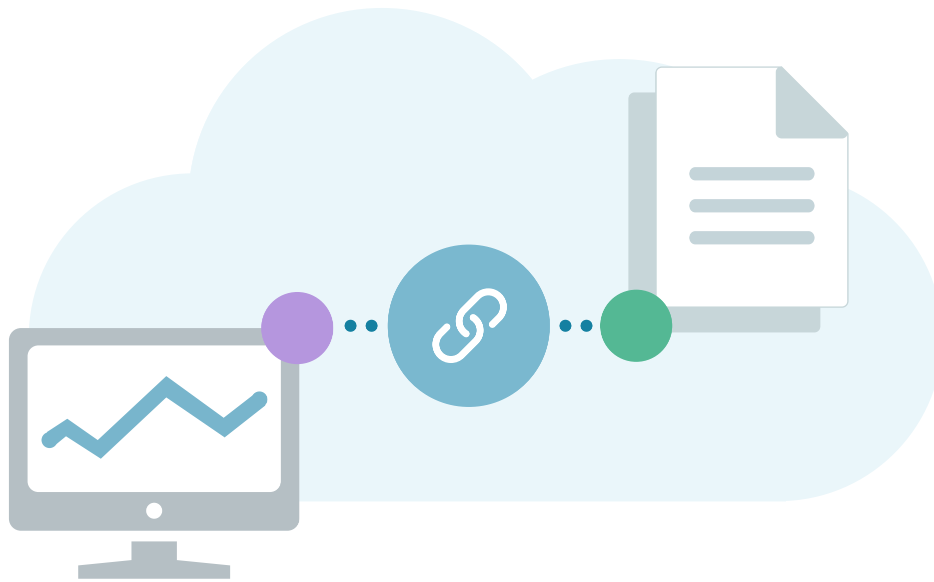
Questrade API Documentation



QUESTRADE®

Contents

Welcome to the Questrade API centre

Before you begin

The Questrade API enables you to develop applications that can access your account, market data and assist you with placing trades through your Questrade brokerage account. This document will provide you with all the resources you need to get started with developing an application using Questrade's API.

If you need assistance, or to see the latest API updates, visit the forums on the Exchange. If you're interested in partnering with Questrade as a company or a developer, send us an email at apisupport@questrade.com.

In order to use Questrade's API, you must have an account in good standing with Questrade. Don't have an account yet? No problem! You can get one easily by filling in your application online at www.questrade.com/account/online.

Key characteristics

The Questrade API currently supports building your API application for the following resources:

- Accounts
- Positions
- Balances
- Executions
- Orders
- Symbols
- Markets

The following key services are supported:

- Retrieving account data
- Getting level 1 quotes
- Placing orders

You can issue REST calls using the following HTTP methods:

- GET: retrieve the existing resource element or collection of elements.
- POST: create a new element of a particular resource.
- DELETE: remove an element of a particular resource.

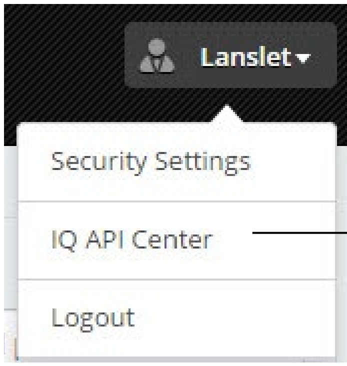
Message formats:

- JSON

Getting started

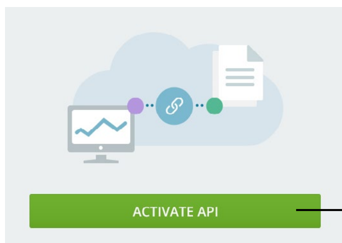
Activating IQ API centre

As a client of Questrade, use your login credentials to login.



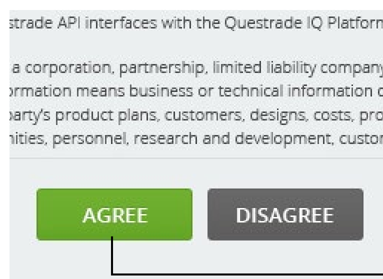
1 API welcome page

In the top right corner where you see your login name, select **API centre** from the drop-down menu. You will be redirected to the API welcome page providing you with more information about the API.



2 Activate API

Once you're ready, click **Activate API.**



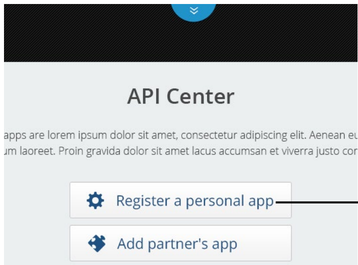
3 API access agreement

The API access agreement appears. You need to read and agree to the terms listed in the agreement to proceed. After clicking **Agree**, you will be one step closer to getting access to Questrade's API.

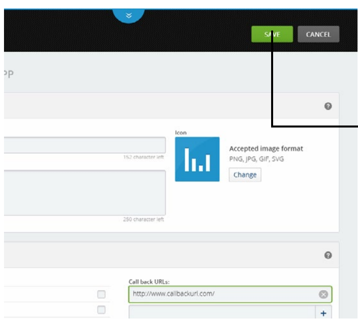
> *i* If you disgaree to the terms, access to Questrade API will not be activated.

Creating your first personal app

Manual authorization flow:



1 In the API Centre, click **Register a personal app**.

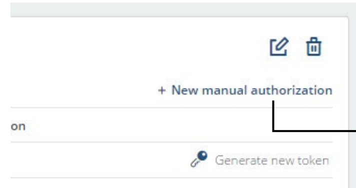


2 In the create personal app section, under basic information, enter a name for your application including a brief description and click **Save**.

Questrade uses the OAuth 2.0 authorization framework for the APIs. We recommend that you familiarize yourself with this framework before proceeding.

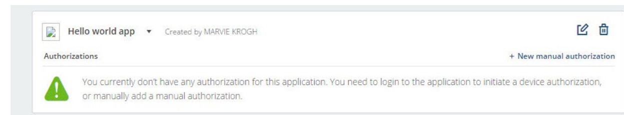
Curl command has been used to represent sample authorization request and responses.

Authorizing your app to access the API

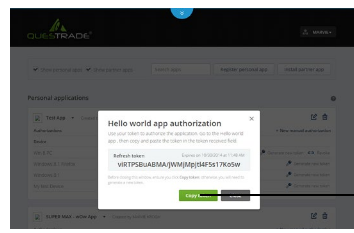


1 In the personal applications page, click **New manual authorization** to generate a personal access token for the corresponding personal app.

The personal app will be visible under personal applications.

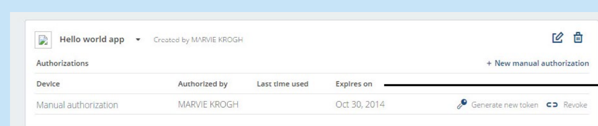


A token is generated and appears in the authorization pop-up window.



2 Click **Copy token** to copy it before clicking **Close.**

For security reasons, Questrade does not store the token information in a decrypted format after it is presented. If you forget to copy the token, you must generate a new token, or follow the revoke authorization process. The generated tokens have a limited time span and expire in 3 days from the date and time the token is generated. The time to expiry can be reviewed from the authorization section on the personal applications page.



