Creating a custom input widget

Yii has a very good set of form widgets, but as with every framework out there, Yii does not have them  
all. In this recipe, we will learn how to create your own input widget. For our example, we will create a  
range input widget.

Getting ready

Create a new application by 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. Create a widget file, @app/components/RangeInputWidget. php, as follows:

<?php

namespace app\components;  
use yii\base\Exception;  
use yii\base\Model;  
use yii\base\Widget;  
use yii\helpers\Html;  
class RangeInputWidget extends Widget  
{

public $model;  
public $attributeFrom;  
public $attributeTo;  
public $htmlOptions = [];  
protected function hasModel()

{

return $this->model instanceof Model&& $this->attributeFrom !== null&&  
$this->attributeTo !== null;}  
public function run()

{

if (!$this->hasModel()) {

throw new Exception('Model must be set');

}

return Html::activeTextInput($this->model, $this->attributeFrom, $this-  
>htmlOptions)

.' &rarr; '

.Html::activeTextInput($this->model, $this->attributeTo, $this-  
>htmlOptions);

}

}

1. Create a controller file, @app/controllers/RangeController . php, as follows:

<?php

namespace app\controllers;  
use Yii;

use yii\web\Controller;  
use app\models\RangeForm;  
class RangeController extends Controller  
{

public function actionIndex()

{

$model = new RangeForm();

if ($model->load(Yii::$app->request->post()) && $model->validate()) {

Yii::$app->session->setFlash('rangeFormSubmitted',

'The form was successfully processed!'

);

}

return $this->render('index', array(

'model' => $model,

));

}

}

1. Create a form file, @app/models/RangeForm.php, as follows:

<?php

namespace app\models;  
use yii\base\Model;  
class RangeForm extends Model  
{

public $from;  
public $to;

public function rules()

{

return [

[['from', 'to'], 'number', 'integerOnly' => true],

['from', 'compare', 'compareAttribute' => 'to', 'operator' => '<='],

];

}

}

1. Create a view file, @app/views/range/index. php, as follows:

<?php

use yii\helpers\Html;

use yii\bootstrap\ActiveForm;

use app\components\RangeInputWidget;

?>

<h1>Range form</h1>

<?php if (Yii::$app->session->hasFlash('rangeFormSubmitted')): ?>

<div class="alert alert-success">

<?= Yii::$app->session->getFlash('rangeFormSubmitted'); ?>

</div>

<?php endif?>

<?= Html::errorSummary($model, ['class'=>'alert alert-danger'])?>

<?php $form = ActiveForm::begin([

'options' => [

'class' => 'form-inline'

]

]); ?>

<div class="form-group">

<?= RangeInputWidget::widget([

'model' => $model,

'attributeFrom' => 'from',

'attributeTo' => 'to',

'htmlOptions' => [

'class' =>'form-control'

]

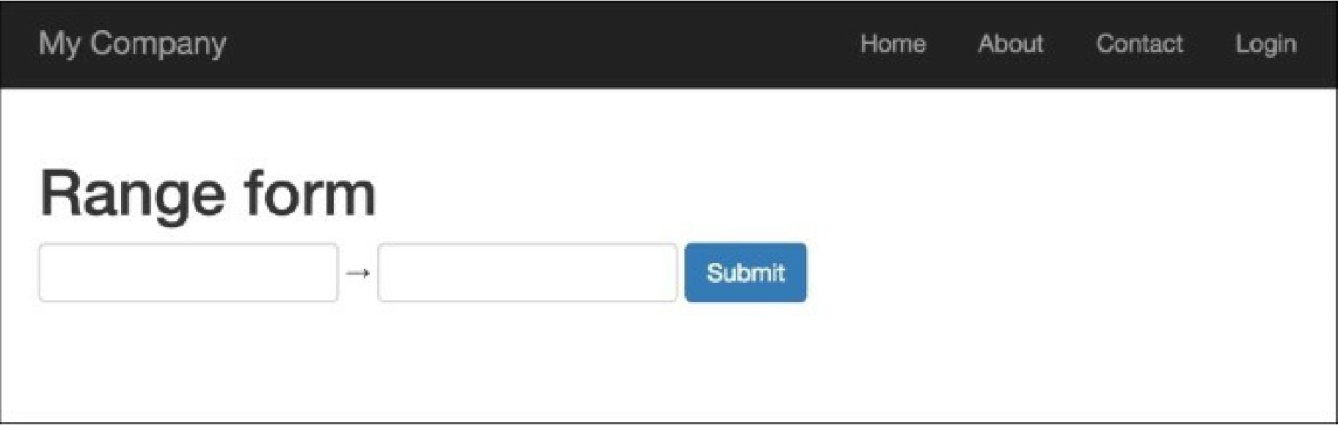
]) ?>

</div>

<?= Html::submitButton('Submit', ['class' => 'btn btn-primary', 'name' =>  
'contact-button']) ?>

