NOREN REST Client API

KAMBALA SOLUTIONS PVT LTD

TABLE OF CONTENTS

CHANGE LOG

Date	Change
20 Jun 2022	1) Added /InteropPositionBook
27 Jun 2022	1) Added /GetOrderReport and /GetTradeReport. 2) Added upldprc and netupldprc in /Positionbook. 3) Changes in /setalert and /modifyalert request. 4) Added hair_cut field in holding response.
Jun 30 2022	1) Added /AMOStatusFlag Req. 2) Added NOREN_KID_ID field in /SingleOrdHist and /OrderBook. 3) Added BLOCK_AMT field in /limits.
Jul 1 2022	1) Added /ValOTPStPwd and /FgtPwdOTP
Jul 02 2022	1) Error messages added in /QuickAuth and /ValOTPStPwd 2) Error messages added in ChangePassword
Jul 11 2022	1) Added uid and brnchid in /QuickAuthBypass and QuickAuthSpl. 2) Added exarr in /ClientDetails . 3) Added algo_id and naic_code in /PlaceOrder
Jul 19 2022	1) Added /GetSubLimits.
Aug 30 2022	1) Added buyavgprc ,sellavgprc,rpnl and netqty in /InteropPositionBook req.
23 Sep 2022	Added seg,exch,prd in GetMaxPayoutAmount req.
17 Oct 2022	Added sms_flag, email_flag and push_flag in set alerts request.
29 Nov 2022	1) Added acct_sts field in /ClientDetails resp. 2) Added new fields in /PositionBook and /InteropPositionBook.
27 Dec 2022	1) Added /GetBrokerage request
2 jan 2023	Added FX (EQT/DER/FX/COM) in below API req: ● Get Max Payout Amount

	● Get Sub Limits ● Limits
11 jan 2023	In Client details API req , brkname field removed. In Place order API req, C/M/H description added.
7 Feb 2023	In GetSecurityInfo API req, uc and lc field added.
10 Feb 2023	In WebSocket Order Update “amo” field added.
11 Feb 2023	In WebSocket Order Update -> tm field added -> kidid,sno_fillid field added
22 Feb 2022	1) Added und_exch,und_tk and ltd in /GetQuotes. 2) Added prc in /TradeBook. 3) Added dob in /ClientDetails . 4) Below fields added in /limits response (mr_eqt_u, mr_der_u, mr_fx_u, mr_com, mr_sell, mr_t1sell, mr_eqt_a, mr_der_a, mr_fx_a, mr_com_a) 5) Added s_prdt_ali field in below api request. (/OrderBook ,/SingleOrdHist, /TradeBook, /PositionBook, /GetOrderReport, /Holdings, /Limits, /GetSubLimits, /GetTradeReport, /InteropPositionBook)
23 Feb 2023	In span cal api , date format updated. Missed ‘Prd’ field updated.
27 Feb 2023	Added close price in /GetQuotes and instrument name in /PositionBook
28 Feb 2023	Added ordersource in /ModifyOrder

INTRODUCTION

Kambala Solutions is a solution oriented firm indulged in serving financial organizations with innovative products and services. We understand the new challenges and obstacles which are continuously arising for businesses with rapid advances in technology and environment, they could only be tackled with efficient and effective approaches. We have expertise in building high throughput, low latency distributed applications that can support a large concurrent user base.

Noren OMS solution is built keeping in mind the large and diverse Indian broking industry. Few of the prominent coverages are Order Execution, Algo Trading, Exchange connectivity and Risk Mitigation. The Noren OMS team strives to serve ever changing requirements with agility and personal touch. Noren OMS is continuously being tested for stability, high scalability and low latency in mind.

This document provides details of APIs to develop Web/Mobile applications for Noren OMS.

Login and User Details

Login

Request to be POSTed to uri : **/NorenWClientTP/QuickAuth**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list

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|vendor_key
imei*		Send mac if users logs in for desktop, imei is from mobile
addldivinf		Optional field, Value must be in below format: iOS - iosInfo.utsname.machine - iosInfo.systemVersion Android - androidInfo.model - androidInfo.version examples: iOS - iPhone 8.0 - 9.0 Android - Moto G - 9 PKQ1.181203.01
ipaddr		Optional field
source	API	

Example:
curl https://apitest.kambala.co.in/NorenWClientTP/QuickAuth \
-d "jData={ \"apkversion\": \"1.0.0\", \"uid\": \"VIDYA\", \"pwd\": \"s3cur3Id\", \"factor2\": \"31-08-2017\", \"imei\": \"134243434\", \"source\": \"API\"}"

Response Details :
Response data will be in json format with below fields.

