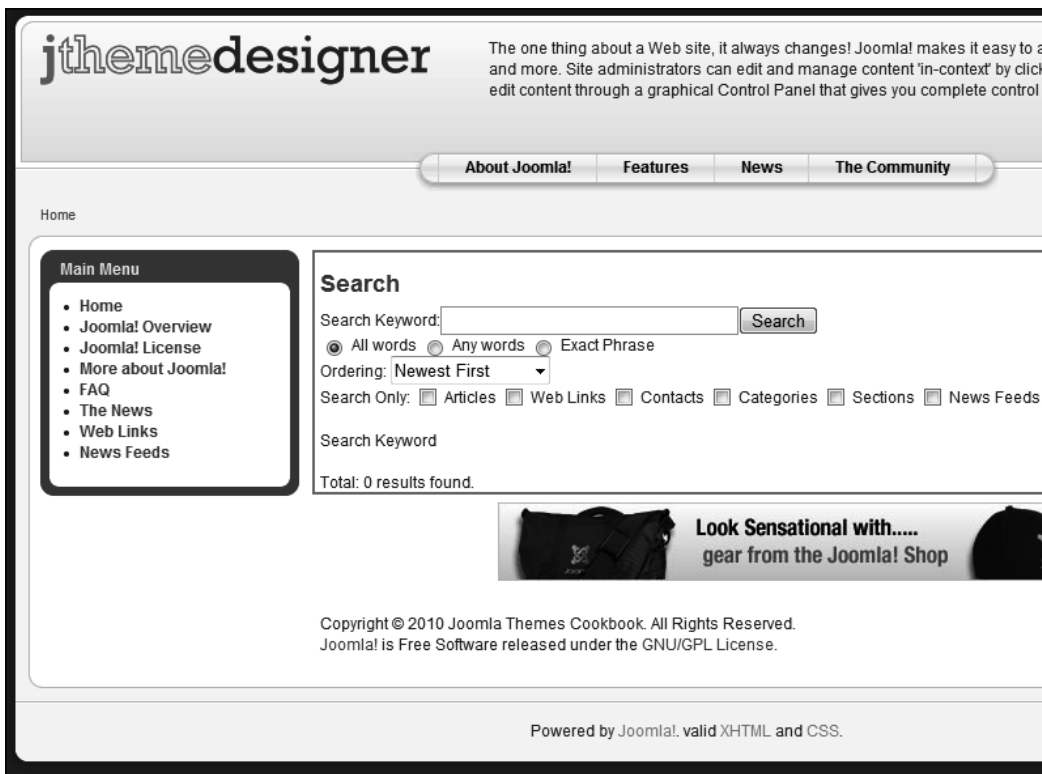
- ▶ *Understanding Joomla! template positions*
- ▶ *Styling the search component*

Styling the search component

Along with providing the search module, which is embedded within your Joomla! template depending on the module position it is assigned to, there is the Joomla! search component.

Getting ready

Firstly, you need to access the search component on your Joomla! website. You can do this by visiting http://example.com/index.php?option=com_search, assuming that your Joomla! installation is installed in the root directory of the `example.com` domain. With the `rhuk_milkyway` template as your currently enabled template, you should see that the search component looks like this:



Open your template's primary CSS file; for our example, this is `templates\rhuk_milkyway\css\template.css`. It is also worth studying the source of the search component page; you'll find that the search form is contained within a `<form>` element identified with an `id` of `searchForm`.

How to do it...

1. In your template's CSS file (`template.css`), begin by styling the overall form first:

```
form#searchForm {
background: #E5F1FD;
border: 1px #0C3A6D solid;
border-radius: 10px;
padding: 10px
}
```



Some browsers do not yet support the `border-radius` property in CSS, so you may just see the search form with squared corners.

