

CoinZoom Public API

Overview

All APIs have a common prefix to their URL.

For testing use: <https://api.stage.coinzoom.com/api/v1/public>
(<https://api.stage.coinzoom.com/api/v1/public>)

For production use: <https://api.coinzoom.com/api/v1/public> (<https://api.coinzoom.com/api/v1/public>)

Each API must include the HTTP header 'Coinzoom-API-Key' with the value of your API key. These keys can be created within the the coinzoom exchange from the 'API Trading' menu. To create your key you will need to complete registration for a CoinZoom Account:

For testing register here: <https://stage.coinzoom.com> (<https://stage.coinzoom.com>)

For production register here: <https://trade.coinzoom.com> (<https://trade.coinzoom.com>)

It is also recommended that you include a user-agent header with all requests that identifies your account. For example:

```
user-agent: XYZ-Corp-Tradebot v1.0.2
```

Streaming (websocket)

The streaming websocket is available at these endpoints

For testing use: <wss://stage.coinzoom.com/api/v1/public/market/data/stream>

For production use: <wss://trade.coinzoom.com/api/v1/public/market/data/stream>

The data items that can be received on this socket are described in the following subsections. You can subscribe to any combination of data items as you want on the socket.

Note: when using the websocket, open a connection and send the relevant subscriptions. After that keep the socket open. It is recommended to that you only close and reopen the socket if it is disconnected unexpectedly. Our firewalls may block repeated opening and closing of socket connections.

Market Ticker

To receive updates on the market for a given symbol send this JSON packet with the appropriate symbol name. This is un-secured request so the api and secret keys are not needed.

```
{"MarketSummaryRequest":{"action":"subscribe","symbol":"BTC/USD"}}
```

To stop receiving this data send the message:

Documentation Settings ▼

```
{ "MarketSummaryRequest": { "action": "unsubscribe", "symbol": "BTC/USD" } }
```

Responses are in the format below. The values in the array are the symbol, opening price, high price, low price, current mid price, volume traded, current best bid price, current best ask price respectively. These values are for a moving 24 hour window ending at the current time:

```
{ "ms": [ "BTC/USD", 6898.3, 7254.76, 6600.36, 7207.06, 0.0, 7195.53, 7218.58 ] }
```

Order Book Ticker

To receive the order book for a given symbol send this JSON packet with the appropriate symbol name. This is un-secured request so the api and secret keys are not needed.

```
{ "OrderBookRequest": {  
  "requestId": "BTC/USD", "action": "subscribe", "symbol": "BTC/USD", "aggregate": false, "depth": 0  
}
```

To stop receiving this data send the message:

```
{ "OrderBookRequest": { "requestId": "BTC/USD", "action": "unsubscribe", "symbol": "BTC/USD" } }
```

The responses come in to forms. A full order book, and incremental updates to this. You should receive a full order book message first.

An example of a full message is below. The **b** and **s** arrays contain the buy and sell sides of the order book. Each individual buy and sell is an array of three values - id, price, amount. The id is used for the incremental updates:

```
{ "ob": "BTC/USD", "b": [ [ "3", 0.01, 0.0100001 ], [ "5", 7193.07, 6.95113491 ],  
  [ "9", 7195.95, 0.69483529 ] ], "s": [ [ "1", 7221.88, 6.92340499 ], [ "2", 7219.0, 0.69261671 ] ] }
```

An example of an incremental message is below. If a buy or sell only contains the id it is an instruction to delete that record from the order book. Otherwise it is an instruction to add it to the book or update it if it is already there.

```
{ "oi": "BTC/USD", "b": [ [ "9" ], [ "5" ], [ "7", 7193.27, 6.95094164 ], [ "8", 7196.15, 0.69481598 ] ], "s":  
  [ [ "2" ], [ "1" ], [ "4", 7222.08, 6.92321326 ], [ "6", 7219.2, 0.69259752 ] ] }
```

Candle Ticker

To receive candles for a given symbol send this JSON packet with the appropriate symbol name. This is un-secured request so the api and secret keys are not needed. The supported intervals are M1 (one minute), M5 (5 minutes), M15 (15 minutes), M30 (30 minutes), H1 (1 hour), H2 (2 hour), H4 (4 hour), D1 (1 day), W1 (1 week), L1 (1 month).

```
{ "CandleRequest": { "action": "subscribe", "symbol": "BTC/USD", "interval": "M1" } }
```

To stop receiving this data send the message:

```
{ "CandleRequest": { "action": "unsubscribe", "symbol": "BTC/USD", "interval": "M1" } }
```

An example of the response message is below. The values in the **c** array are timestamp, opening price, low price, high price, closing price, volume.

```
{"cu": "BTC/USD", "i": "M1", "c": [{"2019-11-26T00:51:00Z", 7140.68, 7149.15, 7140.51, 7145.5, 0.0}]}
```

[Documentation Settings](#) ▼

Trade History

To receive the trade history for a given symbol send this JSON packet with the appropriate symbol name. This is un-secured request so the api and secret keys are not needed.

```
{"TradeSummaryRequest": {"action": "subscribe", "symbol": "BTC/USD"}}
```

