.NET

Dot Net API Document

Product of

Finvasia

Shoonya
Dot Net API

TABLE OF CONTENTS

Version History

Date	Version	Changes	Details
19-04-2021	1.0.0.1	Touchline Broker	Touchline Feed added. Document Format updated
	1.0.0.0	Initial Release	Based on Noren RestAPI v1.10.0

INTRODUCTION

Finvasia Shoonya API provides clients with cutting edge trading technology and a full suite of APIs to enhance their trading capabilities.

About the API

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

API is developed on Visual Studio 2019 and uses .Net Standard 2.0

The dependency libraries are
Newtonsoft.Json 9.0.1
Websocket.Client 4.3.21

API Interface

The namespace ShoonyaRestApiWrapper and class ShoonyaRestApi are of primary use and interest

Initialize

To initialize the api the following are needed
endPoint : The ShoonyaOms address
Appkey : The secret key issued to you

Making Requests

We will be creating an object of ShoonyaRestApi to make requests the callback is taken as an argument in the request method.

```
LoginMessage loginMessage = new LoginMessage();
loginMessage.apkversion = "1.0.0";
```

```
loginMessage.uid = uid;
loginMessage.pwd = pwd;
loginMessage.factor2 = OTP/TOTP;
loginMessage.imei = "134243434";
loginMessage.source = "API";
loginMessage.appkey = appkey;

nApi.SendLogin(Program.OnAppLoginResponse, endPoint, loginMessage);
```

In the above example we are sending the Login request, this takes 3 arguments

a. Callback : this is the function where the application will be handling the response
b. Endpoint: ShoonyaOMS address
c. MessageData: parameters of the request being made.

The Callback is of signature

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

A Typical 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, here LoginResponse
Stat is checked to see if the request was successful.

Login and User Details

Login

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

Request Details : LoginMessage

Fields	Possible value	Description
endPoint		The Server ip and port

Fields	Possible value	Description
apkversion*		Application version.
uid*		User Id of the login user
pwd*		Sha256 of the user entered password.
factor2*		OTP or TOTP as entered by the user.
vc*		Vendor code provided by Finvasia team, along with connection URLs
appkey*		Sha256 of uid|vendor_key
imei*		Send mac if users logs in for desktop, imei is from mobile
ip_address		Optional field

source	WEB / MOB	

Example:
{ \"apkversion\": \"1.0.0\", \"uid\": \"VIDYA\", \"pwd\": \"s3cur3Id\", \"factor2\": \"31-08-2017\", \"imei\": \"134243434\", \"source\": \"MOB\"}“

Response Details : LoginResponse

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.
emsg		This will be present only if Login fails.

Sample Success Response :

```
{
   "request_time": "20:18:47 19-05-2020",
   "stat": "Ok",
   "susertoken": "3b97f4c67762259a9ded6dbd7bfafe2787e662b3870422ddd343a59895f423a0",
   "lastaccesstime": "1589899727"
}
```

Sample Failure Response :

```
{
   "request_time": "20:32:14 19-05-2020",
   "stat": "Not_Ok",
   "emsg": "Invalid Input : Wrong Password"
}
```

Logout

public bool SendLogout(OnResponse response)

Request Details : No Params

Fields	**Possible value**	**Description**
-	-	-

Response Details : LogoutResponse

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.

Sample Success Response :

{
  "stat":"Ok",
  "request_time":"10:43:41 28-05-2020"
}

Sample Failure Response :

{
  "stat":"Not_Ok",
  "emsg":"Server Timeout : "
}

Forgot Password

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

Request Details :

Fields	**Possible value**	**Description**
endpoint		WebApi endpoint

user*		User Id
pan*		Pan of the user
dob*		Date of birth

Response Details : ForgotPasswordResponse.

Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Password reset is Success Or failure status
request_time		Response received time.
emsg		This will be present only if password reset fails. (“Invalid User or User Details”)

Sample Success Response :

{
  "request_time":"10:52:56 28-05-2020",
  "stat":"Ok"
}

Sample Failure Response :

{
  "request_time":"17:42:13 26-05-2020",
  "stat":"Not_Ok",
  "emsg":"Error Occurred : Wrong user id or user details"
}

Change Password

public bool Changepwd(OnResponse response, Changepwd changepwd)

Request Details : Changepwd

Fields	**Possible value**	**Description**
uid*		User Id
oldpwd*		Sha256 of old password
pwd*		New password in plain text

Response Details :ChangepwdResponse

Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Password reset is Success Or failure status
request_time		Response received time.
dmsg		This will be present only in case of success. Number of days to expiry will be present in same.
emsg		This will be present only if password change fails

Sample Success Response :

{
"request_time":"10:20:04 27-05-2020",
"stat":"Ok",
“dmag”:”Password Change Success. Your new password will expire in 15”
}

Sample Failure Response :

{
"request_time":"10:21:09 27-05-2020",
"stat":"Not_Ok",
"emsg":"Error Occurred : Password already used"
}

User Details

public bool SendGetUserDetails(OnResponse response)

Request Details : No Params

Response Details :UserDetailsResponse

Fields	**Possible value**	**Description**

stat	Ok or Not_Ok	User details success or failure indication.
exarr		array of strings with enabled exchange names
orarr		array of strings with enabled price types for user
prarr		array of Product Obj with enabled products, as defined below.
brkname		Broker id
brnchid		Branch id
email		
actid		
uprev		Always it will be an INVESTOR, other types of user not allowed to login using this API.
request_time		It will be present only in a successful response.
emsg		This will be present only in case of errors.