This changes the look of the search form as follows:

Search

Search Keyword:

All words
 Any words
 Exact Phrase

Ordering: ▼

Search Only:
 Articles
 Web Links
 Contacts
 Categories
 Sections
 News Feeds

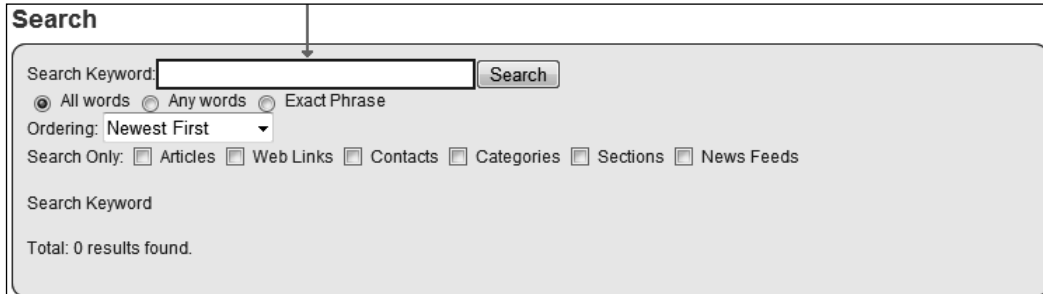
Search Keyword

Total: 0 results found.

2. Next, you'll style the search query field, which is identifiable by the

```
#search_searchword id:
#searchForm #search_searchword {
border: 2px #0C3A6D solid;
color: #0C3A6D
}
```

This helps to distinguish the search field from the other fields in the form:

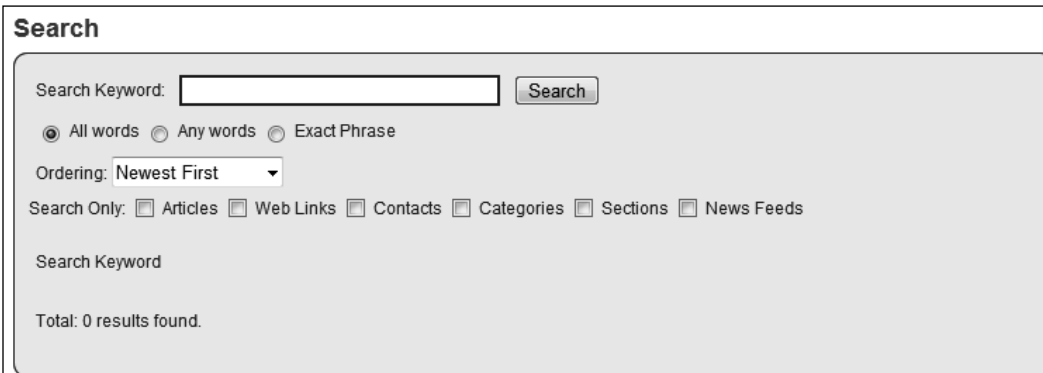


The screenshot shows a search form titled "Search". It contains a "Search Keyword:" input field, a "Search" button, and several options: radio buttons for "All words" (selected), "Any words", and "Exact Phrase"; a dropdown menu for "Ordering:" set to "Newest First"; checkboxes for "Search Only:" including "Articles", "Web Links", "Contacts", "Categories", "Sections", and "News Feeds"; a "Search Keyword:" label; and a "Total: 0 results found." message. A box highlights the search input field, and an arrow points to it from above.

3. Lastly, you'll add some padding to the table cells used to lay out the search component form to provide a little more space between inputs to prevent visitors accidentally clicking:

```
#searchForm table td {  
padding: 5px  
}
```

That's the search form styled!



The screenshot shows the same search form as above, but now with rounded corners and a light gray background. The search input field is now a simple text box with a "Search" button to its right. The form is styled to be more visually appealing and user-friendly.

How it works...

By applying CSS to the relevant elements generated by Joomla!, it's possible to redefine what the search form provided by Joomla!'s search component looks like.

See also

- ▶ *Styling the search module*
- ▶ *Using template overrides in your Joomla! template*

Using template overrides in your Joomla! template

You can use template overrides in your Joomla! template to redefine the output HTML that Joomla! creates for the modules and components on your installation. This can be useful when you are unable to achieve the style you want for your Joomla! template with the markup provided in the base template that you are using.

Getting ready

In this example, you'll be changing the search component's **Search** button, as displayed at http://example.com/index.php?option=com_search, assuming that you have Joomla! installed at example.com:



One such thing that you may want to change is the **Search** button. In this example, we'll change it to a `<input type= "image" />` so that we're able to use an image for the **Search** button rather than the browser's default rendering of the `<button>` element that we can see in the previous screenshot.

How to do it...