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 emsg is "Password Expired" or "Change Password" and stat is Not_Ok]	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
uid		UserId
brnchid		Region
emsg		This will be present only if Login fails. (Redirect to force change password if message is

		“Invalid Input : Password Expired” or “Invalid Input : Change Password”)

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"
}

Error Message list:

Message	Description
"Invalid Input : Missing jData"	Common error message of any of the API (dev issue)
“Invalid Input : Request data is missing.”	Common error message of any of the API (dev issue)
"Invalid Input : jData is not valid json object"	Common error message of any of the API (dev issue)
"Invalid Input : {<Mandatory field name 1>} {or <Mandatory field name n>}... is Missing." Example:	Common error message of any of the API (dev issue)
"Invalid Input : One or more input parameters are not in string format"	Common error message of any of the API (dev issue)
"Invalid Input : Invalid App Key"	API enablement configuration issue (dev/deploy issue)

"Invalid Input : Invalid Vendor code"	API enablement configuration issue (dev/deploy issue)
“Invalid Input : Latest app available, please update”	Version blocked at API server level
"Invalid Input : Wrong PAN/DOB"	
"Invalid Input : Wrong Password"	
"Invalid Input : Invalid User"	
"Invalid Input : Deactivated"	
"Invalid Input : Version blocked: Please download latest version"	Version blocked in OMS level
"Invalid Input : User Blocked due to multiple wrong attempts"	
"Invalid Input : User Not enabled on : WEB" Or "Invalid Input : User Not enabled on : ---"	Depending on custom access type configured in the system message will change
"Error Occurred : 1 "unknown request"" Or “Server Timeout : ”	OMS is down and Web server only up, (EOD/BOD time)
"Invalid Input : Invalid Access Type"	API server instance is not configured to handle input access type. (Mismatch in API URL)
"Invalid Input : Password Expired"	Password expired after configured number of days, Redirect to Change password screen.
"Invalid Input : Change Password"	If password reset by admin/system, Redirect to Change password screen.

Logout

Request to be POSTed to uri : **/NorenWClientTP/Logout**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User Id of the login user

Response Details :
Response data will be in json format with below fields.

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.

Sample Success Response :

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

Sample Failure Response :

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

Forgot Password

Request to be POSTed to uri : **/NorenWClientTP/ForgotPassword**

Request Details :

Parameter Name	Possible value	Description

jData*		Should send json object with fields in below list

Json Fields	Possible value	Description
uid*		User Id
pan*		Pan of the user
dob		Date of birth in DDMMYYYY format

Response Details :
Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	Password reset is Success Or failure status
request_time		Response received time.
emsg	"Error Occurred : Wrong user id or user details" "Server Timeout : "	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"
}
```


Forgot Password OTP

Request to be POSTed to uri : **/NorenWClientTP/FgtPwdOTP**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list

Json Fields	Possible value	Description
uid*		User Id
pan*		Pan of the user Or Sha256 3 times of password

Response Details :
Response data will be in json format with below fields.

Json Fields	Possible value	Description
uid		User Id
ReqStatus		Request status, present only when success. Value will be “OTP generation success”
emsg		Error message : “Error Occurred : Wrong user id or user details”

Sample Success Response :

```
{
  "uid":"user1",
  "ReqStatus":"OTP generation success"
}
```

Sample Failure Response :

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

Validate OTP

Request to be POSTed to uri : **/NorenWClientTP/ValOTPStPwd**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list

Json Fields	Possible value	Description
uid*		User Id
pwd*		New password to be set in plain text
otp*		OTP needs to sent in this field

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
ReqStatus	1) “Password Reset Success” 2) “Otp expired” 3) “Invalid OTP” 4) "Please enter an alphanumeric password of minimum 8 characters. Refer password criteria for more details" 5) “Password couldn't be changed as it is among the previous 3 passwords”	Request status
emsg		Error message

Sample Success Response :

