PHP Binance Api

Generated by Doxygen 1.8.6

Thu May 3 2018 17:45:23

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Namespace Documentation

3.1 Binance Namespace Reference

Classes

- class API

  Main Binance class.

Chapter 4

Class Documentation

4.1 Binance\API Class Reference

Main Binance class.

Public Member Functions

- __construct ()

  Constructor for the class, send as many argument as you want.
- __get (string $member)

  magic get for private and protected members
- __set (string $member, $value)

  magic set for private and protected members
- buy (string $symbol, $quantity, $price, string $type="LIMIT", array $flags=[])

  buy attempts to create a currency order each currency supports a number of order types, such as -LIMIT -MARKET -STOP_LOSS -STOP_LOSS_LIMIT -TAKE_PROFIT -TAKE_PROFIT_LIMIT -LIMIT_MAKER
- buyTest (string $symbol, $quantity, $price, string $type="LIMIT", array $flags=[])

  buyTest attempts to create a TEST currency order
- sell (string $symbol, $quantity, $price, string $type="LIMIT", array $flags=[])

  sell attempts to create a currency order each currency supports a number of order types, such as -LIMIT -MARKET -STOP_LOSS -STOP_LOSS_LIMIT -TAKE_PROFIT -TAKE_PROFIT_LIMIT -LIMIT_MAKER
- sellTest (string $symbol, $quantity, $price, string $type="LIMIT", array $flags=[])

  sellTest attempts to create a TEST currency order
- marketBuy (string $symbol, $quantity, array $flags=[])

  marketBuy attempts to create a currency order at given market price
- marketBuyTest (string $symbol, $quantity, array $flags=[])

  marketBuyTest attempts to create a TEST currency order at given market price
- marketSell (string $symbol, $quantity, array $flags=[])

  marketSell attempts to create a currency order at given market price
- marketSellTest (string $symbol, $quantity, array $flags=[])

  marketSellTest attempts to create a TEST currency order at given market price
- cancel (string $symbol, $orderid)

  cancel attempts to cancel a currency order
- orderStatus (string $symbol, $orderid)

  orderStatus attempts to get orders status
- openOrders (string $symbol=null)

  openOrders attempts to get open orders for all currencies or a specific currency

- orders (string $symbol, int $limit=500, int $fromOrderId=1)

  orders attempts to get the orders for all or a specific currency
- history (string $symbol, int $limit=500, int $fromTradeId=-1)

  history Get the complete account trade history for all or a specific currency
- useServerTime ()

  useServerTime adds the 'useServerTime'=>true to the API request to avoid time errors
- time ()

  time Gets the server time
- exchangeInfo ()

  exchangeInfo Gets the complete exchange info, including limits, currency options etc.
- withdraw (string $asset, string $address, $amount, $addressTag=null)

  withdraw requests a asset be withdrawn from binance to another wallet
- depositAddress (string $asset)

  depositAddress get the deposit address for an asset
- depositHistory (string $asset=null, array $params=[ ])

  depositAddress get the deposit history for an asset
- withdrawHistory (string $asset=null, array $params=[ ])

  withdrawHistory get the withdrawal history for an asset
- withdrawFee (string $asset)

  withdrawFee get the withdrawal fee for an asset
- prices ()

  prices get all the current prices
- bookPrices ()

  bookPrices get all bid/asks prices
- account ()

  account get all information about the api account
- prevDay (string $symbol=null)

  prevDay get 24hr ticker price change statistics for symbols
- aggTrades (string $symbol)

  aggTrades get Market History / Aggregate Trades
- depth (string $symbol)

  depth get Market depth
- balances ($priceData=false)

  balances get balances for the account assets
- getProxyUriString ()

  getProxyUriString get Uniform Resource Identifier string assocaited with proxy config
- setProxy (array $proxyconf)

  setProxy set proxy config by passing in an array of the proxy configuration
- order (string $side, string $symbol, $quantity, $price, string $type="LIMIT", array $flags=[ ], bool $test=false)

  order formats the orders before sending them to the curl wrapper function You can call this function directly or use the helper functions
- candlesticks (string $symbol, string $interval="5m", int $limit=null, $startTime=null, $endTime=null)

  candlesticks get the candles for the given intervals 1m,3m,5m,15m,30m,1h,2h,4h,6h,8h,12h,1d,3d,1w,1M
- cumulative (array $depth)

