



### CakePHP Conventions

#### Naming Conventions

<b>Controllers:</b>	Plural, PascalCase, ends with <code>Controller</code> (e.g., <code>ArticlesController</code> )
<b>Models:</b>	Singular, PascalCase (e.g., <code>Article</code> )
<b>Tables:</b>	Plural, PascalCase, ends with <code>Table</code> (e.g., <code>ArticlesTable</code> )
<b>Views:</b>	Lowercase, underscored (e.g., <code>edit.php</code> , <code>view.php</code> )
<b>Database Tables:</b>	Plural, underscored (e.g., <code>articles</code> , <code>users</code> )
<b>Fields in Database:</b>	Lowercase, underscored (e.g., <code>article_id</code> , <code>created_at</code> )

#### Directory Structure

<code>src/Controller/</code>	- Controller files.
<code>src/Model/Table/</code>	- Table classes (database access).
<code>src/Model/Entity/</code>	- Entity classes (data objects).
<code>templates/</code>	- View files.
<code>config/</code>	- Configuration files.
<code>webroot/</code>	- Publicly accessible assets (CSS, JS, images).

### Core Classes & Usage

#### Controller

<code>\$this-&gt;loadModel('modelName');</code>	Loads a model into the controller.
<code>\$this-&gt;request-&gt;getData();</code>	Accesses POST data.
<code>\$this-&gt;request-&gt;getQuery();</code>	Accesses GET parameters.
<code>\$this-&gt;redirect(['controller' =&gt; '...', 'action' =&gt; '...']);</code>	Redirects to another action.
<code>\$this-&gt;set(compact('variable1', 'variable2'));</code>	Sets variables to be available in the view.
<code>\$this-&gt;Flash-&gt;success('Message');</code>	Displays a success flash message.

#### Table

<code>\$this-&gt;find('all');</code>	Retrieves all records.
<code>\$this-&gt;get(\$id);</code>	Retrieves a single record by ID.
<code>\$entity = \$this-&gt;newEntity(\$data);</code>	Creates a new entity from data.
<code>\$this-&gt;patchEntity(\$entity, \$data);</code>	Patches an entity with new data.
<code>\$this-&gt;save(\$entity);</code>	Saves an entity to the database.
<code>\$this-&gt;delete(\$entity);</code>	Deletes an entity from the database.

#### Entity

Entities represent individual table rows. You can access properties directly (e.g., `$article->title`). Use `_accessible` property to define which fields are mass assignable.

### ORM Features

#### Associations

<code>\$this-&gt;belongsTo('Author');</code>	Defines a belongs-to association.
<code>\$this-&gt;hasMany('Comments');</code>	Defines a has-many association.
<code>\$this-&gt;belongsToMany('Tags');</code>	Defines a belongs-to-many association.
<code>\$this-&gt;hasOne('Profile');</code>	Defines a has-one association.
Eager loading associations	<code>\$this-&gt;Articles-&gt;find()-&gt;contain(['Authors', 'Comments']);</code>

#### Retrieving Data

<code>\$this-&gt;find('all')-&gt;where(['Articles.published' =&gt; true]);</code>	Find all published articles.
<code>\$this-&gt;find('list');</code>	Retrieves data as a key-value list.
<code>\$this-&gt;find('threaded');</code>	Retrieves data as a threaded list.
<code>\$this-&gt;find('first');</code>	Retrieves first element.

#### Query Builder

CakePHP provides a powerful query builder for constructing complex database queries. Use the `->select()`, `->where()`, `->order()`, `->limit()`, and `->join()` methods to customize your queries.

### Common Tasks

## Generating URLs

<pre><code>\$this-&gt;Url-&gt;build(['controller' =&gt; 'Articles', 'action' =&gt; 'view', 1]);</code></pre>	Generates a URL to the <code>view</code> action of the <code>Articles</code> controller with ID 1.
<pre><code>Router::url(['controller' =&gt; 'Articles', 'action' =&gt; 'index']);</code></pre>	Generates a URL using the Router class (outside of views).
<pre><code>\$this-&gt;Url-&gt;asset('img/logo.png');</code></pre>	Generate path to asset.

## Form Helper

<pre><code>echo \$this-&gt;Form-&gt;create(\$article);</code></pre>	Creates a form for the <code>\$article</code> entity.
<pre><code>echo \$this-&gt;Form-&gt;input('title');</code></pre>	Creates an input field for the <code>title</code> property.
<pre><code>echo \$this-&gt;Form-&gt;button('Submit');</code></pre>	Creates a submit button.
<pre><code>echo \$this-&gt;Form-&gt;end();</code></pre>	Closes the form.

## Baking

CakePHP's baking tool can generate code for models, views, and controllers. Use the `bin/cake bake` command to generate code quickly. For example, `bin/cake bake controller Articles`.