1. Save the search button graphic that you want to use in place of the **Search** button in the `templates\rhuk_milkyway\images` directory as `search.png`:



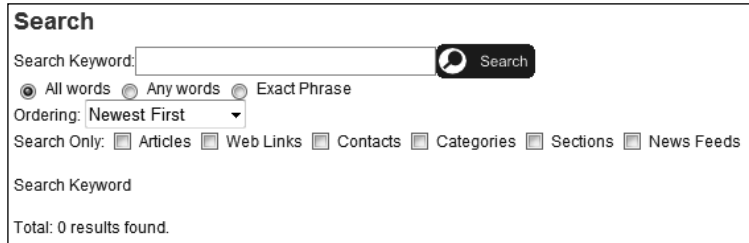
2. Next, you need to copy the contents of the `default_form.php` file located in the `components\com_search\views\search\tmpl` directory of your Joomla! installation into the `templates\rhuk_milkyway\html\com_search\views\search` directory. You may need to create the `com_search` subdirectory and its child directories (`view` and `search`) if they do not already exist. Locate the line that reads:

```
<button name="Search" onclick="this.form.submit()"
        class="button"><?php echo JText::_('Search');?>
</button>
```

3. Replace this line with:

```
<input type="image" name="Search" onclick="this.form.submit()"
      src="<?php echo $this->baseurl ?>templates/rhuk_milkyway/images/
      search.png"
      alt="<?php echo JText::_('Search');?>" />
```

4. Your new search component form now looks like this:



How it works...

By default, Joomla! looks in a component's own directory within the currently enabled Joomla! template for a custom template override. If Joomla! fails to find this file, it looks in the `\components` directory for the relevant file with which to generate the required HTML by the component.

See also

- ▶ *Styling the search module*
- ▶ *Styling the search component*

Customizing the breadcrumb

The larger your website gets, the more important it is to make use of Joomla!'s breadcrumb feature.

Getting ready

To start redefining your breadcrumb's style, open the `template.css` file for your template; use the `rhuk_milkyway` template for this demonstration. This means that your CSS file will be located in the `templates\rhuk_milkyway\css` directory of your Joomla! installation. If you visit a page other than the home page in your Joomla! website, you'll be able to see the breadcrumb.



As you can see, the *rhuk_milkyway* template defines the style for the breadcrumb in the `template.css` file.

```
span.pathway {
  display: block;
  margin: 0 20px;
  height: 16px;
  line-height: 16px;
  overflow: hidden;
}
```

The HTML that defines the breadcrumb for the **Features** page is as shown:

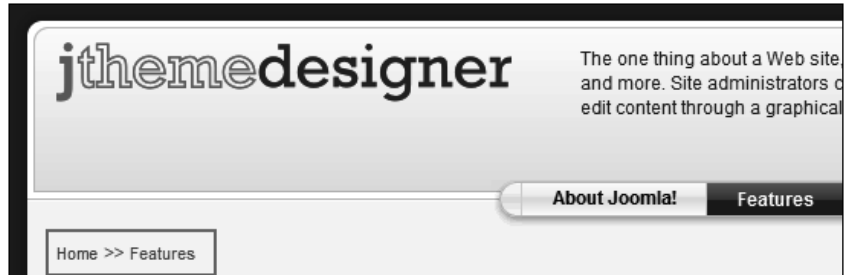
```
<div id="pathway">
  <span class="breadcrumbs pathway">
    <a href="http://example.com/" class="pathway">Home</a>
    
    Features
  </span>
</div>
```

How to do it...

1. You can customize the breadcrumb by changing the CSS, and altering the color and size of the breadcrumb's content:

```
span.pathway {
  color: #666;
  font-size: 90%;
  display: block;
  margin: 0 20px;
  height: 16px;
  line-height: 16px;
  overflow: hidden;
}
```

2. Once the altered CSS file has been uploaded, you can see your changes:



3. The next step to customizing your breadcrumb is to alter the image used for the separator arrows, located at `\templates\rhuk_milkyway\images\arrow.png`. You'll replace this image with your own new one (which has been enlarged in this image to make it easier to view).



4. Once uploaded, your new breadcrumb looks a little more fitting for your website:



How it works...

By targeting specific `ids` and `classes` with CSS and changing an image in the `images` directory of our template, we can subtly change our template to distinguish it from others without a great deal of work.

See also

- ▶ *Styling the search module*
- ▶ *Styling pagination*

Styling pagination

Some content in your Joomla! website may run over multiple pages (for example, some search results). By styling pagination, you can again help to distinguish your Joomla! template from others.

