

Couchbase Cheatsheet

Method	Title
<code>\$object->add(\$key, \$value [, \$expiry] [, \$persistto] [, \$replicateto])</code>	Add a value with the specified key that does not already exist
<code>\$object->append(\$key, \$value [, \$expiry] [, \$casunique] [, \$persistto] [, \$replicateto])</code>	Append a value to an existing key
<code>\$object->cas(\$casunique, \$key, \$value [, \$expiry])</code>	Compare and set a value providing the supplied CAS key matches
<code>\$object->decrement(\$key, \$offset)</code>	Decrement the value of an existing numeric key
<code>\$object->delete(\$key [, \$casunique])</code>	Delete a key/value
<code>\$object->fetch(\$key [, \$keyn])</code>	Fetch the next delayed result set document
<code>\$object->fetchAll(\$key [, \$keyn])</code>	Fetch all the delayed result set documents
<code>\$object->flush()</code>	Deletes all values from the corresponding bucket
<code>\$object->get(\$key [, \$callback] [, \$casunique])</code>	Get one or more key values
<code>\$object->getDelayed(\$keyn [, \$with_cas] [, \$callback])</code>	Get a key to be fetched later
<code>\$object->getMulti(\$keycollection [, \$casarray])</code>	Get one or more key values
<code>\$object->getAndLock(\$key [, \$casarray] [, \$getl-expiry])</code>	Get the value for a key, lock the key from changes
<code>\$object->getAndLockMulti(\$keycollection [, \$casarray] [, \$getl-expiry])</code>	Get the value for a key, lock the key from changes
<code>\$object->getAndTouch(\$key, \$expiry)</code>	Get a value and update the expiration time for a given key

<code>\$object->getAndTouchMulti(\$key, \$expiry)</code>	Get a value and update the expiration time for a given key
<code>\$object->getClientVersion()</code>	Returns the version of the client library
<code>\$object->getNumReplicas()</code>	Returns the number of replicas for the configured bucket
<code>\$object->getOption(\$option)</code>	Retrieve an option
<code>\$object->getStats()</code>	Get the database statistics
<code>\$object->getTimeout()</code>	Get the currently configured timeout value for operations
<code>\$object->getVersion()</code>	Returns the versions of all servers in the server pool
<code>\$object->increment(\$key [, \$offset] [, \$create] [, \$expiry] [, \$initial])</code>	Increment the value of an existing numeric key
<code>\$object->keyDurability(\$key, \$casunique, \$durability)</code>	Wait until the durability of a document has been reached
<code>\$object->keyDurabilityMulti(\$keycollection, \$durability)</code>	Wait until the durability of a document has been reached
<code>\$object->new Couchbase(\$url [, \$username] [, \$password] [, \$bucket] [, \$persistent])</code>	Create connection to Couchbase Server
<code>\$object->observe(\$key, \$casunique, \$observedetails)</code>	Get the durability of a document
<code>\$object->observeMulti(\$keycollection, \$observedetails)</code>	Get the durability of a document
<code>\$object->prepend(\$key, \$value [, \$expiry] [, \$casunique] [, \$persistto] [, \$replicateto])</code>	Prepend a value to an existing key
<code>\$object->replace(\$key, \$value [, \$expiry] [, \$casunique] [, \$persistto] [, \$replicateto])</code>	Update an existing key with a new value

<code>\$object->set(\$key, \$value [, \$expiry] [, \$casunique] [, \$persistto] [, \$replicateto])</code>	Store a value using the specified key
<code>\$object->setMulti(\$kvarray [, \$expiry])</code>	Set multiple key/value items at once
<code>\$object->setOption(\$option, \$mixed)</code>	Specify an option
<code>\$object->setTimeout(\$timeout)</code>	Set the configured timeout value for operations
<code>\$object->touch(\$key, \$expiry)</code>	Update the expiry time of an item
<code>\$object->touchMulti(\$keyarray, \$expiry)</code>	Change the expiration time for multiple documents
<code>\$object->unlock(\$key, \$casunique)</code>	Unlock a previously locked key
<code>\$object->view(\$ddocname [, \$viewname] [, \$viewoptions])</code>	Execute a view request
<code>\$object->view()</code>	Query a view
<code>\$object->viewGenQuery(\$ddocname [, \$viewname] [, \$viewoptions])</code>	Generate a view request, but do not execute the query