To stop receiving this data send the message:

```
{"TradeSummaryRequest": {"action": "unsubscribe", "symbol": "BTC/USD"}}
```

An example of the response message is below. The values in the array are the symbol, price, quantity and timestamp. The data represents a summary of all trades over a time period of approximately half a second. The quantity is the cumulative quantity of all trades in that period. The price is the weighted average price (weighted by trade quantity). If there have been no trades a message will not be sent on the socket.

```
{"ts": ["BTC/USD", 8772.05, 0.01, "2020-01-16T21:02:23Z"]}
```

Order Updates

To receive updates on your orders send this JSON packet

```
{"OrderUpdateRequest": {"action": "subscribe", "apiKey": "your key", "secretKey": "your secret"}}
```

Sample responses for a successfully filled market order are:

```
{"OrderResponse": {"orderId": "300e0bb8-3c6b-417e-b3fb-b6197ff3a907", "clientOrderId": "my-id-3", "symbol": "ETH/USD", "orderType": "MARKET", "orderSide": "BUY", "quantity": 3.14, "executionType": "NEW", "orderStatus": "NEW", "lastQuantity": 0, "leavesQuantity": 3.14, "cumulativeQuantity": 0, "transactTime": "2019-11-14T21:15:55.860601Z"}}
```

```
{"OrderResponse": {"id": "26dbbb12-fd7a-4e0f-9f1a-7ffd1582e5c5", "orderId": "300e0bb8-3c6b-417e-b3fb-b6197ff3a907", "clientOrderId": "my-id-3", "symbol": "ETH/USD", "orderType": "MARKET", "orderSide": "BUY", "quantity": 3.14, "executionType": "FILL", "orderStatus": "FILLED", "lastPrice": 184.96, "averagePrice": 184.96, "lastQuantity": 3.14, "leavesQuantity": 0, "cumulativeQuantity": 3.14, "transactTime": "2019-11-14T21:15:56.280367Z"}}
```

Sample responses for a successfully cancelled order are:

```
{"OrderCancelResponse": {"orderId": "7a8514a8-a977-400c-a628-f70c4c182186", "response": "Cancel Pending", "symbol": "ETH/USD"}}
```

```
{"OrderResponse": {"id": "52e367f7-a81a-4ae8-bff8-3e3fdf95b9d9", "orderId": "7a8514a8-a977-400c-a628-f70c4c182186", "symbol": "ETH/USD", "orderType": "LIMIT", "orderSide": "BUY", "price": 120, "quantity": 10, "executionType": "CANCEL", "orderStatus": "CANCELLED", "lastQuantity": 0, "leavesQuantity": 0, "cumulativeQuantity": 0, "transactTime": "2019-11-14T21:13:21.954453Z"}}
```



GET Get currencies

Documentation Settings ▼

```
{{coinzoomHost}}/api/v1/public/currencies
```

HEADERS

Coinzoom-API-Key

```
{{coinzoomApiKey}}
```

Coinzoom-API-Secret

```
{{coinzoomSecretKey}}
```

Example Request

Get currencies

```
curl --location --request GET '{{coinzoomHost}}/api/v1/public/currencies' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}'
```

GET Get pairs

```
{{coinzoomHost}}/api/v1/public/instruments
```

Example Request

Get pairs

```
curl --location --request GET '{{coinzoomHost}}/api/v1/public/instruments'
```

GET Query ledger balances

{{coinzoomHost}}/api/v1/public/ledger/list

Documentation Settings ▼

HEADERS

Coinzoom-API-Key

{{coinzoomApiKey}}

Content-Type

application/json

Coinzoom-API-Secret

{{coinzoomSecretKey}}

Example Request

Query ledger balances

```
curl --location --request GET '{{coinzoomHost}}/api/v1/public/ledger/list' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw ''
```

POST Place limit order

{{coinzoomHost}}/api/v1/public/orders/new

HEADERS

Coinzoom-API-Key

{{coinzoomApiKey}}

Content-Type

application/json

Coinzoom-API-Secret

{{coinzoomSecretKey}}

```
{
  "symbol": "ZOOM/USD",
  "orderType": "LIMIT",
  "orderSide": "BUY",
  "quantity": 20,
  "price": 0.03,
  "payFeesWithZoomToken": true
}
```

Example Request

Place limit order

```
curl --location --request POST '{{coinzoomHost}}/api/v1/public/orders/new' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw '{
  "symbol": "ZOOM/USD",
  "orderType": "LIMIT",
  "orderSide": "BUY",
  "quantity": 20,
  "price": 0.03.
}
```

[View More](#)

POST Place market order

```
{{coinzoomHost}}/api/v1/public/orders/new
```

HEADERS

Coinzoom-API-Key

```
{{coinzoomApiKey}}
```

Content-Type

```
application/json
```

Coinzoom-API-Secret

```
{{coinzoomSecretKey}}
```

```
{
  "symbol": "ZOOM/USD",
  "orderType": "MARKET",
  "orderSide": "BUY",
  "quantity": 20,
  "payFeesWithZoomToken": true
}
```

