HITBTC

RESET:

|  |
| --- |
| https://api.hitbtc.com/api/2 |
|  |

TICKET: GET /api/2/public/ticker

Gate io

**REST API BaseURL:**

* Live trading: https://api.gateio.ws/api/v4
* Futures TestNet trading: https://fx-api-testnet.gateio.ws/api/v4
* Futures live trading alternative (futures only): <https://fx-api.gateio.ws/api/v4>

**Retrieve ticker information**

GET /spot/tickers

*Retrieve ticker information*

Return only related data if currency\_pair is specified; otherwise return all of them

# coding: utf-8

import requests

host = "https://api.gateio.ws"

prefix = "/api/v4"

headers = {'Accept': 'application/json', 'Content-Type': 'application/json'}

url = '/spot/tickers'

query\_param = ''

r = requests.request('GET', host + prefix + url, headers=headers)

print(r.json())

**Retrieve market trades**

GET /spot/trades

*Retrieve market trades*

You can use from and to to query by time range, or use last\_id by scrolling page. The default behavior is by time range.

Scrolling query using last\_id is not recommended any more. If last\_id is specified, time range query parameters will be ignored.

# coding: utf-8

import requests

host = "https://api.gateio.ws"

prefix = "/api/v4"

headers = {'Accept': 'application/json', 'Content-Type': 'application/json'}

url = '/spot/trades'

query\_param = 'currency\_pair=BTC\_USDT'

r = requests.request('GET', host + prefix + url + "?" + query\_param, headers=headers)

print(r.json())

Bittrex:

* All REST requests must be sent to https://api.bittrex.com/v3 using the application/json content type. Non-HTTPS requests will be redirected to HTTPS, possibly causing functional or performance issues with your application.

GET /markets/tickers

List tickers for all markets.

<https://api.bittrex.com/v3/markets/tickers>

[ { "symbol": "string", "lastTradeRate": "number (double)", "bidRate": "number (double)", "askRate": "number (double)" } ]

HUOBI

Dear API user,

Since the effective day of this notification, Huobi Global is going to add a new RESTful API endpoint – “GET /v2/account/ledger”.

GET /v2/account/ledger

## Overview

| **Category** | **URL Path** | **Description** |
| --- | --- | --- |
| Common | /v1/common/\* | Common interface, including currency, currency pair, timestamp, etc |
| Market Data | /market/\* | Market data interface, including trading, depth, quotation, etc |
| Account | /v1/account/\* /v1/subuser/\* | Account interface, including account information, sub-user ,etc |
| Order | /v1/order/\* | Order interface, including order creation, cancellation, query, etc |
| Margin | /v1/margin/\* | Margin interface, including debit, payment, query, etc |
| Cross Margin | /v1/cross-margin/\* | Cross margin interface, including debit, payment, query, etc |

.

Above is a general category, it doesn't cover all API, you can refer to detailed API document according to your requirement.

New Version Rate limit Rule

• The new version rate limit is applied on UID basis, which means, the overall access rate, from all API keys under same UID, to single endpoint, shouldn’t exceed the rate limit applied on that endpoint.

• It is suggested to read HTTP Header X-HB-RateLimit-Requests-Remain and X-HB-RateLimit-Requests-Expire to get the remaining count of request and the expire time for current rate limit time window, then you can adjust the API access rate dynamically.

Request Format

The API is restful and there are two method: GET and POST.

• GET request: All parameters are included in URL, and do not carry body(content-length>0), in otherwise will return 403 error code.

• POST request: All parameters are formatted as JSON and put int the request body

Response Format

The response is JSON format.There are four fields in the top level: status, ch, ts and data. The first three fields indicate the general status, the business data is is under data field.

Below is an example of response:

{

"status": "ok",

"ch": "market.btcusdt.kline.1day",

"ts": 1499223904680,

"data": // per API response data in nested JSON object

}

Field Data Type Description

status string Status of API response

ch string The data stream. It may be empty as some API doesn't have data stream

ts int The UTC timestamp when API respond, the unit is millisecond

data object The body data in response

Data Type

The JSON data type described in this document is defined as below:

• string: a sequence of characters that are quoted

• int: a 32-bit integer, mainly used for status code, size and count

• long: a 64-bit integer, mainly used for Id and timestamp

• float: a fraction represented in decimal format, mainly used for volume and price, recommend to use high precision decimal data types in program

Best Practice

Security

• It is strongly suggested to bind your IP with your API Key to ensure that your API Key can only be used in your machine. Furthermore, your API Key will be expired after 90 days if it is not binded with any IP.

