# ShoonyaApi-dotnet

[INTRODUCTION](#md_introduction)

[Login and Session](#md_session)

* [Login](#md_login)
* [UserDetails](#md_userdetails)
* [ForgotPassword](#md_forgot)
* [ChangePassword](#md_changepwd)
* [Logout](#md_logout)
* [SetSession](#md_setsession)

[WatchLists](#md_watchlist)

* [GetWatchListNames](#md_getwatchlistnames)
* [GetWatchList](#md_getwatchlist)
* [AddScriptoWatchList](#md_addscripwatchlist)
* [DeleteScriptoWatchList](#md_delscripwatchlist)

[Market](#md_market)

* [SearchScrips](#md_searchscrips)
* [GetSecurityInfo](#md_securityinfo)
* [GetQuote](#_TOC_250012)
* [GetTimePriceData(Chartdata)](#md_tpseries)
* [GetOptionChain](#md_optionchain)
* [GetIndexList](#md_indexlist)
* [ExchMsg](#md_exchmsg)
* [TopListNames](#md_toplistnames)
* [TopList](#md_toplist)

[Orders and Trades](#md_ordersntrades)

* [PlaceOrder](#md_placeorder)
* [ModifyOrder](#md_modifyorder)
* [CancelOrder](#md_cancelorder)
* [ExitSNOOrder](#md_exitorder)
* [ProductConversion](#md_prdconvert)
* [OrderMargin](#md_ordermargin)
* [OrderBook](#md_orderbook)
* [TradeBook](#md_tradebook)
* [SingleOrderHistory](#md_orderhistory)
* [MultiLegOrderBook](#md_mlorderbook)
* [BasketOrderMargin](#_TOC_250024)
* [PositionsBook](#md_positions)
* [Holdings](#md_holdings)
* [Limits](#md_limits)

[Order and MarketData Update](#md_ordermktupdate)

* [Connect](#md_connect)
* [SubscribeMarketData](#md_subscribe)
* [UnSubscribeMarketData](#md_unsubscribe)
* [SubscribeOrderUpdate](#md_subscribeorders)

# INTRODUCTION: About the API

The Api is a dotNet wrapper of the NorenAPI which offers a combination of Rest calls and WebSocket for the purposes of Trading.

API is developed on VisualStudio2019, it uses .NetStandard 2.0 and has a dependency on Newtonsoft.Json 9.0.1

The namespace NorenRestApiWrapper and class NorenRestApi are of primary use and interest

We will be creating an object of NorenRestApi to make requests, the callback is an argument of the request method.

Initialize the api the following are needed

endPoint: The api end point as instructed by your Broker  
Appkey : The secretkey issued to you, donot append the userid to it.

# Login and Session

## Login

###### public bool SendLogin(OnResponse response,string endPoint,LoginMessage login)

connect to the broker, only once this function has returned successfully can any other operations be performed

Loginrequest takes three arguments

1. Callback: this is the function where the application will be handling the response
2. Endpoint: OMS address
3. MessageData: parameters of the request being made.

The Callback is of signature

###### public delegate void OnResponse(NorenResponseMsg Response,bool ok)

Example:

LoginMessage loginMessage = new LoginMessage();  
loginMessage.uid = uid;  
loginMessage.pwd = pwd;  
loginMessage.factor2 = pan;  
loginMessage.imei = "134243434";  
loginMessage.source = "API";  
loginMessage.appkey = appkey;  
  
nApi.SendLogin(Program.OnAppLoginResponse, endPoint, loginMessage);

The callback will be handled as below

public static void OnAppLoginResponse(NorenResponseMsg Response, bool ok)  
{  
 LoginResponse loginResp= Response as LoginResponse;  
  
 if(loginResp.stat=="Ok")  
 {  
 //do all work here  
 }  
}

The Response is casted to expected DataType ie in this example being LoginResponse,

stat is checked to see if the request was successful.

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| apkversion\* |  | Application version. |
| uid\* |  | User Id of the login user |
| pwd\* |  | Sha256 of the user entered password. |
| factor2\* |  | DOB or PAN as entered by the user. (DOB should be in DD-MM-YYYY) |
| vc\* |  | Vendor code provided by noren team, along with connection URLs |
| appkey\* |  | Sha256 of uid |
| imei\* |  | Send mac if users logs in for desktop, imei is from mobile |
| addldivinf |  | Optional field, Value must be in below format: |
| ipaddr |  | Optional field |
| source | API |  |

Response is of type LoginResponse

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Login Success Or failure status |
| susertoken |  | It will be present only on login success. This data to be sent in subsequent requests in jKey field and web socket connection while connecting. |
| lastaccesstime |  | It will be present only on login success. |
| spasswordreset | Y | If Y Mandatory password reset to be enforced. Otherwise the field will be absent. |
| exarr |  | Json array of strings with enabled exchange names |
| uname |  | User name |
| prarr |  | Json array of Product Obj with enabled products, as defined below. |
| actid |  | Account id |
| email |  | Email Id |
| brkname |  | Broker id |
| emsg |  | This will be present only if Login fails. |

## UserDetails

###### public bool SendGetUserDetails(OnResponse response)

Example:

//get user details  
nApi.SendGetUserDetails(Handlers.OnUserDetailsResponse);

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |

Response is of type UserDetailsResponse

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| exarr |  | list of exchanges enabled |
| orarr |  | ordertypes enabled for the user |
| prarr |  | list of product object |
| brkname |  | | Region Category |
| brnchid |  | Branch |
| actid |  | | Account Id |
| email |  | email id |

## Logout

###### public bool SendLogout(OnResponse response)

Terminate the current session with the server.

