PHP Binance Api

Generated by Doxygen 1.8.6

Sun Apr 15 2018 22:27:22

Contents

1 Namespace Index **1**

1.1 Namespace List . . . 1

2 Class Index **3**

2.1 Class List . . . 3

3 Namespace Documentation **5**

3.1 Binance Namespace Reference . . . 5

4 Class Documentation **7**

4.1 Binance\API Class Reference . . . 7

4.1.1 Detailed Description . . . 9

4.1.2 Constructor & Destructor Documentation . . . 10

4.1.2.1 __construct . . . 10

4.1.3 Member Function Documentation . . . 10

4.1.3.1 __get . . . 10

4.1.3.2 __set . . . 10

4.1.3.3 account . . . 10

4.1.3.4 addToTransfered . . . 10

4.1.3.5 aggTrades . . . 11

4.1.3.6 balances . . . 11

4.1.3.7 bookPrices . . . 11

4.1.3.8 buy . . . 12

4.1.3.9 buyTest . . . 12

4.1.3.10 cancel . . . 12

4.1.3.11 candlesticks . . . 13

4.1.3.12 chart . . . 13

4.1.3.13 cumulative . . . 14

4.1.3.14 depositAddress . . . 14

4.1.3.15 depositHistory . . . 14

4.1.3.16 depth . . . 15

4.1.3.17 depthCache . . . 15

4.1.3.18 displayDepth . . . 15
4.1.3.19 exchangeInfo . . . 16
4.1.3.20 first . . . 16
4.1.3.21 getProxyUriString . . . 16
4.1.3.22 getRequestCount . . . 16
4.1.3.23 getTransfered . . . 17
4.1.3.24 highstock . . . 17
4.1.3.25 history . . . 17
4.1.3.26 keepAlive . . . 18
4.1.3.27 kline . . . 18
4.1.3.28 last . . . 18
4.1.3.29 marketBuy . . . 18
4.1.3.30 marketBuyTest . . . 19
4.1.3.31 marketSell . . . 19
4.1.3.32 marketSellTest . . . 19
4.1.3.33 miniTicker . . . 20
4.1.3.34 openOrders . . . 20
4.1.3.35 order . . . 20
4.1.3.36 orders . . . 21
4.1.3.37 orderStatus . . . 21
4.1.3.38 prevDay . . . 22
4.1.3.39 prices . . . 22
4.1.3.40 sell . . . 22
4.1.3.41 sellTest . . . 23
4.1.3.42 setProxy . . . 23
4.1.3.43 sortDepth . . . 23
4.1.3.44 terminate . . . 24
4.1.3.45 ticker . . . 24
4.1.3.46 time . . . 24
4.1.3.47 trades . . . 25
4.1.3.48 userData . . . 26
4.1.3.49 useServerTime . . . 26
4.1.3.50 withdraw . . . 27
4.1.3.51 withdrawHistory . . . 27
4.1.4 Member Data Documentation . . . 27
4.1.4.1 $api_key . . . 27
4.1.4.2 $api_secret . . . 27
4.1.4.3 $base . . . 28
4.1.4.4 $chartQueue . . . 28
4.1.4.5 $charts . . . 28

4.1.4.6 $curlOpts 28
4.1.4.7 $depthCache 28
4.1.4.8 $depthQueue 28
4.1.4.9 $info 28
4.1.4.10 $proxyConf 28
4.1.4.11 $requestCount 28
4.1.4.12 $stream 28
4.1.4.13 $transfered 28
4.1.4.14 $wapi 29

Index **30**

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Binance 5

Chapter 2

Class Index

2.1 Class List

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

Binance\API
Main Binance class 7

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 agument 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)

  depositAddress get the deposit history for an asset
- withdrawHistory (string $asset=null)

  withdrawHistory get the withdrawal history 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

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 agument as you want.

Returns

null

Definition at line 60 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 165 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 178 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 669 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 1426 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 701 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 739 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 657 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 287 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 303 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 421 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 983 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 1713 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 1256 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 587 of file php-binance-api.php.

4.1.3.15 Binance\API::depositHistory (string *$asset* = `null`)

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

Returns

array with error message or array deposit history information

Exceptions

\Exception	

Definition at line 606 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 716 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 1535 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 1345 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 540 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 1315 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 752 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 1414 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 1395 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 1290 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 return the orders from this order onwards

Returns

array with error message or array of orderDetails array

Exceptions

\Exception	

Definition at line 499 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 1835 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 1781 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 1330 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 363 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 377 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 392 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 406 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 1931 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 456 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 924 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 477 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 439 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 682 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 645 of file php-binance-api.php.

4.1.3.40 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 332 of file php-binance-api.php.

4.1.3.41 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 348 of file php-binance-api.php.

4.1.3.42 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 800 of file php-binance-api.php.

4.1.3.43 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 1503 of file php-binance-api.php.

4.1.3.44 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 1823 of file php-binance-api.php.

4.1.3.45 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 1665 of file php-binance-api.php.

4.1.3.46 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 528 of file php-binance-api.php.