  cumulative Converts depth cache into a cumulative array
- highstock (array $chart, bool $include_volume=false)

  highstock Converts Chart Data into array for highstock & kline charts
- first (array $array)

  first Gets first key of an array
- last (array $array)

  last Gets last key of an array

- displayDepth (array $array)

  displayDepth Formats nicely for console output
- getTransfered ()

  getTransfered gets the total transfered in b,Kb,Mb,Gb
- getRequestCount ()

  getRequestCount gets the total number of API calls
- addToTransfered (int $int)

  addToTransfered add interger bytes to the total transfered also incrementes the api counter
- sortDepth (string $symbol, int $limit=11)

  sortDepth Sorts depth data for display & getting highest bid and lowest ask
- depthCache ($symbols, callable $callback)

  depthCache Pulls /depth data and subscribes to WebSocket endpoint Maintains a local Depth Cache in sync via lastUpdateId.
- trades ($symbols, callable $callback)

  trades Trades WebSocket Endpoint
- ticker ($symbol, callable $callback)

  ticker pulls 24h price change statistics via WebSocket
- chart ($symbols, string $interval="30m", callable $callback)

  chart Pulls /kline data and subscribes to WebSocket endpoint
- kline ($symbols, string $interval="30m", callable $callback)

  kline Subscribes to WebSocket endpoint for latest chart data only
- terminate ($endpoint)

  terminate Terminates websocket endpoints.
- keepAlive ()

  keepAlive Keep-alive function for userDataStream
- userData (&$balance_callback, &$execution_callback=false)

  userData Issues userDataStream token and keepalive, subscribes to userData WebSocket
- miniTicker (callable $callback)

  miniTicker Get miniTicker for all symbols
- report ()

  Report.

Protected Attributes

- $base = 'https://api.binance.com/api/'
- $wapi = 'https://api.binance.com/wapi/'
- $stream = 'wss://stream.binance.com:9443/ws/'
- $api_key
- $api_secret
- $depthCache = [ ]
- $depthQueue = [ ]
- $chartQueue = [ ]
- $charts = [ ]
- $curlOpts = [ ]
- $info
- $proxyConf = null
- $transfered = 0
- $requestCount = 0

4.1.1 Detailed Description

Main Binance class.

Eg. Usage: require 'vendor/autoload.php'; $api = new Binance\API();

Definition at line 29 of file php-binance-api.php.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Binance\API::__construct ( )

Constructor for the class, send as many argument as you want.

No arguments - use file setup 1 argument - file to load config from 2 arguments - api key and api secret

Returns

null

Definition at line 64 of file php-binance-api.php.

4.1.3 Member Function Documentation

4.1.3.1 Binance\API::__get (string *$member*)

magic get for private and protected members

Parameters

$file	string the name of the property to return

Returns

null

Definition at line 93 of file php-binance-api.php.

4.1.3.2 Binance\API::__set (string *$member,* *$value*)

magic set for private and protected members

Parameters

$member	string the name of the member property
$value	the value of the member property

Definition at line 107 of file php-binance-api.php.

4.1.3.3 Binance\API::account ( )

account get all information about the api account

$account = $api->account();

Returns

array with error message or array of all the account information

Exceptions

\Exception	

Definition at line 646 of file php-binance-api.php.

4.1.3.4 Binance\API::addToTransfered (int *$int*)

addToTransfered add interger bytes to the total transfered also incrementes the api counter

$apiCount = $api->addToTransfered($int);

Returns

null

Definition at line 1496 of file php-binance-api.php.

4.1.3.5 Binance\API::aggTrades (string *$symbol*)

aggTrades get Market History / Aggregate Trades

$trades = $api->aggTrades("BNBBTC");

Parameters

$symbol	string the symbol to get the trade information for

Returns

array with error message or array of market history

Exceptions

\Exception	

Definition at line 680 of file php-binance-api.php.

4.1.3.6 Binance\API::balances (*$priceData* = `false`)

balances get balances for the account assets

$balances = $api->balances($ticker);

Parameters

bool	*$priceData*	array of the symbols balances are required for

Returns

array with error message or array of balances

Exceptions

\Exception	

Definition at line 721 of file php-binance-api.php.

4.1.3.7 Binance\API::bookPrices ( )

bookPrices get all bid/asks prices

$ticker = $api->bookPrices();

Returns

array with error message or array of all the book prices

Exceptions

\Exception	

Definition at line 633 of file php-binance-api.php.