Example:

ret = nApi.SendLogout()

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |

Response is of type LogoutResponse :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Logout Success Or failure status |
| request\_time |  | It will be present only on successful logout. |
| emsg |  | This will be present only if Logout fails. |

## ForgotPassword

###### public bool SendForgotPassword(OnResponse response,string endpoint,string user,string pan,string dob)

Example:

ret = nApi.SendForgotPassword(OnForgotPwdResponse, endpoint, user, pan, dob);

Request Details :

| Json Fields| Possible value| Description|  
| --- | --- | --- |  
| uid\* | | User Id |  
| pan\* | | pan of the user |  
| dob\* | | Date of birth |

Response is of type ForgotPasswordResponse:

| Json Fields| Possible value| Description|  
| --- | --- | --- |  
| stat | Ok or Not\_Ok | Success Or failure status |  
| emsg | | This will be present only if request fails. |

## ChangePassword

###### public bool Changepwd(OnResponse response,Changepwd changepwd)

Example:

ret = nApi.SendChangepwd(OnChangePwdResponse, changepwd);

Request Details : Changepwd

| Json Fields| Possible value| Description|  
| --- | --- | --- |  
| uid\* | | User Id |  
| oldpwd\* | | old password of the user |  
| pwd\* | | new password of the user |

Response is of type ChangepwdResponse:

| Json Fields| Possible value| Description|  
| --- | --- | --- |  
| stat | Ok or Not\_Ok | Success Or failure status |  
| emsg | | This will be present only if request fails. |

## SetSession

This method initializes the api with an existing session instead of creating a new session with SendLogin.

###### public bool SetSession(string endpoint, string uid, string pwd, string usertoken)

Request Details :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |  
| endpoint || server endpoint|  
|uid || user |  
|pwd || password |  
|usertoken|| session id from loginresponse. |

Response is of type bool :

| Json Fields| Possible value| Description|  
| --- | --- | --- |  
|Value|True/False| |

# WatchLists

## GetWatchListNames

###### public bool SendGetMWList(OnResponseresponse)

Request Details :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

Response is of type MWListResponse :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

## GetWatchList

###### public bool SendGetMarketWatch(OnResponse response,string wlname)

Request Details :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

Response is of type MarketWatchResponse :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

## AddScriptoWatchList

###### public bool SendAddMultiScripsToMW(OnResponse response,string watchlist,string scrips)

Request Details :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

Response is of type StandardResponse :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

## DeleteScriptoWatchList

###### public bool SendDeleteMultiMWScrips( OnResponse response,string watchlist,string scrips)

Request Details :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

Response is of type StandardResponse :  
| Json Fields| Possible value| Description|  
| --- | --- | --- |

## SearchScrips

###### public bool SendSearchScrip(OnResponse response,string exch,string searchtxt)

Search for scrip or contract and its properties

The call can be made to get the exchange provided token for a scrip or alternately can search for a partial string to get a list of matching scrips  
Trading Symbol:

SymbolName + ExpDate + 'F' for all data having InstrumentName starting with FUT

SymbolName + ExpDate + 'P' + StrikePrice for all data having InstrumentName starting with OPT and with OptionType PE

SymbolName + ExpDate + 'C' + StrikePrice for all data having InstrumentName starting with OPT and with OptionType C

For MCX, F to be ignored for FUT instruments

Example:

api.SendSearchScrip(Program.OnResponse, 'NSE', 'REL');

This will reply as following

{  
 "stat": "Ok",  
 "values": [  
 {  
 "exch": "NSE",  
 "token": "18069",  
 "tsym": "REL100NAV-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "24225",  
 "tsym": "RELAXO-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "4327",  
 "tsym": "RELAXOFOOT-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "18068",  
 "tsym": "RELBANKNAV-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "2882",  
 "tsym": "RELCAPITAL-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "18070",  
 "tsym": "RELCONSNAV-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "18071",  
 "tsym": "RELDIVNAV-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "18072",  
 "tsym": "RELGOLDNAV-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "2885",  
 "tsym": "RELIANCE-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "15068",  
 "tsym": "RELIGARE-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "553",  
 "tsym": "RELINFRA-EQ"  
 },  
 {  
 "exch": "NSE",  
 "token": "18074",  
 "tsym": "RELNV20NAV-EQ"  
 }  
 ]  
}

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| uid\* |  | Logged in User Id |
| stext\* |  | Search Text |
| exch |  | Exchange (Select from â€˜exarrâ€™ Array provided in User Details response) |

Response is of type SearchScripResponse:

SearchScripResponse has a list of a ScripItem,

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Market watch success or failure indication. |
| values |  | Array of json objects. (object fields given in below table) |
| emsg |  | This will be present only in case of errors. |

ScripItem is as follows

|  |  |  |
| --- | --- | --- |
| Json Fields of object in values Array | Possible value | Description |
| exch | NSE, BSE, NFO ... | Exchange |
| tsym |  | Trading symbol of the scrip (contract) |
| token |  | Token of the scrip (contract) |
| pp |  | Price precision |
| ti |  | Tick size |
| ls |  | Lot size |

## GetSecurityInfo

###### public bool SendGetSecurityInfo( OnResponse response,string exch,string token)

Example:

ret = nApi.SendGetSecurityInfo(OnSecurityResponse, 'NSE', '22')

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| uid\* |  | Logged in User Id |
| exch |  | Exchange |
| token |  | Contract Token |

Response is of type GetSecurityInfoResponse:

Response data will have below fields.

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| request\_time |  | It will be present only in a successful response. |
| stat | Ok or Not\_Ok | Market watch success or failure indication. |
| exch | NSE, BSE, NFO ... | Exchange |
| tsym |  | Trading Symbol |
| cname |  | Company Name |
| symnam |  | Symbol Name |
| seg |  | Segment |
| exd |  | Expiry Date |
| instname |  | Intrument Name |
| strprc |  | Strike Price |
| optt |  | Option Type |
| isin |  | ISIN |
| ti |  | Tick Size |
| ls |  | Lot Size |
| pp |  | Price precision |
| mult |  | Multiplier |
| gp\_nd |  | gn/gd \* pn/pd |
| prcunt |  | Price Units |
| prcqqty |  | Price Quote Qty |
| trdunt |  | Trade Units |
| delunt |  | Delivery Units |
| frzqty |  | Freeze Qty |
| gsmind |  | scripupdate Gsm Ind |
| elmbmrg |  | Elm Buy Margin |
| elmsmrg |  | Elm Sell Margin |
| addbmrg |  | Additional Long Margin |
| addsmrg |  | Additional Short Margin |
| splbmrg |  | Special Long Margin |
| splsmrg |  | Special Short Margin |
| delmrg |  | Delivery Margin |
| tenmrg |  | Tender Margin |
| tenstrd |  | Tender Start Date |
| tenendd |  | Tender End Eate |
| exestrd |  | Exercise Start Date |
| exeendd |  | Exercise End Date |
| elmmrg |  | Elm Margin |
| varmrg |  | Var Margin |
| expmrg |  | Exposure Margin |
| token |  | Contract Token |
| prcftr\_d |  | ((GN / GD) \* (PN/PD)) |

