Using decorators

In Yii, we can enclose content into a decorator. The common usage of decorators is layout. When you are rendering a view using the render method of your controller, Yii automatically decorates it with the main layout. Let’s create a simple decorator that will properly format quotes.

Getting ready

Create a new application using the Composer package manager, as described in the official guide at <http://www.yiiframework.com/doc-2.0/guide-start-installation.html>.

How to do it...

1. First, we will create a decorator file, @app/views/decorators/quote. php:

<div class="quote">

<h2>&ldquo;<?= $content?>&rdquo;, <?= $author?></h2>

</div>

1. Now, replace the content of @app/views/site/index. php with the following code:

<?php

use yii\widgets\ContentDecorator;

/\* @var \*/

?>

<?php ContentDecorator::begin([

'viewFile' => '@app/views/decorators/quote.php',

'view' => $this,

'params' => ['author' => 'S. Freud']

]

);?>

Time spent with cats is never wasted.

<?php ContentDecorator::end();?>

1. Now, your Home page should look like the following:

How it works.

Decorators are pretty simple. Everything between contentDecorator: : begin() and contentDecorator: : end() is rendered into a $content variable and passed into a decorator template. Then, the decorator template is rendered and inserted in the place where ContentDecorator: :end() was called.

We can pass additional variables into the decorator template using a second parameter of ContentDecorator : : begin(), such as the one we did for the author.

Note that we have used @app/views/decorators/quot e. php as the view path.

See also

* The <http://www.yiiframework.com/doc-2.0/yii-widgets-contentdecorator.html>URL provides more details about decorators:
* The Using the controller context in a view recipe