4.1.3.47 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 1604 of file php-binance-api.php.

4.1.3.48 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 1882 of file php-binance-api.php.

4.1.3.49 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 515 of file php-binance-api.php.

4.1.3.50 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 = "44tLjmXrQNrWJ5NBsEj2R77ZBEgDa3fEe9GLpSf2FRmhexPvfYDUAB7EXX1Hdb3aMQ9FLqdJ56yaAhiXoRsceGJCRS3Jxkn"; $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 564 of file php-binance-api.php.

4.1.3.51 Binance\API::withdrawHistory (string *$asset =* null)

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

Returns

array with error message or array deposit history information

Exceptions

\Exception	

Definition at line 627 of file php-binance-api.php.

4.1.4 Member Data Documentation

4.1.4.1 Binance\API::$api_key [protected]

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

4.1.4.2 Binance\API::$api_secret [protected]

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

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

Definition at line 30 of file php-binance-api.php.

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

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

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

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

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

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

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

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

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

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

4.1.4.9 Binance\API::$info [protected]

Initial value:

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

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

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

Definition at line 43 of file php-binance-api.php.

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

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

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

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

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

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

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

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

Index

$api_key
    Binance::API, 27
$api_secret
    Binance::API, 27
$base
    Binance::API, 27
$chartQueue
    Binance::API, 28
$charts
    Binance::API, 28
$curlOpts
    Binance::API, 28
$depthCache
    Binance::API, 28
$depthQueue
    Binance::API, 28
$info
    Binance::API, 28
$proxyConf
    Binance::API, 28
$requestCount
    Binance::API, 28
$stream
    Binance::API, 28
$transfered
    Binance::API, 28
$wapi
    Binance::API, 28
__construct
    Binance::API, 10
__get
    Binance::API, 10
__set
    Binance::API, 10

account
    Binance::API, 10
addToTransfered
    Binance::API, 10
aggTrades
    Binance::API, 11

balances
    Binance::API, 11
Binance, 5
Binance::API
    $api_key, 27
    $api_secret, 27
    $base, 27
    $chartQueue, 28
    $charts, 28
    $curlOpts, 28
    $depthCache, 28
    $depthQueue, 28
    $info, 28
    $proxyConf, 28
    $requestCount, 28
    $stream, 28
    $transfered, 28
    $wapi, 28
    __construct, 10
    __get, 10
    __set, 10
    account, 10
    addToTransfered, 10
    aggTrades, 11
    balances, 11
    bookPrices, 11
    buy, 12
    buyTest, 12
    cancel, 12
    candlesticks, 13
    chart, 13
    cumulative, 14
    depositAddress, 14
    depositHistory, 14
    depth, 15
    depthCache, 15
    displayDepth, 15
    exchangeInfo, 16
    first, 16
    getProxyUriString, 16
    getRequestCount, 16
    getTransfered, 17
    highstock, 17
    history, 17
    keepAlive, 17
    kline, 18
    last, 18
    marketBuy, 18
    marketBuyTest, 19
    marketSell, 19
    marketSellTest, 19
    miniTicker, 20
    openOrders, 20
    order, 20
    orderStatus, 21
    orders, 21
    prevDay, 22

prices, 22
sell, 22
sellTest, 23
setProxy, 23
sortDepth, 23
terminate, 24
ticker, 24
time, 24
trades, 24
useServerTime, 26
userData, 26
withdraw, 26
withdrawHistory, 27
Binance\API, 7
bookPrices
Binance::API, 11
buy
Binance::API, 12
buyTest
Binance::API, 12

cancel
Binance::API, 12
candlesticks
Binance::API, 13
chart
Binance::API, 13
cumulative
Binance::API, 14

depositAddress
Binance::API, 14
depositHistory
Binance::API, 14
depth
Binance::API, 15
depthCache
Binance::API, 15
displayDepth
Binance::API, 15

exchangeInfo
Binance::API, 16

first
Binance::API, 16

getProxyUriString
Binance::API, 16
getRequestCount
Binance::API, 16
getTransfered
Binance::API, 17

highstock
Binance::API, 17
history
Binance::API, 17

keepAlive
Binance::API, 17
kline
Binance::API, 18

last
Binance::API, 18

marketBuy
Binance::API, 18
marketBuyTest
Binance::API, 19
marketSell
Binance::API, 19
marketSellTest
Binance::API, 19
miniTicker
Binance::API, 20

openOrders
Binance::API, 20
order
Binance::API, 20
orderStatus
Binance::API, 21
orders
Binance::API, 21

prevDay
Binance::API, 22
prices
Binance::API, 22

sell
Binance::API, 22
sellTest
Binance::API, 23
setProxy
Binance::API, 23
sortDepth
Binance::API, 23

terminate
Binance::API, 24
ticker
Binance::API, 24
time
Binance::API, 24
trades
Binance::API, 24

useServerTime
Binance::API, 26
userData
Binance::API, 26

withdraw
Binance::API, 26
withdrawHistory
Binance::API, 27