

# The ATrade Desktop™: Open Broker API™ Class Library

Copyright 2008 All Rights Reserved.

---

Written By Cedric L. Harris, Microsoft Solution Architect/President/CEO  
**ATRADE Investment Technologies, LLC**

## Contents

1.0	Purpose .....	3
2.0	Scope .....	3
3.0	Assumptions .....	3
4.0	Dependencies .....	4
5.0	Revisions .....	4
6.0	AmeritradeBrokerAPI Solution Structure .....	5
7.0	AmeritradeBrokerAPI Primary Properties .....	6
8.0	AmeritradeBrokerAPI Primary Methods .....	7
9.0	Nested ATrade Argument Properties .....	8
10.0	ListItemSorter Class .....	9
11.0	Nested ATrade Argument Types .....	10
12.0	Nested ATrade Argument: ATrade Chart Data Structure .....	11
13.0	Nested ATrade Argument: ATrade Level 2 Data Class .....	11
14.0	Nested ATrade Argument: ATrade Login Status Enumeration .....	12
15.0	Nested ATrade Argument: ATrade Message Enumeration .....	12
16.0	Nested ATrade Argument: Order Action Type Enumeration .....	13
17.0	Nested ATrade Argument: Order Status Enumeration .....	13
18.0	Nested ATrade Argument: Order Type Enumeration .....	13
19.0	Nested ATrade Argument: Trade Details Structure .....	14
20.0	Nested ATrade Argument: Trade Reply Details Structure .....	14
21.0	Order History Enumeration .....	15
22.0	Nested ATrade Argument: Trade Type Enumeration .....	15
23.0	Account Cash Balance Structure .....	15
24.0	Level 1 Quotes Structure .....	16
25.0	Positions Enumeration .....	17
26.0	Demo Application .....	18

## 1.0 Purpose

The Open Broker API™ Class Library was developed and written by Cedric L. Harris, Microsoft Solution Architect, President and CEO of ATrade Investment Technologies, LLC as a 100% .NET/C# implementation of the written TD AMERITRADE® API specification. The written TD AMERITRADE® API specification is provided by TD AMERITRADE® to third-party application developers, and used to integrate with TD AMERITRADE® brokerage account(s) and related brokerage account functionality. TD Ameritrade, QuoteTracker, and the Ameritrade Network are registered trademarks of Ameritrade Securities, INC. The Open Broker API™ and its demo application ("ATrade Investment Technologies, LLC : TD Ameritrade API Test Form") represents technology that is wholly owned and maintained by ATrade Investment Technologies, LLC; All rights are reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## 2.0 Scope

This document and its associated source code and binaries are intended for parties authorized by TD Ameritrade® Securities, INC to access the TD Ameritrade® developer's toolkit.

## 3.0 Assumptions

Individuals using this document and its associated source code should be thoroughly familiar with the Microsoft .NET framework 2.0 and greater, and C#.

This API is designed to be as straightforward as possible. To this end, an exhaustive demo application, the ("ATrade Investment Technologies, LLC: TD Ameritrade API Test Form") is included in place of a detailed explanation of each method and its parameters. Wherever a question does arise, the demo and the following outline of the API's class library should be referenced.

## 4.0 Dependencies

This document and associated source code utilize third-party binary libraries specific to:

1. Charting via the ZedGraph.dll charting library, and
2. NZipLib.dll compression and decompression library.

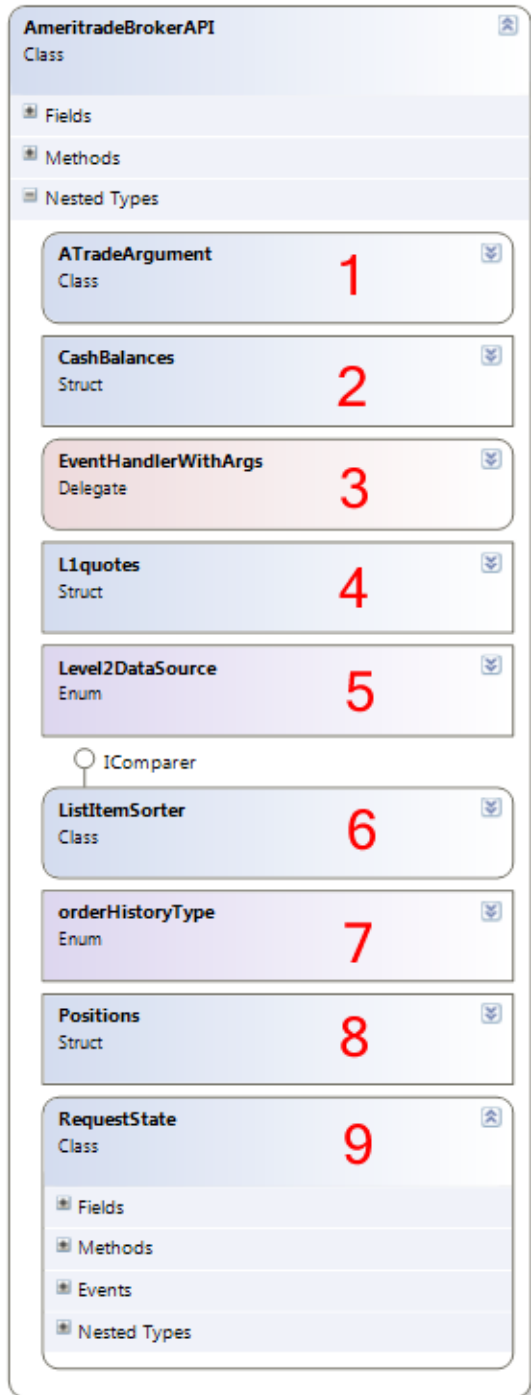
## 5.0 Revisions

This document and its associated source code represent the initial release as a 100% C# implementation of the specification developed by QuoteTracker, INC a wholly owned subsidiary of TD Ameritrade® Securities, INC.

Version	Release Notes
1.00.01	<p><b>Includes:</b></p> <ol style="list-style-type: none"> <li>1. Broker account login functionality</li> <li>2. Broker account logout functionality</li> <li>3. Broker account logout functionality</li> <li>4. Broker account functionality</li> <li>5. Broker account positions functionality</li> <li>6. Broker account trade history functionality</li> <li>7. Asynchronous snapshot of charting data</li> <li>8. Asynchronous streaming of charting data</li> <li>9. Asynchronous snapshot of level 1 data</li> <li>10. Asynchronous streaming of level 1 data</li> <li>11. Asynchronous streaming of level 2 data</li> </ol> <ol style="list-style-type: none"> <li>1. Level 1 stock data lookup functionality</li> <li>2. Evaluating whether a stock is shortable or not shortable</li> <li>3. Stock quote retrieval</li> <li>4. Filter of trade history by trade history types</li> <li>5. Historical stock data retrieval/decompression/parsing functionality</li> <li>6. Charting of historical stock data</li> </ol> <ol style="list-style-type: none"> <li>1. Entering of stock orders</li> <li>2. Modification/editing of stock orders</li> <li>3. Canceling of stock orders</li> </ol> <p><b>Does not include:</b></p> <ol style="list-style-type: none"> <li>1. Stop_Market, Stop_Limit, Trailing Stop Percentage, Trailing Stop Dollar order types</li> <li>2. Fill or Kill (FOK), All or None (AON), Do Not Reduce (DNR), All or None Do Not Reduce (aon_dnr) special instructions</li> </ol>

## 6.0 AmeritradeBrokerAPI Solution Structure

The AmeritradeBrokerAPI represents a single .NET namespace containing several classes, properties, methods, enumerations, and structures used to facilitate access to brokerage account functionality. The following class (AmeritradeBrokerAPI) is used extensively within the test application to demonstrate functionality.



### 1. The ATradeArgument Class

The primary class used to house the bulk of nested classes, enumerations and structures.

### 2. The CashBalances Structure

The structure used to hold and pass data relating to cash balance.

### 3. EventHandlerWithArgs

This class is used to pass data between threads.