Product Obj format

Fields	Possible value	Description
prd		Product name
s_prdt_ali		Product display name
exch		array of strings with enabled, allowed exchange names

Sample Success Response:


```
{
  "request_time": "20:20:04 19-05-2020",
  "prarr": [
            { “prd”:"C",
              “s_prdt_ali” : “Delivery”,
                 “exch” :  [“NSE”, “BSE”]
            },
           { “prd”:"I",
              “s_prdt_ali” : “Intraday”,
                 “exch” : [“NSE”, “BSE”, “NFO”]
```

},
, { “prd”:"H",
“s_prdt_ali” : “High Leverage”,
“exch” : [“NSE”, “BSE”, “NFO”]
},
{ “prd”:"B",
“s_prdt_ali” : “Bracket Order”,
“exch” : [“NSE”, “BSE”, “NFO”]
}
],
"exarr": [
"NSE",
"NFO"
],
"orarr": [
"MKT",
"LMT",
"SL-LMT",
"SL-MKT",
"DS",
"2L",
"3L",
"4L"
],
"brkname": "VIDYA",
"brnchid": "VIDDU",
"email": "gururaj@gmail.com",
"actid": "GURURAJ",
"uprev": "INVESTOR",
"stat": "Ok"
}

Sample Failure Response:

{
"stat": "Not_Ok",
"emsg": "Session Expired : Invalid Session Key"
}

Watch Lists

Get WatchList Names

public bool SendGetMWList(OnResponse response)

Request Details : No Params
Response Details : MWListResponse

Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	MWList success or failure indication.
values		Watch List names as a array of strings.
request_time		It will be present only in a successful response.
emsg		This will be present only in case of errors or No WatchLists are set yet.

Sample Success Response :


```
{
  "request_time": "12:34:52 21-05-2020",
  "values": [
    "default",
    "WL"
  ],
  "stat": "Ok"
}
```


Sample Failure Response :


```
{
  "stat": "Not_Ok",
  "emsg": "Session Expired : Invalid Session Key"
}
```

Get WatchList

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

Request Details :

Fields	Possible value	Description
wlname*		Name of the Watchlist, for which scrip list is required.

Response Details : MarketWatchResponse

Fields	Possible value	Description
stat	Ok or Not_Ok	Market watch success or failure indication.
values		Array of objects. (object fields given in below table)
request_time		It will be present only in a successful response.
emsg		This will be present only in case of errors. That is : 1) Invalid Input : Invalid WatchList Name 2) Session Expired

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

Sample Success Response :


```
{
   "request_time": "13:25:17 21-05-2020",
   "values": [
      {
         "exch": "BSE",
         "token": "972889",
         "tsym": "915PTCIF27"
      },
      {
         "exch": "NSE",
         "token": "13",
         "tsym": "ABB-EQ"
      },
      {
         "exch": "NSE",
         "token": "22",
         "tsym": "ACC-EQ"
      }
   ],
   "stat": "Ok"
}
```


Sample Failure Response :


```
{
  "stat":"Not_Ok",
  "emsg":"Invalid Input : Missing uid or wlname."
}
```


Search Scrips

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

Request Details :

Fields	**Possible value**	**Description**
exch		Exchange (Select from 'exarr' Array provided in User Details response)
stext*		Search Text

Response Details :SearchScripResponse

Fields	Possible value	Description
stat	Ok or Not_Ok	Market watch success or failure indication.
values		Array of ScripItem objects. (object fields given in below table)
emsg		This will be present only in case of errors. That is : 1) Invalid Input 2) Session Expired

ScripItem

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

Sample Success Response :


```
{
  "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"
    }
  ]
}
```


Sample Failure Response :


```
{
  "stat":"Not_Ok",
  "emsg":"No Data : "
}
```


Add Scrip to Watch List

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

Request Details :

Fields	**Possible value**	**Description**
watchlist*		Name of the Watchlist, for which scrip list is required.
scrips*		List of scrips, example format NSE|22#BSE|506734

Response Details :StandardResponse

Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Watch list update success or failure indication.
request_time		It will be present only in a successful response.
emsg		This will be present only in case of errors.

		That is : 1) Invalid Input 2) Session Expired

Sample Success Response :

{
  "request_time": "13:50:40 21-05-2020",
  "stat": "Ok"
}

Sample Failure Response :

{
  "stat":"Not_Ok",
  "emsg":"Session Expired : Invalid Session Key"
}

Delete Scrip to Watch List

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

Request Details :

Fields	Possible value	Description
watchlist*		Name of the Watchlist, for which scrip list is required.
scrips*		List of scrips, example format NSE|22#BSE|506734

Response Details :StandardResponse

Fields	Possible value	Description
stat	Ok or Not_Ok	Watch list update success or failure indication.
request_time		It will be present only in a successful response.
emsg		This will be present only in case of errors. That is : 1) Invalid Input 2) Session Expired

Sample Success Response :

{
  "request_time": "13:50:40 21-05-2020",
  "stat": "Ok"
}

Sample Failure Response :

{
  "stat":"Not_Ok",
  "emsg":"Invalid Input : Missing uid or wlname or scrips."
}

Get SecurityInfo

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

Request Details :

Fields	Possible value	Description
exch		Exchange
token		Contract Token

Response Details :GetSecurityInfoResponse

Response data will have below fields.

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))

Sample Success Response :

{
"request_time": "17:43:38 31-10-2020",
"stat": "Ok",
"exch": "NSE",
"tsym": "ACC-EQ",
"cname": "ACC LIMITED",
"symname": "ACC",
"seg": "EQT",
"instname": "EQ",
"isin": "INE012A01025",
"pp": "2",
"ls": "1",
"ti": "0.05",
"mult": "1",
"prcftr_d": "(1 / 1) * (1 / 1)",
"trdunt": "ACC.BO",
"delunt": "ACC",
"token": "22",
"varmrg": "40.00"
}

Sample Failure Response :

{

"stat":"Not_Ok",
"request_time":"10:50:54 10-12-2020",
"emsg":"Error Occurred : 5 \"no data\""
}

Order and Trades

Place Order

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

Request Details :PlaceOrder

Fields	Possible value	Description
uid*		Logged in User Id
actid*		Login users account ID
exch*	NSE / NFO / BSE / MCX	Exchange (Select from 'exarr' 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 'prarr' 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 “Yes”, 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