Expiry date

3 Use the token you copied to redeem it for an access token and the server URL using the following command:

Sample request:

```
https://login.questrade.com/oauth2/token?client_id=<consumer_
key>&grant_type=refresh_token&refresh_token=<refresh_token>
```

Example:

```
https://login.questrade.com/oauth2/token?grant_type=refresh_
token&client_id=AOHz6Z5StoItJLacCHeq3ADqtgNYEQ&refresh_token=p4VTj45Gh
S8lY7aFoKDNZxB8yQHMOr+f
```

Sample response:

```
{
   “access_token”: ”C3lTUKuNQrAAmSD/TPjuV/HI7aNrAwDp”,
   “token_type”: ”Bearer”,
   “expires_in”: 300,
   “refresh_token”: ”aSBe7wAAdx88QTbwut0tiu3SYic3ox8F”
   “api_server”: ”https://api01.iq.questrade.com/v1”
}
```

4 Attempt your first API call. Use the access token and server URL with the following command to make an authorized call:

Sample request:

```
GET /v1/accounts HTTP/1.1
Host: https://api01.iq.questrade.com
Authorization: Bearer p4VTj45GhS8lY7aFoKDNZxB8yQHMOr+f
```

Sample response:

```
{
    “accounts”: [
              {
                   “type”: “Margin”,
                   “number”: “26598145”,
                   “status”: “Active”,
                   “isPrimary”: true,
                   “isBilling”: true,
                   “clientAccountType”: “Individual”
              },
              ...
    ],
    “userId”: 3000124
}
```

That’s it! You are now ready to start making requests to Questrade’s IQ platform using your personal access token.

> *i* While you are getting accustomed to Questrade’s API, you can control API access by going to **Security settings** in the API centre and setting the **General API access** option to **On** or **Off** as per your preference.

What’s next?

Depending on your comfort level, you can either choose to explore Questrade’s API methods directory to expand your knowledge, or you can directly access the SDKs and the sample code available for C++ and C# that’s included on our website.

Revoking authorization

To revoke authorization, you can do one of the following:

1. Go to **API Centre > Personal applications,** and click **Revoke** to immediately expire the token that was issued previously or click **Delete** to delete the personal app.

OR

2. Use the revoke endpoint to revoke the authorization from your application code as per the example below:

Revoke endpoint URL: https://login.questrade.com/oauth2/revoke

Revoke endpoint example

```
POST /oauth2/revoke HTTP/1.1
Host: https://login.questrade.com
Content-Type: application/x-www-form-urlencoded
token=p4VTj45GhS8lY7aFoKDNZxB8yQHMOr+f
```

Making an authorized request

Authorized requestes can be made toward API servers. The URL of the API servers will be provided to your application as a response to every access token request you make. Once your application obtains an access token and URL of proxy server to contact, it can then make authenticated calls on behalf of the user that authorized the application using a number of REST endpoints.

Your API application must pass the access token in the “Authorization” HTTP header as described in the sample request below:

Sample authorized request

```
GET /v1/accounts HTTP/1.1
Host: https://api01.iq.questrade.com
Authorization: Bearer p4VTj45GhS8lY7aFoKDNZxB8yQHMOr+f
```

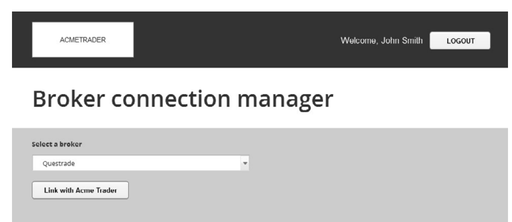
Integrating your application

The Questrade API provides support for web-based, mobile, and desktop applications.

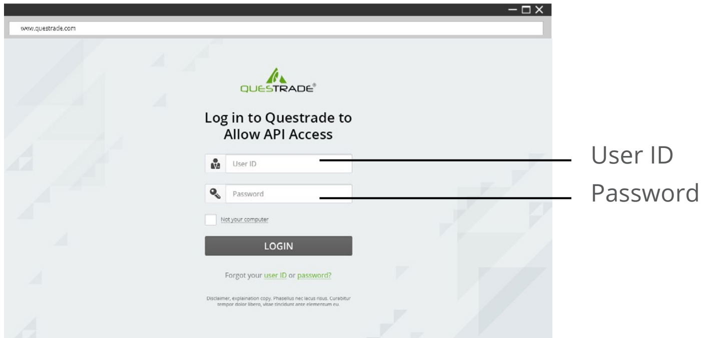
> This section uses a fictional trading application called AcmeTrader to illustrate how IQ API facilitates authorization of any client applications with the IQ trading platform.

Web based application

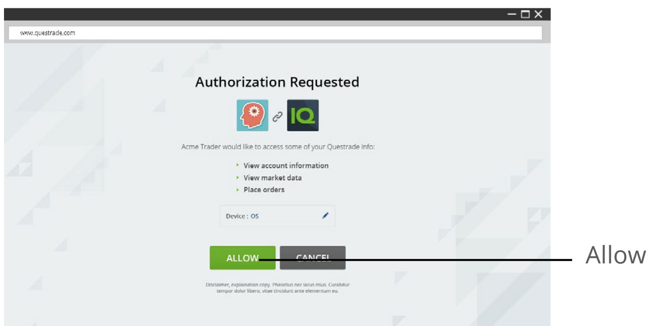
1. Log in to the Acme Trader web application (client) at https://acmetrader.com.
2. Navigate to the broker connection manager, and from the select a broker drop-down menu, choose **Questrade.**
3. Then click **Link with Acme Trader**.



4. A Questrade API login page with the app id as the parameter appears. Enter your **User ID** and **password**. Then click **Login**.



5. The Authorization requested message appears so you can access your accounts at Questrade. Click **Allow access** to proceed.

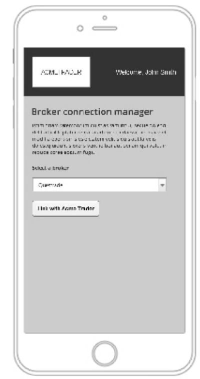


6. You are redirected to the “https://acmetrader.com/cb?code=d78fgasdasdf” web page.
Parse the code from this URL and redeem it for a token.

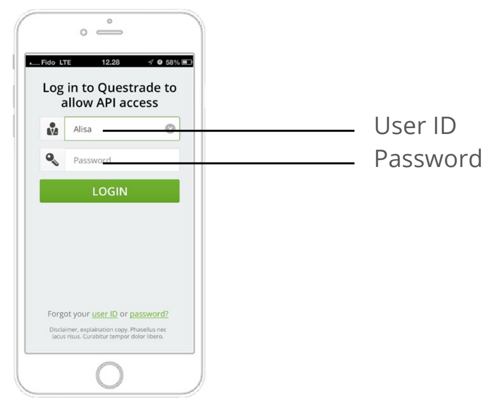
Your API access is now enabled.

Mobile applications (iOS and Android)

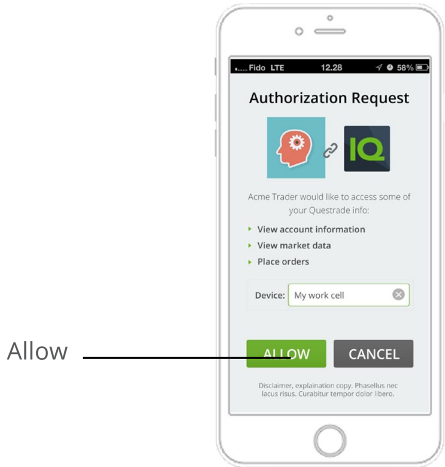
1. Log in to Acme Trader from your mobile device.
2. Register the “mytestapp://” URL scheme with your mobile OS.
3. Register the “mytestapp://” as OAuth redirect URL.
4. In the broker connection manager, from the select a broker drop-down, choose **Questrade.** Then click **Link with Acme Trader**.



5. Then open your web browser. The Questrade API login page with the app id as the parameter appears. Enter your **user ID** and **password**. Then click **Login**.



6 The authorization requested message appears so you can access your accounts at Questrade. Click **Allow access** to proceed.



7 Questrade redirects the web browser to “mytestapp://token= p4VTj45GhS8lY7aFoKDNZxB8yQHMOr+f”.

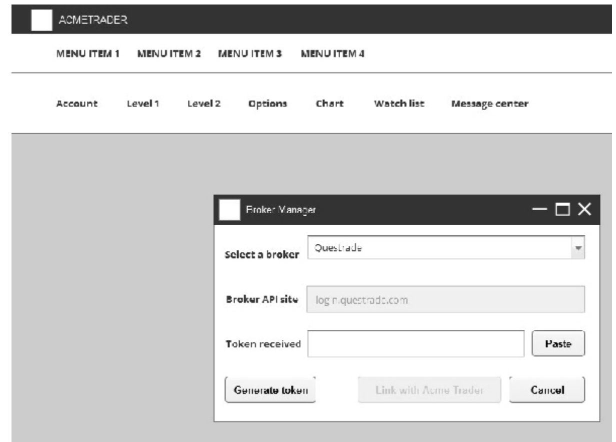
8 The mobile’s OS detects the URL scheme and automatically returns back to the Acme Trader app to pass the URL.