```
{
    "ReqStatus":"Password Reset Success"
}
```

Sample Failure Response :

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

Change Password

Request to be POSTed to uri : **/NorenWClientTP/Changepwd**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list

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

Response Details :
Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	Password reset is Success Or failure status
request_time		Response received time.
dmsg	"Password Change Success. Your new password will expire in 60 days"	This will be present only in case of success. Number of days to expiry will be present in the same.
emsg	1) "Error Occurred : Password couldn't be changed as it is among the previous 3 passwords" 2) "Error Occurred : Please enter an alphanumeric password of minimum 8 characters. Refer password criteria for more details"	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 days”
}

Sample Failure Response :

{
  "request_time":"10:21:09 27-05-2020",
  "stat":"Not_Ok",
  "emsg":"Error Occurred : Password couldn't be changed as it is among the previous 3 passwords"
}

Set Device Pin

Request to be POSTed to uri : **/NorenWClientTP/SetPin**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User Id
imei*		Imei or device unique fingerprint
source*		Access type (API)
dpin*		New pin in plain text

Response Details :
Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	If Pin setting is Success Or failure status

request_time		This will be present only if password change succeeds.
emsg		This will be present only if password change fails

Sample Success Response :

{
  "request_time":"14:59:43 27-05-2020",
  "stat":"Ok"
}

Sample Failure Response :

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

Login with Device Pin

Request to be POSTed to uri : **/NorenWClientTP/PinAuth**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list

Json Fields	**Possible value**	**Description**
uid*		User Id
imei*		Imei or device unique fingerprint
source*		Access type (API)
dpin*		sha256 of entered device pin
vc*		Vendor code provided by noren team, along with connection URLs
appkey*		Sha256 of uid|vendor_key
apkversion*		Application version number

ipaddr		global Ip of internet access

Response Details :
Response data will be in json format with below fields.

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.
request_time		Response received time.
emsg	"Invalid Input : MPIN not enabled for user", Also refer QuickAuth Error Messages list	This will be present only if Login fails.

Sample Success Response :


```
{
  "request_time":"17:01:45 27-05-2020",
  "stat":"Ok",
  "susertoken":"b0856b3f6c4bac657417fc95de3e2060567b8bd80665e0a8ab82bbde5c434936",
  "lastaccesstime":"1590579105"
}
```


Sample Failure Response :