### 4. The L1Quotes Structure (Level 1)

The structure used to hold and pass data relating to the retrieval of Level 1 stock quotes.

### 5. Level2DataSource

This enumeration holds the list of market sources for Level 2 data.

### 6. The ListItem Sorter Class

The class used to sort and filter trade history by a selected trade history type.

### 7. The OrderHistory Type Enumeration

The enumeration used to classify order types.

### 8. The Positions Structure

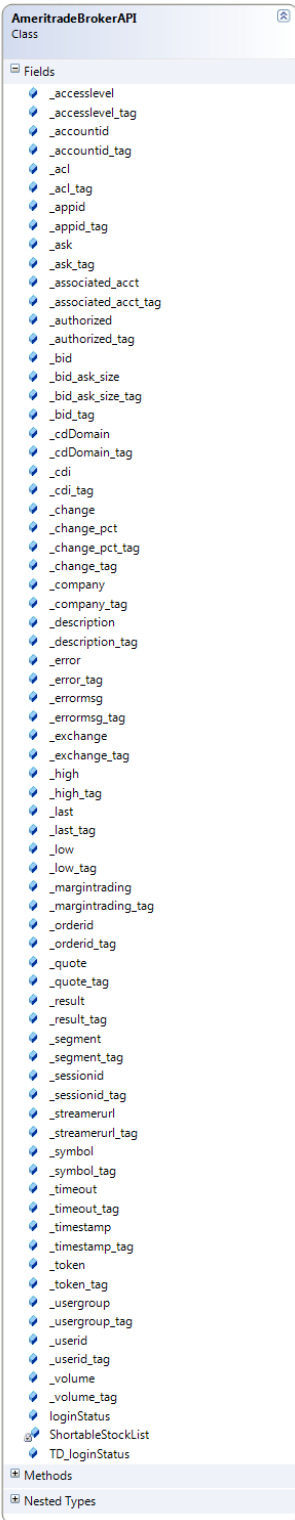
The structure used to hold and pass data relating to stock position(s).

### 9. RequestState Class

The RequestState Class I used by as part of the asynchronous request(s) to pass data to threads created dynamically.

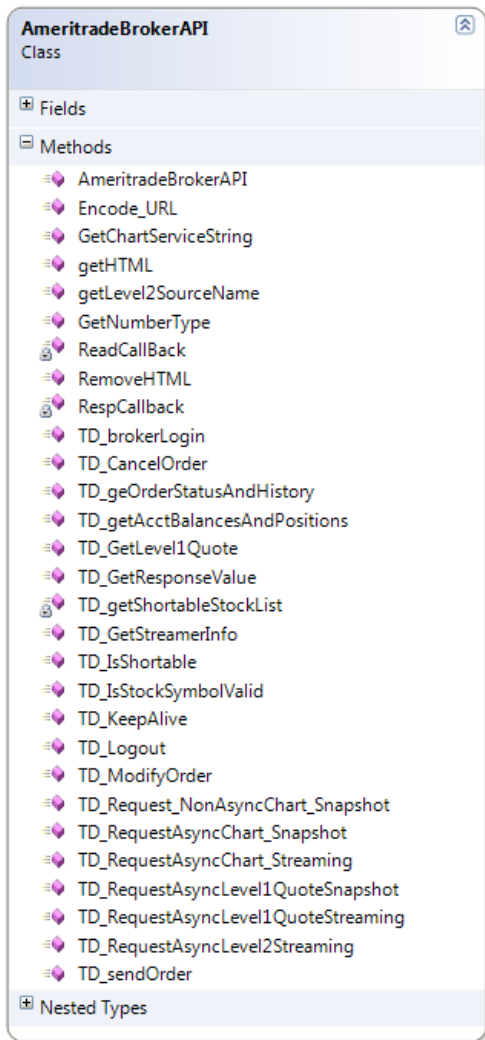
## 7.0 AmeritradeBrokerAPI Primary Properties

At the top of the class structure lie the AmeritradeBrokerAPI properties. These properties contain all relevant information retrieved during login and access to the TD Ameritrade Streamer®.