• It is strongly suggested not to share your API Key with any body or third-party software, otherwise your personal information and asset may be stolen. If your expose your API Key by accident, please do delete the API Key and create a new one.

General

API Access

• It is suggested not to use temporary domain or proxy, which may be not stable.

• It is suggested to use AWS Japan to access API for lower latency

• It is suggested to connect to domain api-aws.huobi.pro if your server is based on AWS, because this domain is optimized for AWS client, the latency will be lower.

New Version Rate limit Rule

• Only those endpoints marked with rate limit value separately are applied with new rate limit rule.

• It is suggested to read HTTP Header X-HB-RateLimit-Requests-Remain and X-HB-RateLimit-Requests-Expire to get the remaining count of request and the expire time for current rate limit time window, then you can adjust the API access rate dynamically.

• The overall access rate, from all API keys under same UID, to single endpoint, shouldn’t exceed the rate limit applied on that endpoint.

Market

Market data

• It is suggested to use WebSocket interface to subscribe the market update and then cache the data locally, because WebSocket notification has lower latency and not have rate limit.

• It is suggested not to subscribe too many topics in a single websocket connection, it may generate more notifications and cause network latency and disconnection.

Latest trade

• It is suggested to subscribe WebSocket topic market.$symbol.trade.detail, the response field price represents the latest price, and it has lower latency.

• It is suggested to use tradeId to de-duplicate if you subscribe WebSocket topic market.$symbol.trade.detail.

Depth

• It is suggested to subscribe WebSocket topic market.$symbol.bbo if you only need the best bid and best offer.

• It is suggested to subscribe WebSocket topic market.$symbol.depth.$type if you need multiple bid and offer with normal latency.

• It is suggested to subscribe WebSocket topic market.$symbol.mbp.$level if you need multiple bid and offer with lower latency

• It is suggested to use version field to de-duplicate and discard the smaller data if you use Rest interface /market/depth and WebSocket topic market.$symbol.depth.$type. It is suggest to use seqNum to de-duplicate and discard the smaller data if yo subscribe WebSocket topic market.$symbol.mbp.$levels.

Order

Place an order (/v1/order/orders/place)

• It is suggested to follow the symbol reference (/v1/common/symbols) to validate the amount and value before placing the older, otherwise you may place an invalid order and waste your time.

• It is suggested to provide an unique client-order-id field when placing the order, it is useful to track your orders status if you fail to get the order id response. Later you can use the client-order-id to match the WebSocket order notification or query order detail by interface /v1/order/orders/getClientOrder.The uniqueness of the clientOrderId passed in when you place an order will no longer be verified. We recommend you to manage clientOrderId by yourself to ensure its uniqueness. If multiple orders use the same clientOrderId, the latest order corresponding to the clientOrderId will be returned when querying/canceling an order.

Search history olders (/v1/order/orders)

• It is recommended to use start-time and end-time to query, that are two timestamps with 13 digits (millisecond). The maximum query time window is 48 hours (2 days), the more precision you provide, the better performance you will get. You can query for multiple iterations.

Order update

• It is suggested to subscribe WebSocket topic orders.$symbol, which has lower latency and more accurate sequence.

Account

Asset update

• It is suggested to subscribe both WebSocket topic orders.$symbol and account.update#${mode}. The former one tells the order status update and arrives earlier than the latter one, and the latter one confirms the final asset balance.

• It is suggested not to subscribe WebSocket topic accounts, which is replaced by accounts.update#${mode}, and will be retired later.

API Explorer

API Explorer allows user to invoke and observe each API request and response without writing any program. The UI is designed as the same as document, which has input parameters and response description, user can use it easily without any additional user guide.

This Explorer encapsulates a shared API Key, and will show the signature calcuation steps and request parameters when it invokes API. If you encounter signature problem, you can copy the API Key and timestamp to your program and compare with the result in Explore

Market data APIs provide public market information such as varies of candlestick, depth and trade information.