9 Parse the token out of the URL to access Questrade’s API.

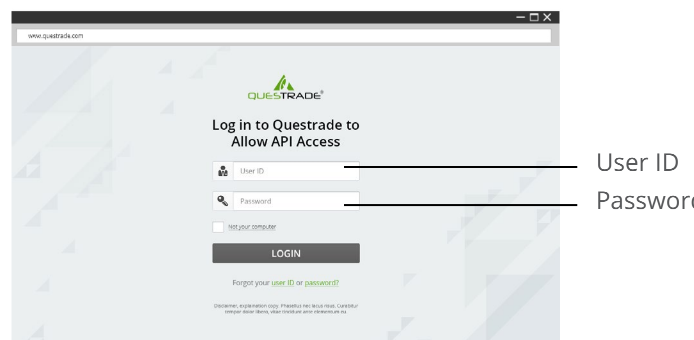
Desktop application

1 Launch the Acme Trader desktop application.

2 Navigate to the **Broker Manager.** Then from the select a broker drop-down list, choose **Questrade** and click **Generate token**.

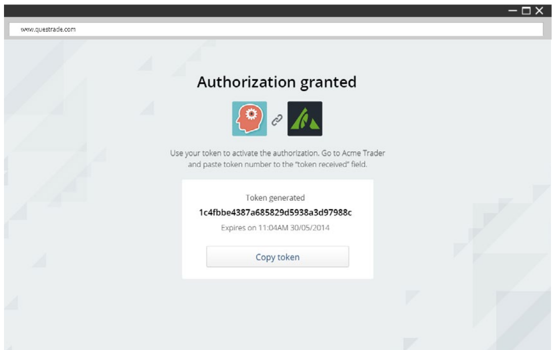


3 You are redirected to Questrade’s API login page. In the login window, enter your **User ID** and **password**. Then click **Log in**.



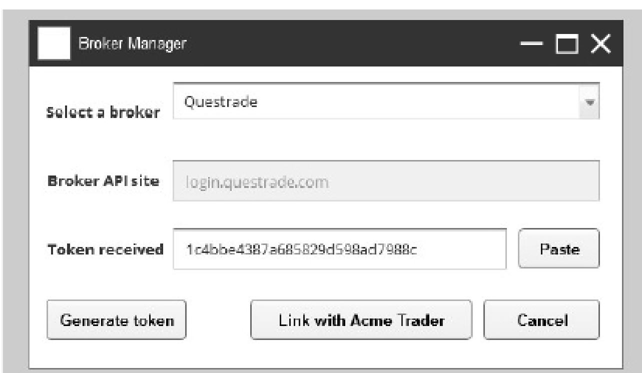
4. The authorization requested message opens so you can access your accounts at Questrade. Click **Allow access** to proceed.

5. Questrade grants you authorization through the AcmeTrader app and displays a token.



6. Click **Copy token**. Then create a new connection within the AcmeTrader desktop application.

7. Paste the token in the **Token received** field in the broker manager window.



Your access to Questrade's API through AcmeTrader is now enabled.

Section 1 — Rate limiting

The API implements a rate limiting system for all authenticated calls. The purpose of this functionality is to protect servers from abuse and to provide a clear expectation of the level of service that the API commits to fulfill.

Limits

Rate limits are defined across categories of API calls as described in the following table.

Category	API calls	Maximum allowed requests per second	Maximum allowed requests per hour
Account calls	GET time GET accounts GET accounts/:id/positions GET accounts/:id/balances GET accounts/:id/executions GET accounts/:id/orders GET symbols/:id GET symbols/:id/options	100	10,000
Market Data calls	GET markets GET markets/quotes/:id GET markets/candles/:id	100	10,000
Order calls	POST accounts/:id/orders[/:orderId] POST accounts/:id/orders[:/orderId]/impact DELETE accounts/:id/orders/:orderId	100	10,000

Rate limit headers

The API communicates rate limit information in the headers of responses to all calls that are limited.

Specifically, the following headers are provided:

- X-RateLimit-Remaining – number of requests allowed against a the current limit
- X-RateLimit-Reset – time when the current limit will expire (Unix timestamp)

Sample rate limit headers

```
HTTP/1.1 200 OK
Content-Length: 123
Content-Type: application/json; charset=utf-8
Date: Fri, 18 Aug 2014 22:17:16 GMT
X-RateLimit-Remaining: 100
X-RateLimit-Reset: 1300286940
```

Exceeding limits

If one exceeds the above rate limits, then the API server will respond with a rate limit error message with the HTTP status code 429 (Too Many Requests) and the same rate limit headers one would receive in a normal response to limited call.

Section 2 — Order processing

The following sections describe the order lifecycle as well as the order routing capabilities of the IQ platform that the Questrade APIs allow you to connect to.

IQ order lifecycle

The IQ order lifecycle explains the different states in which an order may exist once it is placed, as well as the possible transitions between the order states. You need to familiarize yourself with the different order states to understand the responses you may receive when you place an order using the Questrade API.

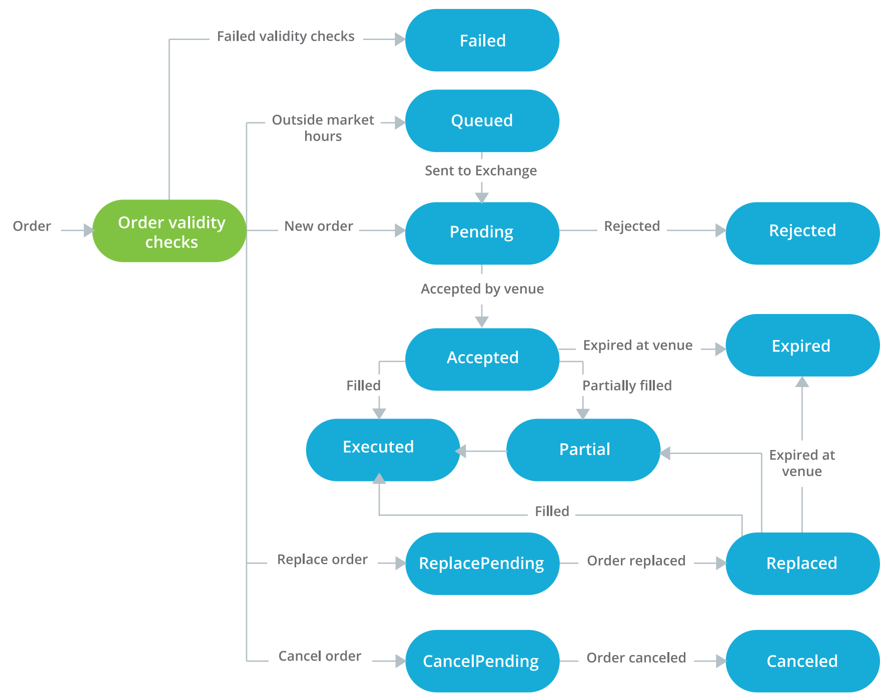


Figure 1: Lifecycle of an order in the IQ platform

Note: figure 1 does not show every single transition for the sake of readability.

Order routing

On IQ, route is defined as the destination where the order is sent for execution. Additionally, route is further divided into two categories: primary and secondary.

The primary route is the trading venue that Questrade clients can directly send orders to (for example, to listing exchanges like NYSE, an ECN like ATG, or a black-box execution venue like MNGD). The primary route eliminates the need to go through a Smart Order Route (SOR).

