

CakePHP Console Cheat Sheet



Details extracted from CakePHP 3.2.x command prompt.

Installation

```
composer require --dev cakephp/bake :~1.0
```

PhpStorm comes with **composer** built in.
To install CakePHP via PhpStorm, use the File → New → Composer Project.
Then filter for *CakePHP* and select the `cakephp\app\composer` project.

CakePHP Console

Run the CakePHP console from project root

```
bin/cake (use bin\cake for Windows)
```

Possible "shells"

```
bake, migrations, i18n, orm_cache, plugin, routes, server, behavior_time, console, orm
```

Run Cake Server
(usually on localhost port 8765)

```
bin/cake server
```

Run migrations shell
(with help)

```
bin/cake migrations -h
```

Run bake with plugin prefix

```
bin/cake bake.bake -h
```

⚠ PHP CLI must be installed and in the path.

Bake Shell Common Options

help	--help or -h
Verbose output	--verbose or -v
Quiet output	--quiet or -q
Plugin to bake into	--plugin or -p
Force overwrite of existing files	--force or -f

Bake Shell Common Options (cont)

Datasource connection to get data from (default: default)	--connection or -c
Theme to use when baking	--theme or -t
Do not bake a test skeleton	--no-test
Do not bake generic CRUD action methods	--no-actions

❗ These options are common to most of the cake shell commands.

Server Shell

Run server shell with
(defaults)

```
bin/cake server
```

Run server shell and get help

```
bin/cake server --help
```

Run server shell on port 9999

```
bin/cake server --port 9999
```

Run server shell using IP of 10.0.0.2

```
bin/cake server --host 10.0.0.2
```

Run server shell with document root dummy/-app/root

```
bin/cake server --document_root dummy/app
```

Run server shell with verbose output

```
bin/cake server --verbose
```

... example

```
bin/cake -H 10.0.0.2 -p 9999 -d dummy/app -v
```

Bake Shell: Model

```
Run bake    bin/cake bake model
model
shell and
list all
tables
```

```
Bake        bin/cake bake model Friends
model
class for
Friends
table
```

```
Run bake    bin/cake bake model --help
model
shell and
get help
```

```
Genera-     bin/cake bake model  [subcommand] [options,
lised use:  >]
```

Options *other than the default ones*

```
Table       --table <Ta ble Nam e>
name if "-
non-co-
nventiona-
l" table
naming
used
```

```
Disable     --no-e ntity
generation
of Entity
class
```



By **Ady G** (AdyG)
cheatography.com/adyg/

Not published yet.
Last updated 13th May, 2016.
Page 1 of 4.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Bake Shell: Model (cont)

Disable generation of Table class

`--no-table`

Disable generation of validation rules

`--no-validation`

Disable generation of a rules checker

`--no-rules`

Disable generation of associations

`--no-associations`

Disable generation of fields

`--no-fields`

Comma separated list of fields to make accessible

`--fields <field name>[,<field name>...]`

Comma separated list of hidden fields

`--hidden <field name>[,<field name>...]`

Set the primary key field or fields (comma separated)

`--primary-key <key-field>[,<key-field>...]`

Display field if you would like to select one

`--display -field <field-name>`

Cake Routes Shell

Bake Shell: Controller

Bake a controller skeleton

`cake bake controller [subcommand] [option]`

Bake the My controller with CRUD actions

`bin/cake bake controller MyController`

Bake all controllers with CRUD methods

`cake bake controller all`

⚠ Beware:

This may overwrite existing controllers

Options

Plugin to bake into

`--plugin or -p`

Force overwrite of existing files

`--force or -f`

Datasource connection to get data from

`--connection or -c`

(default: default)

Theme to use when baking

`--theme or -t`

Components to use (comma separated)

`--components`

Helpers to use (comma separated)

`--helpers`

Namespace/routing prefix to use

`--prefix`

Do not bake a test skeleton

`--no-test`

Do not bake generic CRUD action methods

`--no-actions`

List of all Routes	<code>bin/cake routes</code>	<i><name></i> is the name of the controller to bake, and you may use <i>Plugin.name</i> to bake controllers into plugins (optional).									
URL Parsing Testing	<code>bin/cake routes check route- and -op tions</code>										
*** example	<code>bin/bake routes check /bookm ark s/e dit/1</code>	<table><tr><th colspan="3">Version Control</th></tr><tr><th>Version</th><th>Date</th><th>Notes</th></tr><tr><td>1.0</td><td>25 April 2016</td><td>Initial version, there are many more commands to add</td></tr></table>	Version Control			Version	Date	Notes	1.0	25 April 2016	Initial version, there are many more commands to add
Version Control											
Version	Date	Notes									
1.0	25 April 2016	Initial version, there are many more commands to add									
URL Generation Testing	<code>bin/cake routes generate contro ller:contro ller name action:action other opti</code>	ⓘ This will be updated each time we add new content to the cheat sheet.									
*** example	<code>bin/cake routes generate contro lle r:B ook marks action :edit 1</code>										
*** example with string parameters	<code>bin/cake routes check " /bo okm ark s/? pag e=1 &s ort =ti tle &d ire cti on= -des c"</code>										



By **Ady G** (AdyG)
cheatography.com/adyg/

Not published yet.
Last updated 13th May, 2016.
Page 2 of 4.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>