4.1.3.8 Binance\API::buy (string *$symbol,* *$quantity,* *$price,* string *$type =* "LIMIT", array *$flags =* [])

buy attempts to create a currency order each currency supports a number of order types, such as -LIMIT -MARKET -STOP_LOSS -STOP_LOSS_LIMIT -TAKE_PROFIT -TAKE_PROFIT_LIMIT -LIMIT_MAKER

You should check the

See Also

exchangeInfo for each currency to determine what types of orders can be placed against specific pairs

$quantity = 1; $price = 0.0005; $order = $api->buy("BNBBTC", $quantity, $price);

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$price	string price per unit you want to spend
$type	string type of order
$flags	array addtional options for order type

Returns

array with error message or the order details

Definition at line 220 of file php-binance-api.php.

4.1.3.9 Binance\API::buyTest (string *$symbol,* *$quantity,* *$price,* string *$type =* "LIMIT", array *$flags =* [])

buyTest attempts to create a TEST currency order

See Also

buy()

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$price	string price per unit you want to spend
$type	string config
$flags	array config

Returns

array with error message or empty or the order details

Definition at line 237 of file php-binance-api.php.

4.1.3.10 Binance\API::cancel (string *$symbol*, *$orderid*)

cancel attempts to cancel a currency order

$orderid = "123456789"; $order = $api->cancel("BNBBTC", $orderid);

Parameters

$symbol	string the currency symbol
$orderid	string the orderid to cancel

Returns

array with error message or the order details

Exceptions

\Exception	

Definition at line 362 of file php-binance-api.php.

4.1.3.11 Binance\API::candlesticks (string *$symbol,* string *$interval* = `"5m"`, int *$limit* = `null`, *$startTime* = `null`, *$endTime* = `null`)

candlesticks get the candles for the given intervals 1m,3m,5m,15m,30m,1h,2h,4h,6h,8h,12h,1d,3d,1w,1M

$candles = $api->candlesticks("BNBBTC", "5m");

Parameters

$symbol	string to query
$interval	string to request
$limit	int limit the amount of candles
$startTime	string request candle information starting from here
$endTime	string request candle information ending here

Returns

array containing the response

Exceptions

\Exception	

Definition at line 1005 of file php-binance-api.php.

4.1.3.12 Binance\API::chart (*$symbols,* string *$interval* = `"30m"`, callable *$callback*)

chart Pulls /kline data and subscribes to WebSocket endpoint

$api->chart(["BNBBTC"], "15m", function($api, $symbol, $chart) { echo "{$symbol} chart update\n"; print_r($chart); });

Parameters

$symbols	string required symbols
$interval	string time inteval
$callback	callable closure

Returns

null

Exceptions

\Exception	

Definition at line 1815 of file php-binance-api.php.

4.1.3.13 Binance\API::cumulative (array *$depth*)

cumulative Converts depth cache into a cumulative array

$cumulative = $api->cumulative($depth);

Parameters

$depth	array cache array

Returns

array cumulative depth cache

Definition at line 1314 of file php-binance-api.php.

4.1.3.14 Binance\API::depositAddress (string *$asset*)

depositAddress get the deposit address for an asset

$depositAddress = $api->depositAddress("VEN");

Parameters

$asset	string the currency such as BTC

Returns

array with error message or array deposit address information

Exceptions

\Exception	

Definition at line 543 of file php-binance-api.php.

4.1.3.15 Binance\API::depositHistory (string *$asset* = `null`, array *$params* = `[]`)

depositAddress get the deposit history for an asset

$depositHistory = $api->depositHistory();

$depositHistory = $api->depositHistory("BTC");

Parameters

$asset	string empty or the currency such as BTC
$params	array optional startTime, endTime, status parameters

Returns

array with error message or array deposit history information

Exceptions

\Exception	

Definition at line 564 of file php-binance-api.php.

4.1.3.16 Binance\API::depth (string *$symbol*)

depth get Market depth

$depth = $api->depth("ETHBTC");

Parameters

$symbol	string the symbol to get the depth information for

Returns

array with error message or array of market depth

Exceptions

\Exception	

Definition at line 696 of file php-binance-api.php.

4.1.3.17 Binance\API::depthCache (*$symbols,* callable *$callback*)

depthCache Pulls /depth data and subscribes to WebSocket endpoint Maintains a local Depth Cache in sync via lastUpdateId.