# Order and Trades

## PlaceOrder

###### public bool SendPlaceOrder( OnResponse response,PlaceOrder order )

Example:Place an order as follows

PlaceOrder order = new PlaceOrder();  
 order.uid = uid;  
 order.actid = actid;  
 order.exch = "NSE";  
 order.tsym = "M&M-EQ";  
 order.qty = "10";  
 order.dscqty = "0";  
 order.prc = "100.5";  
 order.prd = "I";  
 order.trantype = "B";  
 order.prctyp = "LMT";  
 order.ret = "DAY";  
 order.ordersource = "API";  
  
 nApi.SendPlaceOrder(Handlers.OnResponseNOP, order);

Place a Cover Order as follows

PlaceOrder order = new PlaceOrder();  
 order.uid = uid;  
 order.actid = actid;  
 order.exch = "CDS";  
 order.tsym = "USDINR27JAN21F";  
 order.qty = "10";  
 order.dscqty = "0";  
 order.prc = "76.0025";  
 order.blprc = "74.0025";  
 order.prd = "H";  
 order.trantype = "B";  
 order.prctyp = "LMT";  
 order.ret = "DAY";  
 order.ordersource = "API";  
  
 nApi.SendPlaceOrder(Handlers.OnResponseNOP, order);

Place a Bracket Order as follows

PlaceOrder order = new PlaceOrder();  
 order.uid = uid;  
 order.actid = actid;  
 order.exch = "NSE";  
 order.tsym = "INFY-EQ";  
 order.qty = "10";  
 order.dscqty = "0";  
 order.prc = "2800";  
 order.blprc = "2780";  
 order.bpprc = "2820";  
 order.prd = "B";  
 order.trantype = "B";  
 order.prctyp = "LMT";  
 order.ret = "DAY";  
 order.ordersource = "API";  
  
 nApi.SendPlaceOrder(Handlers.OnResponseNOP, order);

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| uid\* |  | Logged in User Id |
| actid\* |  | Login users account ID |
| exch\* | NSE / NFO / BSE / MCX | Exchange (Select from â€˜exarrâ€™ Array provided in User Details response) |
| tsym\* |  | Unique id of contract on which order to be placed. (use url encoding to avoid special char error for symbols like M&M) |
| qty\* |  | Order Quantity |
| prc\* |  | Order Price |
| trgprc |  | Only to be sent in case of SL / SL-M order. |
| dscqty |  | Disclosed quantity (Max 10% for NSE, and 50% for MCX) |
| prd\* | C / M / H | Product name (Select from â€˜prarrâ€™ Array provided in User Details response, and if same is allowed for selected, exchange. Show product display name, for user to select, and send corresponding prd in API call) |
| trantype\* | B / S | B -> BUY, S -> SELL |
| prctyp\* | LMT / MKT / SL-LMT / SL-MKT / DS / 2L / 3L |  |
| ret\* | DAY / EOS / IOC | Retention type (Show options as per allowed exchanges) |
| remarks |  | Any tag by user to mark order. |
| ordersource | MOB / WEB / TT | Used to generate exchange info fields. |
| bpprc |  | Book Profit Price applicable only if product is selected as B (Bracket order ) |
| blprc |  | Book loss Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| trailprc |  | Trailing Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| amo |  | Yes , If not sent, of Not â€œYesâ€, will be treated as Regular order. |
| tsym2 |  | Trading symbol of second leg, mandatory for price type 2L and 3L (use url encoding to avoid special char error for symbols like M&M) |
| trantype2 |  | Transaction type of second leg, mandatory for price type 2L and 3L |
| qty2 |  | Quantity for second leg, mandatory for price type 2L and 3L |
| prc2 |  | Price for second leg, mandatory for price type 2L and 3L |
| tsym3 |  | Trading symbol of third leg, mandatory for price type 3L (use url encoding to avoid special char error for symbols like M&M) |
| trantype3 |  | Transaction type of third leg, mandatory for price type 3L |
| qty3 |  | Quantity for third leg, mandatory for price type 3L |
| prc3 |  | Price for third leg, mandatory for price type 3L |

Response will be of type PlaceOrderResponse :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Place order success or failure indication. |
| request\_time |  | Response received time. |
| norenordno |  | It will be present only on successful Order placement to OMS. |
| emsg |  | This will be present only if Order placement fails |

## ModifyOrder

###### bool SendModifyOrder( OnResponse response,ModifyOrder order )

To Modify an order, use the OrderNumber returned in place order, you can only modify an open order(Status: New).