The secondary route on IQ only serves as a preferred destination for excess liquidity. An order to a secondary route on IQ always goes through a SOR, where the nationwide liquidity is examined (for example, a limit order for the same security sent to all trading venues) to obtain executions that are attractive in price (for example, the lowest priced sell orders for the same security first). If the nationwide liquidity is not found, a fraction of the order remains unfilled as the SOR sends the remainder to the preferred destination that the client or the IQ servers selected.

Due to the difference in securities law in the U.S. and in Canada, only U.S. routes can act as primary routes. As per guidelines set by IIROC, for Canadian routes, all orders are sent by default through the SOR by selecting “auto” as the default selection on the IQ platform.

The diagram below illustrates the possible paths of client orders through IQ to the supported trading venues:

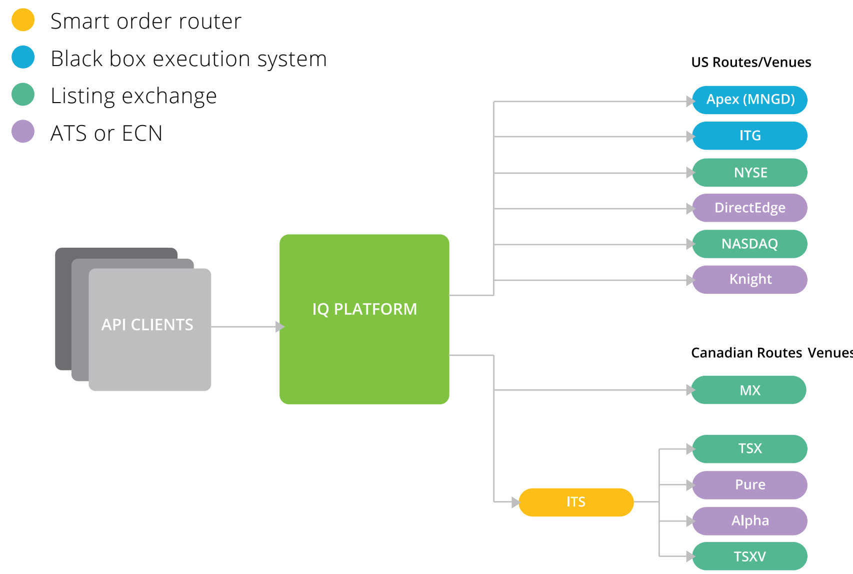


Figure 2: Questrade IQ trading platform to market connectivity

Section 3 — Error handling

Error format

In case of not being able to successfully service a request the API will return an error response.

The API offers two categories of errors:

1. General Error – errors that do NOT result in creation of an order with an internal identifier.

2. Order Processing Error – errors that occur during processing of trade calls (e.g., placing, replacing, canceling an order) that result in an order being created in the platform.

General error response

General error message will consist of properties specified in the table below.

Property	Type	Description
code	String	Error code.
message	String	Error description.

Sample

```
HTTP/1.1 400 Bad Request
Content-Length: 63
Content-Type: application/json; charset=utf-8
Date: Fri, 18 Aug 2014 22:17:16 GMT
{
    "code" : 3139,
    "message" : "Invalid route"
}
```

Order processing error response

Whenever a trading call results in the creation of an order with a unique internal identifier, the platform will return an error response with additional properties (orderId and orders) in addition to standard error fields (code and message) in the response.

Property	Type	Description
code	String	Error code.
message	String	Error message.
orderId	Integer	Internal unique identifier assigned to the order.
orders	Complex	List of Order records.
L orders	Complex	See Order record in the above sections of this document.

Sample order error response (no order created)

```
HTTP/1.1 400 Bad Request
Content-Length: 63
Content-Type: application/json; charset=utf-8
Date: Fri, 18 Aug 2014 22:17:16 GMT
{
    "code" : 3139,
    "message" : "Invalid route"
}
```

Sample order error response (order created)

```
HTTP/1.1 200 OK
Content-Length: 63
Content-Type: application/json; charset=utf-8
Date: Fri, 18 Aug 2014 22:17:16 GMT
```

```
{
    "code" : 3054,
    "message" : "Order was rejected by the exchange",
    "orderId" : 134353223,
    "orders" : [
            …
    ]
}
```

HTTP status code	HTTP status message	Error code	Error message
404	Not Found	1001	Invalid endpoint.
400	Bad Request	1002	Invalid or malformed argument.
400	Bad Request	1003	Argument length exceeds imposed limit.
400	Bad Request	1004	Missing required argument.
413	Request Entity Too Large	1005	Request length exceeds imposed limit.
429	Too Many Requests	1006	Rate limit exceeded.
500	Internal Server Error	1007	IQ servers responded with a business error.
500	Internal Server Error	1008	IQ servers responded with a technical error.
500	Internal Server Error	1009	IQ servers responded with an unexpected error.
502	Bad Gateway	1010	IQ servers produced an invalid response.
503	Gateway Timeout	1011	IQ servers did not produce a response before a timeout.
405	Method Not Allowed	1012	Method unsupported by endpoint (e.g., GET vs. POST).
400	Bad Request	1013	Requesting anything other than 'application/json'.
401	Unauthorized	1014	Missing authorization header.

HTTP status code	HTTP status message	Error code	Error message
400	Bad Request	1015	Malformed authorization header.
403	Forbidden	1016	Request is out-of-allowed OAuth scopes.
401	Unauthorized	1017	Access token is invalid.
404	Not Found	1018	Account number not found.
404	Not Found	1019	Symbol not found.
404	Not Found	1020	Order not found.
500	Internal Server Error	1021	Unexpected error (with undefined handling).

Section 4 — REST operations

IQ API endpoints

The following table describes the endpoints that Questrade API currently supports:

IQ API endpoint	Description
GET time	Retrieves current server time.
GET accounts	Retrieves a list of accounts.
GET accounts/:id/positions	Retrieves positions for a specific account.
GET accounts/:id/balances	Retrieves balances for a specific account.
GET accounts/:id/executions	Retrieves executions for a specific account.
GET accounts/:id/orders	Retrieves orders for a specific account.
GET symbols/:id	Retrieves information about one or more symbols.
GET symbols/search	Retrieves symbol using a full-text search.
GET symbols/:id/options	Retrieves options for a specific underlying symbol.
GET markets	Retrieves trading markets supported by the platform.
GET markets/quotes/:id	Retrieves a level 1 market data quote.
GET markets/candles/:id	Retrieves OHLC historical data for a given symbol.
POST accounts/:id/orders[/:orderId]	Places a new order or replaces an existing order.
POST accounts/:id/orders[/:orderId]/impact	Estimates the impact a prospective new order, or an order replacing another existing order will have on an account.
DELETE accounts/:id/orders/:orderId	Cancels an existing order.

IQ API OAuth scopes

As part of its OAuth 2.0 implementation, IQ API defines OAuth scopes – permissions that the account holders grants to the authorized API client application. Each API call belongs to one and only scope.

The following table describes scopes that the API provides and the mapping of API calls to these scopes.

Scope	Scope Identifier	API calls
Read account information	read_acc	GET time GET accounts GET accounts/:id/positions GET accounts/:id/balances GET accounts/:id/executions GET accounts/:id/orders GET symbols/:id GET symbols/:id/options
Read market data	read_md	GET markets GET markets/quotes/:id GET markets/candles/:id
Trade	trade	POST accounts/:id/orders[/:orderId] POST accounts/:id/orders[:/orderId]/impact DELETE accounts/:id/orders/:orderId

Account calls

GET time

Retrieves current server time.

Request parameters

No parameters.

Response properties

Parameter	Type	Description
time	DateTime	Current server time in ISO format and Eastern time zone.

Sample request

```
GET https://api01.iq.questrade.com/v1/time
```

Sample JSON response

```
{
    "time": "2014-10-24T12:14:42.730000-04:00"
}
```

GET accounts

Retrieves the accounts associated with the user on behalf of which the API client is authorized.

Request parameters

No parameters.