See depth() and depthHandler()

$api->depthCache(["BNBBTC"], function($api, $symbol, $depth) { echo "{$symbol} depth cache update".PHP_EOL; //print_r($depth); // Print all depth data $limit = 11; // Show only the closest asks/bids $sorted = $api->sortDepth($symbol, $limit); $bid = $api->first($sorted['bids']); $ask = $api->first($sorted['asks']); echo $api->displayDepth($sorted); echo "ask: {$ask}".PHP_EOL; echo "bid: {$bid}".PHP_EOL; });

Parameters

$symbol	string optional array of symbols
$callback	callable closure

Returns

null

Definition at line 1617 of file php-binance-api.php.

4.1.3.18 Binance\API::displayDepth (array *$array*)

displayDepth Formats nicely for console output

$outputString = $api->displayDepth($array);

Parameters

$array	array

Returns

string of the depth information

Definition at line 1409 of file php-binance-api.php.

4.1.3.19 Binance\API::exchangeInfo ( )

exchangeInfo Gets the complete exchange info, including limits, currency options etc.

$info = $api->exchangeInfo();

Returns

array with error message or exchange info array

Exceptions

\Exception	

Definition at line 494 of file php-binance-api.php.

4.1.3.20 Binance\API::first (array *$array*)

first Gets first key of an array

$first = $api->first($array);

Parameters

$array	array

Returns

string key or null

Definition at line 1377 of file php-binance-api.php.

4.1.3.21 Binance\API::getProxyUriString ( )

getProxyUriString get Uniform Resource Identifier string assocaited with proxy config

$balances = $api->getProxyUriString();

Returns

string uri

Definition at line 747 of file php-binance-api.php.

4.1.3.22 Binance\API::getRequestCount ( )

getRequestCount gets the total number of API calls

$apiCount = $api->getRequestCount();

Returns

int get the total number of api calls

Definition at line 1483 of file php-binance-api.php.

4.1.3.23 Binance\API::getTransfered ( )

getTransfered gets the total transfered in b,Kb,Mb,Gb

$transfered = $api->getTransfered();

Returns

string showing the total transfered

Definition at line 1463 of file php-binance-api.php.

4.1.3.24 Binance\API::highstock (array *$chart*, bool *$include_volume* = `false`)

highstock Converts Chart Data into array for highstock & kline charts

$highstock = $api->highstock($chart, $include_volume);

Parameters

$chart	array
$include_volume	bool for inclusion of volume

Returns

array highchart data

Definition at line 1349 of file php-binance-api.php.

4.1.3.25 Binance\API::history (string *$symbol*, int *$limit* = `500`, int *$fromTradeId* = `-1`)

history Get the complete account trade history for all or a specific currency

$allHistory = $api->history(); $BNBHistory = $api->history("BNBBTC"); $limitBNBHistory = $api->history("BNBBTC",5); $limitBNBHistoryFromId = $api->history("BNBBTC",5,3);

Parameters

$symbol	string the currency symbol
$limit	int the amount of orders returned
$fromTradeId	int (optional) return the orders from this order onwards. negative for all

Returns

array with error message or array of orderDetails array

Exceptions

\Exception	

Definition at line 444 of file php-binance-api.php.

4.1.3.26 Binance\API::keepAlive ( )

keepAlive Keep-alive function for userDataStream

$api->keepAlive();

Returns

null

Definition at line 1956 of file php-binance-api.php.

4.1.3.27 Binance\API::kline (*$symbols*, string *$interval* = "30m", callable *$callback*)

kline Subscribes to WebSocket endpoint for latest chart data only

$api->kline(["BNBBTC"], "15m", function($api, $symbol, $chart) { echo "{$symbol} chart update\n"; print_r($chart); });

Parameters

$symbols	string required symbols
$interval	string time inteval
$callback	callable closure

Returns

null

Exceptions

\Exception	

Definition at line 1896 of file php-binance-api.php.

4.1.3.28 Binance\API::last (array *$array*)

last Gets last key of an array

$last = $api->last($array);

Parameters

$array	array

Returns

string key or null

Definition at line 1393 of file php-binance-api.php.

4.1.3.29 Binance\API::marketBuy (string *$symbol*, *$quantity*, array *$flags* = [])

marketBuy attempts to create a currency order at given market price

$quantity = 1; $order = $api->marketBuy("BNBBTC", $quantity);

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$flags	array addtional options for order type

Returns

array with error message or the order details

Definition at line 300 of file php-binance-api.php.