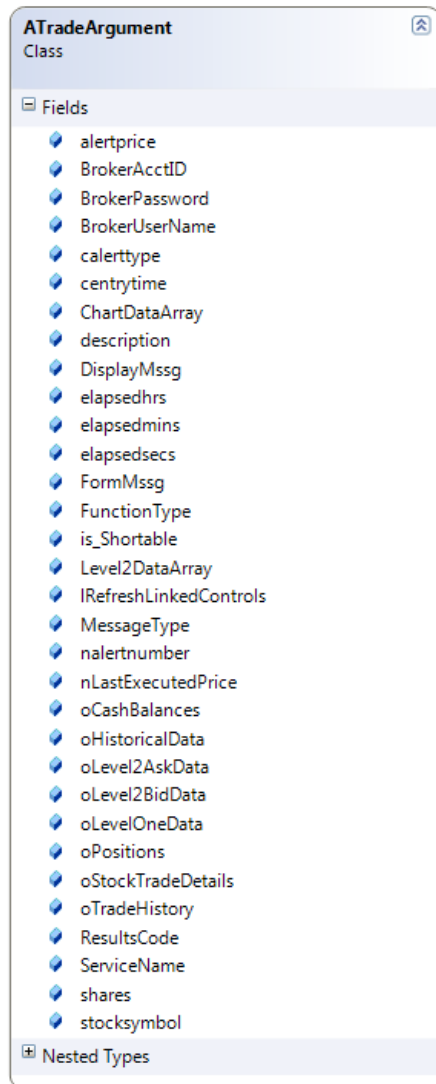
## 8.0 AmeritradeBrokerAPI Primary Methods

Within the API exists a small set of methods used to facilitate access to your account(s). These methods are both directly related to account functional and peripheral utility functions that support primary methods such as encrypting URLs.



## 9.0 Nested ATrade Argument Properties

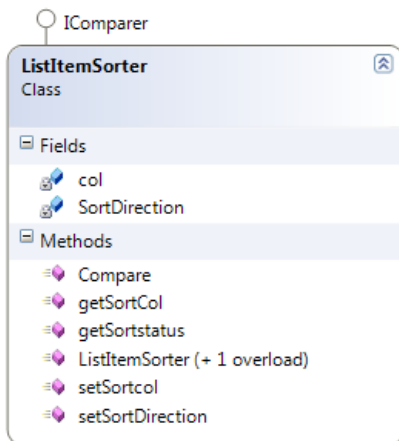
The ATradeArgument represents the primary object within the AmeritradeBrokerAPI Class. As such, this class maintains the majority of properties and structures used to encapsulate data retrieved when communicating with the TD Ameritrade servers. Just a few of these variables are represented below. Where it made sense to encapsulate data within collections, these variables are defined as collections of sorted data.