Response properties

Property	Type	Description
accounts	Complex	List of account records.
L Account	Complex	
L type	String	Type of the account (e.g., "Cash", "Margin").
L number	String	Eight-digit account number (e.g., "26598145").
L status	Enumeration	Status of the account (e.g., Active).
L isPrimary	Boolean	Whether this is a primary account for the holder.
L isBilling	Boolean	Whether this account is one that gets billed for various expenses such as inactivity fees, market data, etc.
L clientAccountType	String	Type of client holding the account (e.g., "individual").
userId	Integer	Internal identifier of the user making the request.

Sample request

```
GET https://api01.iq.questrade.com/v1/accounts
```

Sample JSON response


```
{
    "accounts": [
            {
                "type": "Margin",
                "number": "26598145",
                "status": "Active",
                "isPrimary": true,
                "isBilling": true,
                "clientAccountType": "Individual"
            },
            ...
    ],
    "userId": 3000124
}
```


GET accounts/:id/positions

Retrieves positions in a specified account.

Request parameters

Parameter	Type	Description
id	String	Account number. Can occur in the 'location' header only.

Response properties

Property	Type	Description
positions	Complex	List of position records.
L Position	Complex	
L symbol	String	Position symbol.
L symbolId	Integer	Internal symbol identifier.
L openQuantity	Double	Position quantity remaining open.
L closedQuantity	Double	Portion of the position that was closed today.
L currentMarketValue	Double	Market value of the position (quantity x price).
L currentPrice	Double	Current price of the position symbol.
L averageEntryPrice	Double	Average price paid for all executions constituting the position.
L closedPnL	Double	Realized profit/loss on this position.
L openPnL	Double	Unrealized profit/loss on this position.
L totalCost	Double	Total cost of the position.
L isRealTime	Boolean	Designates whether real-time quote was used to compute PnL.
L isUnderReorg	Boolean	Designates whether a symbol is currently undergoing a reorg.

Sample request

```
GET https://api01.iq.questrade.com/v1/accounts/26598145/positions
```

Sample JSON response

```
{
    "positions": [
            {
                "symbol": "THI.TO",
                "symbolId": 38738,
                "openQuantity": 100,
                "closedQuantity": 0,
                "currentMarketValue": 6017,
                "currentPrice": 60.17,
                "averageEntryPrice": 60.23,
                "closedPnl": 0,
                "openPnl": -6,
                "totalCost": 6023,
                "isRealTime": false,
                "isUnderReorg": false,
            },
            ...
    ]
}
```

GET accounts/:id/balances

Retrieves per-currency and combined balances for a specified account.

Request parameters

Parameter	Type	Description
id	String	Account number. Can occur in the 'Location' header only.

Response properties

Property	Type	Description
perCurrencyBalances	Complex	List of Balance records.
L Balance	Complex	See below.
combinedBalances	Complex	List of Balance records.
L Balance	Complex	See below.
sodPerCurrencyBalances	Complex	List of Balance records.
L Balance	Complex	See below.
sodCombinedBalances	Complex	List of Balance records.
L Balance	Complex	
L currency	Enumeration	Currency of the balance figure (e.g., "USD" or "CAD").
L cash	Double	Balance amount.
L marketValue	Double	Market value of all securities in the account in a given currency.
L totalEquity	Double	Equity as a difference between cash and marketValue properties.
L buyingPower	Double	Buying power for that particular currency side of the account.
L maintenanceExcess	Double	Maintenance excess for that particular side of the account.
L isRealTime	Boolean	Whether real-time data was used to calculate the above values.

Sample request

```
GET https://api01.iq.questrade.com/v1/accounts/26598145/balances
```

Sample JSON response

```
{
    "perCurrencyBalances": [
            {
                "currency": "CAD",
                "cash": 243971.7,
                "marketValue": 6017,
                "totalEquity": 249988.7,
                "buyingPower": 496367.2,
                "maintenanceExcess": 248183.6,
                "isRealTime": false
            },
            {
                "currency": "USD",
                "cash": 198259.05,
                "marketValue": 53745,
                "totalEquity": 252004.05,
                "buyingPower": 461013.3,
                "maintenanceExcess": 230506.65,
                "isRealTime": false
            }
    ],
    "combinedBalances": [
            ...
    ],
    "sodPerCurrencyBalances": [
            ...
    ],
    "sodCombinedBalances": [
            ...
    ]
}
```

GET accounts/:id/executions

Retrieves executions for a specific account.

Request parameters

Parameter	Type	Description
id	String	Account number. Can occur in the 'location' header only.
startTime	DateTime	Start of time range in ISO format. By default – start of today, 12:00am.
endTime	DateTime	End of time range in ISO format. By default – end of today, 11:59pm

Response properties

Property	Type	Description
executions	Complex	List of execution records.
L Execution	Complex	
L symbol	String	Execution symbol.
L symbolId	Integer	Internal symbol identifier.
L quantity	Integer	Execution quantity.
L side	String	Client side of the order to which execution belongs.
L price	Double	Execution price.
L id	Integer	Internal identifier of the execution.

Property	Type	Description
L orderId	Integer	Internal identifier of the order to which the execution belongs.
L orderChainId	Integer	Internal identifier of the order chain to which the execution belongs.
L exchangeExecId	String	Identifier of the execution at the market where it originated.
L timestamp	DateTime	Execution timestamp.
L notes	String	Manual notes that may have been entered by Trade Desk staff.
L venue	String	Trading venue where execution originated.
L totalCost	Double	Execution cost (price x quantity).
L orderPlacementCommission	Double	Questrade commission for orders placed with Trade Desk.
L commission	Double	Questrade commission.
L executionFee	Double	Liquidity fee charged by execution venue.
L secFee	Double	SEC fee charged on all sales of US securities.
L canadianExecutionFee	Integer	Additional execution fee charged by TSX (if applicable).
L parentId	Integer	Internal identifier of the parent order.
L strategyType	Enumeration	Type of order strategy to which execution belongs.

Sample request

```
GET https://api01.iq.questrade.com/v1/accounts/26598145/executions
```

Sample JSON response


```
{
    "executions": [
        {
            "symbol": "AAPL",
            "symbolId": 8049,
            "quantity": 10,
            "side": "Buy",
            "price": 536.87,
            "id": 53817310,
            "orderId": 177106005,
            "orderChainId": 177106005,
            "exchangeExecId": "XS1771060050147",
            "timestamp": 2014-03-31T13:38:29.000000-04:00,
            "notes": "",
            "venue": "LAMP",
            "totalCost": 5368.7,
            "orderPlacementCommission": 0,
            "commission": 4.95,
            "executionFee": 0,
            "secFee": 0,
            "canadianExecutionFee": 0,
            "parentId": 0,
            "strategyType": "SingleLeg"

        },
        ...
    ]
}
```

GET accounts/:id/orders [/:orderId]

Retrieves orders for specified account.

Request parameters

Parameter	Type	Description
id	String	Account number. Can occur in the 'Location' header only.
startTime	DateTime	Start of the time rang in ISO format. By default – start of today, 12:00am.
endTime	DateTime	End of the time range in ISO format. By default – end of today, 11:59pm.
stateFilter	Enum	All, Open, Closed – retrieve all, active or closed orders
orderId	Integer	Retrieve single order details

Response properties

Property	Type	Description
orders	Complex	List of order records.
L Order		
L id	Integer	Internal order identifier.
L symbol	String	Symbol that follows Questrade symbology (e.g., "TD.TO").
L symbolId	Integer	Internal symbol identifier.
L totalQuantity	Integer	Total quantity of the order.
L openQuantity	Integer	Unfilled portion of the order quantity.
L filledQuantity	Integer	Filled portion of the order quantity.
L canceledQuantity	Integer	Unfilled portion of the order quantity after cancellation.
L side	Enumeration	Client view of the order side (e.g., "buy-to-open").
L orderType	Enumeration	Order price type (e.g., "market").
L limitPrice	Integer	Limit price.
L stopPrice	Double	Stop price.
L isAllOrNone	Boolean	Specifies all-or-none special instruction.
L isAnonymous	Boolean	Specifies Anonymous special instruction.
L icebergQuantity	Integer	Specifies Iceberg special instruction.
L minQuantity	Integer	Specifies Minimum special instruction.
L avgExecPrice	Double	Average price of all executions received for this order.
L lastExecPrice	Double	Price of the last execution received for the order in question.
L source	Enumeration	Platform client from which the order originated (e.g., "TradingAPI").
L timeInForce	Enumeration	Time-in-force instruction (e.g., "Day").
L gtdDate	DateTime	Good-Till-Date marker and date parameter.
L state	Enumeration	Order execution status.
L clientReasonStr	String	Human readable order rejection reason message.