Example:

ModifyOrder order = new ModifyOrder();  
 order.norenordno = ordno;  
 order.exch = "NSE";  
 order.tsym = "M&M-EQ";  
 order.qty = "15";  
 order.prc = "100.5";  
 order.prd = "I";   
 order.prctyp = "LMT";  
 order.ret = "DAY";  
   
 nApi.SendModifyOrder(Handlers.OnResponseNOP, order);

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| exch\* |  | Exchange |
| norenordno\* |  | Noren order number, which needs to be modified |
| prctyp | LMT / MKT / SL-MKT / SL-LMT | This can be modified. |
| prc |  | Modified / New price |
| qty |  | Modified / New Quantity |
| tsym\* |  | Unque id of contract on which order was placed. Canâ€™t be modified, must be the same as that of original order. (use url encoding to avoid special char error for symbols like M&M) |
| ret | DAY / IOC / EOS | New Retention type of the order |
|  |  |  |
| trgprc |  | New trigger price in case of SL-MKT or SL-LMT |
| uid\* |  | User id of the logged in user. |
| bpprc |  | Book Profit Price applicable only if product is selected as B (Bracket order ) |
| blprc |  | Book loss Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| trailprc |  | Trailing Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |

Response will be of type ModifyOrderResponse :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Modify order success or failure indication. |
| result |  | Noren Order number of the order modified. |
| request\_time |  | Response received time. |
| emsg |  | This will be present only if Order modification fails |

## CancelOrder

###### public bool SendCancelOrder( OnResponse response,string norenordno)

To cancel an order, send the OrderNumber returned by PlaceOrder

Example:

nApi.SendCancelOrder(Handlers.OnResponseNOP, order);

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| norenordno\* |  | Noren order number, which needs to be modified |
| uid\* |  | User id of the logged in user. |

Response is of type CancelOrderResponse:

Response data will be in json format with below fields.

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Cancel order success or failure indication. |
| result |  | Noren Order number of the canceled order. |
| request\_time |  | Response received time. |
| emsg |  | This will be present only if Order cancelation fails |

## ExitSNOOrder

###### public bool SendExitSNOOrder( OnResponse response,string norenordno,string product)

Example:

public class ExitSNOOrder : NorenMessage  
 {  
 public string norenordno;  
 public string prd;  
 public string uid;  
 }

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| norenordno\* |  | Noren order number, which needs to be modified |
| prd\* | H / B | Allowed for only H and B products (Cover order and bracket order) |
| uid\* |  | User id of the logged in user. |

Response is of type ExitSNOOrderResponse :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Cancel order success or failure indication. |
| dmsg |  | Display message, (will be present only in case of success). |
| request\_time |  | Response received time. |
| emsg |  | This will be present only if Order cancelation fails |

## OrderMargin

###### public bool SendGetOrderMargin( OnResponse response,OrderMargin ordermargin)

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

Response is of type OrderMarginResponse:

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

## OrderBook

###### public bool SendGetOrderBook( OnResponse response,string product)

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| No Parameters |  |  |  |

Response is of type OrderBookResponse:

OrderBookResponse has a list of OrderBookItem

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Order book success or failure indication. |
| exch |  | Exchange Segment |
| tsym |  | Trading symbol / contract on which order is placed. |
| norenordno |  | Noren Order Number |
| prc |  | Order Price |
| qty |  | Order Quantity |
| prd |  | Display product alias name, using prarr returned in user details. |
| status |  |  |
| trantype | B / S | Transaction type of the order |
| prctyp | LMT / MKT | Price type |
| fillshares |  | Total Traded Quantity of this order |
| avgprc |  | Average trade price of total traded quantity |
| rejreason |  | If order is rejected, reason in text form |
| exchordid |  | Exchange Order Number |
| cancelqty |  | Canceled quantity for order which is in status cancelled. |
| remarks |  | Any message Entered during order entry. |
| dscqty |  | Order disclosed quantity. |
| trgprc |  | Order trigger price |
| ret | DAY / IOC / EOS | Order validity |
| uid |  |  |
| actid |  |  |
| bpprc |  | Book Profit Price applicable only if product is selected as B (Bracket order ) |
| blprc |  | Book loss Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| trailprc |  | Trailing Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| amo |  | Yes / No |
| pp |  | Price precision |
| ti |  | Tick size |
| ls |  | Lot size |
| token |  | Contract Token |
| norentm |  |  |
| ordenttm |  |  |
| exch\_tm |  |  |
| snoordt |  | 0 for profit leg and 1 for stoploss leg |
| snonum |  | This field will be present for product H and B; and only if it is profit/sl order. |

Response in case of failure:

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Not\_Ok | Order book failure indication. |
| request\_time |  | Response received time. |
| emsg |  | Error message |

## MultiLegOrderBook

###### public bool SendGetMultiLegOrderBook( OnResponse response,string product)

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| No Parameters |  |  |  |

Response is of type MultiLegOrderBookResponse:

MultiLegOrderBookResponse has a list of MultiLegOrderBookItem

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

## SingleOrderHistory

###### public bool SendGetOrderHistory( OnResponse response,string norenordno)

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| norenordno\* |  | Noren Order Number |  |

Response is of type OrderHistoryResponse:

OrderHistoryResponse has a list of SingleOrdHistItem

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Order book success or failure indication. |
| exch |  | Exchange Segment |
| tsym |  | Trading symbol / contract on which order is placed. |
| norenordno |  | Noren Order Number |
| prc |  | Order Price |
| qty |  | Order Quantity |
| prd |  | Display product alias name, using prarr returned in user details. |
| status |  |  |
| rpt |  | (fill/complete etc) |
| trantype | B / S | Transaction type of the order |
| prctyp | LMT / MKT | Price type |
| fillshares |  | Total Traded Quantity of this order |
| avgprc |  | Average trade price of total traded quantity |
| rejreason |  | If order is rejected, reason in text form |
| exchordid |  | Exchange Order Number |
| cancelqty |  | Canceled quantity for order which is in status cancelled. |
| remarks |  | Any message Entered during order entry. |
| dscqty |  | Order disclosed quantity. |
| trgprc |  | Order trigger price |
| ret | DAY / IOC / EOS | Order validity |
| uid |  |  |
| actid |  |  |
| bpprc |  | Book Profit Price applicable only if product is selected as B (Bracket order ) |
| blprc |  | Book loss Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| trailprc |  | Trailing Price applicable only if product is selected as H and B (High Leverage and Bracket order ) |
| amo |  | Yes / No |
| pp |  | Price precision |
| ti |  | Tick size |
| ls |  | Lot size |
| token |  | Contract Token |
| norentm |  |  |
| ordenttm |  |  |
| exch\_tm |  |  |

Response data in case of failure:

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Not\_Ok | Order book failure indication. |
| request\_time |  | Response received time. |
| emsg |  | Error message |

## TradeBook

###### public bool SendGetTradeBook( OnResponse response,string account)

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| uid\* |  | Logged in User Id |
| actid\* |  | Account Id of logged in user |

Response is of type TradeBookResponse:

TradeBookResponse has a list of TradeBookItem

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Order book success or failure indication. |
| exch |  | Exchange Segment |
| tsym |  | Trading symbol / contract on which order is placed. |
| norenordno |  | Noren Order Number |
| qty |  | Order Quantity |
| prd |  | Display product alias name, using prarr returned in user details. |
| trantype | B / S | Transaction type of the order |
| prctyp | LMT / MKT | Price type |
| fillshares |  | Total Traded Quantity of this order |
| avgprc |  | Average trade price of total traded quantity |
| exchordid |  | Exchange Order Number |
| remarks |  | Any message Entered during order entry. |
| ret | DAY / IOC / EOS | Order validity |
| uid |  |  |
| actid |  |  |
| pp |  | Price precision |
| ti |  | Tick size |
| ls |  | Lot size |
| cstFrm |  | Custom Firm |
| fltm |  | Fill Time |
| flid |  | Fill ID |
| flqty |  | Fill Qty |
| flprc |  | Fill Price |
| ordersource |  | Order Source |
| token |  | Token |

Response data will be in json format with below fields in case of failure:

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Not\_Ok | Order book failure indication. |
| request\_time |  | Response received time. |
| emsg |  | Error message |

## ExchMsg

###### public bool SendGetExchMsg( OnResponse response,ExchMsg exchmsg)

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| exch |  | Exchange Segment |  |

Response is of type ExchMsgResponse:

ExchMsgResponse has a list of ExchMsg

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

## PositionsBook

###### public bool SendGetPositionBook( OnResponse response,string account)

retrieves the overnight and day positions as a list

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

Response is of type PositionBookResponse:

PositionBookResponse - list of PositionBookItem which is as follows,

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| stat | string | False | Position book success or failure indication. |
| exch | string | False | Exchange segment |
| tsym | string | False | Trading symbol / contract. |
| token | string | False | Contract Token |
| uid | string | False | User Id |
| actid | string | False | Account Id |
| prd | string | False | Product name |
| netqty | string | False | Net Position Quantity |
| netavgprc | string | False | Net Position Average Price |
| daybuyqty | string | False | Day Buy Quantity |
| daysellqty | string | False | Day Sell Quantity |
| daybuyavgprc | string | False | Day Buy Average Price |
| daysellavgprc | string | False | Day Sell Average Price |
| daybuyamt | string | False | Day Buy Amount |
| daysellamt | string | False | Day Sell Amount |
| cfbuyqty | string | False | Carry Forward Sell Quantity |
| cforgavgprc | string | False | Original Average Price |
| cfsellqty | string | False | Carry Forward Sell Quantity |
| cfbuyavgprc | string | False | Carry Forward Buy Average Price |
| cfsellavgprc | string | False | Carry Forward Sell Average Price |
| cfbuyamt | string | False | Carry Forward Buy Amount |
| cfsellamt | string | False | Carry Forward Sell Amount |
| lp | string | False | LTP |
| rpnl | string | False | Realized Profit and Loss |
| urmtom | string | False | UnRealized Mark To Market (Can be recalculated in LTP update : = netqty \* (lp from web socket - netavgprc) \* prcftr |
| bep | string | False | Breakeven Price |
| openbuyqty | string | False | Open Buy Order Quantity |
| opensellqty | string | False | Open Sell Order Quantity |
| openbuyamt | string | False | Open Buy Order Amount |
| opensellamt | string | False | Open Sell Order Amount |
| openbuyavgprc | string | False |  |
| opensellavgprc | string | False |  |
| mult | string | False |  |
| pp | string | False |  |
| prcftr | string | False |  |
| ti | string | False |  |
| ls | string | False |  |
| request\_time | string | False |  |

Response data in case of failure:

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Not\_Ok | Position book request failure indication. |
| request\_time |  | Response received time. |
| emsg |  | Error message |

## ProductConversion

###### public bool SendProductConversion( OnResponse response,ProductConversion prdConv)

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| exch\* |  | Exchange |
| tsym\* |  | Unique id of contract on which order was placed. Canâ€™t be modified, must be the same as that of original order. (use url encoding to avoid special char error for symbols like M&M) |
| qty\* |  | Quantity to be converted. |
| uid\* |  | User id of the logged in user. |
| actid\* |  | Account id |
| prd\* |  | Product to which the user wants to convert position. |
| prevprd\* |  | Original product of the position. |
| trantype\* |  | Transaction type |
| postype\* | Day / CF | Converting Day or Carry forward position |
| ordersource | MOB | For Logging |

Response Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Position conversion success or failure indication. |
| emsg |  | This will be present only if Position conversion fails. |

# Holdings and Limits

## Holdings

###### public bool SendGetHoldings( OnResponse response,string account,string product)

retrieves the holdings as a list

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| uid\* |  | Logged in User Id |
| actid\* |  | Account id of the logged in user. |
| prd\* |  | Product name |

Response is of type HoldingsResponse :

HoldingsResponse has a list of HoldingsItem

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok or Not\_Ok | Holding request success or failure indication. |
| exch\_tsym |  | Array of objects exch\_tsym objects as defined below. |
| holdqty |  | Holding quantity |
| dpqty |  | DP Holding quantity |
| npoadqty |  | Non Poa display quantity |
| colqty |  | Collateral quantity |
| benqty |  | Beneficiary quantity |
| unplgdqty |  | Unpledged quantity |
| brkcolqty |  | Broker Collateral |
| btstqty |  | BTST quantity |
| btstcolqty |  | BTST Collateral quantity |
| usedqty |  | Holding used today |
| upldprc |  | Average price uploaded along with holdings |