Example:
{\"uid\":\"VIDYA\", \"actid\":\"CLIENT1\", \"exch\":\"NSE\", \"tsym\":\"ACC-EQ\", \"qty\":\"50\", \"price\":\"1400\", \"prd\":\"H\", \"trantype\":\"B\", \"prctyp\":\"LMT\", \"ret\":\"DAY\"}“ \

Response Details :PlaceOrderResponse

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

Sample Success Response:

{
  "request_time": "10:48:03 20-05-2020",
  "stat": "Ok",
  "norenordno": "20052000000017"
}

Sample Error Response :

{
  "stat": "Not_Ok",
  "request_time": "20:40:01 19-05-2020",
  "emsg": "Error Occurred : 2 \"invalid input\""
}

Modify Order

```
public bool SendModifyOrder(OnResponse response, ModifyOrder order)
```

Request Details :ModifyOrder

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. Can'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 Details :ModifyOrderResponse

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

Sample Success Response :


```
{
    "request_time":"14:14:08 26-05-2020",
    "stat":"Ok",
    "result":"20052600000103"
}
```

Sample Failure Response :

{
  "request_time":"16:03:29 28-05-2020",
  "stat":"Not_Ok",
  "emsg":"Rejected : ORA:Order not found"
}

Cancel Order

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

Request Details :

Fields	Possible value	Description
norenordno*		Noren order number, which needs to be modified

Response Details :CancelOrderResponse

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

Sample Success Response :

{
  "request_time":"14:14:10 26-05-2020",
  "stat":"Ok",
  "result":"20052600000103"
}

Sample Failure Response :

{
  "request_time":"16:01:48 28-05-2020",
  "stat":"Not_Ok",

"emsg":"Rejected : ORA:Order not found to Cancel"
}

Exit SNO Order

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

Request Details :

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)

Response Details :ExitSNOOrderResponse

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

Order Margin

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

Request Details :OrderMargin

Fields	Possible value	Description
uid*		Logged in User Id
actid*		Login users account ID

exch*	NSE / NFO / BSE / MCX	Exchange (Select from 'exarr' 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.
prd*	C / M / H	Product name (Select from 'prarr' 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	
blprc		Book loss Price applicable only if product is selected as H and B (High Leverage and Bracket order)
rorgqty		Optional field. Application only for modify order, open order quantity
fillshares		Optional field. Application only for modify order, quantity already filled.
rorgprc		Optional field. Application only for modify order, open order price
orgtrgprc		Optional field. Application only for modify order, open order trigger price
norenordno		Optional field. Application only for H or B order modification
snonum		Optional field. Application only for H or B order modification

Response Details :OrderMarginResponse

Fields	Possible	Description

	value	
stat	Ok or Not_Ok	Place order success or failure indication.
request_time		Response received time.
remarks		This field will be available only on success.
cash		Total credits available for order
marginused		Total margin used.
emsg		This will be present only if Order placement fails

Order Book

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

Request Details :

Fields	Possible value	Description
prd	H / M / ...	Product name

Response Details : OrderBookResponse list of OrderBookItem

Response data will be in Array of objects with below fields in case of success.

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		Order 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
orddttm		
ordenttm		
extm		

Response data will be as below fields in case of failure:

Fields	**Possible value**	**Description**
stat	Not_Ok	Order book failure indication.
request_time		Response received time.
emsg		Error message

Sample Success Output :

Success response :

[
{
“stat” : “Ok”,
“exch” : “NSE” ,
“tsym” : “ACC-EQ” ,
“norenordno” : “20062500000001223”,
“prc” : “127230”,
“qty” : “100”,
“prd” : “C”,
“status”: “Open”,
“trantype” : “B”,
“prctyp” : ”LMT”,
“fillshares” : “0”,
“avgprc” : “0”,
“exchordid” : “250620000000343421”,
“uid” : “VIDYA”,
“actid” : “CLIENT1”,
“ret” : “DAY”,
“amo” : “Yes”
},
{
“stat” : “Ok”,
“exch” : “NSE” ,
“tsym” : “ABB-EQ” ,

"norenordno" : "20062500000002543",
"prc" : "127830",
"qty" : "50",
"prd" : "C",
"status": "REJECT",
"trantype" : "B",
"prctyp" : "LMT",
"fillshares" : "0",
"avgprc" : "0",
"rejreason" : "Insufficient funds"
"uid" : "VIDYA",
"actid" : "CLIENT1",
"ret" : "DAY",
"amo" : "No"
}
]

Sample Failure Response :

{
"stat":"Not_Ok",
"emsg":"Session Expired : Invalid Session Key"
}

Multi Leg Order Book

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

Request Details :

Fields	Possible value	Description
prd	H / M / ...	Product name

Response Details :MultiLegOrderBookResponse list of MultiLegOrderBookItem

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		Order 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
tsym2		Trading symbol of second leg, mandatory for price type 2L and 3L
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
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
fillshares2		Total Traded Quantity of 2nd Leg
avgprc2		Average trade price of total traded quantity for 2nd leg
fillshares3		Total Traded Quantity of 3rd Leg
avgprc3		Average trade price of total traded quantity for 3rd leg

Response data will be as below fields in case of failure:

Fields	**Possible value**	**Description**
stat	Not_Ok	Order book failure indication.
request_time		Response received time.
emsg		Error message

Single Order History

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

Request Details :

Fields	Possible value	Description
norenordno*		Noren Order Number

Response Details :OrderHistoryResponse list of SingleOrdHistItem

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		Order status
rpt		Report Type (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
orddttm		
ordenttm		
extm		

Response data will be as below fields in case of failure:

Fields	**Possible value**	**Description**
stat	Not_Ok	Order book failure indication.