4.1.3.30 Binance\API::marketBuyTest (string *$symbol*, *$quantity*, array *$flags* = `[]`)

marketBuyTest attempts to create a TEST currency order at given market price

See Also

marketBuy()

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$flags	array addtional options for order type

Returns

array with error message or the order details

Definition at line 315 of file php-binance-api.php.

4.1.3.31 Binance\API::marketSell (string *$symbol*, *$quantity*, array *$flags* = `[]`)

marketSell attempts to create a currency order at given market price

$quantity = 1; $order = $api->marketSell("BNBBTC", $quantity);

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$flags	array addtional options for order type

Returns

array with error message or the order details

Definition at line 331 of file php-binance-api.php.

4.1.3.32 Binance\API::marketSellTest (string *$symbol*, *$quantity*, array *$flags* = `[]`)

marketSellTest attempts to create a TEST currency order at given market price

See Also

marketSellTest()

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$flags	array addtional options for order type

Returns

array with error message or the order details

Definition at line 346 of file php-binance-api.php.

4.1.3.33 Binance\API::miniTicker (callable *$callback*)

miniTicker Get miniTicker for all symbols

$api->miniTicker(function($api, $ticker) { print_r($ticker); });

Parameters

$callback	callable function closer that takes 2 arguments, $pai and $ticker data

Returns

null

Definition at line 2055 of file php-binance-api.php.

4.1.3.34 Binance\API::openOrders (string *$symbol* = `null`)

openOrders attempts to get open orders for all currencies or a specific currency

$allOpenOrders = $api->openOrders(); $allBNBOrders = $api->openOrders("BNBBTC");

Parameters

$symbol	string the currency symbol

Returns

array with error message or the order details

Exceptions

\Exception	

Definition at line 399 of file php-binance-api.php.

4.1.3.35 Binance\API::order (string *$side,* string *$symbol,* *$quantity,* *$price,* string *$type* = `"LIMIT"`, array *$flags* = `[]`, bool *$test* = `false`)

order formats the orders before sending them to the curl wrapper function You can call this function directly or use the helper functions

See Also

buy()
sell()
marketBuy()
marketSell() $this->httpRequest("https://api.binance.com/api/v1/ticker/24hr");

Parameters

$side	string typically "BUY" or "SELL"
$symbol	string to buy or sell
$quantity	string in the order
$price	string for the order
$type	string is determined by the symbol bu typicall LIMIT, STOP_LOSS_LIMIT etc.
$flags	array additional transaction options
$test	bool whether to test or not, test only validates the query

Returns

array containing the response

Exceptions

\Exception	

Definition at line 943 of file php-binance-api.php.

4.1.3.36 Binance\API::orders (string *$symbol*, int *$limit* = `500`, int *$fromOrderId* = `1`)

orders attempts to get the orders for all or a specific currency

$allBNBOrders = $api->orders("BNBBTC");

Parameters

$symbol	string the currency symbol
$limit	int the amount of orders returned
$fromOrderId	string return the orders from this order onwards

Returns

array with error message or array of orderDetails array

Exceptions

\Exception	

Definition at line 421 of file php-binance-api.php.

4.1.3.37 Binance\API::orderStatus (string *$symbol*, *$orderid*)

orderStatus attempts to get orders status

$orderid = "123456789"; $order = $api->orderStatus("BNBBTC", $orderid);

Parameters

$symbol	string the currency symbol
$orderid	string the orderid to cancel

Returns

array with error message or the order details

Exceptions

\Exception	

Definition at line 381 of file php-binance-api.php.

4.1.3.38 Binance\API::prevDay (string *$symbol* = `null`)

prevDay get 24hr ticker price change statistics for symbols

$prevDay = $api->prevDay("BNBBTC");

Parameters

$symbol	(optional) symbol to get the previous day change for

Returns

array with error message or array of prevDay change

Exceptions

\Exception	

Definition at line 660 of file php-binance-api.php.

4.1.3.39 Binance\API::prices ( )

prices get all the current prices

$ticker = $api->prices();

Returns

array with error message or array of all the currencies prices

Exceptions

\Exception	

Definition at line 620 of file php-binance-api.php.

4.1.3.40 Binance\API::report ( )

Report.

Definition at line 2133 of file php-binance-api.php.

