Pagination and sorting data

In the latest Yii releases, the focus was moved from using Active Record directly, to grids, lists, and data providers. Still, sometimes it is better to use Active Record directly. Let’s see how to list paginated AR records with the ability to sort them. In this section, we would like to create a list of films and sort them by some attributes from a database. In our example, we will sort our films by film title and rental rate attributes.

Getting ready

1. 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>.
2. Download the Sakila database from <http://dev.mysql.com/doc/index-other.html>.
3. Execute the downloaded SQLs; first schema, then data.
4. Configure the DB connection in config/main. php to use the Sakila database.
5. Use Gii to create the Film model.

How to do it...

1. First, you need to create @app/controllers/FilmController. php:

<?php

namespace app\controllers;

use app\models\Film; use yii\web\Controller; use yii\data\Pagination; use yii\data\Sort;

class FilmController extends Controller {

public function actionIndex()

{

$query = Film::find();

$countQuery = clone $query;

$pages = new Pagination(['totalCount' => $countQuery->count()]); $pages->pageSize = 5;

$sort = new Sort([

'attributes' => [

'title',

'rental\_rate'

]

]);

$models = $query->offset($pages->offset)

->limit($pages->limit)

->orderBy($sort->orders)

->all();

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

'models' => $models,

'sort' => $sort,

'pages' => $pages

]);

}

}

1. Now, let’s implement @app/views/film/index. php, as follows:

<?php

use yii\widgets\LinkPager;

/\*\*

* @var \app\models\Film $models
* @var \yii\web\View $this
* @var \yii\data\Pagination $pages
* @var \yii\data\Sort $sort \*/

?>

<h1>Films List</h1>

<p><?=$sort ->link('title')?> | <?=$sort->link('rental\_rate')?></p>

<?php foreach ($models as $model): ?>

<div class="list-group">

<h4 class="list-group-item-heading"> <?=$model->title ?>

<label class="label label-default"> <?=$model->rental\_rate ?>

</label>

</h4>

<p class="list-group-item-text"><?=$model->description ?></p>

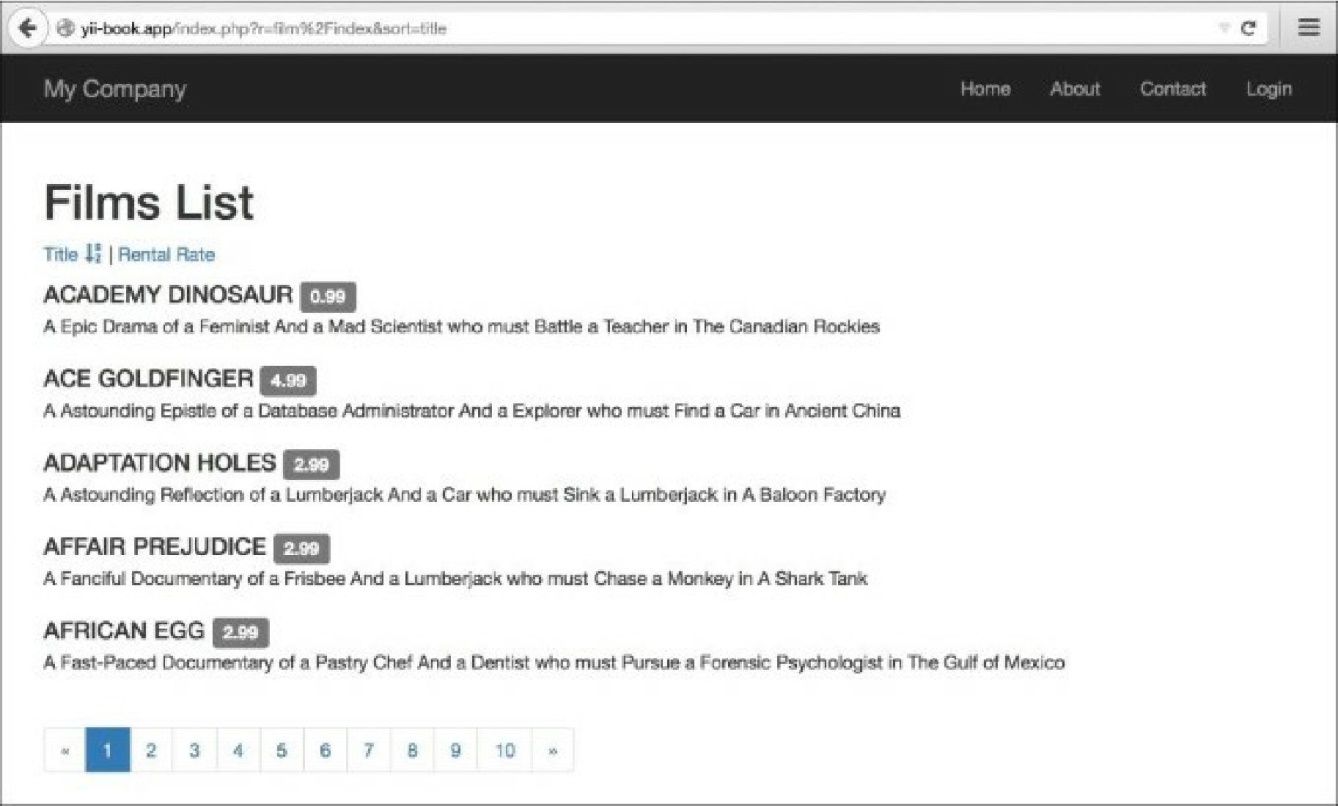
</div>

<?php endforeach ?>

<?=LinkPager::widget([

'pagination' => $pages ]); ?>

1. Try to load <http://yii-book.app/index>. php?r=film/index. You should get a working pagination and links that allow sorting of list by the film’s title or by rental rate:



How it works...

First, we got the total models count and initialized the new pagination component instance with it by passing the totalcount variable to our Pagination instance. Then, we used the $pages->pageSize field to set up the page size for our pagination. After that, we created a sorter instance for the model, specifying model attributes we wanted to sort by and applying order conditions to the query by calling orderBy and passing $sort->orders as a parameter. Then, we called all() to get records from the DB.

At this point, we have the models list, pages, and data used for the link pager, and the sorter that we use to generate sorting links.

In a view, we use the data we have gathered. First, we generate links with the Sort: :link method. Then, we list the models. Finally, using the LinkPager widgets, we render the pagination control.

See also

Visit the following links to get more information about pagination and sorting:

* <http://www.yiiframework.com/doc-2.0/yii-data-pagination.html>
* [http://www.yiiframework.com/doc-2.0/yii-data-sort.html](http://www.yiiframework.com/doc-2.0/yii-data-pagination.html)
* [http://www.yiiframework.com/doc-2.0/guide-output-pagi](http://www.yiiframework.com/doc-2.0/yii-data-sort.html)nation.html [http://www.yiiframework.com/doc-2.0/guide-output-sorting.html](http://www.yiiframework.com/doc-2.0/guide-output-pagination.html)