Getting ready

Open your template's primary stylesheet; generally, this will be called `template.css`, and is located in the `templates\rhuk_milkyway\css\` directory if we are using the `rhuk_milkyway` template (as we are for this demonstration).

It is also worth bearing in mind the typical structure of the pagination feature within the HTML. We can find this by searching for a common word such as "the" or "Joomla!" on our website.

```
<span class="pagination">

    <span>&laquo;</span>
    <span>Start</span>
    <span>Prev</span><strong>
    <span>1</span></strong>

    <strong>
    <a href=" index.php?searchword=Joomla&searchphrase=all&
      Itemid=1&option=com_search&limitstart=20" title="2">2</a>
    </strong>

    <strong>
    <a href=" index.php?searchword=Joomla&searchphrase=all&
      Itemid=1&option=com_search&limitstart=40" title="3">3
    </a></strong>

    <a href=" index.php?searchword=Joomla&searchphrase=all&
      Itemid=1&option=com_search&limitstart=20" title="Next">
    Next </a>

    <a href=" index.php?searchword=Joomla&searchphrase=all&
      Itemid=1&option=com_search&limitstart=40"
      title="End">End</a>

    <span>&raquo;</span>

</span>
```

Our primary interest in the previous part is the `.pagination` class assigned to the `` element that contains the pagination feature's content. By default, the pagination (as seen on the search results page) looks like this:

« Start Prev 1 2 3 Next End »

How to do it...

1. Now that you are aware of the relevant class to style, you can add it to your template's stylesheet, with the aim of making the pagination less obtrusive with the surrounding content of your pages:

```
.pagination {
  color: #666;
  font-size: 90%
}

.pagination a {
  color: #F07 !important /* pink */
}
```

2. Once you've uploaded the newer stylesheet, you'll be able to see the new pagination style, which will appear smaller than before, and with pink-colored links.

« Start Prev 1 2 3 Next End »

Producing more semantic markup for pagination

As you can see in the previous section, the HTML that Joomla! currently generates for the pagination feature is quite verbose—unnecessarily long and untidy. We'll change our template's `pagination.php` file to use more semantic (meaningful) HTML for this feature by adding each item to a list item within an unordered list element (``).

The HTML for pagination is defined within the `pagination.php` file, located in the `html` directory of your template. If you edit this file, remember that you need to escape certain characters such as double quotation marks, that is, `<div class="list-footer">` should read `<div class=\"list-footer\">`. You can mitigate the need to do this by using single quotation marks (`'`) rather than double quotation marks (`"`).



Other templates using semantic markup

The **Beez template** that comes with the default Joomla! installation already provides semantically correct markup for pagination.

1. Open the `pagination.php` file and you will see four PHP functions (assuming that you are looking within the `rhuk_milkyway` template), but the function which is of interest to us is the `pagination_list_render` PHP function. Currently, the code for this function looks like this:

```
function pagination_list_render($list)
{
    // Initialize variables
    $html = "<span class=\"pagination\">";
    $html .= '<span>&laquo;</span>'. $list['start']['data'];
    $html .= $list['previous']['data'];

    foreach( $list['pages'] as $page )
    {
        if($page['data']['active']) {
            $html .= '<strong>';
        }

        $html .= $page['data'];

        if($page['data']['active']) {
            $html .= '</strong>';
        }
    }

    $html .= $list['next']['data'];
    $html .= $list['end']['data'];
    $html .= '<span>&raquo;</span>';

    $html .= "</span>";
    return $html;
}
```

2. You can see that Joomla! builds up the HTML to insert into the page by using the `$html` PHP variable. All you need to change is the HTML you can see:

```
function pagination_list_render($list)
{
    // Initialize variables
    $html = "<ul class=\"pagination\">";
    $html .= '<li class="page-previous">&laquo;</li>' . '<li>' .
        $list['start']['data'] . '</li>';
    $html .= '<li>' . $list['previous']['data'] . '</li>';

    foreach( $list['pages'] as $page )
    {
```

```

        if($page['data']['active']) {
            $html .= '<li>';
        }

        $html .= '<strong class="active">' . $page['data'] .
            '</strong>';

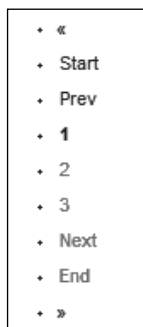
        if($page['data']['active']) {
            $html .= '</li>';
        }
    }

    $html .= '<li>' . $list['next']['data'] . '</li>';
    $html .= '<li>' . $list['end']['data'] . '</li>';
    $html .= '<li class="page-next">&raquo;</li>';

    $html .= "</ul>";
    return $html;
}