```
{
  "request_time":"11:19:56 28-05-2020",
  "stat":"Not_Ok",
  "emsg":"Invalid Input : Mpin Invalid"
}
```

Get HS Token

Request to be POSTed to uri : **/NorenWClientTP/GetHsToken**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User Id

Response Details :
Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	If Pin setting is Success Or failure status
request_time		This will be present only if password change succeeds.
hstk		One time Token to be sent to BackOffice or third party link
emsg		This will be present only if password change fails

Validate HS Token

Request to be POSTed to uri : **/NorenWClientTP/ValidateHsToken**
(To be used only from server, Call this url from Browser / Client Side APKs)

Request Details :

Parameter Name	Possible value	Description

LoginId*		Send sLoginId received from Initiator site
token*		Key Obtained on login success.0

Response Details :
Response data will be in plain text format TRUE if Token is valid and FALSE for invalid User Id or Token.

External Integration (Backoffice Url..etc) Flow:

1. Trading site will call the third party url on user clicking the specified link (eg:Back Office login)
2. Trading site will pass the User id , Token and Client ID to the the third party url
3. Third Party application/web server will make a server call to our web server using this “Validate HS Token” Url.
4. If Trading site web server says ok then Third party application will provide access to the user/client

User Details

Request to be POSTed to uri : **/NorenWClientTP/UserDetails**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	User details success or failure indication.

exarr		Json array of strings with enabled exchange names
orarr		Json array of strings with enabled price types for user
prarr		Json array of Product Obj with enabled products, as defined below.
brkname		Broker id
brnchid		Branch id
email		
actid		Account Id
uid		User Id
m_num		Mobile Number
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

Json Fields	**Possible value**	**Description**
prd		Product name
s_prdt_ali		Product display name
exch		Json 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"
}

Client Details

Request to be POSTed to uri : **/NorenWClientTP/ClientDetails**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	User details success or failure indication.
actid		Account ID
creatdte		Creation date
creattme		Creation time
m_num		Mobile Number
email		Email ID
pan		PAN
dob		Date of birth in DDMMYYYY format

act_sts		Account Status
addr		Address
addroffice		Office address
addrcity		City
addrstate		State
bankdetails		Array Object, details given below.
dp_acct_num		Array Object, details given below.
exarr	["CDS","NSE", "NFO","MCX"," BSE","NCX"," BSTAR","BCD"]	Json array of strings with enabled exchange names
mandate_id_list		Mandate Id List [Array Object, details given below.]
request_time		It will be present only in a successful response.
emsg		This will be present only in case of errors.

bankdetails Obj format

Json Fields	Possible value	Description
bankn		Bank name
acctnum		Account number

dp_acct_num Obj format

Json Fields	Possible value	Description
dpnum		Dp account number

mandate_id_list Obj format

Json Fields	Possible value	Description

mandate_id		Mandate Id

Save FCM token

Request to be POSTed to uri : **/NorenWClientTP/SaveFMCToken**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User Id
fcmtkn*		FCM token collected from device

Response Details :
Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	If save FCM token is Success Or failure status
request_time		This will be present only if password change succeeds.
emsg		This will be present only if save token fails

Sample Success Response :

{
  "request_time":"14:59:43 27-05-2020",
  "stat":"Ok"
}

Sample Failure Response :

{

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

Watch Lists

Get WatchList Names

Request to be POSTed to uri : **/NorenWClientTP/MWList**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	MWList success or failure indication.
values		Watch List names as a json 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

Request to be POSTed to uri : **/NorenWClientTP/MarketWatch**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
wlname*		Name of the Watchlist, for which scrip list is required.

Response Details :

Response data will be in json format with below fields.

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)
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

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
dname		Broker specific contract display name, present only if applicable.

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

Request to be POSTed to uri : **/NorenWClientTP/SearchScrip**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

Response Details :

Response data will be in json format with below fields.

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. That is : 1) Invalid Input 2) Session Expired

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
weekly		Weekly Option, ‘W1’, ‘W2’, ‘W3’, ‘W4’ th week
nontrd		Non tradable instruments

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

Request to be POSTed to uri : **/NorenWClientTP/AddMultiScripsToMW**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
wlname*		Name of the Watchlist, for which scrip list is required.
scrips*		List of scrips, example format NSE|22#BSE|506734

Response Details :

Response data will be in json format with below fields.

Json 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

Request to be POSTed to uri : **/NorenWClientTP/DeleteMultiMWScrips**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
wlname*		Name of the Watchlist, for which scrip list is required.
scrips*		List of scrips, example format NSE|22#BSE|506734

Response Details :

Response data will be in json format with below fields.

Json 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

Request to be POSTed to uri : **/NorenWClientTP/GetSecurityInfo**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description

uid*		Logged in User Id
exch		Exchange
token		Contract Token

Example:
jData={"uid":"{{USER_ID}}", "exch":"NSE", "token":"22"}&jKey=c180c60ec8f7870490ec27f4accdba179a149a5e5b1d503a0682d8b72acb24b3

Response Details :

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 (text format)
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
mkt_t		Market type
issue_d		Issue date
listing_d		Listing date

last_trd_d		last trading date
elmmrg		Elm Margin
varmrg		Var Margin
expmrg		Exposure Margin
token		Contract Token
prcftr_d		((GN / GD) * (PN/PD)) (actual value for calculations)
weekly		Weekly Option, ‘W1’, ‘W2’, ‘W3’, ‘W4’ th week
nontrd		Non tradable instruments
dname		Broker specific contract display name, present only if applicable.
uc		Upper circuit limitlc
lc		Lower circuit limit

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\""
}

Get Quotes

Request to be POSTed to uri : **/NorenWClientTP/GetQuotes**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

Example:

jData={"uid":"{{USER_ID}}", "exch":"NSE", "token":"22"}&jKey=4c258343e010b21851856cadb188e0adc24e173cfbf945e7beda7bc045f745 48

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible	Description

	value	
stat	Ok or Not_Ok	Watch list update success or failure indication.
request_time		It will be present only in a successful response.
exch	NSE, BSE, NFO ...	Exchange
tsym		Trading Symbol
cname		Company Name
symname		Symbol Name
seg		Segment
instname		Instrument Name
isin		ISIN
pp		Price precision
ls		Lot Size
ti		Tick Size
mult		Multiplier
uc		Upper circuit limitlc
lc		Lower circuit limit
prcftr_d		Price factor ((GN / GD) * (PN/PD))
token		Token
lp		LTP
o		Open Price
c		Close Price
h		Day High Price

l		Day Low Price
v		Volume
ap		Average trade price or VWAP for day
ltq		Last trade quantity
ltt		Last trade time
ltd	dd-mm-yy	Last Trade Date
bp1		Best Buy Price 1
sp1		Best Sell Price 1
bp2		Best Buy Price 2
sp2		Best Sell Price 2
bp3		Best Buy Price 3
sp3		Best Sell Price 3
bp4		Best Buy Price 4
sp4		Best Sell Price 4
bp5		Best Buy Price 5
sp5		Best Sell Price 5
bq1		Best Buy Quantity 1
sq1		Best Sell Quantity 1
bq2		Best Buy Quantity 2
sq2		Best Sell Quantity 2
bq3		Best Buy Quantity 3
sq3		Best Sell Quantity 3
bq4		Best Buy Quantity 4

sq4		Best Sell Quantity 4
bq5		Best Buy Quantity 5
sq5		Best Sell Quantity 5
bo1		Best Buy Orders 1
so1		Best Sell Orders 1
bo2		Best Buy Orders 2
so2		Best Sell Orders 2
bo3		Best Buy Orders 3
so3		Best Sell Orders 3
bo4		Best Buy Orders 4
so4		Best Sell Orders 4
bo5		Best Buy Orders 5
so5		Best Sell Orders 5
und_exch		Underlying Exch seg
und_tk		Underlying Token

Sample Success Response :

{
"request_time":"12:05:21 18-05-2021",
"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",
"uc":"2093.95",
"lc":"1713.25",
"prcftr_d":"(1 / 1) * (1 / 1)",
"token":"22",
"lp":"0.00",
"h":"0.00",
"l":"0.00",
"v":"0",
"ltq":"0",
"ltt":"05:30:00",
"bp1":"2000.00",
"sp1":"0.00",
"bp2":"0.00",
"sp2":"0.00",
"bp3":"0.00",
"sp3":"0.00",
"bp4":"0.00",
"sp4":"0.00",
"bp5":"0.00",
"sp5":"0.00",
"bq1":"2",
"sq1":"0",
"bq2":"0",
"sq2":"0",
"bq3":"0",
"sq3":"0",
"bq4":"0",
"sq4":"0",
"bq5":"0",
"sq5":"0",
"bo1":"2",
"so1":"0",
"bo2":"0",
"so2":"0",
"bo3":"0",
"so3":"0",
"bo4":"0",
"so4":"0",
"bo5":"0",
"So5":"0"
}

Sample Failure Response :

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

Get list of predefined MWs

Request to be POSTed to uri : **/NorenWClientTP/PreDefinedMWList**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
values		Watch List names as a json array of strings.

Example:
jData={"uid":"{{USER_ID}}"}&jKey=552636ffd5d8f659235e4af6dab0bccdcd5915d26ca07a074a4912e506ea960f

Sample Success Response:

{
  "values": [
     "NIFTYBANK",
     "NIFTY50"
  ]

}

Get list of predefined MW scrips

Request to be POSTed to uri : /**NorenWClientTP/PreDefinedMW**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	**Possible value**	**Description**
uid*		Logged in User Id
wlname*		Name of the Watchlist, for which scrip list is required.

Response Details :

Response data will be in json format with below fields.

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)

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

Example:

jData={"uid":"{{USER_ID}}","wlname":"NIFTY50"}&jKey=552636ffd5d8f659235e4af6dab0bccdcd5915d26ca07a074a4912e506ea960f

Sample Success Response:


```
{
  "stat": "Ok",
  "values": [
    {
      "exch": "NSE",
      "token": "15083",
      "tsym": "ADANIPORTS-EQ",
      "pp": "2",
      "ls": "1",
      "ti": "0.05"
    },
    {
      "exch": "NSE",
      "token": "236",
      "tsym": "ASIANPAINT-EQ",
      "pp": "2",
      "ls": "1",
      "ti": "0.05"
    },
    {
      "exch": "NSE",
      "token": "5900",
      "tsym": "AXISBANK-EQ",
      "pp": "2",
      "ls": "1",
      "ti": "0.05"
    },
    {
      "exch": "NSE",
      "token": "16669",
      "tsym": "BAJAJ-AUTO-EQ",
      "pp": "2",
      "ls": "1",
      "ti": "0.05"
    },
```

```
    {
        "exch": "NSE",
        "token": "16675",
        "tsym": "BAJAJFINSV-EQ",
        "pp": "2",
        "ls": "1",
        "ti": "0.05"
    },
    {
        "exch": "NSE",
        "token": "317",
        "tsym": "BAJFINANCE-EQ",
        "pp": "2",
        "ls": "1",
        "ti": "0.05"
    },
....
]
}
```


Order and Trades

Place Order

Request to be POSTed to uri : **/NorenWClientTP/PlaceOrder**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json 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-LMT / SL-MKT order.
dscqty		Disclosed quantity (Max 10% for NSE, and 50% for MCX)
prd*	C / M / H (C-Delivery , M-Margin , H-Cover Order)	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	
mkt_protection		Market order protection percentage. Applicable only for MKT orders in BSE/BFO/BCS and MCX segments.
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. [Optional field else it will take login access type]
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)
ext_remarks		External remarks
cl_ord_id		Cli Order Id

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
algo_id		Exchange approved algo id
naic_code		

Example:

```
curl https://apitest.kambala.co.in/NorenWClientTP/PlaceOrder \
   -d "jData={\"uid\":\"VIDYA\", \"actid\":\"CLIENT1\", \"exch\":\"NSE\", \"tsym\":\"ACC-EQ\", \"qty\":\"50\", \"price\":\"1400\", \"prd\":\"H\", \"trantype\":\"B\", \"prctyp\":\"LMT\", \"ret\":\"DAY\"}“ \
   -d “jKey=GHUDWU53H32MTHPA536Q32WR”
```

Response Details :

Response data will be in json format with below fields.

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

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

Request to be POSTed to uri : **/NorenWClientTP/ModifyOrder**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
exch*		Exchange
norenordno*		Noren order number, which needs to be modified
prctyp*	LMT / MKT /	This can be modified.

	SL-MKT / SL-LMT	
prc*		Modified / New price
qty*		Modified / New Quantity / old quantity if not modified Quantity to Fill / Order Qty - This is the total qty to be filled for the order. Its Open Qty/Pending Qty plus Filled Shares (cumulative for the order) for the order. * Please do not send only the pending qty in this field
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
mkt_protection		Market order protection percentage. Applicable only for MKT orders in BSE/BFO/BCS and MCX segments.
trgprc		New trigger price in case of SL-MKT or SL-LMT
dscqty		Disclosed quantity (Max 10% for NSE, and 50% for MCX)
ext_remarks		External remarks
cl_ord_id		Cli Order Id
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)
ordersource	MOB / WEB / TT	Used to generate exchange info fields. [Optional field else it will take login access type]

Response Details :

Response data will be in json format with below fields.

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

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

Request to be POSTed to uri : **/NorenWClientTP/CancelOrder**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

uid*		User id of the logged in user.
ext_remarks		External remarks
cl_ord_id		Cli Order Id

Response Details :

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

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

Request to be POSTed to uri : **/NorenWClientTP/ExitSNOOrder**

Request Details :

Parameter Name	Possible value	Description

jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	**Possible value**	**Description**
norenordno*		'snonum' from order book if available, else Noren order number.
prd*	H / B	Allowed for only H and B products (Cover order and bracket order)
uid*		User id of the logged in user.

Response Details :

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.
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

Request to be POSTed to uri : **/NorenWClientTP/GetOrderMargin**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json 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-LMT / SL-MKT 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 :

Response data will be in json format with below fields.

Json Fields	**Possible value**	**Description**
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		Meaning changes with remarks as explained below.
ordermargin		Margin required for this order.
marginusedprev		Margin used excluding this order.
emsg		This will be present only if Order placement fails

Possible Response Cases for Order margin:

“remarks”:”Order Success” => Order will go through “cash” -> field will have total cash available with the user. “marginused” -> field will have Margin used if the user places order.
“remarks”:”Insufficient Balance” => Order will get rejected. “cash” -> field will have total cash available for this **type of order**. “marginused” -> field will have Additional margin required for this order to pass through. (Show caption as **Shortfall margin**)
“remarks”:”Order Success : Eligible Sell:<total qty available>” => Order will go through
“remarks”:”Eligible Sell: <total qty available>” => Order will get rejected.
“remarks”:”No Holdings uploaded” => Order will get rejected, no holding for this symbol
“remarks”:”Invalid scrip” or "RED is under Reconciliation" => Order will get rejected.

“remarks”:”Squareoff Order” => Order will go through

Basket Margin

Request to be POSTed to uri : **/NorenWClientTP/GetBasketMargin**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json 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
basketlists		Optional field. Array of json objects. (object fields given in below table)

Json Fields of object in values Array	**Possible value**	**Description**
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	

Response Details :

Response data will be in json format with below fields.

Json Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Place order success or failure indication.
request_time		Response received time.
remarks		This field will contain rejection reason.
marginused		Total margin used.
marginusedtrade		Margin used after trade.
emsg		This will be present only if Order placement fails

Order Book

Request to be POSTed to uri : **/NorenWClientTP/OrderBook**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

uid*		Logged in User Id
prd	H / M / ...	Product name

Example:
curl https://apitest.kambala.co.in/NorenWClientTP/OrderBook \
  -d "jData={\"uid\":\"VIDYA\"}“ \
  -d “jKey=GHUDWU53H32MTHPA536Q32WR”

Response Details :

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

Json Fields	**Possible value**	**Description**
stat	Ok or Not_Ok	Order book success or failure indication.
uid		Logged in User Id
actid		Login users account ID
exch		Exchange Segment
tsym		Trading symbol / contract on which order is placed.
norenordno		Noren Order Number
prc		Order Price
qty		Order Quantity
mkt_protection		Market Protection percentage
prd		Display product alias name, using prarr returned in user details.
s_prdt_ali		Product display name
status		[Order status](#)
trantype	B / S	Transaction type of the order
prctyp	LMT / MKT / SL-MKT /	Price type

	SL-LMT	
fillshares		Total Traded Quantity of this order (will not be present if no trades for this order)
avgprc		Average trade price of total traded quantity (will not be present if no trades for this order)
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
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		Noren time stamp
ordenttm		Order entry time
exch_tm		Exchange update time Format: dd-mm-YYYY hh:MM:SS

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.
sno_fillid		SNO fill id
prcftr		Contract price factor (GN*PN)/(GD*PD), (used for order value calculation)
mult		Contract price multiplier, (used for order value calculation)
dname		Broker specific contract display name, present only if applicable.
rqty		To be used in get margin from modify window.
rprc		To be used in get margin from modify window.
rtrgprc		To be used in get margin from modify window, for H/B products only
rblprc		To be used in get margin from modify window, for H/B products only
rorgqty		To be used in get margin from modify window.
rorgprc		To be used in get margin from modify window.
orgtrgprc		To be used in get margin from modify window, for H/B products only
orgblprc		To be used in get margin from modify window, for H/B products only
st_intrn		

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

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

Request to be POSTed to uri : **/NorenWClientTP/MultiLegOrderBook**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
prd	H / M / ...	Product name

Example:

curl https://apitest.kambala.co.in/NorenWClientTP/MultiLegOrderBook \
   -d "jData={\"uid\":\"VIDYA\"}“ \
   -d “jKey=GHUDWU53H32MTHPA536Q32WR”

Response Details :

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

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		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
dname2		Broker specific contract display name of second leg, present only if applicable.
dname3		Broker specific contract display name of third leg, present only if applicable.

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

Single Order History

Request to be POSTed to uri : **/NorenWClientTP/SingleOrdHist**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	**Possible value**	**Description**
uid*		Logged in User Id
norenordno*		Noren Order Number

- **Example:**

curl https://apitest.kambala.co.in/NorenWClientTP/SingleOrdHist \
-d "jData={\"uid\":\"VIDYA\"}“ \
-d “jKey=GHUDWU53H32MTHPA536Q32WR”

Response Details :

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

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.
s_prdt_ali		Product display name
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
norentm		
ordenttm		
exch_tm		Format: dd-mm-YYYY hh:MM:SS

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

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",
    "dscqty": "0",
    "prd": "M",
    "status": "COMPLETE",
    "rpt": "Fill",
    "fillshares": "180",
    "norentm": "19:59:32 13-12-2020",
    "exch_tm": "01-01-1980 00:00:00",
    "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",
    "dscqty": "0",
    "prd": "M",
    "status": "OPEN",
    "rpt": "New",
    "norentm": "19:59:32 13-12-2020",
    "exch_tm": "01-01-1980 00:00:00",
    "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",
    "dscqty": "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

Request to be POSTed to uri : **/NorenWClientTP/TradeBook**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

Example:

curl https://apitest.kambala.co.in/NorenWClientTP/TradeBook \
  -d "jData={\"uid\":\"VIDYA\", \“actid\”:\”DEMO1\”}“ \
  -d “jKey=GHUDWU53H32MTHPA536Q32WR”

Response Details :

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

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.
s_prdt_ali		Product display name
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
norentm		Noren time stamp
exch_tm		Exchange update time Format: dd-mm-YYYY hh:MM:SS
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.
remarks		Any message Entered during order entry.
prc		Order Price

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

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": "01-01-1980 00:00:00",
    "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": "01-01-1980 00:00:00",
        "remarks": "WC TEST Order",
        "exchordid": "6858"
    }
]
```