Notes:  
Valuation : btstqty + holdqty + brkcolqty + unplgdqty + benqty + Max(npoadqty, dpqty) - usedqty  
Salable: btstqty + holdqty + unplgdqty + benqty + dpqty - usedqty

Exch\_tsym object:

|  |  |  |
| --- | --- | --- |
| Json Fields of object in values Array | Possible value | Description |
| exch | NSE, BSE, NFO ... | Exchange |
| tsym |  | Trading symbol of the scrip (contract) |
| token |  | Token of the scrip (contract) |
| pp |  | Price precision |
| ti |  | Tick size |
| ls |  | Lot size |

Response data will be in json format with below fields in case of failure:

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Not\_Ok | Position book request failure indication. |
| request\_time |  | Response received time. |
| emsg |  | Error message |

## Limits

###### public bool SendGetLimits(OnResponse response,string account,string product = "", string segment = ""; string exchange = "")

retrieves the margin and limits set

Request Details:

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| product\_type | string | True | retreives the delivery holdings or for a given product |
| segment | string | True | CM / FO / FX |
| exchange | string | True | Exchange NSE/BSE/MCX |

Response is of type LimitsResponse:

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| stat | Ok or Not\_Ok | False | Limits request success or failure indication. |
| actid | string | True | Account id |
| prd | string | True | Product name |
| seg | string | True | Segment CM / FO / FX |
| exch | string | True | Exchange |
| -------------------------Cash Primary Fields------------------------------- |  |  |  |
| cash | string | True | Cash Margin available |
| payin | string | True | Total Amount transferred using Payins today |
| payout | string | True | Total amount requested for withdrawal today |
| -------------------------Cash Additional Fields------------------------------- |  |  |  |
| brkcollamt | string | True | Prevalued Collateral Amount |
| unclearedcash | string | True | Uncleared Cash (Payin through cheques) |
| daycash | string | True | Additional leverage amount / Amount added to handle system errors - by broker. |
| -------------------------Margin Utilized---------------------------------- |  |  |  |
| marginused | string | True | Total margin / fund used today |
| mtomcurper | string | True | Mtom current percentage |
| -------------------------Margin Used components--------------------- |  |  |  |
| cbu | string | True | CAC Buy used |
| csc | string | True | CAC Sell Credits |
| rpnl | string | True | Current realized PNL |
| unmtom | string | True | Current unrealized mtom |
| marprt | string | True | Covered Product margins |
| span | string | True | Span used |
| expo | string | True | Exposure margin |
| premium | string | True | Premium used |
| varelm | string | True | Var Elm Margin |
| grexpo | string | True | Gross Exposure |
| greexpo\_d | string | True | Gross Exposure derivative |
| scripbskmar | string | True | Scrip basket margin |
| addscripbskmrg | string | True | Additional scrip basket margin |
| brokerage | string | True | Brokerage amount |
| collateral | string | True | Collateral calculated based on uploaded holdings |
| grcoll | string | True | Valuation of uploaded holding pre haircut |
| -------------------------Additional Risk Limits--------------------------- |  |  |  |
| turnoverlmt | string | True |  |
| pendordvallmt | string | True |  |
| -------------------------Additional Risk Indicators--------------------------- |  |  |  |
| turnover | string | True | Turnover |
| pendordval | string | True | Pending Order value |
| -------------------------Margin used detailed breakup fields------------------------- |  |  |  |
| rzpnl\_e\_i | string | True | Current realized PNL (Equity Intraday) |
| rzpnl\_e\_m | string | True | Current realized PNL (Equity Margin) |
| rzpnl\_e\_c | string | True | Current realized PNL (Equity Cash n Carry) |
| rzpnl\_d\_i | string | True | Current realized PNL (Derivative Intraday) |
| rzpnl\_d\_m | string | True | Current realized PNL (Derivative Margin) |
| rzpnl\_f\_i | string | True | Current realized PNL (FX Intraday) |
| rzpnl\_f\_m | string | True | Current realized PNL (FX Margin) |
| rzpnl\_c\_i | string | True | Current realized PNL (Commodity Intraday) |
| rzpnl\_c\_m | string | True | Current realized PNL (Commodity Margin) |
| uzpnl\_e\_i | string | True | Current unrealized MTOM (Equity Intraday) |
| uzpnl\_e\_m | string | True | Current unrealized MTOM (Equity Margin) |
| uzpnl\_e\_c | string | True | Current unrealized MTOM (Equity Cash n Carry) |
| uzpnl\_d\_i | string | True | Current unrealized MTOM (Derivative Intraday) |
| uzpnl\_d\_m | string | True | Current unrealized MTOM (Derivative Margin) |
| uzpnl\_f\_i | string | True | Current unrealized MTOM (FX Intraday) |
| uzpnl\_f\_m | string | True | Current unrealized MTOM (FX Margin) |
| uzpnl\_c\_i | string | True | Current unrealized MTOM (Commodity Intraday) |
| uzpnl\_c\_m | string | True | Current unrealized MTOM (Commodity Margin) |
| span\_d\_i | string | True | Span Margin (Derivative Intraday) |
| span\_d\_m | string | True | Span Margin (Derivative Margin) |
| span\_f\_i | string | True | Span Margin (FX Intraday) |
| span\_f\_m | string | True | Span Margin (FX Margin) |
| span\_c\_i | string | True | Span Margin (Commodity Intraday) |
| span\_c\_m | string | True | Span Margin (Commodity Margin) |
| expo\_d\_i | string | True | Exposure Margin (Derivative Intraday) |
| expo\_d\_m | string | True | Exposure Margin (Derivative Margin) |
| expo\_f\_i | string | True | Exposure Margin (FX Intraday) |
| expo\_f\_m | string | True | Exposure Margin (FX Margin) |
| expo\_c\_i | string | True | Exposure Margin (Commodity Intraday) |
| expo\_c\_m | string | True | Exposure Margin (Commodity Margin) |
| premium\_d\_i | string | True | Option premium (Derivative Intraday) |
| premium\_d\_m | string | True | Option premium (Derivative Margin) |
| premium\_f\_i | string | True | Option premium (FX Intraday) |
| premium\_f\_m | string | True | Option premium (FX Margin) |
| premium\_c\_i | string | True | Option premium (Commodity Intraday) |
| premium\_c\_m | string | True | Option premium (Commodity Margin) |
| varelm\_e\_i | string | True | Var Elm (Equity Intraday) |
| varelm\_e\_m | string | True | Var Elm (Equity Margin) |
| varelm\_e\_c | string | True | Var Elm (Equity Cash n Carry) |
| marprt\_e\_h | string | True | Covered Product margins (Equity High leverage) |
| marprt\_e\_b | string | True | Covered Product margins (Equity Bracket Order) |
| marprt\_d\_h | string | True | Covered Product margins (Derivative High leverage) |
| marprt\_d\_b | string | True | Covered Product margins (Derivative Bracket Order) |
| marprt\_f\_h | string | True | Covered Product margins (FX High leverage) |
| marprt\_f\_b | string | True | Covered Product margins (FX Bracket Order) |
| marprt\_c\_h | string | True | Covered Product margins (Commodity High leverage) |
| marprt\_c\_b | string | True | Covered Product margins (Commodity Bracket Order) |
| scripbskmar\_e\_i | string | True | Scrip basket margin (Equity Intraday) |
| scripbskmar\_e\_m | string | True | Scrip basket margin (Equity Margin) |
| scripbskmar\_e\_c | string | True | Scrip basket margin (Equity Cash n Carry) |
| addscripbskmrg\_d\_i | string | True | Additional scrip basket margin (Derivative Intraday) |
| addscripbskmrg\_d\_m | string | True | Additional scrip basket margin (Derivative Margin) |
| addscripbskmrg\_f\_i | string | True | Additional scrip basket margin (FX Intraday) |
| addscripbskmrg\_f\_m | string | True | Additional scrip basket margin (FX Margin) |
| addscripbskmrg\_c\_i | string | True | Additional scrip basket margin (Commodity Intraday) |
| addscripbskmrg\_c\_m | string | True | Additional scrip basket margin (Commodity Margin) |
| brkage\_e\_i | string | True | Brokerage (Equity Intraday) |
| brkage\_e\_m | string | True | Brokerage (Equity Margin) |
| brkage\_e\_c | string | True | Brokerage (Equity CAC) |
| brkage\_e\_h | string | True | Brokerage (Equity High Leverage) |
| brkage\_e\_b | string | True | Brokerage (Equity Bracket Order) |
| brkage\_d\_i | string | True | Brokerage (Derivative Intraday) |
| brkage\_d\_m | string | True | Brokerage (Derivative Margin) |
| brkage\_d\_h | string | True | Brokerage (Derivative High Leverage) |
| brkage\_d\_b | string | True | Brokerage (Derivative Bracket Order) |
| brkage\_f\_i | string | True | Brokerage (FX Intraday) |
| brkage\_f\_m | string | True | Brokerage (FX Margin) |
| brkage\_f\_h | string | True | Brokerage (FX High Leverage) |
| brkage\_f\_b | string | True | Brokerage (FX Bracket Order) |
| brkage\_c\_i | string | True | Brokerage (Commodity Intraday) |
| brkage\_c\_m | string | True | Brokerage (Commodity Margin) |
| brkage\_c\_h | string | True | Brokerage (Commodity High Leverage) |
| brkage\_c\_b | string | True | Brokerage (Commodity Bracket Order) |
| peak\_mar | string | True | Peak margin used by the client |
| request\_time | string | True | This will be present only in a successful response. |
| emsg | string | True | This will be present only in a failure response. |