Property	Type	Description
L chainId	Integer	Internal identifier of a chain to which the order belongs.
L creationTime	DateTime	Order creation time.
L updateTime	DateTime	Time of the last update.
L notes	String	Notes that may have been manually added by Questrade staff
L primaryRoute	Enumeration	Direct destination for execution.
L secondaryRoute	Enumeration	Preferred destination for execution.
L orderRoute	Enumeration	Order route name.
L venueHoldingOrder	Enumeration	Venue where non-marketable portion of the order was booked.
L commissionCharged	Double	Total commission amount charged for this order.
L exchangeOrderId	String	Identifier assigned to this order by exchange where it was routed.
L isSignificantShareholder	Boolean	Whether user that placed the order is a significant shareholder.
L isInsider	Boolean	Whether user that placed the order is an insider.
L isLimitOffsetInDollar	Boolean	Whether limit offset is specified in dollars (vs. percent).
L userId	Integer	Internal identifier of user that placed the order.
L placementCommission	Double	Commission for placing the order via the Trade Desk over the phone.
L legs	Complex	List of OrderLeg elements.
L OrderLeg	Complex	
L strategyType	Enumeration	Multi-leg strategy to which the order belongs.
L triggerStopPrice	Double	Stop price at which order was triggered.
L orderGroupId	Integer	Internal identifier of the order group.
L orderClass	Enumeration	Bracket Order class. Primary, Profit or Loss.

Sample request

```
GET https://api01.iq.questrade.com/v1/accounts/26598145/orders
```

Sample request

```
GET https://api01.iq.questrade.com/v1/accounts/26598145/
orders?ids=173577870,...
```

Sample JSON response


```
{
    "orders": [
        {   "id": 173577870,
            "symbol": "AAPL",
            "symbolId": 8049,
            "totalQuantity": 100,
            "openQuantity": 100,
            "filledQuantity": 0,
            "canceledQuantity": 0,
            "clientSide": "Buy",
            "type": "Limit",
            "limitPrice": 500.95,
            "stopPrice": null,
            "isAllOrNone": false,
            "isAnonymous": false,
            "icebergQty": null,
            "minQuantity": null,
            "avgExecPrice": null,
            "lastExecPrice": null,
            "source": "TradingAPI",
            "timeInForce": "Day",
            "gtdDate": null,
            "state": "Canceled",
            "clientReasonStr": "",
            "chainId": 173577870,
            "creationTime": 2014-10-23T20:03:41.636000-04:00,
            "updateTime": 2014-10-23T20:03:42.890000-04:00,
            "notes": "",
            "primaryRoute": "AUTO",
            "secondaryRoute": "",
            "orderRoute": "LAMP",
            "venueHoldingOrder": "",
            "comissionCharged": 0,
            "exchangeOrderId": "XS173577870",
            "isSignificantShareHolder": false,
            "isInsider": false,
            "isLimitOffsetInDollar": false,
            "userId": 3000124,
            "placementCommission": null,
            "legs": [],
            "strategyType": "SingleLeg",
            "triggerStopPrice": null,
            "orderGroupId": 0,
            "orderClass": null,
        },
        ...
    ]
}
```


Market calls

GET symbols/:id

Retrieves detailed information about one or more symbol.

Request parameters

Parameter	Type	Description
id	Integer	Internal symbol identifier. Can occur in the 'Location' header only. Mutually exclusive with 'ids' parameter.
ids	Complex	Comma-separated list of symbol ids.
L id	Integer	Internal symbol identifier.

Response properties

Property	Type	Description
symbol	String	Symbol that follows Questrade symbology (e.g., "TD.TO").
symbolId	Integer	Symbol identifier.
prevDayClosePrice	Double	Closing trade price from the previous trading day.
highPrice52	Double	52-week high price.
lowPrice52	Double	52-week low price.
averageVol3Months	Integer	Average trading volume over trailing 3 months.

Property	Type	Description
averageVol20Days	Integer	Average trading volume over trailing 20 days.
outstandingShares	Integer	Total number of shares outstanding.
eps	Double	Trailing 12-month earnings per share.
pe	Double	Trailing 12-month price to earnings ratio.
dividend	Double	Dividend amount per share.
yield	Double	Dividend yield (dividend / prevDayClosePrice).
exDate	DateTime	Dividend ex-date.
marketCap	Double	Market capitalization (outstandingShares * prevDayClosePrice).
optionType	Enumeration	Option type (e.g., “call”).
optionDurationType	Enumeration	Option duration type (e.g., “weekly”).
optionRoot	String	Option root symbol (e.g., “MSFT”).
optionContractDeliverables	Complex	Option contract deliverables.
L underlyings	Complex	List of UnderlyingMultiplierPair records.
L UnderlyingMultiplierPair	Complex	
L multiplier	Integer	Number of shares deliverable per contract (e.g., 100).
L underlyingSymbol	String	Underlying symbol for the deliverable (e.g., “MSFT”).
L underlyingSymbolId	String	Underlying symbol id for the deliverable (e.g., 2345343).
L cashInLieu	Double	Amount of cash in lieu deliverable per contract.
optionExerciseType	Enumeration	Option exercise style (e.g., “American”).

Property	Type	Description
listingExchange	Enumeration	Primary listing exchange.
description	String	Symbol description (e.g., “Microsoft Corp.”).
securityType	Enumeration	Security type (e.g., “Stock”).
optionExpiryDate	Date time	Option expiry date.
dividendDate	Date time	Dividend declaration date.
optionStrikePrice	Double	Option strike price.
isQuotable	Boolean	Indicates whether the symbol is actively listed.
hasOptions	Boolean	Indicates whether the symbol is an underlying option.
currency	String	String Currency code (follows ISO format).
minTicks	Complex	List of MinTickData records.
L MinTickData	Complex	
L pivot	Double	Beginning of the interval for a given minimum price increment.
L minTick	Double	Minimum price increment.

Sample request

```
GET https://api01.iq.questrade.com/v1/symbols/8049
```

Sample request

```
GET https://api01.iq.questrade.com/v1/symbols?ids=8049,...
```

Sample JSON response


```
{
    "symbols": [
        {
            "symbol": "AAPL",
            "symbolId": 8049,
            "prevDayClosePrice": 102.5,
            "highPrice52": 102.9,
            "lowPrice52": 63.89,
            "averageVol3Months": 43769680,
            "averageVol20Days": 12860370,
            "outstandingShares": 5987867000,
            "eps": 6.2,
            "pe": 16.54,
            "dividend": 0.47,
            "yield": 1.84,
            "exDate": "2014-08-07T00:00:00.000000-04:00",
            "marketCap": 613756367500,
            "tradeUnit": 1,
            "optionType": null,
            "optionDurationType": null,
            "optionRoot": "",
            "optionContractDeliverables": {
                "underlyings": [],
                "cashInLieu": 0
            },
            "optionExerciseType": null,
            "listingExchange": "NASDAQ",
            "description": "APPLE INC",
            "securityType": "Stock",
            "optionExpiryDate": null,
            "dividendDate": "2014-08-14T00:00:00.000000-04:00",
            "optionStrikePrice": null,
            "isTradable": true,
            "isQuotable": true,
            "hasOptions": true,
            "currency": "USD",
            "minTicks": [
                {
                    "pivot": 0,
                    "minTick": 0.0001
                },
                {
                    "pivot": 1,
                    "minTick": 0.01
                }
            ]
        },
        ...
    ]
}
```


GET symbols/search

Retrieves symbol(s) using several search criteria.