```

3. If you now upload the `pagination.php` file and refresh the page, you'll see that the previous style that you had defined only partially styles the newer HTML:



4. If you add the following CSS to your template's `template.css` file, everything will be styled as you intended before:

```

ul.pagination {
    list-style-type: none
}

ul.pagination li {
    display: inline
}

```

5. Once uploaded, your new pagination is complete:

```
« Start Prev 1 2 3 Next End »
```

How it works...

By applying CSS to the relevant `ids` and `classes` specified in the HTML for Joomla!'s pagination feature, it's possible to quite drastically alter the pagination's appearance for your Joomla! template. It's also possible to change the HTML generated by Joomla! in some circumstances, as we saw with our template's `pagination.php` file. One benefit of changing the HTML that Joomla! outputs is that you can add `classes` and `ids` that make it easier to style elements of your Joomla! website with CSS.

See also

- ▶ *Customizing the breadcrumb*

Linking back to the top of your page

If your website contains a number of very long articles or pieces of content, it's wise to include a **back to the top** link at the bottom of each page.

Getting ready

Open your Joomla! template's `index.php` file. For the purposes of this example, we'll be using the *rhuk_milkyway* template, though the technique can be applied to any Joomla! template.

How to do it...

1. Search for the following snippet of code in your template:

```
<table class="nopad">
<tr valign="top">
  <td>
    <jdoc:include type="component" />
    <jdoc:include type="modules" name="footer" style="xhtml"/>
  </td>
```

2. Create an anchor called `top` above the line where the statement `<jdoc:include type="component" />` is included in your page:

```
<table class="nopad">
<tr valign="top">
  <td>
    <a name="top">&nbsp;</a>
    <jdoc:include type="component" />
    <jdoc:include type="modules" name="footer" style="xhtml"/>
  </td>
```

3. Beneath the `include` statement that includes the content, now add the link to the top of the page:

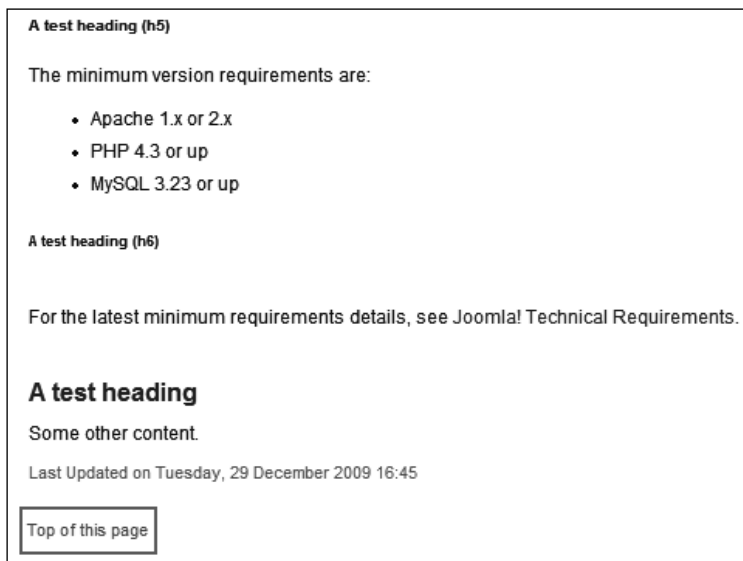
```
<table class="nopad">
<tr valign="top">
  <td>
    <a name="top">&nbsp;</a>
    <jdoc:include type="component" />
    <p class="top-of-page">
      <a href="#top" title="Top of this page">Top of this page</a>
    </p>
    <jdoc:include type="modules" name="footer" style="xhtml"/>
  </td>
```

4. You can now add a little style for your link by defining CSS for `p.top-of-page` in your template's `template.css` file:

```
p.top-of-page {
  color: #666;
  font-size: 90%;
}

p.top-of-page a {
  color: #09C /* blue */
}
```

- Once the changed files have been uploaded, you can see your link appear towards the bottom of every page in your Joomla! website:



How it works...

