

PHP WALLET CLASS

Class Description

Wallet

Php Wallet class is an easy to integrate class that introduces finance cash flow (**Credit & Debit**) in your system. Once incorporated, it creates the capability of your system to be able to:

- ❖ **Receive Money**
- ❖ **Withdraw Money**
- ❖ **Check account balance**
- ❖ **Add funds with different payment gateways**
- ❖ **Delete/Edit the money records**
- ❖ **Check credit/debit statements.**

You can integrate with a good number of Php frameworks. The class has wise prepared SQL statements that won't make you to hard code the table columns
It comes with three (3) database tables whereby you can replace the user's table with yours and maintain the other two(2).

Scenario

Let's say you are building a system that at some point you will need to receive money from different sources, spend it, show balance or continue to offer services based on the users and then produce a statement on the usage.

For instance

You have a system that will involve users to subscribe to your services and you send them messages. Basically you want to track the services offered to the user by crediting the Cash out & debiting the Cash in. At the end of this all you can fetch the balance based on the user, date of transaction or even other options. You can use this class to control the whole wallet in the back end or by user in the front end.

WALLET METHODS USED

The wallet class has several functions/Methods that describes how it functions

- ◆ `,walletRead`
- ◆ `,walletTopUp`
- ◆ `,walletEdit`
- ◆ `,walletDelete`
- ◆ `,walletWithdrawal`
- ◆ `,walletBalance`
- ◆ `,walletStatement`
- ◆ `,arrayToObject`
- ◆ `,objectToArray`

1. WalletRead()

Located at [class/Wallet.php](#)

WalletRead() Method

public array	<pre>walletRead (\$table, Array \$where = array(), Array \$options_ = array())</pre> <p>Selects data from the wallets depending on the columns provided in the options and the where clause.</p> <p>Parameters</p> <p>\$table The source table eg. <code>\$table = 'wallet_in'</code></p> <p>Array \$where = array() Specifies the row(s) to be selected Array items MUST be assigned eg. <code>\$where = array('userId'=>'2')</code></p> <p>Array \$options_ = array() Specifies the column(s) to be selected. Array items MUST not be assigned eg. <code>\$options_ = array('id','userId','amount','source')</code></p> <p>Returns array rows.</p> <p>Throws Wallet\walletException</p> <p>See Resource coming soon...</p>
--------------	--

2. WalletTopUp()

Located at [class/Wallet.php](#)

WalletTopUp() Method

public array	<pre>walletTopUp (\$table, Array \$options_ = array())</pre> <p>Adds funds to the wallet. just ensure that you focus on the amount & other related columns</p> <p>Parameters</p> <p>\$table The targeted table.</p> <p>Array \$options_ = array() The array contains the columns and their assigned values e.g. <code>array('amount'=>'300','source'=>'paypal','date'=>'10-12-2017')</code></p> <p>Returns array message & fields .</p> <p>Throws Wallet\walletException</p> <p>See Resource coming soon...</p>
--------------	---

3. WalletEdit()

Located at [class/Wallet.php](#)

WalletTopUp() Method

public array	<pre>walletEdit(\$table, \$where, \$options_ = array())</pre> <p>Makes changes on the wallet</p> <p>Parameters</p> <p>\$table The source table to be edited e.g <code>\$table='wallet_out'</code></p> <p>\$where Points the exactly the row to edit. This must be one(1) array item . e.g <code>\$where = array</code></p> <p>\$options_ = array() This is an array containing the columns with their respective values e.g <code>array('amount'=>'500', 'source'=>'Paypal')</code></p> <p>Returns array message,fields,where.</p> <p>Throws Wallet\walletException</p> <p>See reference coming soon...</p>
--------------	--

4. WalletDelete()

Located at [class/Wallet.php](#)

WalletDelete() Method

public array	<pre>walletDelete(\$table, Array \$where)</pre> <p>Delete Wallet records from any specified table and the WHERE clause</p> <p>Parameters</p> <p>\$table Either source table from which delete has to be done. e.g <code>\$table ='wallet_in'</code></p> <p>\$where Array with one(1) item which MUST be assigned to a VALUE, this makes delete specific. e.g <code>array('userId'=>'1')</code></p> <p>Returns Object message of affected rows.</p> <p>Throws Wallet\walletException</p> <p>See reference coming soon...</p>
--------------	--

5. WalletWithdrawal()

Located at [class/Wallet.php](#)

WalletWithdrawal() Method

public array	<pre>walletWithdrawal (\$table , Array \$options_ = array())</pre> <p>Use money from the wallet, this is basically crediting the wallet out</p> <p>Parameters</p> <p>\$table The source table. e.g <code>\$table= 'wallet_out'</code></p> <p>\$options_ = array() This is basically the array of fields & the values to be credited in wallet_out e.g <code>array=('userId'=>'2', 'amount'=>'200')</code></p> <p>Returns array message, sql response</p> <p>Throws Wallet\walletException</p> <p>See Resource coming soon...</p>
--------------	--

6. WalletStatement()

Located at [class/Wallet.php](#)

WalletStatement() Method

public array	<pre>walletStatement (\$table1, \$table2, Array \$where = array(), Array \$options_ = array())</pre> <p>Selects values from the wallet for the statement depending on the columns provided in the options and the where clause.</p> <p>Parameters</p> <p>\$table1 Cash In source table eg. <code>\$table1 = 'wallet_in'</code></p> <p>\$table2 Cash out source table eg. <code>\$table2 = 'wallet_out'</code></p> <p>Array \$where = array() Specifies the row(s) to be selected Array items MUST be assigned eg. <code>\$where = array('userId'=>'2')</code></p> <p>Array \$options_ = array() Specifies the column(s) to be selected. Array items MUST not be assigned eg. <code>\$options_ = array('id', 'userId', 'amount', 'source')</code></p> <p>Returns array rows.</p> <p>Throws Wallet\walletException</p> <p>See Resource coming soon...</p>
--------------	--

7. arrayToObject()

WalletarrayToObject() Method

public function

[arrayToObject](#) (**\$array**)

Convert the returned array in to an object

Parameters

\$array

Array to be converted to object

Returns

Object

array

Throws

[Nothing to throw](#)

See

reference coming soon

8. objectToArray()

WalletobjectToArray() Method

public function

[objectToArray](#) (**\$object**)

Convert the returned object to an array.

Parameters

\$object

Object to be returned.

Returns

array

Object

Throws

[Nothing to throw](#)

See

reference coming soon