Example Request

Place market order

```
curl --location --request POST '{{coinzoomHost}}/api/v1/public/orders/new' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw '{
  "symbol": "ZOOM/USD",
  "orderType": "MARKET",
  "orderSide": "BUY",
  "quantity": 20,
  "payFeesWithZoomToken": true
}
```

[View More](#)

POST Place stop limit order

```
{{coinzoomHost}}/api/v1/public/orders/new
```

HEADERS

Coinzoom-API-Key

```
{{coinzoomApiKey}}
```

Content-Type

```
application/json
```

Coinzoom-API-Secret

```
{{coinzoomSecretKey}}
```

```
{
  "symbol": "BTC/USD",
  "orderType": "STOP_LIMIT",
  "orderSide": "BUY",
  "quantity": 7.46,
  "price": 3,
  "stopPrice": 2,
  "payFeesWithZoomToken": true
}
```

Example Request

Place stop limit order

```
curl --location --request POST '{{coinzoomHost}}/api/v1/public/orders/new' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw '{
  "symbol": "BTC/USD",
  "orderType": "STOP_LIMIT",
  "orderSide": "BUY",
  "quantity": 7.46,
  "price": 3.
}
```

[View More](#)

POST Cancel order

```
{{coinzoomHost}}/api/v1/public/orders/cancel
```

HEADERS

Coinzoom-API-Key

```
{{coinzoomApiKey}}
```

Content-Type

```
application/json
```

Coinzoom-API-Secret

```
{{coinzoomSecretKey}}
```

```
{
  "orderId": "a5531ebf-e9c8-4907-9c22-444a76534298",
  "symbol": "BTC/USD"
}
```

Example Request

[Cancel order](#)

```
curl --location --request POST '{{coinzoomHost}}/api/v1/public/orders/cancel' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw '{
  "orderId": "a5531ebf-e9c8-4907-9c22-444a76534298",
  "symbol": "BTC/USD"
}'
```

POST Query open orders

```
{{coinzoomHost}}/api/v1/public/orders/list
```

This request returns a list of your open orders. The values in the orderStatus array can be adjusted to get orders in other statuses. The supported values are NEW, PARTIALLY_FILLED, FILLED, CANCELLED, REJECTED.

The orderSide be one of BUY or SELL. If it is not specified orders from both sides will be returned.

The data is returned in pages. For the initial request pass `null` in the bookmarkOrderId fields. To get the next page of data pass the orderId of the last entry in the previous call as the bookmarkOrderId.

HEADERS

Coinzoom-API-Key

```
{{coinzoomApiKey}}
```

Content-Type

```
application/json
```

Coinzoom-API-Secret

{{coinzoomSecretKey}}



Documentation Settings ▼

BODY raw

```
{
  "symbol": null,
  "orderSide": null,
  "orderStatuses": ["NEW", "PARTIALLY_FILLED"],
  "size": 100,
  "bookmarkOrderId": null
}
```

Example Request

Query open orders

```
curl --location --request POST '{{coinzoomHost}}/api/v1/public/orders/list' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw '{
  "symbol": null,
  "orderSide": null,
  "orderStatuses": ["NEW", "PARTIALLY_FILLED"],
  "size": 100,
  "bookmarkOrderId": null
}'
```

[View More](#)

POST Query historic orders

```
{{coinzoomHost}}/api/v1/public/orders/list
```

This request returns a list of your historic orders. The values in the orderStatus array can be adjusted to get orders in other statuses. The supported values are NEW, PARTIALLY_FILLED, FILLED, CANCELLED, REJECTED.

The orderSide be one of BUY or SELL. If it is not specified orders from both sides will be returned.

The data is returned in pages. For the initial request pass `null` in the bookmarkOrderId fields. To get the next page of data pass the orderId of the last entry in the previous call as the bookmarkOrderId.

HEADERS

Coinzoom-API-Key

{{coinzoomApiKey}}

Content-Type

application/json

Documentation Settings ▼

Coinzoom-API-Secret

{{coinzoomSecretKey}}

BODY raw

```
{
  "symbol": null,
  "orderSide": null,
  "orderStatuses": ["FILLED", "CANCELLED", "REJECTED"],
  "size": 100,
  "bookmarkOrderId": null
}
```

Example Request

Query historic orders

```
curl --location --request POST '{{coinzoomHost}}/api/v1/public/orders/list' \
--header 'Coinzoom-API-Key: {{coinzoomApiKey}}' \
--header 'Content-Type: application/json' \
--header 'Coinzoom-API-Secret: {{coinzoomSecretKey}}' \
--data-raw '{
  "symbol": null,
  "orderSide": null,
  "orderStatuses": ["FILLED", "CANCELLED", "REJECTED"],
  "size": 100,
  "bookmarkOrderId": null
}
```

[View More](#)