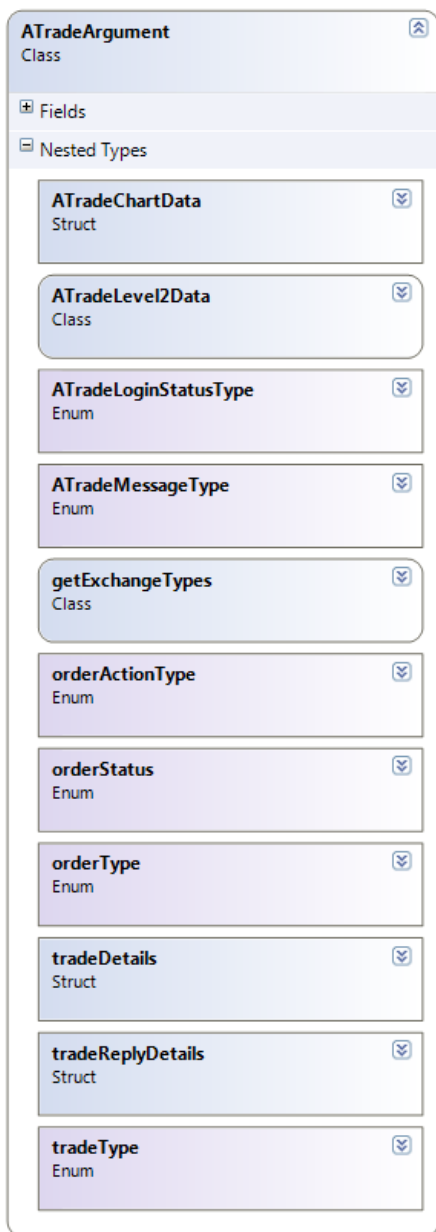
## 10.0 ListItemSorter Class

One of the peripheral classes included for the express purpose of sorting data used to pass to the implementing application is the ***“ListItemSorter”*** Class. This class contains underlying methods used to affect sorting and filtering of specific trade history types.



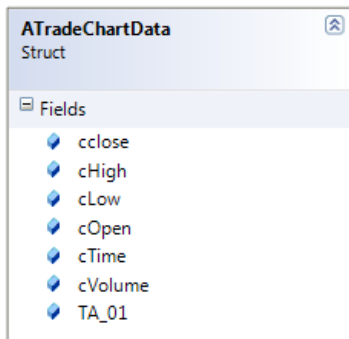
## 11.0 Nested ATrade Argument Types

Within the primary AmeritradeBrokerAPI lay child classes, structures, and enumerations. These additional entities are contained within the ATradeArgument Class as shown in the below image. These entities are at the core of how the primary methods retain and pass data to and from the TD Ameritrade® network.



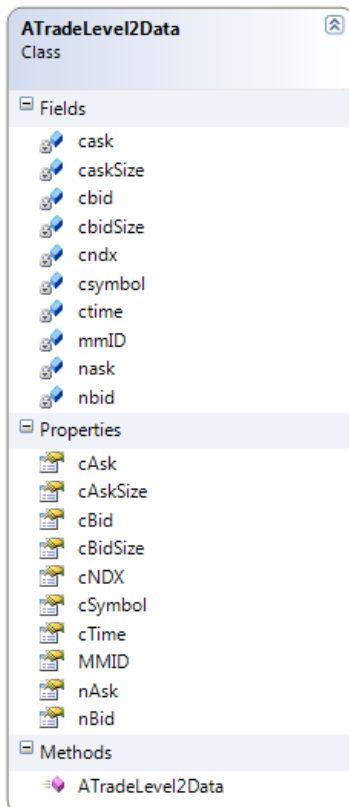
## 12.0 Nested ATrade Argument: ATrade Chart Data Structure

Used to hold relevant data used to chart stock symbols.



## 13.0 Nested ATrade Argument: ATrade Level 2 Data Class

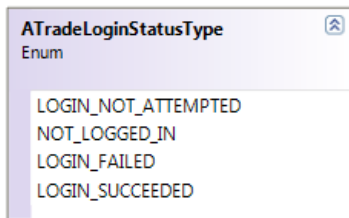
Although Level 2 stock quote data is not implemented with this release of the AmeritradeBrokerAPI, this structure will be used in the subsequent releases to hold relevant data for the future implementation.



## 14.0 Nested ATrade Argument: ATrade Login Status Enumeration

Used internally by the ATrade application to determine the stage of the login attempt.

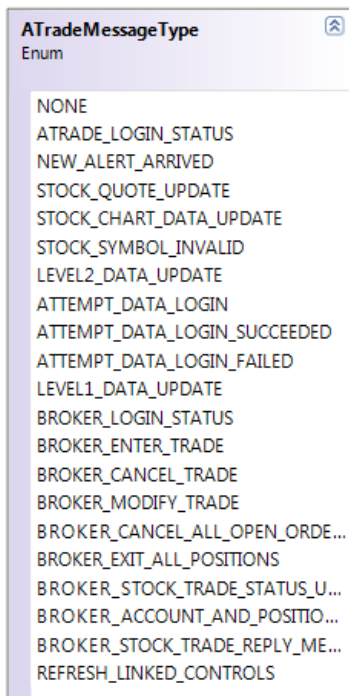
*Note: Not application.*



## 15.0 Nested ATrade Argument: ATrade Message Enumeration

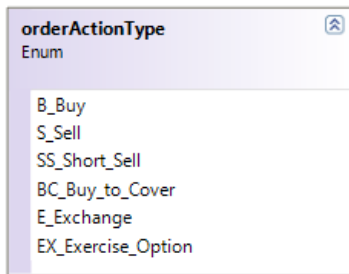
Used internally by the ATrade application to send and receive thread messages where needed.