4.1.3.41 Binance\API::sell (string *$symbol*, *$quantity*, *$price*, string *$type* = `"LIMIT"`, array *$flags* = `[]`)

sell attempts to create a currency order each currency supports a number of order types, such as -LIMIT -MARKET -STOP_LOSS -STOP_LOSS_LIMIT -TAKE_PROFIT -TAKE_PROFIT_LIMIT -LIMIT_MAKER

You should check the

See Also

exchangeInfo for each currency to determine what types of orders can be placed against specific pairs

$quantity = 1; $price = 0.0005; $order = $api->sell("BNBBTC", $quantity, $price);

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$price	string price per unit you want to spend
$type	string type of order
$flags	array addtional options for order type

Returns

array with error message or the order details

Definition at line 267 of file php-binance-api.php.

4.1.3.42 Binance\API::sellTest (string *$symbol*, *$quantity*, *$price*, string *$type* = `"LIMIT"`, array *$flags* = `[]`)

sellTest attempts to create a TEST currency order

See Also

sell()

Parameters

$symbol	string the currency symbol
$quantity	string the quantity required
$price	string price per unit you want to spend
$type	array config
$flags	array config

Returns

array with error message or empty or the order details

Definition at line 284 of file php-binance-api.php.

4.1.3.43 Binance\API::setProxy (array *$proxyconf*)

setProxy set proxy config by passing in an array of the proxy configuration

$proxyConf = [’proto’ => ’tcp’, ’address’ => ’192.168.1.1’, ’port’ => ’8080’, ’user’ => ’dude’, ’pass’ => ’d00d’];

$api->setProxy($proxyconf);

Returns

null

Definition at line 799 of file php-binance-api.php.

4.1.3.44 Binance\API::sortDepth (string *$symbol*, int *$limit* = `11`)

sortDepth Sorts depth data for display & getting highest bid and lowest ask

$sorted = $api->sortDepth($symbol, $limit);

Parameters

$symbol	string to sort
$limit	int depth

Returns

null

Definition at line 1584 of file php-binance-api.php.

4.1.3.45 Binance\API::terminate (*$endpoint*)

terminate Terminates websocket endpoints.

View endpoints first: print_r($api->subscriptions)

$api->terminate('ethbtc_kline@5m');

Returns

null

Definition at line 1943 of file php-binance-api.php.

4.1.3.46 Binance\API::ticker (*$symbol,* callable *$callback*)

ticker pulls 24h price change statistics via WebSocket

$api->ticker(false, function($api, $symbol, $ticker) { print_r($ticker); });

Parameters

$symbol	string optional symbol or false
$callback	callable closure

Returns

null

Definition at line 1765 of file php-binance-api.php.

4.1.3.47 Binance\API::time ( )

time Gets the server time

$time = $api->time();

Returns

array with error message or array with server time key

Exceptions

\Exception	

Definition at line 481 of file php-binance-api.php.

4.1.3.48 Binance\API::trades (*$symbols,* callable *$callback*)

trades Trades WebSocket Endpoint

$api->trades(["BNBBTC"], function($api, $symbol, $trades) { echo "{$symbol} trades update".PHP_EOL; print_-r($trades); });

Parameters

$symbols	
$callback	callable closure

Returns

null

Definition at line 1697 of file php-binance-api.php.

4.1.3.49 Binance\API::userData (& *$balance_callback,* & *$execution_callback* = `false`)

userData Issues userDataStream token and keepalive, subscribes to userData WebSocket

$balance_update = function($api, $balances) { print_r($balances); echo "Balance update".PHP_EOL; };

$order_update = function($api, $report) { echo "Order update".PHP_EOL; print_r($report); $price = $report['price']; $quantity = $report['quantity']; $symbol = $report['symbol']; $side = $report['side']; $orderType = $report['orderType']; $orderId = $report['orderId']; $orderStatus = $report['orderStatus']; $executionType = $report['orderStatus']; if($executionType == "NEW") { if($executionType == "REJECTED") { echo "Order Failed! Reason: {$report['rejectReason']}".PHP_EOL; } echo "{$symbol} {$side} {$orderType} ORDER #{$orderId} ({$orderStatus})".PHP_EOL; echo "..price: {$price}, quantity: {$quantity}".PHP_EOL; return; }

//NEW, CANCELED, REPLACED, REJECTED, TRADE, EXPIRED echo "{$symbol} {$side} {$executionType} {$orderType} ORDER #{$orderId}".PHP_EOL; }; $api->userData($balance_update, $order_update);

Parameters

	$balance_callback	callable function
bool	*$execution_callback*	callable function

Returns

null

Exceptions

\Exception	

Definition at line 2004 of file php-binance-api.php.