The market data is updated **once per second**.

Get Klines(Candles)

This endpoint retrieves all klines in a specific range.

HTTP Request

GET /market/history/kline

BIT GET

Public

Get Server Time

Rate Limit: 20 times/1s (IP)

**HTTP Request**

Obtain server time

GET /api/spot/v1/public/time

Get All Tickers

Rate Limit: 20 times/1s (IP)

**HTTP Request**

GET /api/spot/v1/market/tickers

coinex

API Basic Information

Base URL：https://api.coinex.com/v1

All API request headers need to add Content-Type: "application/json" and User-Agent

APIs that require signature need to add authorization in the header

* Please first check the HTTP status code of Response, and make sure that it indicates 200 before parsing Response
* All Requests and Responses use json.
* Account and transaction-related API endpoints require signatures, while market-related API endpoints do not

# **Market Data API**

## Acquire All Market List

Example:

{

"code": 0,

"data": [

"LTCBCH",

"ETHBCH",

"ZECBCH",

"DASHBCH"

],

"message": "Ok"

}

GET /market/list

Get all market list, applicable to spot and margin markets

Crypto.com

REST API v1 and v2 have separate endpoints.

Future versions will share the same endpoint as v2, but with the version incremented.

 Note: REST API V2 requests need to be sent as "Content Type: application/json"

Documentation for the old REST API V1 can be found [here](https://crypto.com/exchange-docs-v1).

### UAT Sandbox

#### REST API v2

**https://uat-api.3ona.co/v2/{method}**

#### REST API v1

**https://uat-api.3ona.co/v1/{method}**

### Production

#### REST API v2

**https://api.crypto.com/v2/{method}**

#### REST API v1

**https://api.crypto.com/v1/{method}**

## ublic/get-ticker

Request Sample

https://{URL}/v2/public/get-ticker?instrument\_name=BTC\_USDT

Response Sample

{

"code":0,

"method":"public/get-ticker",

"result":{

"data": [

{"i":"CRO\_BTC","b":0.00000890,"k":0.00001179,"a":0.00001042,"t":1591770793901,"v":14905879.59,"h":0.00,"l":0.00,"c":0.00},

{"i":"EOS\_USDT","b":2.7676,"k":2.7776,"a":2.7693,"t":1591770798500,"v":774.51,"h":0.05,"l":0.05,"c":0.00},

{"i":"BCH\_USDT","b":247.49,"k":251.73,"a":251.67,"t":1591770797601,"v":1.01693,"h":0.01292,"l":0.01231,"c":-0.00047},

...

...

{"i":"ETH\_USDT","b":239.92,"k":242.59,"a":240.30,"t":1591770798701,"v":0.97575,"h":0.01236,"l":0.01199,"c":-0.00018},

{"i":"ETH\_CRO","b":2693.11,"k":2699.84,"a":2699.55,"t":1591770795053,"v":95.680,"h":8.218,"l":7.853,"c":-0.050}

]

}

}

Fetches the public tickers for an instrument (e.g. BTC\_USDT).

instrument\_name can be omitted to show tickers for **all** instruments

UPBIT

# Creating an Authorizable Request

[Suggest Edits](https://global-docs.upbit.com/edit/create-authorization-request)

## Make a REST API Request

For POST, PUT, and DELETE requests, set the HTTP body value as JSON format if exists. An example of the valid content type is:

Content-Type: application/json; charset=utf-8

# Tickers

GEThttps://EXCHANGE-REGION-ENDPOINT-URL.com/v1/ticker

MEXC

API Library

We provide developers with SDKs in five languages: Python, DotNET, Java, Javascript, and Go, and provide users with methods to call APIs directly through the SDK. Currently supports all interfaces in spot.

<https://github.com/mxcdevelop/mexc-api-sdk>

**Test Connectivity**

Response

{}

* **GET** /api/v3/ping

Test connectivity to the Rest API.

