Cheatography

RedBeanPHP Class Reference Cheat Sheet

by Richard Holloway (richardjh) via cheatography.com/478/cs/347/

About RedBeanPHP ORM

Easy ORM for PHP and MySQL, PostgreSQL and many other database systems.
Use the simplicity of NoSQL with the power of SQL.
http://redbeanphp.com/

Class Reference pt1

addDatabase ($key, $dsn, $user=null, $pass=null, $frozen=false)

Adds a database to the facade, afterwards you can select the database using selectDatabase($key).

addTags ($bean, $tagList)

Part of RedBeanPHP Tagging API. Adds tags to a bean. If $tagList is a comma separated list of tags all tags will be associated with the bean. You may also pass an array instead of a string.

areRelated($bean1, $bean2)

Checks whether a pair of beans is related N-M. This function does not check whether the beans are related in N:1 way.

associate($bean1, $bean2, $extra=null)

Associates two Beans. This method will associate two beans with eachother. You can then get one of the beans by using the related() function and providing the other bean.

batch($type, $ids)

Returns an array of beans. Pass a type and a series of ids and this method will bring you the correspondig beans.

begin()

Facade Convience method for adapter transaction system. Begins a transaction.

clearRelations($bean, $type)

Clears all associated beans. Breaks all many-to-many associations of a bean and a specified type.

Class Reference pt1 (cont)

close()

Closes the database connection.

commit()

Facade Convience method for adapter transaction system. Commits a transaction.

configureFacadeWithToolbox($tb)

Configures the facade, want to have a new Writer? A new Object Database or a new Adapter and you want it on-the-fly? Use this method to hot-swap your facade with a new toolbox.

convertToBeans($type, $rows)

Converts a series of rows to beans.

count($beanType)

Counts beans

debug($tr=true)

Toggles DEBUG mode. In Debug mode all SQL that happens under the hood will be printed to the screen.

dependencies($dep)

Sets a list of dependencies. A dependency list contains an entry for each dependent bean. A dependent bean will be removed if the relation with one of the dependencies gets broken.

dispense($type, $num)

Dispenses a new RedBean OODB Bean for use with the rest of the methods.

dispenseLabels($type, $labels)

A label is a bean with only an id, type and name property. This function will dispense beans for all entries in the array. The values of the array will be assigned to the name property of each individual bean.

dup($bean, $trail, $pid=false)

Makes a copy of a bean. This method makes a deep copy of the bean.

exec($sql, $values=array())

Convenience function to execute Queries directly. Executes SQL.

Class Reference pt1 (cont)

exportAll($beans)

Exports a collection of beans. Handy for XML/JSON exports with a Javascript framework like Dojo or ExtJS.

find($type, $sql=null, $values=array())

Finds a bean using a type and a where clause (SQL). As with most Query tools in RedBean you can provide values to be inserted in the SQL statement by populating the value array parameter; you can either use the question mark notation or the slot-notation (:keyname).

findAll($type, $sql=null, $values=array())

Finds a bean using a type and a where clause (SQL). As with most Query tools in RedBean you can provide values to be inserted in the SQL statement by populating the value array parameter; you can either use the question mark notation or the slot-notation (:keyname). The findAll() method differs from the find() method in that it does not assume a WHERE-clause.

Class Reference pt2

findAndExport($type, $sql=null, $value=array())

Finds a bean using a type and a where clause (SQL). As with most Query tools in RedBean you can provide values to be inserted in the SQL statement by populating the value array parameter; you can either use the question mark notation or the slot-notation (:keyname). The variation also exports the beans (i.e. it returns arrays).

findLast($type, $sql=null, $values=array())

Finds a bean using a type and a where clause (SQL). As with most Query tools in RedBean you can provide values to be inserted in the SQL statement by populating the value array parameter; you can either use the question mark notation or the slot-notation (:keyname). This variation returns the last bean only.

By **Richard Holloway** (richardjh)
cheatography.com/richardjh/
richardjh.org

Published 16th April, 2012.
Last updated 5th June, 2014.

Cheatography

RedBeanPHP Class Reference Cheat Sheet

by Richard Holloway (richardjh) via cheatography.com/478/cs/347/

Class Reference pt2 (cont)

findOne($type, $sql=null, $values=array())

Finds a bean using a type and a where clause (SQL). As with most Query tools in RedBean you can provide values to be inserted in the SQL statement by populating the value array parameter; you can either use the question mark notation or the slot-notation (:keyname). This variation returns the first bean only.

findOrDispense($type, $sql, $values)

Convenience method. Tries to find beans of a certain type, if no beans are found, it dispenses a bean of that type.

freeze($tr=true)

Toggles fluid or frozen mode. In fluid mode the database structure is adjusted to accomodate your objects. In frozen mode this is not the case.

gatherLabels($beans)

Gathers labels from beans. This function loops through the beans, collects the values of the name properties of each individual bean and stores the names in a new array. The array then gets sorted using the default sort function of PHP (sort).

genSlots($array)

Generates question mark slots for an array of values.

getAll($sql, $values=array())

Convenience function to execute Queries directly. Executes SQL.