request_time		Response received time.
emsg		Error message

Sample Success Output :


```
[
  {
      "stat": "Ok",
      "norenordno": "20121300065716",
      "uid": "DEMO1",
      "actid": "DEMO1",
      "exch": "NSE",
      "tsym": "ACCELYA-EQ",
      "qty": "180",
      "trantype": "B",
      "prctyp": "LMT",
      "ret": "DAY",
      "token": "7053",
      "pp": "2",
      "ls": "1",
      "ti": "0.05",
      "prc": "800.00",
      "avgprc": "800.00",
      "discqty": "0",
      "prd": "M",
      "status": "COMPLETE",
      "rpt": "Fill",
      "fillshares": "180",
      "norentm": "19:59:32 13-12-2020",
      "exch_tm": "00:00:00 01-01-1980",
      "remarks": "WC TEST Order",
      "exchordid": "6858"
  },
  {
      "stat": "Ok",
      "norenordno": "20121300065716",
```

```
    "uid": "DEMO1",
    "actid": "DEMO1",
    "exch": "NSE",
    "tsym": "ACCELYA-EQ",
    "qty": "180",
    "trantype": "B",
    "prctyp": "LMT",
    "ret": "DAY",
    "token": "7053",
    "pp": "2",
    "ls": "1",
    "ti": "0.05",
    "prc": "800.00",
    "discqty": "0",
    "prd": "M",
    "status": "OPEN",
    "rpt": "New",
    "norentm": "19:59:32 13-12-2020",
    "exch_tm": "00:00:00 01-01-1980",
    "remarks": "WC TEST Order",
    "exchordid": "6858"
},
{
    "stat": "Ok",
    "norenordno": "20121300065716",
    "uid": "DEMO1",
    "actid": "DEMO1",
    "exch": "NSE",
    "tsym": "ACCELYA-EQ",
    "qty": "180",
    "trantype": "B",
    "prctyp": "LMT",
    "ret": "DAY",
    "token": "7053",
    "pp": "2",
    "ls": "1",
    "ti": "0.05",
```

```
        "prc": "800.00",
        "discqty": "0",
        "prd": "M",
        "status": "PENDING",
        "rpt": "PendingNew",
        "norentm": "19:59:32 13-12-2020",
        "remarks": "WC TEST Order"
    },
    {
        "stat": "Ok",
        "norenordno": "20121300065716",
        "uid": "DEMO1",
        "actid": "DEMO1",
        "exch": "NSE",
        "tsym": "ACCELYA-EQ",
        "qty": "180",
        "trantype": "B",
        "prctyp": "LMT",
        "ret": "DAY",
        "token": "7053",
        "pp": "2",
        "ls": "1",
        "ti": "0.05",
        "prc": "800.00",
        "prd": "M",
        "status": "PENDING",
        "rpt": "NewAck",
        "norentm": "19:59:32 13-12-2020",
        "remarks": "WC TEST Order"
    }
]
```


Trade Book

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

Request Details :

Fields	Possible value	Description
actid*		Account Id of logged in user

Response Details :TradeBookResponse list of TradeBookItem

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 as below fields in case of failure:

Fields	Possible value	Description
stat	Not_Ok	Order book failure indication.
request_time		Response received time.
emsg		Error message

Sample Success Output :


```
[
   {
       "stat": "Ok",
       "norenordno": "20121300065715",
       "uid": "GURURAJ",
       "actid": "GURURAJ",
       "exch": "NSE",
       "prctyp": "LMT",
       "ret": "DAY",
       "prd": "M",
       "flid": "102",
       "fltm": "01-01-1980 00:00:00",
       "trantype": "S",
```

"tsym": "ACCELYA-EQ",
"qty": "180",
"token": "7053",
"fillshares": "180",
"flqty": "180",
"pp": "2",
"ls": "1",
"ti": "0.05",
"prc": "800.00",
"flprc": "800.00",
"norentm": "19:59:32 13-12-2020",
"exch_tm": "00:00:00 01-01-1980",
"remarks": "WC TEST Order",
"exchordid": "6857"
},
{
"stat": "Ok",
"norenordno": "20121300065716",
"uid": "GURURAJ",
"actid": "GURURAJ",
"exch": "NSE",
"prctyp": "LMT",
"ret": "DAY",
"prd": "M",
"flid": "101",
"fltm": "01-01-1980 00:00:00",
"trantype": "B",
"tsym": "ACCELYA-EQ",
"qty": "180",
"token": "7053",
"fillshares": "180",
"flqty": "180",
"pp": "2",
"ls": "1",
"ti": "0.05",
"prc": "800.00",
"flprc": "800.00",

```
        "norentm": "19:59:32 13-12-2020",
        "exch_tm": "00:00:00 01-01-1980",
        "remarks": "WC TEST Order",
        "exchordid": "6858"
    }
]
```

Exch Msg

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

Request Details :ExchMsg

Fields	Possible value	Description
uid*		Logged in User Id
exch		Exchange (Select from 'exarr' Array provided in User Details response)

Response Details :ExchMsgResponse list of ExchMsgItem

Response data will be as below fields in case of success.

Fields	Possible value	Description
stat	Ok	Whi Exch Msg success or failure indication.
exchmsg		It will be present only in a successful response.
exchtm		Exchange Time

Response data will be as below fields in case of failure:

Fields	Possible	Description

	value	
stat	Not_Ok	Order book failure indication.
request_time		Response received time.
emsg		Error message

Order Margin

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

Request Details :OrderMargin

Fields	Possible value	Description
actid*		Login users account ID
exch*	NSE / NFO / BSE / MCX	Exchange (Select from 'exarr' Array provided in User Details response)
qty*		Order Quantity
prc*		Order Price
trgprc		Only to be sent in case of SL / SL-M order.
prd*	C / M / H	Product name (Select from 'prarr' 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	
orgqty		Org Quantity

orgprc		Org Price
token		Unique id of contract on which order to be placed.
flqty		Fill Quantity
srcuid		Source User ID
srcbkrid		Source Broker ID

Response Details :OrderMarginResponse

Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Place order success or failure indication.
request_time		Response received time.
remarks	Insufficient Balance/ Order Success/ Invalid scrip, RED is under Reconciliation/ Squareoff Order	Any message Entered during order entry.
cash	optional	Total Cash
marginused	optional	Margin Used

Positions Book

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

Request Details :

Fields	Possible value	Description
actid*		Account id of the logged in user.