4.1.3.50 Binance\API::useServerTime ( )

useServerTime adds the 'useServerTime'=>true to the API request to avoid time errors

$api->useServerTime();

Returns

null

Exceptions

\Exception	

Definition at line 465 of file php-binance-api.php.

4.1.3.51 Binance\API::withdraw (string *$asset,* string *$address,* *$amount,* *$addressTag* = null)

withdraw requests a asset be withdrawn from binance to another wallet

$asset = "BTC"; $address = "1C5gqLRs96Xq4V2ZZAR1347yUCpHie7sa"; $amount = 0.2; $response = $api->withdraw($asset, $address, $amount);

$address = "44tLjmXrQNrWJ5NBsEj2R77ZBEgDa3fEe9GLpSf2FRmhexPvfYDUAB7EXX1Hdb3aMQ9FLqdJ56ya-AhiXoRsceGJCRS3Jxkn"; $addressTag = "0e5e38a01058dbf64e53a4333a5acf98e0d5feb8e523d32e3186c664a9c762c1"; $amount = 0.1; $response = $api->withdraw($asset, $address, $amount, $addressTag);

Parameters

$asset	string the currency such as BTC
$address	string the addressed to whihc the asset should be deposited
$amount	double the amount of the asset to transfer
$addressTag	string adtional transactionid required by some assets

Returns

array with error message or array transaction

Exceptions

\Exception	

Definition at line 519 of file php-binance-api.php.

4.1.3.52 Binance\API::withdrawFee (string *$asset*)

withdrawFee get the withdrawal fee for an asset

$withdrawFee = $api->withdrawFee("BTC");

Parameters

$asset	string currency such as BTC

Returns

array with error message or array containing withdrawFee

Exceptions

\Exception	

Definition at line 603 of file php-binance-api.php.

4.1.3.53 Binance\API::withdrawHistory (string *$asset* = null, array *$params* = [])

withdrawHistory get the withdrawal history for an asset

$withdrawHistory = $api->withdrawHistory();

$withdrawHistory = $api->withdrawHistory("BTC");

Parameters

$asset	string empty or the currency such as BTC
$params	array optional startTime, endTime, status parameters

Returns

array with error message or array deposit history information

Exceptions

\Exception	

Definition at line 585 of file php-binance-api.php.

4.1.4 Member Data Documentation

4.1.4.1 Binance\API::$api_key [protected]

Definition at line 34 of file php-binance-api.php.

4.1.4.2 Binance\API::$api_secret [protected]

Definition at line 35 of file php-binance-api.php.

4.1.4.3 Binance\API::$base = 'https://api.binance.com/api/' [protected]

Definition at line 31 of file php-binance-api.php.

4.1.4.4 Binance\API::$chartQueue = [ ] [protected]

Definition at line 38 of file php-binance-api.php.

4.1.4.5 Binance\API::$charts = [ ] [protected]

Definition at line 39 of file php-binance-api.php.

4.1.4.6 Binance\API::$curlOpts = [ ] [protected]

Definition at line 40 of file php-binance-api.php.

4.1.4.7 Binance\API::$depthCache = [ ] [protected]

Definition at line 36 of file php-binance-api.php.

4.1.4.8 Binance\API::$depthQueue = [ ] [protected]

Definition at line 37 of file php-binance-api.php.

4.1.4.9 Binance\API::$info [protected]

Initial value:

```
= [
        "timeOffset" => 0
```

Definition at line 41 of file php-binance-api.php.

4.1.4.10 Binance\API::$proxyConf = null [protected]

Definition at line 44 of file php-binance-api.php.

4.1.4.11 Binance\API::$requestCount = 0 [protected]

Definition at line 46 of file php-binance-api.php.

4.1.4.12 Binance\API::$stream = 'wss://stream.binance.com:9443/ws/' [protected]

Definition at line 33 of file php-binance-api.php.

4.1.4.13 Binance\API::$transfered = 0 [protected]

Definition at line 45 of file php-binance-api.php.

4.1.4.14 Binance\API::$wapi = 'https://api.binance.com/wapi/' [protected]

Definition at line 32 of file php-binance-api.php.

Index