Model

* Src\Model

CakePHP’s models are composed of:

* Table
* Table objects provide access to the collection of entities stored in a specific table.
* Entity

Entity

* Create new entity

$article = $this->Articles->newEntity();

* Set data article from request data

$article = $this->Articles->patchEntity($article, $this->request->getData());

* Save ~ insert or update

If ($this->Articles->save($article) {

//ok

}

* Delete

$this->request->allowMethod('post');

$article = $this->Articles->findBySlug($slug)->firstOrFail();

if ($this->Articles->delete($article)) {

$this->Flash->success(\_\_('The article has been deleted.'));

} else {

$this->Flash->error(\_\_('The article could not be deleted. Please, try again.'));

}

return $this->redirect(['action' => 'index']);

method:

* Get
* Getting a single entity by primary key
* Syntax:

Cake\ORM\Table::get($id, $options = [])

* If don’t exits bug error “Record not found in table "[users]"” at

Cake\Datasource\Exception\RecordNotFoundException. Try catch to use

try {

$user = $userTable->get(2334);

} catch (\Cake\Datasource\Exception\RecordNotFoundException $e) {

echo 'not found';

}

* Example:
* Get a user from user id

$user = $this->Users->get($id);

* Get a user from id with all comments

$user = $this->Users->get($id, [

'contain' => ['Comments']

]);

* Using cache
* Use any cache config or CacheEngine instance and a generated key

$user = $users->get($id, [

'cache' => 'custom',

]);

* Use any cache config or CacheEngine instance and specific key

$user = $users->get($id, [

'cache' => 'custom',

'key' => 'mykey'

]);

* Disable caching

$user = $users->get($id, [

'cache' => false

]);