

Module Hello World 1

Last Updated Wednesday, 11 July 2007

IntroductionThis tutorial aims to give you a grounding in the basic concepts for writing Joomla! modules. It will develop a very simple Hello World module and then extend it using patTemplate, a powerful set of functions that help separate the presentation (HTML) from the application logic (PHP) using templates. Full documentation for patTemplate can be found on their website.

RequirementsYou need for this tutorial:

- Joomla! 1.0 or greater

Let's RollWe will be creating two files in this tutorial in the folder called /modules/. Let's look at the files we need.

mod_helloworld.php

This file is the actually engine for the module.

mod_helloworld.xmlThis

file is the XML setup file for the module. It defines the information required for the module to be installed.

Installing the Basic ModuleYou cannot create a module from scratch from the Joomla! Administrator, so we have to make some basic files first and then

install them as a module.

Let's make the actual module first. Save the following code as mod_helloworld.php.

```
<?php
/**
 * @version 1.0 $
 * @package HelloWorld
 * @copyright (C) 2005 Andrew Eddie
 * @license http://www.gnu.org/copyleft/gpl.html GNU/GPL
 */

/** ensure this file is being included by a parent file */
defined( '_VALID_MOS' ) or die( 'Direct Access to this location is not allowed.' );

?>
<h1>Hello World</h1>
```

Next save the following code in a file named mod_helloworld.xml.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<mosinstall type="module" version="4.5.2">
  <name>Hello World</name>
  <author>Andrew Eddie</author>
  <creationDate>February 2005</creationDate>
  <copyright>(C) 2005 Andrew Eddie</copyright>
  <license>http://www.gnu.org/copyleft/gpl.html GNU/GPL</license>
  <version>1.0</version>
  <description>A module that says hello</description>
  <files>
    <filename module="mod_helloworld">mod_helloworld.php</filename>
  </files>
  <params />
</mosinstall>
```

The first line of the file is a definition statement. You do not have to worry to much about what it means, but it must be in the file and it must be the first line (there cannot be any spaces before it). The other tags mean:

mosinstall

This is the parent tag that defines the rest of the installer file for Joomla!. It has an attribute for type which in this case is module. It also takes a value for the version of Joomla! it can run on. name This is the name of your module. author This is the name of the author for the module. creationDate This is the date the module was created. copyright This is the copyright holder of the module's code. license This is the name of, or a reference to, the license under which the module is released. version This is the version of the module. description This is a free text description of the module. files This is a collection of the files included with the module. filename This is a file that is used by the module. Any number of files can be listed, including files in a subdirectory. The file that Joomla! calls to invoke the module must contain the module attribute that takes a value of the name of the file without the .php extension.

Now, zip these two files up into a file called mod_helloworld.zip or you can download a copy here (mod_helloworld.zip).

Follow these instructions to install the basic module:

- Log into the Joomla! Administrator.
 - Select Installers -> Modules from the menu.
 - In the Upload Package File area, click the Browse button and select the zip file you just created or downloaded. Then click the Upload File and Install button.
- If all goes well, you should now see a message indicating a successful install. Click on the Continue link.

While you are still logged into the Joomla! Administrator, select Modules -> Site Modules from the Menu. Scroll down until you see the listing for Hello World. You will see that the module is unpublished and assigned to the left position in the template. Click on the red X in the published column to publish the module.

Preview your site. You should see your module saying hi to you.

Congratulations, you have built and deployed your first module. Now that it is installed we can modify the files directly to add more features.

Improving the Presentation

You will find that your new module has been installed in the /modules directory of your site. First we will separate the presentation layer from the module file.

Create a new directory under /modules called /mod_helloworld. In this new directory create a file called default.html. Copy the following code in this file:

```
<mos:comment>
@version 4.5.2
@package HelloWorld
@copyright (C) 2005 Andrew Eddie
@license http://www.gnu.org/copyleft/gpl.html GNU/GPL
</mos:comment>
<mos:tmpl name="helloworld">
  <h1>Hello World</h1>
  The time is {TIME}
</mos:tmpl>
```

We have added a little extra code here to demonstrate how to add a variable to your module template. You will also notice that the HTML is wrapped in a mos:tmpl tag. This defines a template and we have given the template the name of helloworld.

Now, find mod_helloworld.php in the /modules directory and open it in your editor.

Tip: There are many quality editors that are available for free, PSPad and HTML-Kit to name a few. For something a little more powerful, you might like to try out Eclipse.

Delete the existing code and replace it with the following:

```
<?php
```

```

/**
 * @version 1.0
 * @package HelloWorld
 * @copyright (C) 2005 Andrew Eddie
 * @license http://www.gnu.org/copyleft/gpl.html GNU/GPL
 */

/** ensure this file is being included by a parent file */
defined( '_VALID_MOS' ) or die( 'Direct Access to this location is not allowed.' );

// load the patTemplate library
require_once( $mosConfig_absolute_path . '/includes/patTemplate/patTemplate.php' );

// create the template
$tpl =& patFactory::createTemplate( "", false, false );

// set the path to look for html files
$tpl->setRoot( dirname( __FILE__ ) . '/mod_helloworld' );

$tpl->readTemplatesFromInput( 'default.html' );

$tpl->addVar( 'helloworld', 'time', date( 'H:i:s' ) );

$tpl->displayParsedTemplate( 'helloworld' );
?>Let's examine what is happening in this file.

```

- The comment block at the top of the file defines meta data about the file, in particular the license and the copyright. This block has some special notations that are able to be parsed by an application called phpDocumentor. It is used to assemble API (Application Programming Interface) documentation. The important thing here is to have a version and explicitly state the copyright and license terms for the file.

- Next we look for a constant called `_VALID_MOS`. Because this constant is defined elsewhere in Joomla!, it ensures that only Joomla! is able to access this file. It prevents a user from opening the file explicitly in a browser.

- We then include the patTemplate library.

- The patTemplate object is created and then we set the root directory for template (HTML) files to the /mod_helloworld directory in your module.

- Because we have set the root directory, we can just read in the default.html file with the readTemplatesFromInput method.

- Next we want to give the {TIME} variable in our template a value. To do this we use the addVar method of the template object. The method takes three arguments. The first is the name of the template, the second is the name of the template variable and the last is the actual value to assign to it.

- Last of all we display the helloworld template.

Save all your files and refresh your browser. You should see that the module now displays the time.

You can download the files for the final part of the tutorial here (mod_helloworld_1.zip). Note that the XML file has been updated to include the new HTML file.

To suggest a change to this page please [click here](#).