Parameter：

NONE

**Check Server Time**

Response

{

"serverTime" : 1645539742000

}

* **GET** /api/v3/time

Parameter：

NONE

**Exchange Information**

Response

{

"symbol": "TOMO3LUSDT",

"status": "ENABLED",

"baseAsset": "TOMO3L",

"baseAssetPrecision": 2,

"quoteAsset": "USDT",

"quotePrecision": 3,

"quoteAssetPrecision": 3,

"baseCommissionPrecision": 2,

"quoteCommissionPrecision": 3,

"orderTypes": [

"LIMIT",

"LIMIT\_MAKER"

],

"quoteOrderQtyMarketAllowed": false,

"isSpotTradingAllowed": false,

"isMarginTradingAllowed": false,

"quoteAmountPrecision": "5",

"baseSizePrecision": "0.0001",

"permissions": [

"SPOT",

"MARGIN"

],

"filters": [],

"maxQuoteAmount": "5000000",

"makerCommission": "0.002",

"takerCommission": "0.002"

}

* **GET** /api/v3/exchangeInfo

Current exchange trading rules and symbol information

**Parameter**：

There are 3 possible options:

| **Method** | **Example** |
| --- | --- |
| No parameter | curl -X GET "https://api.mexc.com/api/v3/exchangeInfo" |
| symbol | curl -X GET "https://api.mexc.com/api/v3/exchangeInfo?symbol=MXUSDT" |
| symbols | curl -X GET "https://api.mexc.com/api/v3/exchangeInfo?symbols=MXUSDT,BTCUSDT" |

**Order Book**

Response

{

"lastUpdateId": 1112416,

"bids": [

["15.00000", "49999.00000"]

],

"asks": [

["14.0000", "1.0000"]

]

}

**GET** /api/v3/depth

Parameter：

| **name** | **Type** | **Mandatory** | **Description** | **Scope** |
| --- | --- | --- | --- | --- |
| symbol | string | YES | Symbol |  |
| limit | integer | NO | Returen number | default 100; max 5000 |

Response：

| **name** | **Type** | **Description** |
| --- | --- | --- |
| lastUpdateId | long | Last Update Id |
| bids | list | Bid [Price, Quantity ] |
| asks | list | Ask [Price, Quantity ] |

**Recent Trades List**

Response

[

{

"id": null,

"price": "23",

"qty": "0.478468",

"quoteQty": "11.004764",

"time": 1640830579240,

"isBuyerMaker": true,

"isBestMatch": true

}

]

**GET** /api/v3/trades

Parameter：

| **name** | **Type** | **Mandatory** | **Description** | **Scope** |
| --- | --- | --- | --- | --- |
| symbol | string | YES |  |  |
| limit | integer | NO |  | Default 500; max 1000 |

Response:

| **name** | **Description** |
| --- | --- |
| id | Trade id |
| price | Price |
| qty | Number |
| quoteQty | Trade total |
| time | Trade time |
| isBuyerMaker | Was the buyer the maker? |
| isBestMatch | Was the trade the best price match? |

**Old Trade Lookup**

Response

[

{

"id": null,

"price": "23",

"qty": "0.478468",

"quoteQty": "11.004764",

"time": 1640830579240,

"isBuyerMaker": true,

"isBestMatch": true

}

]

**GET** /api/v3/historicalTrades

Parameters:

| **name** | **Type** | **Mandatory** | **Description** | **Scope** |
| --- | --- | --- | --- | --- |
| symbol | string | YES | Symbol |  |
| limit | integer | NO | Return number | Default 500; max 1000 |

Response:

| **name** | **Description** |
| --- | --- |
| id | Trade id |
| price | Price |
| qty | Number |
| quoteQty | Trade total |
| time | Trade time |
| isBuyerMaker | Was the buyer the maker? |
| isBestMatch | Was the trade the best price match? |

**Compressed/Aggregate Trades List**

Response

[

{

"a": null,

"f": null,

"l": null,

"p": "46782.67",

"q": "0.0038",

"T": 1641380483000,

"m": false,

"M": true

}

]

**GET** /api/v3/aggTrades

Get compressed, aggregate trades. Trades that fill at the time, from the same order, with the same price will have the quantity aggregated.