Response Details :PositionBookResponse list of PositionBookItem

Fields	Possible value	Description
stat	Ok or Not_Ok	Position book success or failure indication.
exch		Exchange segment
tsym		Trading symbol / contract.
token		Contract token
uid		User Id
actid		Account Id
prd		Product name to be shown.
netqty		Net Position quantity
netavgprc		Net position average price
daybuyqty		Day Buy Quantity
daysellqty		Day Sell Quantity
daybuyavgprc		Day Buy average price
daysellavgprc		Day buy average price
daybuyamt		Day Buy Amount
daysellamt		Day Sell Amount
cfbuyqty		Carry Forward Buy Quantity
cforgavgprc		Original Avg Price
cfsellqty		Carry Forward Sell Quantity
cfbuyavgprc		Carry Forward Buy average price

cfsellavgprc		Carry Forward Buy average price
cfbuyamt		Carry Forward Buy Amount
cfsellamt		Carry Forward Sell Amount
lp		LTP
rpnl		RealizedPNL
urmtom		UnrealizedMTOM. (Can be recalculated in LTP update : = netqty * (lp from web socket - netavgprc) * prcftr
bep		Break even price
openbuyqty		
opensellqty		
openbuyamt		
opensellamt		
openbuyavgprc		
opensellavgprc		
mult		
pp		
prcftr		gn*pn/(gd*pd).
ti		Tick size
ls		Lot size
request_time		This will be present only in a failure response.

Response data will be as below fields in case of failure:

Fields	**Possible value**	**Description**

stat	Not_Ok	Position book request failure indication.
request_time		Response received time.
emsg		Error message

Sample Success Response :

[
{
"stat":"Ok",
"uid":"POORNA",
"actid":"POORNA",
"exch":"NSE",
"tsym":"ACC-EQ",
"prarr":"C",
"pp":"2",
"ls":"1",
"ti":"5.00",
"mult":"1",
"prcftr":"1.000000",
"daybuyqty":"2",
"daysellqty":"2",
"daybuyamt":"2610.00",
"daybuyavgprc":"1305.00",
"daysellamt":"2610.00",
"daysellavgprc":"1305.00",
"cfbuyqty":"0",
"cfsellqty":"0",
"cfbuyamt":"0.00",
"cfbuyavgprc":"0.00",
"cfsellamt":"0.00",
"cfsellavgprc":"0.00",
"openbuyqty":"0",
"opensellqty":"23",
"openbuyamt":"0.00",
"openbuyavgprc":"0.00",
"opensellamt":"30015.00",
"opensellavgprc":"1305.00",
"netqty":"0",
"netavgprc":"0.00",
"lp":"0.00",
"urmtom":"0.00",
"rpnl":"0.00",
"cforgavgprc":"0.00"

  }
]

Sample Failure Response :

{
  "stat":"Not_Ok",
  "request_time":"14:14:11 26-05-2020",
  "emsg":"Error Occurred : 5 \"no data\""
}

Product Conversion

```
public bool SendGetOrderMargin(OnResponse response, ProductConversion prdConv)
```

Request Details :ProductConversion

Fields	Possible value	Description
exch*		Exchange
tsym*		Unique id of contract on which order was placed. Can'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 :ProductConversionResponse

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.

Sample Success Response :

{
  "request_time":"10:52:12 02-06-2020",
  "stat":"Ok"
}

Sample Failure Response :

{
  "stat":"Not_Ok",
  "emsg":"Invalid Input : Invalid Position Type"
}

Holdings and Limits

Holdings

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

Request Details :

Fields	Possible value	Description
actid*		Account id of the logged in user.
prd*		Product name

Response Details : HoldingsResponse list of HoldingsItem

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

colqty		Collateral quantity
btstqty		BTST quantity
btstcolqty		BTST Collateral quantity
usedqty		Holding used today
upldprc		Average price uploaded along with holdings

Exch_tsym object:

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 as below fields in case of failure:

Fields	**Possible value**	**Description**
stat	Not_Ok	Position book request failure indication.
request_time		Response received time.
emsg		Error message

Sample Success Response :

```
[
    {
        "stat":"Ok",
        "exch_tsym":[
                        {
```

```
                    "exch":"NSE",
                    "token":"13",
                         "tsym":"ABB-EQ"
                     }
                ],
        "holdqty":"2000000",
        "colqty":"200",
        "btstqty":"0",
        "btstcolqty":"0",
        "usedqty":"0",
        "upldprc" : "1800.00"
    },
    {
        "stat":"Ok",
        "exch_tsym":[
                    {
                         "exch":"NSE",
                         "token":"22",
                         "tsym":"ACC-EQ"
                    }
                ],
        "holdqty":"2000000",
        "colqty":"200",
        "btstqty":"0",
        "btstcolqty":"0",
        "usedqty":"0",
        "upldprc" : "1400.00"
    }
]
```

Sample Failure Response :

```
{
  "stat":"Not_Ok",
  "emsg":"Invalid Input : Missing uid or actid or prd."
}
```

Limits

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

Request Details :

Fields	Possible value	Description
actid*		Account id of the logged in user.

