Creating helpers

There are a lot of built-in framework helpers such as stringHelper in the yii\helpers namespace. These contain sets of helpful static methods for manipulating strings, files, arrays, and other subjects.

In many cases, for additional behavior you can create a own helper and put any static function into one. For example, we implement the number helper in this recipe.

Getting ready

Create a new yii2 -app-basic application using the composer package manager as described in the official guide at [http://www.yiiframework.com/doc-2.0/guide-start-insta]lation.html](http://www.yiiframework.com/doc-2.0/guide-start-installation.html).

How to do it...

1. Create the helpers directory in your project and write the NumberHelper class:

<?php

namespace app\helpers;

class NumberHelper {

public static function format($value, $decimal = 2)

{

return number\_format($value, $decimal, '.', ',');

}

}

2. Add the actionNumbers method to SiteController:

<?php...

class SiteController extends Controller {

public function actionNumbers()

{

return $this->render('numbers', ['value' => 18878334526.3]);

}

}

3. Add the views/site/numbers. php view:

<?php

use app\helpers\NumberHelper; use yii\helpers\Html;

/\* @var $this yii\web\View \*/

/\* @var $value float \*/

$this->title = 'Numbers';

$this->params['breadcrumbs'][] = $this->title;

?>

<div class="site-numbers">

<h1><?= Html::encode($this->title) ?></h1> <p>