Parameters:

| **name** | **Type** | **Mandatory** | **Description** | **Scope** |
| --- | --- | --- | --- | --- |
| symbol | string | YES |  |  |
| startTime | long | NO | Timestamp in ms to get aggregate trades from INCLUSIVE. |  |
| endTime | long | NO | Timestamp in ms to get aggregate trades until INCLUSIVE. |  |
| limit | integer | NO |  | Default 500; max 1000. |

Response:

| **name** | **Description** |
| --- | --- |
| a | Aggregate tradeId |
| f | First tradeId |
| l | Last tradeId |
| p | Price |
| q | Quantity |
| T | Timestamp |
| m | Was the buyer the maker? |
| M | Was the trade the best price match? |

**Kline/Candlestick Data**

Response

[

[

1640804880000,

"47482.36",

"47482.36",

"47416.57",

"47436.1",

"3.550717",

1640804940000,

"168387.3"

]

]

**GET** /api/v3/klines

Kline/candlestick bars for a symbol. Klines are uniquely identified by their open time.

Parameters:

| **name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| symbol | string | YES |  |
| interval | ENUM | YES | ENUM: Kline interval |
| startTime | long | NO |  |
| endTime | long | NO |  |
| limit | integer | NO | Default 500; max 1000. |

Response:

| **Index** | **Description** |
| --- | --- |
| 0 | Open time |
| 1 | Open |
| 2 | High |
| 3 | Low |
| 4 | Close |
| 5 | Volume |
| 6 | Close time |
| 7 | Quote asset volume |

**Current Average Price**

Response

{

"mins": 5,

"price": "9.35751834"

}

**GET** /api/v3/avgPrice

Parameters:

| **name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| symbol | string | YES |  |

Response:

| **name** | **Description** |
| --- | --- |
| mins | Average price time frame |
| price | Price |

**24hr Ticker Price Change Statistics**

Response

{

"symbol": "BTCUSDT",

"priceChange": "184.34",

"priceChangePercent": "0.00400048",

"prevClosePrice": "46079.37",

"lastPrice": "46263.71",

"bidPrice": "46260.38",

"bidQty": "",

"askPrice": "46260.41",

"askQty": "",

"openPrice": "46079.37",

"highPrice": "47550.01",

"lowPrice": "45555.5",

"volume": "1732.461487",

"quoteVolume": null,

"openTime": 1641349500000,

"closeTime": 1641349582808,

"count": null

}

or

[

{

"symbol": "BTCUSDT",

"priceChange": "184.34",

"priceChangePercent": "0.00400048",

"prevClosePrice": "46079.37",

"lastPrice": "46263.71",

"bidPrice": "46260.38",

"bidQty": "",

"askPrice": "46260.41",

"askQty": "",

"openPrice": "46079.37",

"highPrice": "47550.01",

"lowPrice": "45555.5",

"volume": "1732.461487",

"quoteVolume": null,

"openTime": 1641349500000,

"closeTime": 1641349582808,

"count": null

},

{

"symbol": "ETHUSDT",

"priceChange": "184.34",

"priceChangePercent": "0.00400048",

"prevClosePrice": "46079.37",

"lastPrice": "46263.71",

"bidPrice": "46260.38",

"bidQty": "",

"askPrice": "46260.41",

"askQty": "",

"openPrice": "46079.37",

"highPrice": "47550.01",

"lowPrice": "45555.5",

"volume": "1732.461487",

"quoteVolume": null,

"openTime": 1641349500000,

"closeTime": 1641349582808,

"count": null

}

]

**GET** /api/v3/ticker/24hr

Parameters:

| **name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| symbol | string | NO | If the symbol is not sent, tickers for all symbols will be returned in an array. |

Response:

| **name** | **Description** |
| --- | --- |
| symbol | Symbol |
| priceChange | price Change |
| priceChangePercent | price change percent |
| prevClosePrice | Previous close price |
| lastPrice | Last price |
| lastQty | Last quantity |
| bidPrice | Bid best price |
| bidQty | Bid best quantity |
| askPrice | Ask best price |
| askQty | Ask best quantity |
| openPrice | Open |
| highPrice | High |
| lowPrice | Low |
| volume | Deal volume |
| quoteVolume | Quote asset volume |
| openTime | Start time |
| closeTime | Close time |
| count |  |