The `<jdoc:include type="component" />` statement inserts the page's content into our Joomla! template, so this is where we want to allow our website's visitors to skip back to the top. Giving the value `#top` for the `href` attribute in the `link` element (inserted below the content) creates a link to an anchor called `top` within the page, which we create above the content block in the form ` `.

There's more...

There is an extension available for Joomla! that provides similar functionality; it's called **Return to Top for Content Items**, and is available from the Joomla! website at <http://extensions.joomla.org/extensions/structure-a-navigation/site-navigation/5617>.



Downloading the extension requires you to be logged into the extension developer's website.

Rather than inserting a **Return to top** link in every page on your website, it allows you to insert these links wherever you need to within your content by inserting `{rt}` into your page's content.

However, adding the link to your template is better for your website's performance, as every module or plugin that your Joomla! website has enabled will slow down the loading time of the website for your visitors.

See also

- ▶ *Styling the search module*
- ▶ *Styling pagination*

Adding a random background image to your Joomla! template

In distinguishing your Joomla! template from others, there are a number of extensions for Joomla! to help you, including one that allows you to display a random image as your template's background image for the <body> element.

Getting ready

You need to install the extension called **Random Background**. You can find the file's download link on the Joomla! website at <http://extensions.joomla.org/extensions/style-a-design/templating/6054>. Once you have saved the extension files somewhere on your computer, log in to your website's Joomla! administration panel (if Joomla! is installed at `example.com`, the administration panel is typically accessible at `example.com/administrator`), and select the **Install/Uninstall** option from the **Extensions** option in the primary navigation:

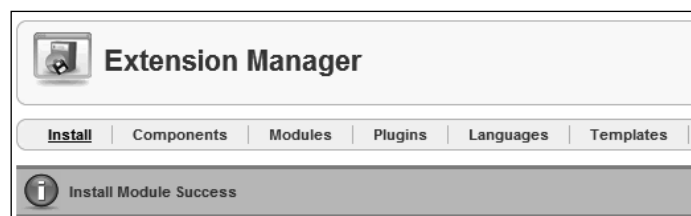


You will then be presented with a form, from where you can upload the extension's .zip file. Select the file from your computer, and then click on the **Upload File & Install** button:



The screenshot shows the Joomla! Extension Manager interface. At the top, there is a navigation menu with tabs for Site, Menus, Content, Components, Extensions, Tools, and Help. Below this, the 'Extension Manager' section is active, with sub-tabs for Install, Components, Modules, Plugins, Languages, and Templates. The 'Upload Package File' section is highlighted, showing a text input field for 'Package File' containing the path 'c:\program_data\joomla\components\mod_background_1.5.x_1.4.zip' and a 'Choose...' button. To the right of the input field is a button labeled 'Upload File & Install'. Below this, the 'Install from Directory' section shows an 'Install Directory' field with the path 'public/joomla/tmp' and an 'Install' button. The 'Install from URL' section shows an 'Install URL' field with the text 'http://' and an 'Install' button.

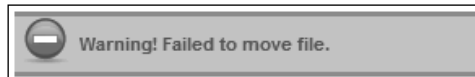
Once complete, you should receive a confirmation message:



The screenshot shows the Joomla! Extension Manager interface after a successful installation. The 'Install' sub-tab is active. A confirmation message is displayed at the bottom of the page, featuring an information icon and the text 'Install Module Success'.

Setting relevant permissions for installing the module

If you have problems installing the module, you may receive an error message like the following one:



The error is most likely because two directories on your server do not have sufficient permissions:

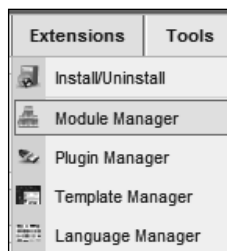
- ▶ \tmp
- ▶ \modules

Use Joomla!'s FTP layer to manage the necessary file permissions for you. You can edit Joomla!'s configuration file, which is called `configuration.php`, in the root of your Joomla! website. Simply add these variables into the file if they don't exist already:

```
var $ftp_host = ''; // your FTP host, e.g. ftp.example.com or just
example.com, depending on your host
var $ftp_port = ''; // usually 21
var $ftp_user = ''; // your FTP username
var $ftp_pass = ''; // your FTP password
var $ftp_root = ''; // usually / or the directory of your Joomla!
install
var $ftp_enable = '1'; // 1 = enabled
```

How to do it...