Positions Book

Request to be POSTed to uri : **/NorenWClientTP/PositionBook**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
actid*		Account id of the logged in user.

Example:

curl https://apitest.kambala.co.in/NorenWClientTP/PositionBook \
-d "jData={\"uid\":\"VIDYA\", \"actid\":\"ACCT_1\"}" \
-d "jKey=GHUDWU53H32MTHPA536Q32WR"

Response Details :

Response data will be in json format with Array of Objects with below fields in case of success.

Json 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.
s_prdt_ali		Product display name
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
totbuyamt		Total Buy Amount
totsellamt		Total Sell Amount

totbuyavgprc		Total Buy Avg Price
totsellavgprc		Total Sell Avg Price
lp		LTP
rpnl		RealizedPNL
urmtom		UnrealizedMTOM. (Can be recalculated in LTP update : = netqty * (lp from web socket - netavgprc) * prcftr
bep		Break even price
openbuyqty		Open Buy Quantity
opensellqty		Open Sell Quantity
openbuyamt		Open Buy Amount
opensellamt		Open Sell Amount
openbuyavgprc		Open Buy Average Price
opensellavgprc		Open Sell Average Price
mult		Contract price multiplier, (used for order value calculation)
pp		Price precision
prcftr		gn*pn/(gd*pd).
ti		Tick size
ls		Lot size
instname		Instrument Name
upldprc		Upload price
netupldprc		Net Upload Price
request_time		This will be present only in a failure response.
dname		Broker specific contract display name, present only if applicable.