**Symbol Price Ticker**

Response

{

"symbol": "BTCUSDT",

"price": "184.34"

}

or

[

{

"symbol": "BTCUSDT",

"price": "6.65"

},

{

"symbol": "ETHUSDT",

"price": "5.65"

}

]

**GET** /api/v3/ticker/price

Parameters:

| **name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| symbol | string | NO | If the symbol is not sent, all symbols will be returned in an array. |

Response:

| **name** | **Description** |
| --- | --- |
| symbol |  |
| price | Last price |

**Symbol Order Book Ticker**

Response

{

"symbol": "AEUSDT",

"bidPrice": "0.11001",

"bidQty": "115.59",

"askPrice": "0.11127",

"askQty": "215.48"

}

OR

[

{

"symbol": "AEUSDT",

"bidPrice": "0.11001",

"bidQty": "115.59",

"askPrice": "0.11127",

"askQty": "215.48"

},

{

"symbol": "AEUSDT",

"bidPrice": "0.11001",

"bidQty": "115.59",

"askPrice": "0.11127",

"askQty": "215.48"

}

]

**GET** /api/v3/ticker/bookTicker

Best price/qty on the order book for a symbol or symbols.

Parameters:

| **name** | **Type** | **Mandatory** | **Description** |
| --- | --- | --- | --- |
| symbol | string | NO | If the symbol is not sent, all symbols will be returned in an array. |

Response:

| **name** | **Description** |
| --- | --- |
| symbol | Symbol |
| bidPrice | Best bid price |
| bidQty | Best bid quantity |
| askPrice | Best ask price |
| askQty | Best ask quantity |

BITFINEX

### URL

Public endpoints should use the domain:  
[https://api-pub.bitfinex.com](https://api-pub.bitfinex.com/)  
Authenticated endpoints should use the domain:  
[https://api.bitfinex.com](https://api.bitfinex.com/)

### Parameters:

To build a request, use the following structure to append parameters:

HTML

Public:

https://api-pub.bitfinex.com/v2/<endpoint>?parameter=value&parameter=value&....

Authenticated:

https://api.bitfinex.com/v2/<endpoint>?parameter=value&parameter=value&....

TICKER

The ticker endpoint provides a high level overview of the state of the market for a specified pair. It shows the current best bid and ask, the last traded price, as well as information on the daily volume and price movement over the last day.

JavaScriptShellPython

const ws = require('ws')

const w = new ws('wss://api-pub.bitfinex.com/ws/2')

w.on('message', (msg) => console.log(msg))

let msg = JSON.stringify({

event: 'subscribe',

channel: 'ticker',

symbol: 'tBTCUSD'

})

w.on('open', () => w.send(msg))

Request / ResponseSnapshotUpdate

// request

{

event: "subscribe",

channel: "ticker",

symbol: SYMBOL

}

// response - trading

{

event: "subscribed",

channel: "ticker",

chanId: CHANNEL\_ID,

symbol: SYMBOL,

pair: PAIR

}

{"event":"subscribed","channel":"ticker","chanId":224555,"symbol":"tBTCUSD","pair":"BTCUSD"}

// response - funding

{

event: "subscribed",

channel: "fticker",

chanId: CHANNEL\_ID,

symbol: SYMBOL

}

{"event":"subscribed","channel":"ticker","chanId":232591,"symbol":"fUSD","currency":"USD"}

KRAKEN

<https://docs.kraken.com/rest/> FIJATE PORQUE CREO QUE HAY QUE DESCARGAR EL ARCHIVO DEL REST

[https://docs.kraken.com/rest/#tag/Market-Data](https://docs.kraken.com/rest/" \l "tag/Market-Data)

COINBASE

# Requests

All requests and responses are application/json content type and follow typical HTTP response status codes for success and failure.

Note: Request URLs must be lowercase as URLs are [case-sensitive](https://www.w3.org/TR/WD-html40-970708/htmlweb.html).

## Errors

{

"message": "Invalid Price"

}

NO ENCONTRE LOS TICKET

GEMINI

<https://docs.gemini.com/rest-api/>

<https://docs.gemini.com/websocket-api/#market-data>

CEXIO

<https://cex.io/rest-api>

AHI SALEN LAS DOS