prd		Product name
seg	CM / FO / FX	Segment
exch		Exchange

Response Details :LimitsResponse

Fields	Possible value	Description
stat	Ok or Not_Ok	Limits request success or failure indication.
actid		Account id
prd		Product name
seg	CM / FO / FX	Segment
exch		Exchange
---------------------------------Cash Primary Fields--		
cash		Cash Margin available
payin		Total Amount transferred using Payins today
payout		Total amount requested for withdrawal today
---------------------------------Cash Additional Fields--		
brkcollamt		Prevalued Collateral Amount
unclearedcash		Uncleared Cash (Payin through cheques)
daycash		Additional leverage amount / Amount added to handle system errors - by broker.
---------------------------------Margin Utilized--		
marginused		Total margin / fund used today
mtomcurper		Mtom current percentage
---------------------------------Margin Used components----------------------------		

cbu		CAC Buy used
csc		CAC Sell Credits
rpnl		Current realized PNL
unmtom		Current unrealized mtom
marprt		Covered Product margins
span		Span used
expo		Exposure margin
premium		Premium used
varelm		Var Elm Margin
grexpo		Gross Exposure
greexpo_d		Gross Exposure derivative
scripbskmar		Scrip basket margin
addscripbskmrg		Additional scrip basket margin
brokerage		Brokerage amount
collateral		Collateral calculated based on uploaded holdings
grcoll		Valuation of uploaded holding pre haircut
------------------------------Additional Risk Limits-----------------------------------		
turnoverlmt		
pendordvallmt		
------------------------------Additional Risk Indicators-----------------------------------		
turnover		Turnover
pendordval		Pending Order value
------------------------------Margin used detailed breakup fields------------------------------		
rzpnl_e_i		Current realized PNL (Equity Intraday)
rzpnl_e_m		Current realized PNL (Equity Margin)

rzpnl_e_c		Current realized PNL (Equity Cash n Carry)
rzpnl_d_i		Current realized PNL (Derivative Intraday)
rzpnl_d_m		Current realized PNL (Derivative Margin)
rzpnl_f_i		Current realized PNL (FX Intraday)
rzpnl_f_m		Current realized PNL (FX Margin)
rzpnl_c_i		Current realized PNL (Commodity Intraday)
rzpnl_c_m		Current realized PNL (Commodity Margin)
uzpnl_e_i		Current unrealized MTOM (Equity Intraday)
uzpnl_e_m		Current unrealized MTOM (Equity Margin)
uzpnl_e_c		Current unrealized MTOM (Equity Cash n Carry)
uzpnl_d_i		Current unrealized MTOM (Derivative Intraday)
uzpnl_d_m		Current unrealized MTOM (Derivative Margin)
uzpnl_f_i		Current unrealized MTOM (FX Intraday)
uzpnl_f_m		Current unrealized MTOM (FX Margin)
uzpnl_c_i		Current unrealized MTOM (Commodity Intraday)
uzpnl_c_m		Current unrealized MTOM (Commodity Margin)
span_d_i		Span Margin (Derivative Intraday)
span_d_m		Span Margin (Derivative Margin)
span_f_i		Span Margin (FX Intraday)
span_f_m		Span Margin (FX Margin)
span_c_i		Span Margin (Commodity Intraday)
span_c_m		Span Margin (Commodity Margin)
expo_d_i		Exposure Margin (Derivative Intraday)
expo_d_m		Exposure Margin (Derivative Margin)
expo_f_i		Exposure Margin (FX Intraday)

expo_f_m		Exposure Margin (FX Margin)
expo_c_i		Exposure Margin (Commodity Intraday)
expo_c_m		Exposure Margin (Commodity Margin)
premium_d_i		Option premium (Derivative Intraday)
premium_d_m		Option premium (Derivative Margin)
premium_f_i		Option premium (FX Intraday)
premium_f_m		Option premium (FX Margin)
premium_c_i		Option premium (Commodity Intraday)
premium_c_m		Option premium (Commodity Margin)
varelm_e_i		Var Elm (Equity Intraday)
varelm_e_m		Var Elm (Equity Margin)
varelm_e_c		Var Elm (Equity Cash n Carry)
marprt_e_h		Covered Product margins (Equity High leverage)
marprt_e_b		Covered Product margins (Equity Bracket Order)
marprt_d_h		Covered Product margins (Derivative High leverage)
marprt_d_b		Covered Product margins (Derivative Bracket Order)
marprt_f_h		Covered Product margins (FX High leverage)
marprt_f_b		Covered Product margins (FX Bracket Order)
marprt_c_h		Covered Product margins (Commodity High leverage)
marprt_c_b		Covered Product margins (Commodity Bracket Order)
scripbskmar_e_i		Scrip basket margin (Equity Intraday)
scripbskmar_e_m		Scrip basket margin (Equity Margin)
scripbskmar_e_c		Scrip basket margin (Equity Cash n Carry)
addscripbskmrg_d_i		Additional scrip basket margin (Derivative Intraday)

addscripbskmrg_d_m		Additional scrip basket margin (Derivative Margin)
addscripbskmrg_f_i		Additional scrip basket margin (FX Intraday)
addscripbskmrg_f_m		Additional scrip basket margin (FX Margin)
addscripbskmrg_c_i		Additional scrip basket margin (Commodity Intraday)
addscripbskmrg_c_m		Additional scrip basket margin (Commodity Margin)
brkage_e_i		Brokerage (Equity Intraday)
brkage_e_m		Brokerage (Equity Margin)
brkage_e_c		Brokerage (Equity CAC)
brkage_e_h		Brokerage (Equity High Leverage)
brkage_e_b		Brokerage (Equity Bracket Order)
brkage_d_i		Brokerage (Derivative Intraday)
brkage_d_m		Brokerage (Derivative Margin)
brkage_d_h		Brokerage (Derivative High Leverage)
brkage_d_b		Brokerage (Derivative Bracket Order)
brkage_f_i		Brokerage (FX Intraday)
brkage_f_m		Brokerage (FX Margin)
brkage_f_h		Brokerage (FX High Leverage)
brkage_f_b		Brokerage (FX Bracket Order)
brkage_c_i		Brokerage (Commodity Intraday)
brkage_c_m		Brokerage (Commodity Margin)
brkage_c_h		Brokerage (Commodity High Leverage)
brkage_c_b		Brokerage (Commodity Bracket Order)

request_time		This will be present only in a successful response.
emsg		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 : "
}

Market Info

Get Index List

Request Details :

Fields	**Possible value**	**Description**

exch*		Exchange

Response Details :

Fields	Possible value	Description
stat	Ok or Not_Ok	TopListNames success or failure indication.
values		Array Of Basket, Criteria pair.
request_time		This will be present only in a successful response.
emsg		This will be present only in case of errors.