*Note: Not application.*



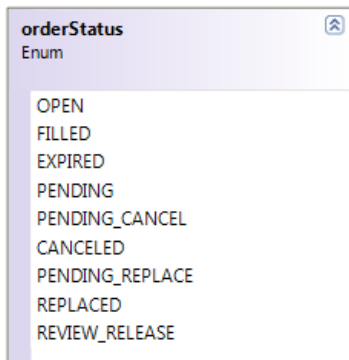
## 16.0 Nested ATrade Argument: Order Action Type Enumeration

This enumeration is used to determine the type of trade being submitted.



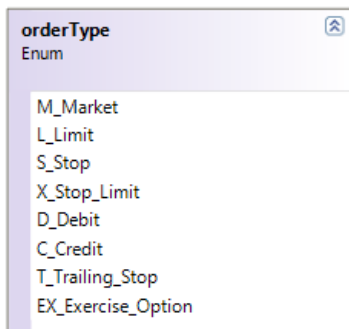
## 17.0 Nested ATrade Argument: Order Status Enumeration

This enumeration is used to determine the status of submitted orders.



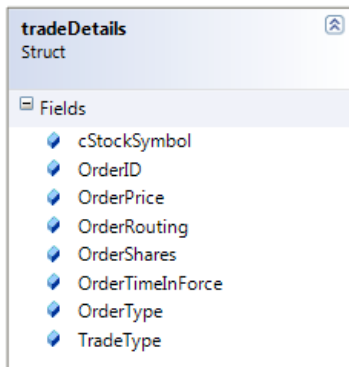
## 18.0 Nested ATrade Argument: Order Type Enumeration

This enumeration is used to determine the type of order being submitted.



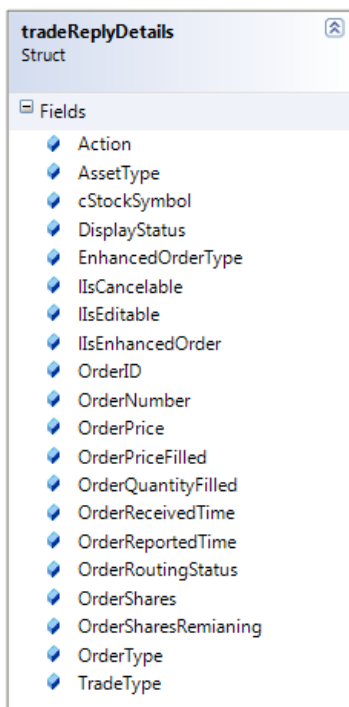
## 19.0 Nested ATrade Argument: Trade Details Structure

This structure is used to hold and pass data relating to the trade history details.



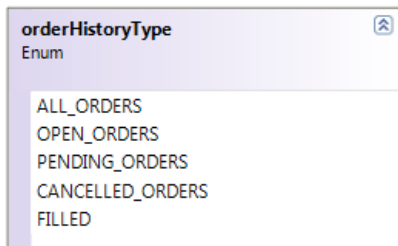
## 20.0 Nested ATrade Argument: Trade Reply Details Structure

This structure is used to hold the reply details of a request to get the trade history for various types of trade. These trade types are defined within the Order History Enumerations.



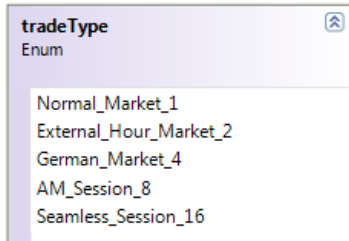
## 21.0 Order History Enumeration

This enumeration is used by the Trade Reply details to filter and sort.