1. Firstly, you need to enable the extension from Joomla!'s **Module Manager** in the administration panel:



- From the list of available extensions on your website, you'll need to check the box next to the **Background** extension:

Module Manager

Enable
 Disable

Site | Administrator

Filter: - Select Template - ▾ - Select Position

#	<input type="checkbox"/>	Module Name	Enabled	Order	Access Level
1	<input type="checkbox"/>	Breadcrumbs	✓	<input type="text" value="1"/>	Public
2	<input type="checkbox"/>	Banners	✓	<input type="text" value="1"/>	Public
3	<input type="checkbox"/>	Footer	✓	<input type="text" value="2"/>	Public
4	<input type="checkbox"/>	testmenu	✘	<input type="text" value="1"/>	Public
5	<input type="checkbox"/>	Main Menu	✓	<input type="text" value="2"/>	Public
6	<input type="checkbox"/>	Resources	✓	<input type="text" value="3"/>	Public
7	<input type="checkbox"/>	Key Concepts	✓	<input type="text" value="4"/>	Public
8	<input type="checkbox"/>	User Menu	✓	<input type="text" value="5"/>	Registered
9	<input type="checkbox"/>	Example Pages	✓	<input type="text" value="6"/>	Public
10	<input type="checkbox"/>	Statistics	✘	<input type="text" value="7"/>	Public
11	<input type="checkbox"/>	Login Form	✓	<input type="text" value="8"/>	Public
12	<input type="checkbox"/>	Archive	✘	<input type="text" value="9"/>	Public
13	<input type="checkbox"/>	Sections	✘	<input type="text" value="10"/>	Public
14	<input type="checkbox"/>	Related Items	✘	<input type="text" value="11"/>	Public
15	<input type="checkbox"/>	Wrapper	✘	<input type="text" value="12"/>	Public
16	<input type="checkbox"/>	Feed Display	✘	<input type="text" value="13"/>	Public
17	<input type="checkbox"/>	Background	✘	<input type="text" value="14"/>	Public

- Now click on the **Background** link to view the extension's settings. Notice that the extension needs to be assigned to a position within your template to function.

The screenshot shows the Joomla! Module Settings interface for a module of type 'mod_background'. The title is 'Background'. The 'Show Title' and 'Enabled' options are both set to 'Yes'. The 'Position' is set to 'left' and the 'Order' is '14::Background'. The 'Access Level' is set to 'Public'.

Module Type:	mod_background
Title:	Background
Show Title:	<input type="checkbox"/> No <input checked="" type="checkbox"/> Yes
Enabled:	<input type="checkbox"/> No <input checked="" type="checkbox"/> Yes
Position:	left
Order:	14::Background
Access Level:	Public

- Towards the right of the screen, there are options for the extension. Of particular interest to us is the value of the **Choose random image** option that we need to change to **Yes**.

The screenshot shows the Joomla! Module Parameters interface for the 'Background' module. The 'Choose random image' option is set to 'Yes'. The 'Background image' is set to '- None Selected -'. The 'Images directory' is '/images/stories'. The 'Additional CSS' field contains the following code: `background-repeat: no-repeat !important;`

Module Class Suffix	
Background color	#ffffff
ID/Tag	
Background image	- None Selected -
Choose random image	<input type="checkbox"/> No <input checked="" type="checkbox"/> Yes
Images directory	/images/stories
Additional CSS	background-repeat: no-repeat !important;

- Once the extension has been installed and enabled, you can upload the images you want to appear as background images on your Joomla! website by using Joomla!'s **Media Manager** feature. You can also customize the CSS in the **Additional CSS** field to allow the background image to repeat either vertically (`background-repeat: repeat-y`) or horizontally (`background-repeat: repeat-x`). When you refresh the frontend of your Joomla! website (that is, not the administration panel), you'll see a random image appear at the top-left of the template.



See also

- ▶ *Styling the search module*
- ▶ *Styling pagination*

Where to buy this book

You can buy Joomla! 1.5 Templates Cookbook from the Packt Publishing website:
<https://www.packtpub.com/joomla-1-5-templates-cookbook/book>

Free shipping to the US, UK, Europe and selected Asian countries. For more information, please read our [shipping policy](#).

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/joomla-1-5-templates-cookbook/book