Basket, Criteria pair Object :

Fields	Possible value	Description
idxname		Index Name
token		Index token used to subscribe

Sample Output:


```
{
   "request_time": "20:12:29 13-12-2020",
   "values": [
       {
           "idxname": "HangSeng BeES-NAV",
           "token": "26016"
       },
       {
           "idxname": "India VIX",
           "token": "26017"
       },
       {
           "idxname": "Nifty 50",
           "token": "26000"
```

```json
        },
        {
            "idxname": "Nifty IT",
            "token": "26008"
        },
        {
            "idxname": "Nifty Next 50",
            "token": "26013"
        },
        {
            "idxname": "Nifty Bank",
            "token": "26009"
        },
        {
            "idxname": "Nifty 500",
            "token": "26004"
        },
        {
            "idxname": "Nifty 100",
            "token": "26012"
        },
        {
            "idxname": "Nifty Midcap 50",
            "token": "26014"
        },
        {
            "idxname": "Nifty Realty",
            "token": "26018"
        },
    ]
}
```

Get Top List Names

Request Details :

Fields	Possible value	Description
exch*		Exchange

Response Details :

Fields	Possible value	Description
stat	Ok or Not_Ok	TopListNames success or failure indication.
values		Array Of Basket, Criteria pair.
request_time		This will be present only in a successful response.
emsg		This will be present only in case of errors.

Basket, Criteria pair Object :

Fields	Possible value	Description
bskt		Basket name
crt		criteria

Sample Success Response :


```
{
   "request_time":"13:08:22 03-06-2020",
  "values":[
        {
            "bskt":"NSEBL",
            "crt":"VOLUME"
        },
        {
            "bskt":"NSEBL",
            "crt":"LTP"
        },
        {
            "bskt":"NSEBL",
            "crt":"VALUE"
        },
        {
```

"bskt":"NSEEQ",
"crt":"VOLUME"
},
{
"bskt":"NSEEQ",
"crt":"LTP"
},
{
"bskt":"NSEEQ",
"crt":"VALUE”
},
{
"bskt":"NSEALL",
"crt":"VOLUME"
},
{
"bskt":"NSEALL",
"crt":"LTP"
},
{
"bskt":"NSEALL",
"crt":"VALUE"
}
]
}

Sample Failure Response :

{
"stat":"Not_Ok",
"emsg":"Session Expired : Invalid Session Key"
}

Get Top List

Request Details :

Fields	**Possible value**	**Description**
exch*		Exchange
tb*	T or B	Top or Bottom
bskt*		Basket name

crt*		criteria

Response Details :

Fields	Possible value	Description
stat	Ok or Not_Ok	TopList success or failure indication.
values		Array of top / bottom contracts object
request_time		This will be present only in a successful response.
emsg		This will be present only in case of errors.

top / bottom contracts object :

Fields	Possible value	Description
tsym		Trading symbol
lp		LTP
c		Previous Close price
v		volume
value		Total traded value
oi		Open interest
pc		LTP percentage change

Sample Success Response :


```
[
    {
        "stat":"Ok",
        "request_time":"15:44:45 03-06-2020",
        "values":[
                {
                        "tsym":"AIRAN-EQ",
                        "lp":"950.00",
                        "c":"915.00",
                        "v":"42705",
```

"value":"40185405.00",
"oi":"0",
"Pc":"3.83"
},
{
"tsym":"SHRENIK-EQ",
"lp":"1850.00",
"c":"1785.00",
"v":"206846",
"value":"368806418.00",
"oi":"0",
"Pc":"3.64”
},
{
"tsym":"REMSONSIND-EQ",
"lp":"6000.00",
"c":"5795.00",
"v":"3948",
"value":"22752324.00",
"Oi":"0",
"pc":"3.54"
},
{
"tsym":"AXISNIFTY-EQ",
"lp":"106700.00",
"c":"103301.00",
"v":"422",
"value":"43825544.00",
"oi":"0",
"Pc":"3.29"
}
]
}
]

Sample Failure Response :

{
"stat":"Not_Ok",
"emsg":"Invalid Input : Missing uid or exch or bskt or tb or crt"
}

Get Time Price Data (Chart data)

Request Details :

Fields	Possible value	Description

exch*		Exchange
token*		

Response Details :

Fields	Possible value	Description
stat	Not_Ok	TPData failure indication.
emsg		This will be present only in case of errors.

Response data will be as follows in case for success.

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

Sample Success Response :

```
[
  {
    "stat":"Ok",
    "time":"02-06-2020 15:46:23",
    "into":"0.00",
    "inth":"0.00",
    "intl":"0.00",
    "intc":"0.00",
    "intvwap":"0.00",
    "intv":"0",
    "intoi":"0",
    "v":"980515",
    "oi":"128702"
  },
  {
    "stat":"Ok",
    "time":"02-06-2020 15:45:23",
    "into":"0.00",
    "inth":"0.00",
    "intl":"0.00",
    "intc":"0.00",
    "intvwap":"0.00",
    "intv":"0",
    "intoi":"0",
    "v":"980515",
    "oi":"128702"
  },
  {
    "stat":"Ok",
    "time":"02-06-2020 15:44:23",
    "into":"0.00",
    "inth":"0.00",
    "intl":"0.00",
    "intc":"0.00",
    "intvwap":"0.00",
    "intv":"0",
    "intoi":"0",
    "v":"980515",
    "oi":"128702"
  },
  {
    "stat":"Ok",
    "time":"02-06-2020 15:43:23",
    "into":"1287.00",
    "inth":"1287.00",
    "intl":"0.00",
    "intc":"1287.00",
    "intvwap":"128702.00",
```