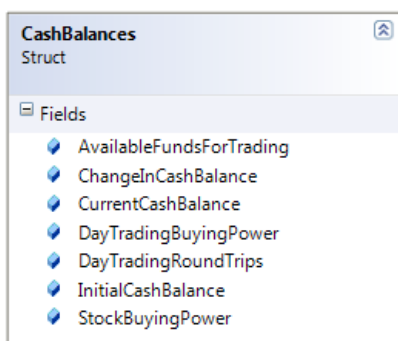
## 22.0 Nested ATrade Argument: Trade Type Enumeration

This enumeration is used to determine the expiration/life of a submitted order.



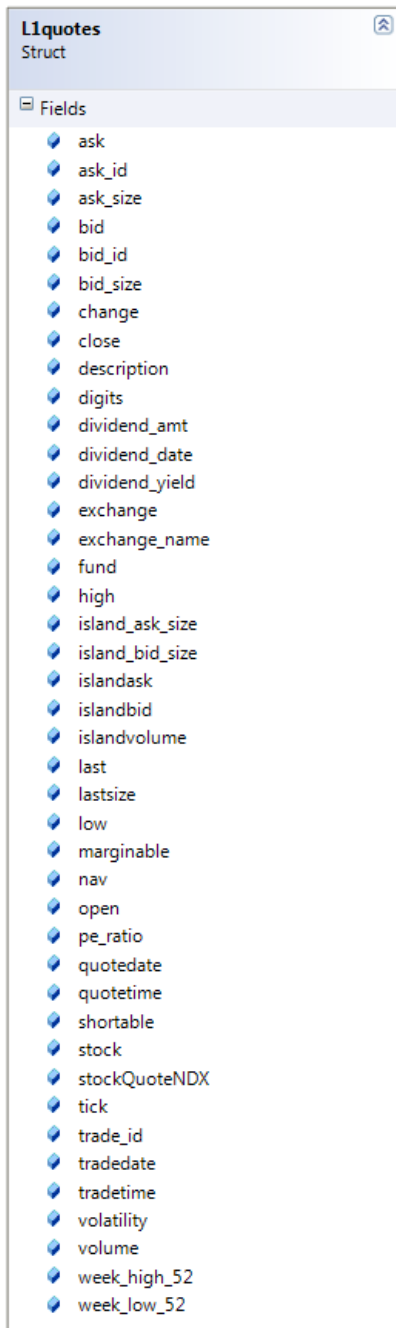
## 23.0 Account Cash Balance Structure

This structure is used to store the relevant account balance information.



## 24.0 Level 1 Quotes Structure

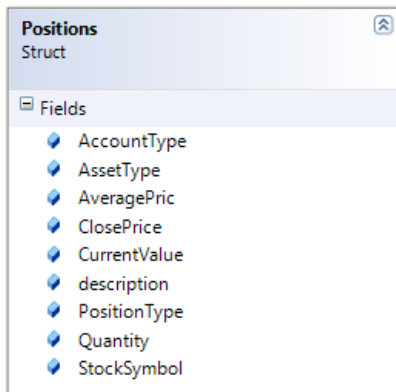
This structure is used to retrieve basic stock quote information and company lookup information.





## 25.0 Positions Enumeration

This enumeration is used to hold and pass data relating to current positions.



## 26.0 Demo Application

This demo application represents the full implementation of the AmeritradeBrokerAPI. Using the API, first, requires a TD Ameritrade® assigned Source ID used to uniquely identify each application utilizing the TD Ameritrade® network.

Once the Source ID, Ameritrade Broker Username and password have been entered, simply click the button labeled “Login & Get Data”. This will activate several method calls to the API and populate all relevant data and enable disabled controls. Once the Account Information and Order Entry controls have been activated, you may access additional API functionality.