Sample Success Response :  
{  
"request\_time":"18:07:31 29-05-2020",  
"stat":"Ok",  
"cash":"1500000000000000.00",  
"payin":"0.00",  
"payout":"0.00",  
"brkcollamt":"0.00",  
"unclearedcash":"0.00",  
"daycash":"0.00",  
"turnoverlmt":"50000000000000.00",  
"pendordvallmt":"2000000000000000.00",  
"turnover":"3915000.00",  
"pendordval":"2871000.00",  
"marginused":"3945540.00",  
"mtomcurper":"0.00",  
"urmtom":"30540.00",  
"grexpo":"3915000.00",  
"uzpnl\_e\_i":"15270.00",  
"uzpnl\_e\_m":"61080.00",  
"uzpnl\_e\_c":"-45810.00"  
}

Sample Failure Response :  
{  
"stat":"Not\_Ok",  
"emsg":"Server Timeout : "  
}

# MarketInfo

## GetIndexList

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| exch |  | Exchange Segment |  |

Response is of type IndexListResponse:

IndexListResponse has a list of Index

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

## GetTopListNames

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| exch |  | Exchange Segment |  |

Response is of type TopListNamesResponse:

TopListNamesResponse has a list of Index

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

## GetTopList

Request Details :

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| exch |  | Exchange Segment |  |

Response is of type TopListResponse:

TopListResponse has a list of Index

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |

## GetTimePriceData /ChartData

###### public bool SendGetTPSeries(OnResponse response, string exch, string token, string starttime = null, string endtime = null, string interval = null)

Request Details :

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| uid\* |  | Logged in User Id |
| exch\* |  | Exchange |
| token\* |  |  |
| st |  | Start time (seconds since 1 jan 1970) |
| et |  | End Time (seconds since 1 jan 1970) |
| intrv | â€œ1â€, â€3â€, â€œ5â€, â€œ10â€, â€œ15â€, â€œ30â€, â€œ60â€, â€œ120â€, â€œ240â€ | Candle size in minutes (optional field, if not given assume to be â€œ1â€) |

Response Details :

Response data will be in json format in case for failure.