"intv":"4",
"intoi":"128702",
"v":"980515",
"oi":"128702"
},
{
"stat":"Ok",
"time":"02-06-2020 15:42:23",
"into":"0.00",
"inth":"0.00",
"intl":"0.00",
"intc":"0.00",
"intvwap":"0.00",
"intv":"0",
"intoi":"0",
"v":"980511",
"oi":"128702"
}
]

Sample Failure Response :

{
"stat":"Not_Ok",
"emsg":"Session Expired : Invalid Session Key"
}

Get Option Chain

Request Details :

Fields	Possible value	Description
uid*		Logged in User Id
tsym*		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)
exch*		Exchange (UI need to check if exchange in NFO / CDS / MCX / or any other exchange which has options, if not don't allow)
strprc*		Mid price for option chain selection

cnt*		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)

Response Details :

Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Market watch success or failure indication.
values		Array of objects. (object fields given in below table)
emsg		This will be present only in case of errors. That is : 1) Invalid Input 2) Session Expired

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)
optt		Option Type
strprc		Strike price
pp		Price precision
ti		Tick size
ls		Lot size

Order Updates and MarketData Update

This Api allows you to receive updates receive the 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)

Request:

Fields	**Possible value**	**Description**
uri		Server EndPoint
OnFeed		Application Callback for MarketData
OnOrderFeed		Callback for Order Updates

Subscribe Market Data

public bool SubscribeToken(***string exch***, ***string token***)
Request :

Fields	**Possible value**	**Description**
exch	NSE, BSE, NFO ...	Exchange
token		Scrip Token

Market Data Updates :

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 ..	Exchange name
tk	22	Scrip Token
lp		LTP
pc		Percentage change
v		volume
o		Open price
h		High price
l		Low price
c		Close price
ap		Average trade price

UnSubscribe Market Data

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

Request :

Fields	Possible value	Description
exch	NSE, BSE, NFO ...	Exchange
token		Scrip Token

Subscribe Market Data Depth

public bool SubscribeTokenDepth(***string exch***, ***string token***)

Request :

Fields	Possible value	Description
exch	NSE, BSE, NFO ...	Exchange
token		Scrip Token

Depth subscription Updates :

Fields	Possible value	Description
t	df	‘df’ represents depth feed
e	NSE, BSE, NFO ..	Exchange name
tk	22	Scrip Token
lp		LTP
pc		Percentage change
v		volume
o		Open price
h		High price
l		Low price
c		Close price
ap		Average trade price
ltt		Last trade time
ltq		Last trade quantity
tbq		Total Buy Quantity
tsq		Total Sell Quantity
bq1		Best Buy Quantity 1
bq2		Best Buy Quantity 2
bq3		Best Buy Quantity 3
bq4		Best Buy Quantity 4

bq5		Best Buy Quantity 5
bp1		Best Buy Price 1
bp2		Best Buy Price 2
bp3		Best Buy Price 3
bp4		Best Buy Price 4
bp5		Best Buy Price 5
bo1		Best Buy Orders 1
bo2		Best Buy Orders 2
bo3		Best Buy Orders 3
bo4		Best Buy Orders 4
bo5		Best Buy Orders 5
sq1		Best Sell Quantity 1
sq2		Best Sell Quantity 2
sq3		Best Sell Quantity 3
sq4		Best Sell Quantity 4
sq5		Best Sell Quantity 5
sp1		Best Sell Price 1
sp2		Best Sell Price 2
sp3		Best Sell Price 3
sp4		Best Sell Price 4
sp5		Best Sell Price 5
so1		Best Sell Orders 1
so2		Best Sell Orders 2
so3		Best Sell Orders 3
so4		Best Sell Orders 4

so5		Best Sell Orders 5
lc		Lower Circuit Limit
uc		Upper Circuit Limit
52h		52 week high low in other exchanges, Life time high low in mcx
52l		52 week high low in other exchanges, Life time high low in mcx

Sample Message :


```
{
    "t": "df",
    "e": "NSE",
    "tk": "22",
    "o": "1166.00",
    "h": "1179.00",
    "l": "1145.35",
    "c": "1152.65",
    "ap": "1159.74",
    "v": "819881",
    "tbq": "120952",
    "tsq": "131730",
    "bp1": "1156.00",
    "sp1": "1156.50",
    "bp2": "1155.80",
    "sp2": "1156.55",
    "bp3": "1155.75",
    "sp3": "1156.65",
    "bp4": "1155.70",
    "sp4": "1156.70",
    "bp5": "1155.65",
    "sp5": "1156.75",
    "bq1": "4",
    "sq1": "10",
    "bq2": "67",
    "sq2": "63",
    "bq3": "83",
    "sq3": "1",
    "bq4": "139",
    "sq4": "53",
    "bq5": "393",
```

```
    "sq5": "94"
}
```

Unsubscribe Depth

public bool Un***SubscribeTokenDepth***(***string exch***, ***string token***)

Request :

Fields	**Possible value**	**Description**
exch	NSE, BSE, NFO ...	Exchange
token		Scrip Token

Subscribe Order Update

public bool SubscribeOrders(OnOrderFeed orderFeed, string account)

Request :

Fields	Possible value	Description
actid		Account id based on which order updated to be sent.

Order Update subscription Updates : **NorenOrderFeed**

Fields	Possible value	Description
t	om	'om' represents touchline feed
norenordno		Noren Order Number
uid		User Id
actid		Account ID
exch		Exchange
tsym		Trading symbol
qty		Order quantity
prc		Order Price
prd		Product
status		Order status (open, complete, rejected etc)
reporttype		Order event for which this message is sent out. (fill, rejected, canceled)
trantype		Order transaction type, buy or sell
prctyp		Order price type (LMT, MKT, SL-LMT, SL-MKT)
ret		Order retention type (DAY, EOS, IOC,...)
fillshares		Filled shares

avgprc		Average fill price
rejreason		Order rejection reason, if rejected
exchordid		Exchange Order ID
cancelqty		Canceled quantity, in case of canceled order
remarks		User added tag, while placing order
dscqty		Disclosed quantity
trgprc		Trigger price for SL orders