Class Reference pt2 (cont)

getAssoc($sql, $values=array())

Convenience function to execute Queries directly. Executes SQL. Results will be returned as an associative array. The first column in the select clause will be used for the keys in this array and the second column will be used for the values. If only one column is selected in the query, both key and value of the array will have the value of this field for each row.

getCell($sql, $values=array())

Convenience function to execute Queries directly. Executes SQL.

getCol($sql, $values=array())

Convenience function to execute Queries directly. Executes SQL.

getColumns($table)

Returns a list of columns. Format of this array: array(fieldname => type) Note that this method only works in fluid mode because it might be quite heavy on production servers!

getRow($sql, $values=array())

Convenience function to execute Queries directly. Executes SQL.

getVersion()

Get version

graph($array, $filterEmpty=false)

facade method for Cooker Graph.

hasTags($bean, $tags, $all=false)

Part of RedBeanPHP Tagging API. Tests whether a bean has been associated with one ore more of the listed tags. If the third parameter is TRUE this method will return TRUE only if all tags that have been specified are indeed associated with the given bean, otherwise FALSE. If the third parameter is FALSE this method will return TRUE if one of the tags matches, FALSE if none match.

Class Reference pt3

isoDate($time=null)

Simple convenience function, returns ISO date formatted representation of $time

isoDateTime($time=null)

Simple convenience function, returns ISO date time formatted representation of $time.

load($type, $id)

Loads the bean with the given type and id and returns it.

log($filename)

Activates TimeLine Schema Alteration monitoring and Query logging.

nuke()

Nukes the entire database

query($method, $sql, $values)

Internal Query function, executes the desired query. Used by all facade query functions. This keeps things DRY.

related($bean, $type, $sql=null, $values=array())

Returns all the beans associated with $bean. This method will return an array containing all the beans that have been associated once with the associate() function and are still associated with the bean specified.

relatedOne($bean, $type, $sql=null, $values=array())

Returns only single associated bean.

rollback()

Facade Convience method for adapter transaction system. Rolls back a transaction.

selectDatabase($key)

Selects a different database for the Facade to work with.

By **Richard Holloway** (richardjh)
cheatography.com/richardjh/
richardjh.org

Published 16th April, 2012.
Last updated 5th June, 2014.

Cheatography

RedBeanPHP Class Reference Cheat Sheet

by Richard Holloway (richardjh) via cheatography.com/478/cs/347/

Class Reference pt3 (cont)

setup($dsn=null, $username=null, $password=null)

> Kickstarts redbean for you. This method should be called before you start using RedBean. The Setup() method can be called without any arguments, in this case it will try to create a SQLite database in /tmp called red.db (this only works on UNIX-like systems).

store($bean)

> Stores a RedBean OODB Bean and returns the ID.

storeAll($beans)

> Short hand function to store a set of beans at once, IDs will be returned as an array. For information please consult the R::store() function. A loop saver.

swap($beans, $property)

> Given an array of two beans and a property, this method swaps the value of the property. This is handy if you need to swap the priority or orderNo of an item (i.e. bug-tracking, page order).

tag($bean, $tagList=null)

> Part of RedBeanPHP Tagging API. Tags a bean or returns tags associated with a bean. If $tagList is null or omitted this method will return a comma separated list of tags associated with the bean provided. If $tagList is a comma separated list (string) of tags all tags will be associated with the bean. You may also pass an array instead of a string.

tagged($beanType, $tagList)

> Part of RedBeanPHP Tagging API. Returns all beans that have been tagged with one of the tags given.

trash($bean)

> Deletes the specified bean.

Class Reference pt3 (cont)

trashAll($beans)

> Short hand function to trash a set of beans at once. For information please consult the R::trash() function. A loop saver.

unassociate($bean1, $bean2, fast=false)

> Breaks the association between two beans. This functions breaks the association between a pair of beans. After calling this functions the beans will no longer be associated with eachother. Calling related() with either one of the beans will no longer return the other bean.

unrelated($bean, $type, $sql=null, $values=array())

> The opposite of related(). Returns all the beans that are not associated with the bean provided.

untag($bean, $tagList)

> Part of RedBeanPHP Tagging API. Removes all sepcified tags from the bean. The tags specified in the second parameter will no longer be associated with the bean.

wipe($beanType)

> Wipes all beans of type $beanType.

Class Reference: See Also

Converted from the comments in the source code.
See http://www.redbeanphp.com/api/d9/daa/class_red_bean___facade.html for more details.

By **Richard Holloway** (richardjh)
cheatography.com/richardjh/
richardjh.org

Published 16th April, 2012.
Last updated 5th June, 2014.