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

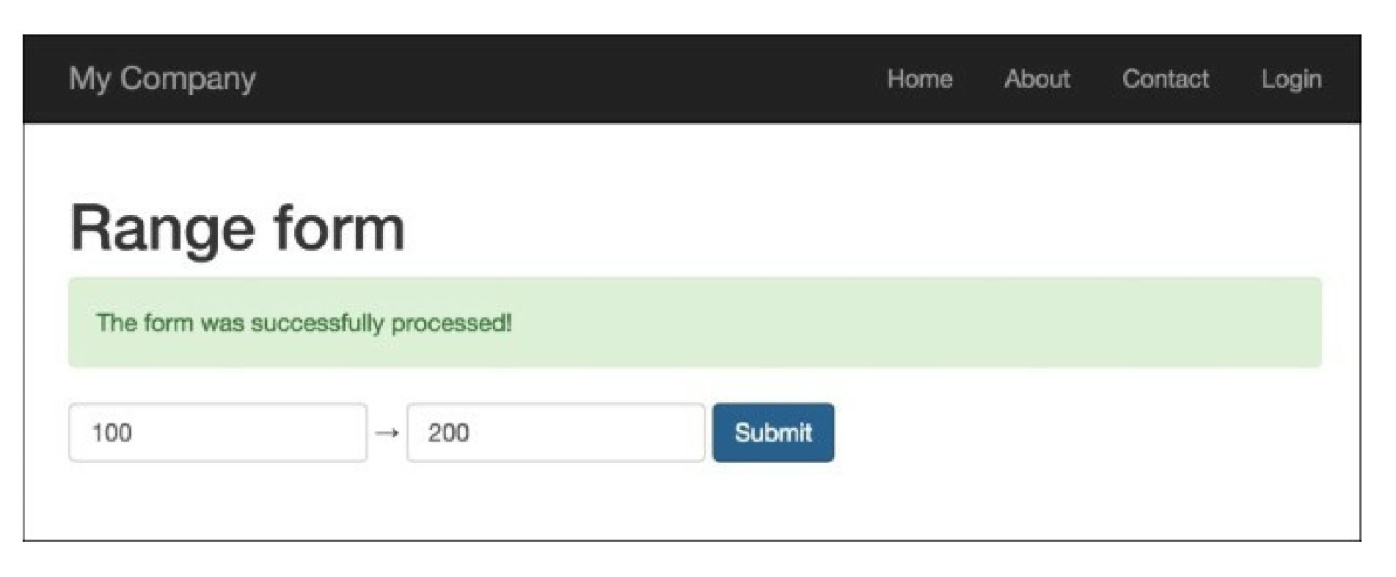
1. Run a range controller by opening index. php?r=range and you’ll get the following:



**6**. Enter 200 in the first text input field and 300 in the second, and you’ll get the following:



**7**. The widget outputs an error if the first value is bigger than the second; that is it. Try to input correct  
values, 100 and 200, for the first and second inputs, respectively:



How it works...

We write the range input widget, which requires four parameters:

* model: If it is not set, an exception will be thrown
* **attributeFrom:** This is used to set **minim**um range value
* **attributeTo:** This is used to set ma**xim**um range value
* **htmloptions:** It is passed to each input

This widget is used in form validation, and is set to check that the first value is less than or equal to the  
second value.

There’s more.

The Yii2 framework has an official Twitter Bootstrap extension that provides you with a pack of PHP  
wrappers over Twitter Bootstrap widgets. Before you write your own widget, check whether a Bootstrap  
widget exists at <http://www.yiiframework.com/doc-2.0/extbootstrap-index.html>**.**

See also

In order to learn more about widgets, you can use the following resources:

* [http://www.viiframework.com/doc-2.0/yii-base-wirlget.html](http://www.yiiframework.com/doc-2.0/yii-base-widget.html)
* [https://github.com/viisoft/yii2-bootstrap/blob/master/docs/guide/usage-widgets.md](https://github.com/yiisoft/yii2-bootstrap/blob/master/docs/guide/usage-widgets.md)