zf-tutorial/

/module

/Album

/config

/src

/Controller

/Form

/Model

/view

/album

/album

module.config.php

namespace Album;

use Zend\Router\Http\Segment;

use Zend\ServiceManager\Factory\InvokableFactory;

return [

'controllers' => [

'factories' => [

Controller\AlbumController::class => InvokableFactory::class,

],

],

// The following section is new and should be added to your file:

'router' => [

'routes' => [

'album' => [

'type' => Segment::class,

'options' => [

'route' => '/album[/:action[/:id]]',

'constraints' => [

'action' => '[a-zA-Z][a-zA-Z0-9\_-]\*',

'id' => '[0-9]+',

],

'defaults' => [

'controller' => Controller\AlbumController::class,

'action' => 'index',

],

],

],

],

],

'view\_manager' => [

'template\_path\_stack' => [

'album' => \_\_DIR\_\_ . '/../view',

],

],

];

Album.php under module/Album/src/Model:

namespace Album\Model;

class Album

{

public $id;

public $artist;

public $title;

public function exchangeArray(array $data)

{

$this->id = !empty($data['id']) ? $data['id'] : null;

$this->artist = !empty($data['artist']) ? $data['artist'] : null;

$this->title = !empty($data['title']) ? $data['title'] : null;

}

}

AlbumTable.php file in module/Album/src/Model directory like this:

namespace Album\Model;

use RuntimeException;

use Zend\Db\TableGateway\TableGatewayInterface;

class AlbumTable

{

private $tableGateway;

public function \_\_construct(TableGatewayInterface $tableGateway)

{

$this->tableGateway = $tableGateway;

}

public function fetchAll()

{

return $this->tableGateway->select();

}

public function getAlbum($id)

{

$id = (int) $id;

$rowset = $this->tableGateway->select(['id' => $id]);

$row = $rowset->current();

if (! $row) {

throw new RuntimeException(sprintf(

'Could not find row with identifier %d',

$id

));

}

return $row;

}

public function saveAlbum(Album $album)

{

$data = [

'artist' => $album->artist,

'title' => $album->title,

];

$id = (int) $album->id;

if ($id === 0) {

$this->tableGateway->insert($data);

return;

}

if (! $this->getAlbum($id)) {

throw new RuntimeException(sprintf(

'Cannot update album with identifier %d; does not exist',

$id

));

}

$this->tableGateway->update($data, ['id' => $id]);

}

public function deleteAlbum($id)

{

$this->tableGateway->delete(['id' => (int) $id]);

}

}

module/Album/src/Module.php

namespace Album;

// Add these import statements:

use Zend\Db\Adapter\AdapterInterface;

use Zend\Db\ResultSet\ResultSet;

use Zend\Db\TableGateway\TableGateway;

use Zend\ModuleManager\Feature\ConfigProviderInterface;

class Module implements ConfigProviderInterface

{

// getConfig() method is here

// Add this method:

public function getServiceConfig()

{

return [

'factories' => [

Model\AlbumTable::class => function($container) {

$tableGateway = $container->get(Model\AlbumTableGateway::class);

return new Model\AlbumTable($tableGateway);

},

Model\AlbumTableGateway::class => function ($container) {

$dbAdapter = $container->get(AdapterInterface::class);

$resultSetPrototype = new ResultSet();

$resultSetPrototype->setArrayObjectPrototype(new Model\Album());

return new TableGateway('album', $dbAdapter, null, $resultSetPrototype);

},

],

];

}

public function getControllerConfig()

{

return [

'factories' => [

Controller\AlbumController::class => function($container) {

return new Controller\AlbumController(

$container->get(Model\AlbumTable::class)

);

},

],

];

}

}

module/Album/src/Controller/AlbumController.php

namespace Album\Controller;

// Add the following import:

use Album\Model\AlbumTable;

use Zend\Mvc\Controller\AbstractActionController;

use Zend\View\Model\ViewModel;

class AlbumController extends AbstractActionController

{

// Add this property:

private $table;

// Add this constructor:

public function \_\_construct(AlbumTable $table)

{

$this->table = $table;

}

public function indexAction()

{

return new ViewModel([

'albums' => $this->table->fetchAll(),

]);

}

public function addAction()

{

}

public function editAction()

{

}

public function deleteAction()

{

}

}

index.phtml

**<?php**

// module/Album/view/album/album/index.phtml:

$title = 'My albums';

$this->headTitle($title);

**?>**

<h1>**<?=** $this->escapeHtml($title) **?>**</h1>

<p>

<a href="**<?=** $this->url('album', ['action' => 'add']) **?>**">Add new album</a>

</p>

<table class="table">

<tr>

<th>Title</th>

<th>Artist</th>

<th>&nbsp;</th>

</tr>

**<?php** foreach ($albums as $album) : **?>**

<tr>

<td>**<?=** $this->escapeHtml($album->title) **?>**</td>

<td>**<?=** $this->escapeHtml($album->artist) **?>**</td>

<td>

<a href="**<?=** $this->url('album', ['action' => 'edit', 'id' => $album->id]) **?>**">Edit</a>

<a href="**<?=** $this->url('album', ['action' => 'delete', 'id' => $album->id]) **?>**">Delete</a>

</td>

</tr>

**<?php** endforeach; **?>**

</table>