Request parameters

Parameter	Type	Description
prefix	String	Prefix of a symbol or any word in the description.
offset	Integer	Offset in number of records from the beginning of a result set.

Response properties

Property	Type	Description
symbols	Complex	List of EquitySymbol elements.
L EquitySymbol	Complex	
L symbol	String	Symbol name.
L symbolId	Integer	Internal unique symbol identifier.
L description	String	Symbol description.
L securityType	Enumeration	Symbol security type.
L listingExchange	Enumeration	Primary listing exchange of the symbol.
L isQuotable	Boolean	Whether a symbol has live market data.
L isTradable	Boolean	Whether a symbol is tradable on the platform.
L currency	String	Symbol currency (ISO format).

Sample request

```
GET https://api01.iq.questrade.com/v1/symbols/search?prefix=BMO
```

Sample JSON response


```
{
    "symbol": [
        {
            "symbol": "BMO",
            "symbolId": 9292,
            "description": "BANK OF MONTREAL",
            "securityType": "Stock",
            "listingExchange": "NYSE",
            "isTradable": true,
            "isQuotable": true,
            "currency": "USD"
        },
        {
            "symbol": "BMO.PRJ.TO",
            "symbolId": 9300,
            "description": "BANK OF MONTREAL CL B SR 13",
            "securityType": "Stock",
            "listingExchange": "TSX",
            "isTradable": true,
            "isQuotable": true,
            "currency": "CAD"
        }
    ]
}
```


GET symbols/:id/options

Retrieves an option chain for a particular underlying symbol.

Request parameters

Parameter	Type	Description
id	Integer	Internal symbol identifier. Can occur in the 'Location' header only.

Response properties

Property	Type	Description
symbols	Complex	List of ChainPerExpiryDate elements.
L ChainPerExpiryDate	Complex	
L expiryDate	DateTime	Option expiry date.
L description	String	Description of the underlying option.
L optionType	Enumeration	Option type (e.g., "call").
L listingExchange	Enumeration	Primary listing exchange.
L optionExerciseType	Enumeration	Option exercise style (e.g., "American").
L chainPerRoot	Complex	List of ChainPerRoot elements.
L ChainPerRoot	Complex	
L optionRoot	String	Option root symbol.
L chainPerStrikePrice	Complex	List of ChainPerStrikePrice elements.
L ChainPerStrikePrice	Complex	
L strikePrice	Double	Option strike price.
L callSymbolId	Integer	Internal identifier of the call option symbol.
L putSymbolId	Integer	Internal identifier of the put option symbol.

Sample request

```
GET https://api01.iq.questrade.com/v1/symbols/9291/options
```

Sample JSON response


```
{
    "options": [
        {       "expiryDate": "2015-01-17T00:00:00.000000-05:00",
                "description": "BANK OF MONTREAL",
                "optionType": "Call",
                "listingExchange": "MX",
                "optionExerciseType": "American",
                "chainPerRoot": [
                    {
                        "root": "BMO",
                        "chainPerStrikePrice": [
                        {
                            "strikePrice": 60,
                            "callSymbolId": 6101993,
                            "putSymbolId": 6102009
                        },
                        {
                            "strikePrice": 62,
                            "callSymbolId": 6101994,
                            "putSymbolId": 6102010
                        },
                        {
                            "strikePrice": 64,
                            "callSymbolId": 6101995,
                            "putSymbolId": 6102011

                        },
                        ...

                    ],
                    "multiplier": 100
                }
            ]
        },
    ]
}
```


GET markets

Retrieves information about supported markets.

Request parameters

No parameters.

Response properties

Property	Type	Description
markets	Complex	List of market records.
L Market	Complex	
L name	String	Market name.
L tradingVenues	Complex	List of trading venue codes.
L code	Enumeration	Order execution venue code.
L defaultTradingVenue	Enumeration	Default trading venue code.
L primaryOrderRoutes	Complex	List of primary order route codes.
L code	Enumeration	Order route code.
L secondaryOrderRoutes	Complex	List of secondary order route codes.
L code	Enumeration	Order route code.
L level1Feeds	Complex	List of level 1 market data feed codes.
L code	Enumeration	Level 1 market data feed code.
L level2Feeds	Complex	List of level 2 market data feed codes.
L code	Enumeration	Level 2 market data feed code.
L extendedStartTime	DateTime	Pre-market opening time for current trading date.
L startTime	DateTime	Regular market opening time for current trading date.

Property	Type	Description
L endTime	DateTime	Regular market closing time for current trading date.
L extendedEndTime	DateTime	Extended market closing time for current trading date.
L currency	Enumeration	Currency code (ISO format).
L snapQuotesLimit	Integer	Number of snap quotes that the user can retrieve from a market.

Sample request

```
GET https://api01.iq.questrade.com/v1/markets
```

Sample JSON response


```
{
    "markets": [
            {
                "name": "TSX",
                "tradingVenues": [
                    "TSX",
                    "ALPH",
                    "CHIC",
                    "OMGA",
                    "PURE"
                ],
                "defaultTradingVenue": "AUTO",
                "primaryOrderRoutes": [
                    "AUTO"
                ],
                "secondaryOrderRoutes": [
                    "TSX",
                    "AUTO"
                ],
                "level1Feeds": [
                    "ALPH",
                    "CHIC",
                    "OMGA",
                    "PURE",
                    "TSX"
                ],
                "level2Feeds": [
                    "ALPH",
                    "CHIC",
                    "OMGA",
                    "PURE",
                    "TSX"
                ],
                "extendedStartTime": "2014-10-06T07:00:00.000000-04:00",
                "startTime": "2014-10-06T09:30:00.000000-04:00",
                "endTime": "2014-10-06T16:00:00.000000-04:00",
                "currency": "CAD",
                "snapQuotesLimit": 99999
            },
            ...
    ]
}
```


GET markets/quotes/:id

Retrieves a single Level 1 market data quote for one or more symbols.

Request parameters

Parameter	Type	Description
id	Integer	Internal symbol identifier. Can occur in the 'Location' header only.
ids	Complex	Comma-separated list of symbol ids.
L id	Integer	Internal symbol identifier.

Response properties

Property	Type	Description
quotes	Complex	List of Quote records.
L Quote	Complex	
L symbol	Integer	Symbol name following Questrade's symbology.
L symbolId	String	Internal symbol identifier.
L tier	Enumeration	Market tier.
L bidPrice	Double	Bid price.
L bidSize	Integer	Bid quantity.
L askPrice	Double	Ask price.
L askSize	Integer	Ask quantity.
L lastTradeTrHrs	Double	Price of the last trade during regular trade hours.
L lastTradePrice	Double	Price of the last trade.
L lastTradeSize	Integer	Quantity of the last trade.
L lastTradeTick	Enumeration	Trade direction.
L volume	Integer	Volume.
L openPrice	Double	Opening trade price.
L highPrice	Double	Daily high price.
L lowPrice	Double	Daily low price.
L delay	Boolean	Whether a quote is delayed (true) or real-time.
L isHalted	Boolean	Whether trading in the symbol is currently halted.

Sample request

```
GET https://api01.iq.questrade.com/v1/markets/quotes/38738
```

Sample request

```
GET https://api01.iq.questrade.com/v1/markets/quotes?names=THI.TO
```

Sample request

```
GET https://api01.iq.questrade.com/v1/markets/quotes?ids=38738,...
```

Sample JSON response


```
{
    "quotes": [
        {
            "symbol": "THI.TO",
            "symbolId": 38738,
            "tier": "",
            "bidPrice": 83.65,
            "bidSize": 6500,
            "askPrice": 83.67,
            "askSize": 9100,
            "lastTradePriceTrHrs": 83.66,
            "lastTradePrice": 83.66,
            "lastTradeSize": 3100,
            "lastTradeTick": "Equal",
            "lastTradeTime": 2014-10-24T20:06:40.131000-04:00,
            "volume": 80483500,
            "openPrice": 83.66,
            "highPrice": 83.86,
            "lowPrice": 83.66,
            "delay": 0,
            "isHalted": false,
        },
        ...
    ]
}
```

GET markets/candles/:id

Retrieves historical market data in the form of OHLC candlesticks for a specified symbol.

