Analyzing the Yii error stack trace

When an error occurs, Yii can display the error stack trace along with the error. A stack trace is especially helpful when we need to know what really caused an error rather than just the fact that an error occurred.

Getting ready

1. Create a new yii2 -app-basic application by using the Composer package manager, as described in the official guide at [http://www. yiiframework. c om/doc-2.0/guide -start-installation .html](http://www.yiiframework.com/doc-2.0/guide-start-installation.html).
2. Configure a database and import the following migration:

<?php

use yii\db\Migration;

class m160308\_093234\_create\_article\_table extends Migration {

public function up()

{

$this->createTable('{{%article}}', [

'id' => $this->primaryKey(),

'alias' => $this->string()->notNull(),

'title' => $this->string()->notNull(),

'text' => $this->text()->notNull(),

]);

}

public function down()

{

$this->dropTable('{{%article}}');

}

}

1. Generate an Article model using Yii.

How to do it...

Carry out the following steps:

1. Now we will need to create some code to work with. Create protected/controllers/ErrorController .php as follows:

<?php

namespace app\controllers;

use app\models\Article; use yii\web\Controller;

class ErrorController extends Controller {

public function actionIndex()

{

$article = $this->findModel('php'); return $article->title;

} private function findModel($alias)