The screenshot shows the 'ATrade Investment Technologies, LLC: TD Ameritrade API Test Form' window. It contains the following sections:

- TD Ameritrade API Login Details:** Fields for Source ID, User Name, Password, and a Symbol field with 'HES' entered. There is an 'Encrypt' checkbox.
- Stock Symbol Info:** Buttons for 'Login', 'Get Data', and 'Log Out'.
- Level I Data:** Fields for Company Name, Shortable Status, Last Executed Price, Exchange, Today's Volume, Percent Changed, Price Changed, Today's Low, Today's High, Bid, and Ask.
- Order Entry:** Fields for Order ID, Symbol, Price, and Shares. Dropdowns for Trade Type (set to 'buy'), Order Type (set to 'limit'), Order Routing (set to 'auto'), and Time In Force (set to 'day'). Buttons for 'Enter Order', 'Modify Order', and 'Cancel Order'.
- Balances:** A table showing account balances:
 

Category	Value
Current Cash Balance	\$0.00
Initial Cash Balance	\$0.00
\$ Change In Cash Balance	\$0.00
% Change In Cash Balance	0.00%
Stock Buying Power	\$0.00
Day Trading Buying Power	\$0.00
Available Funds For Trading	\$0.00
Day Trading Round Trips	0
- Historical Stock Data Test:** Includes a 'History View' dropdown set to 'ALL', a 'Records' field, and a 'Chart Availability' dropdown. Below is a table for 'Uncompressed Historical (Chart) Data':
 

Time	Hi	Low	Open	Close	Volume
- Level II Data:** Two tables for 'Buy Orders (BID)' and 'Sell Orders (ASK)'. Each table has columns for MMID, Size, and Price. Below these is a 'Streaming Chart Data Record(s)' section with a 'Streaming Chart Data' table.

ATrade Investment Technologies, LLC : TD Ameritrade API Test Form

Functions | Charting

TD Ameritrade API Login Details

TD Ameritrade Source ID: FM  
 TD Ameritrade User Name: \*\*\*\*\*  
 TD Ameritrade Password: \*\*\*\*\*  
☒ Encrypt    Symbol: HES

Stock Symbol Info

Login    Get Data    Log Out

Level I Data

Company Name: HESS CORP COM  
 Last Executed Price: 123.53  
 Exchange: NYSE  
 Percent Changed: -2.00  
 Today's Low: 120.06  
 Bid: 124.5

Shortable Status: Shortable  
 Today's Volume: 5226528  
 Price Changed: -2.4664  
 Today's High: 124.93  
 Ask: 124.8

Order Entry

Order ID:   
 Symbol:   
 Price:   
 Shares:   
 Trade Type: buy  
 Order Type: limit  
 Order Routing: auto  
 Time In Force: day

Enter Order    Modify Order    Cancel Order

Level II Data

Buy Orders (BID) 3			Sell Orders (ASK) 3		
MMID	Size	Price	MMID	Size	Price
NASDAQ	124.5	100	PACX	124.8	100
PACX	122.07	200	NASDAQ	125.0	100

Streaming Chart Data Record(s)

Streaming Chart Data

HES,505,123.5336,123.5336,123.5336,123.5336,6500,59129,14027,

Balances    Positions    Trade History

History View: ALL

Current Cash Balance: \$79.56  
 Initial Cash Balance: \$79.56  
 \$ Change In Cash Balance: \$0.00  
 % Change In Cash Balance: 0.00%  
 Stock Buying Power: \$79.56  
 Day Trading Buying Power: \$0.00  
 Available Funds For Trading: \$79.56  
 Day Trading Round Trips: 0

Historical Stock Data Test (For use in charting of stock symbols)

Records: 401    Chart Availability: Chart available

Uncompressed Historical (Chart) Data

Time	Hi	Low	Open	Close	Volume
1612	125.23	122.56	122.7	125.23	1550
1614	122.7	122.7	122.7	122.7	8900
1615	124.93	124.88	124.93	124.88	400
1616	124.86	124.86	124.86	124.86	100
1619	124.02	124.02	124.02	124.02	200
1621	123.07	123.07	123.07	123.07	6000
1625	123.53	123.53	123.53	123.53	6500