This call is limited to returning 2,000 candlesticks in a single response.

Request parameters

Parameter	Type	Description
symbolId	Integer	Internal symbol identifier. Can occur in the 'Location' header only.
startTime	DateTime	Beginning of the candlestick range.
endTime	DateTime	End of the candlestick range.
interval	Enumeration	Interval of a single candlestick.

Response properties

Property	Type	Description
candles	Complex	List of Candle records.
L Candle	Complex	
L start	DateTime	Candlestick start timestamp (in ISO format).
L end	DateTime	Candlestick end timestamp (in ISO format).
L open	Double	Opening price.
L high	Double	High price.
L low	Double	Low price.
L close	Double	Closing price.
L volume	Integer	Trading volume.

Sample request

```
GET https://api01.iq.questrade.com/v1/markets/candles/38738?startTime=
2014100&endTime=20141020&interval=OneDay
```

Sample JSON response


```
{
    "candles": [
        {
            "start": 2014-01-02T00:00:00.000000-05:00,
            "end": 2014-01-03T00:00:00.000000-05:00,
            "low": 70.3,
            "high": 70.78,
            "open": 70.68,
            "close": 70.73,
            "volume": 983609
        },
        ...
    ]
}
```

Order calls

POST accounts/:id/orders[/:orderId]

Allows to (re)place an order against a certain account or check impact of possible order on the account.

Request parameters

Parameter	Type	Description
id	String	Account number against which order is being submitted. Can occur in the 'Location' header only.
orderId	Integer	Optional – order id of the order to be replaced.
symbolId	Integer	Internal symbol identifier.
quantity	Integer	Order quantity.
icebergQuantity	Integer	Iceberg instruction quantity.
limitPrice	Double	Limit price.
stopPrice	Double	Stop price.
isAllOrNone	Boolean	Identifies whether the all-or-none instruction is enabled.
isAnonymous	Boolean	Identifies whether the anonymous instruction is enabled.
type	Enumeration	Order type (e.g., "market").
action	Enumeration	Order side (e.g., "buy").
primaryRoute	Enumeration	Primary order route (e.g., "AUTO").

Response properties

Property	Type	Description
orderId	Integer	Internal order identifier.
orders	Complex	List of order objects.
L Order	Complex	

Sample JSON request

```
POST https://api01.iq.questrade.com/v1/accounts/26598145/orders

{
    "symbolId": 8049,
    "quantity": 10,
    "icebergQuantity": 1,
    "limitPrice": 537,
    "isAllOrNone": true,
    "isAnonymous": false,
    "type": "Limit",
    "timeInForce": "GoodTillCanceled",
    "action": "Buy",
    "primaryRoute": "LAMP"
}
```

Sample JSON response


```
{
    "orderId": 177106005,
    "orders": [
        {
            "id": 177106005,
            "symbol": "AAPL",
            "symbolId": 8049,
            "totalQuantity": 10,
            "openQuantity": 10,
            "filledQuantity": 0,
            "canceledQuantity": 0,
            "side": "Buy",
            "type": "Limit",
            "limitPrice": 537,
            "stopPrice": null,
            "isAllOrNone": true,
            "isAnonymous": false,
            "icebergQty": 1,
            "minQuantity": null,
            "avgExecPrice": null,
            "lastExecPrice": null,
            "source": "TradingAPI",
            "timeInForce": "GoodTillCanceled",
            "gtdDate": null,
            "state": "Pending",
            "clientReasonStr": "",
            "chainId": 177106005,
            "creationTime": "2014-10-24T17:48:20.546000-04:00",
            "updateTime": "2014-10-24T17:48:20.876000-04:00",
            "notes": "",
            "primaryRoute": "LAMP",
            "secondaryRoute": "",
            "orderRoute": "LAMP",
            "venueHoldingOrder": "",
            "comissionCharged": 0,
            "exchangeOrderId": "",
            "isSignificantShareHolder": false,
            "isInsider": false,
            "isLimitOffsetInDollar": false,
            "userId": 3000124,
            "placementCommission": null,
            "legs": [],
            "strategyType": "SingleLeg",
            "triggerStopPrice": null
            "orderGroupId": 0,
            "orderClass": null,

        },
        ...
    ]
}
```


POST accounts/:id/orders/[/:orderId]/impact

Allows to calculate the impact that a given order will have on the account without placing it.

Request parameters

Parameter	Type	Description
id	String	Account number against which order is being submitted.
orderId	Integer	Optional – Order Id of order to be replace estimated
symbolId	Integer	Internal symbol identifier.
quantity	Integer	Order quantity.
icebergQuantity	Integer	Iceberg instruction quantity.
limitPrice	Double	Limit price.
stopPrice	Double	Stop price.
isAllOrNone	Boolean	Identifies whether All-or-None instruction is enabled.
isAnonymous	Boolean	Identifies whether Anonymous instruction is enabled.
type	Enumeration	Order type (e.g., "Market").
action	Enumeration	Order side (e.g., "Buy").
primaryRoute	Enumeration	Primary order route (e.g., "AUTO").
secondaryRoute	Enumeration	Secondary order route (e.g., "NYSE").

Response properties

Property	Type	Description
estimatedCommissions	Double	Estimate of commissions to be charged on the order.
buyingPowerEffect	Double	Estimate of change in buying power from the order.
buyingPowerResult	Double	Estimate of buying power in which order will result.
maintExcessEffect	Double	Estimate of change in maintenance excess from the order.
maintExcessResult	Double	Estimate of maintenance excess in which the order will result.
side	Enumeration	Client view of the order side (e.g., "Buy-to-Open").
tradeValueCalculation	String	Estimate of the order execution value.
price	Double	Estimated average fill price.

Sample JSON request

```
POST https://api01.iq.questrade.com/v1/accounts/26598145/orders/impact

{
    "symbol": "BMO.TO",
    "quantity": 5,
    "limitPrice": 100,
    "type": "Limit",
    "timeInForce": "Day",
    "action": "Buy",
    "primaryRoute": "AUTO",
    "secondaryRoute": "AUTO"
}
```

Sample JSON response

```
{
    "estimatedCommissions": 4.9675,
    "buyingPowerEffect": -235.475,
    "buyingPowerResult": 1999764.525,
    "maintExcessEffect": -117.7375,
    "maintExcessResult": 999882.2625,
    "side": "Buy",
    "tradeValueCalculation": "5 x 75.18 = 375.90 CAD",
    "price": 102.51
}
```

DELETE accounts/:id/orders/:orderId

Allows to cancel an existing order.

Request parameters

Parameter	Type	Description
id	String	Account number against which order is being submitted. Can occur in the 'Location' header only.
orderId	Integer	Internal identifier of the order. Can occur in the 'Location' header only.

Response properties

Property	Type	Description
orderId	Integer	Internal identifier of the order.

Sample request

```
DELETE https://api01.iq.questrade.com/v1/accounts/26598145/
orders/203526220
```

Sample JSON response


```
{
    "orderId": 203526220
}
```

Contact us

By phone

Client services
Monday to Friday,
8 a.m. to 8 p.m. ET

French speaking support
Monday to Friday,
8 a.m. to 8 p.m. ET

Toll free
Canada: 1.888.783.7866
U.S.: 1.416.227.9876
Outside Canada and the U.S.:
(001) 416.227.9876

Trade desk
Monday to Friday, 8:30 a.m. to 4:30 p.m. ET
Toll free: 1.866.980.9590
From the U.S.: 1.416.227.6911
Outside Canada and the U.S.:
(001) 416.227.6911

Email

API support:
apisupport@questrade.com

New accounts:
newaccounts@questrade.com

Security:
security@questrade.com

Media enquiries:
marketing@questrade.com

The Exchange:
community@questrade.com

Important note: do not send account instructions by email. Email should be used for general enquiries only, and Questrade will not act on any instructions sent using email. Please use myQuestrade to manage your account. Questrade will not accept any financial liability for any act or failure to act on the instructions in an email.

QUESTRADE®