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Not\_Ok | TPData failure indication. |
| emsg |  | This will be present only in case of errors. |

Response data will be in json format in case for success.

|  |  |  |
| --- | --- | --- |
| Json Fields | Possible value | Description |
| stat | Ok | TPData success indication. |
| time |  | DD/MM/CCYY hh:mm:ss |
| into |  | Interval open |
| inth |  | Interval high |
| intl |  | Interval low |
| intc |  | Interval close |
| intvwap |  | Interval vwap |
| intv |  | Interval volume |
| v |  | volume |
| intoi |  | Interval io change |
| oi |  | oi |

## GetOptionChain

gets the contracts of related strikes

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| exchange | string | False | Exchange (UI need to check if exchange in NFO / CDS / MCX / or any other exchange which has options, if not don't allow) |
| tradingsymbol | string | False | Trading symbol of any of the option or future. Option chain for that underlying will be returned. (use url encoding to avoid special char error for symbols like M&M) |
| strikeprice | float | False | Mid price for option chain selection |
| count | int | True | Number of strike to return on one side of the mid price for PUT and CALL. (example cnt is 4, total 16 contracts will be returned, if cnt is is 5 total 20 contract will be returned) |

the response is as follows,

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| stat | string | True | ok or Not\_ok |
| values | string | True | properties of the scrip |
| emsg | string | False | Error Message |

|  |  |  |  |
| --- | --- | --- | --- |
| Param | Type | Optional | Description |
| exch | string | False | Exchange |
| tsym | string | False | Trading Symbol of Contract |
| token | string | False | Contract token |
| optt | string | False | Option type |
| strprc | string | False | Strike Price |
| pp | string | False | Price Precision |
| ti | string | False | Tick Size |
| ls | string | False | Lot Size |

# OrderUpdates and MarketDataUpdate

This Api allows you to receive updates receivethe marketdata and order updates in the application callbacks as an option, to do so connect as follows.

Api.OnFeedCallback += Application.OnFeedHandler;  
Api.OnOrderCallback += Application.OnOrderHandler;

## Connect

###### public bool ConnectWatcher(string uri,OnFeed marketdata Handler,OnOrderFeed orderHandler)

starts the websocket, WebSocket feed has 2 types of ticks( t=touchline d=depth)and 2 stages (k=acknowledgement, f=further change in tick).

## SubscribeMarketData

###### public bool SubscribeToken(string exch,string token)

t='tk' is sent once on subscription for each instrument. this will have all the fields with the most recent value  
thereon t='tf' is sent for fields that have changed.

For example  
quote event: 03-12-2021 11:54:44{'t': 'tk', 'e': 'NSE', 'tk': '11630', 'ts': 'NTPC-EQ', 'pp': '2', 'ls': '1', 'ti': '0.05', 'lp': '118.55', 'h': '118.65', 'l': '118.10', 'ap': '118.39', 'v': '162220', 'bp1': '118.45', 'sp1': '118.50', 'bq1': '26', 'sq1': '6325'}  
quote event: 03-12-2021 11:54:45{'t': 'tf', 'e': 'NSE', 'tk': '11630', 'lp': '118.45', 'ap': '118.40', 'v': '166637', 'sp1': '118.55', 'bq1': '3135', 'sq1': '30'}  
quote event: 03-12-2021 11:54:46{'t': 'tf', 'e': 'NSE', 'tk': '11630', 'lp': '118.60'}

in the example above we see first message t='tk' with all the values, 2nd message has lasttradeprice avg price and few other fields with value changed.. note bp1 isnt sent as its still 118.45  
in the next tick ( 3rd message) only last price is changed to 118.6

##### Request:

|  |  |  |
| --- | --- | --- |
| Fields | Possible value | Description |
| exch | NSE,BSE,NFO... | Exchange |
| token |  |  |
| ScripToken |  |  |

##### MarketDataUpdates:

Accept for t, e,and tk other fields may/may not be present.

|  |  |  |
| --- | --- | --- |
| Fields | Possible value | Description |
| t | tf | tf represents touchline feed |
| e | NSE,BSE,NFO.. | Exchangename |
| tk | 22 | ScripToken |
| lp |  | LTP |
| pc |  | Percentagechange |
| v |  | volume |
| o |  | Openprice |
| h |  | Highprice |
| l |  | Lowprice |
| c |  | Closeprice |
| ap |  | Averagetradeprice |
| oi |  | Open interest |
| poi |  | Previous day closing Open Interest |
| to1 |  | Total open interest for underlying |
| bq1 |  | Best Buy Quantity 1 |
| bp1 |  | Best Buy Price 1 |
| sq1 |  | Best Sell Quantity 1 |
| sp1 |  | Best Sell Price 1 |

## UnSubscribeMarketData

###### public bool UnSubscribeToken(string exch,string token)

##### Request:

|  |  |  |
| --- | --- | --- |
| Fields | Possible value | Description |
| exch | NSE,BSE,NFO... | Exchange |
| token |  | ScripToken |

## SubscribeOrderUpdate

This is auto subscribed by the api

##### Order Updates : NorenOrderFeed

|  |  |  |
| --- | --- | --- |
| Fields | Possible value | Description |
| t | om | "om" represents touchlinefeed |
| norenordno |  | NorenOrderNumber |
| uid |  | UserId |
| actid |  | AccountID |
| exch |  | Exchange |
| tsym |  |  |
| qty |  | Orderquantity |
| prc |  | OrderPrice |
| prd |  | Product |
| status |  | Orderstatus(open,complete,rejectedetc) |
| reporttype |  | Ordereventforwhichthismessageissentout.(fill,rejected,canceled) |
| trantype |  | Ordertransactiontype,buyorsell |
| prctyp |  | Orderpricetype(LMT,MKT,SL-LMT,SL-MKT) |
| ret |  | Orderretentiontype(DAY,EOS,IOC,...) |
| fillshares |  | Filledshares |
| avgprc |  | Averagefillprice |
| rejreason |  | Orderrejectionreason,ifrejected |
| exchordid |  | ExchangeOrderID |
| cancelqty |  | Canceledquantity,incaseofcanceledorder |
| remarks |  | Useraddedtag,whileplacingorder |
| dscqty |  | Disclosedquantity |
| trgprc |  | TriggerpriceforSLorders |