Models

Overview

The concept of model gets its name because this class is intended to represent (or 'model') some entity.

Creating A Model

All Joomla! models extend the JModel class. The naming convention for models in the Joomla! framework is that the class name starts with the name of the component, followed by 'model', followed by the model name. Therefore, our model class is called HelloModelHello.

```
<?php
// No direct access
defined( '_JEXEC' ) or die( 'Restricted access' );

jimport( 'joomla.application.component.model' );

/**
 * Hello Model
 */
class HelloModelHello extends JModel
{
    /**
    * Gets the greeting
    * @return string The greeting to be displayed to the user
    */
    function getGreeting()
    {
        return 'Hello, World!';
    }
}
```

You will notice a line that starts with jimport. The jimport function is used to load files from the Joomla! framework that are required for our component. This particular statement will load the file /libraries/joomla/application/component/model.php. The '.'s are used as directory separators and the last part is the name of the file to load. All files are loaded relative to the libraries directory. This particular file contains the class definition for the JModel class, which is necessary because our model extends this class.

Using A Model

The Joomla! framework is setup in such a way that the controller will automatically load the model that has the same name as the view and will push it into the view. We can easily retrieve a reference to our model using the JView::getModel() method. If the model had not followed this convention, we could have passed the model name to JView::getModel().

Here's an example of using a model with our Hello component (com_hello).

```
<?php
// No direct access
defined( '_JEXEC' ) or die( 'Restricted access' );

jimport( 'joomla.application.component.view');

/**
 * HTML View class for the HelloWorld Component
 *
 * @package    HelloWorld
 */

class HelloViewHello extends JView
{
 function display($tpl = null)
 {
  $model = &$this->getModel();
  $greeting = $model->getGreeting();
  $this->assignRef( 'greeting', $greeting );

  parent::display($tpl);
 }
}
```