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

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\""
}

Interop Positions Book

Request to be POSTed to uri : **/NorenWClientTP/InteropPositionBook**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

Example:

```
curl https://apitest.kambala.co.in/NorenWClientTP/InteropPositionBook \
   -d "jData={ \"actid\":\"ACCT_1\"}" \
   -d "jKey=GHUDWU53H32MTHPA536Q32WR"
```

Response Details :

Response data will be in json format with Array of Objects with below fields in case of success.

Json 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.
s_prdt_ali		Product display name
daybuyqty		Day Buy Quantity
daysellqty		Day Sell Quantity
daybuyamt		Day Buy Amount
daysellamt		Day Sell Amount
cfbuyqty		Carry Forward Buy Quantity
cfsellqty		Carry Forward Sell Quantity
cfbuyamt		Carry Forward Buy Amount
cfsellamt		Carry Forward Sell Amount
openbuyqty		Open Buy Quantity
opensellqty		Open Sell Quantity
openbuyamt		Open Buy Amount
opensellamt		Open Sell Amount
instname		Instrument Name
upload_prc		Upload Price
buyavgprc		Buy Average Price [(daybuyamt + cfbuyamt) / (daybuyqty + cfbuyqty)]
sellavgprc		Sell Average Price

		[(daysellamt + cfsellamt) / (daysellqty + cfsellqty)]
rpnl		Realized panel
netqty		Net Quantity [daybuyqty + cfbuyqty - daysellqty - cfsellqty]
totbuyamt		Total Buy Amount
totsellamt		Total Sell Amount
totbuyavgprc		Total Buy Avg Price
totsellavgprc		Total Sell Avg Price
openbuyavgprc		Open Buy Average Price
opensellavgprc		Open Sell Average Price
cfbuyavgprc		Carry Forward Buy average price
cfsellavgprc		Carry Forward Buy average price
child_orders		Array Object ,Details given below

child_orders Obj format

Json Fields	**Possible value**	**Description**
exch		Exchange segment
token		Contract token
daybuyqty		Day Buy Quantity
daysellqty		Day Sell Quantity
daybuyamt		Day Buy Amount
daysellamt		Day Sell Amount
cfbuyqty		CF Buy Quantity
cfsellqty		CF Sell Quantity
cfbuyamt		CF Buy Amount

cfsellamt		CF Sell Amount
openbuyqty		Open Buy Quantity
opensellqty		Open Sell Quantity
openbuyamt		Open Buy Amount
opensellamt		Open Sell Amount
upload_prc		Upload Price
totbuyamt		Total Buy Amount
totsellamt		Total Sell Amount
totbuyavgprc		Total Buy Avg Price
totsellavgprc		Total Sell Avg Price
openbuyavgprc		Open Buy Average Price
opensellavgprc		Open Sell Average Price
cfbuyavgprc		Carry Forward Buy average price
cfsellavgprc		Carry Forward Buy average price

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

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

Sample Success Response :

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

"actid":"TESTINV1",
"exch":"EQT",
"token":"PRAKASH",
"prd":"I",
"openbuyqty":"00",
"opensellqty":"09",
"openbuyamt":"00",
"opensellamt":"54000",
"daybuyqty":"01",
"daysellqty":"01",
"daybuyamt":"6000",
"daysellamt":"6000",
"cfbuyqty":"00",
"cfsellqty":"00",
"cfbuyamt":"00",
"cfsellamt":"6000",

"child_orders":
[
{
"exch":"NSE",
"token":"2708",
"openbuyqty":"00",
"opensellqty":"09",
"openbuyamt":"00",
"opensellamt":"54000",
"daybuyqty":"00",
"daysellqty":"01",
"daybuyamt":"00",
"daysellamt":"6000",
"cfbuyqty":"00",
"cfsellqty":"00",
"cfbuyamt":"00",
"cfsellamt":"00",
"upload_prc":"00"
},
{
"exch":"BSE",
"token":"506022",
"openbuyqty":"00",
"opensellqty":"00",
"openbuyamt":"00",
"opensellamt":"00",
"daybuyqty":"01",
"daysellqty":"00",

"daybuyamt":"6000",
"daysellamt":"00",
"cfbuyqty":"00",
"cfsellqty":"00",
"cfbuyamt":"00",
"cfsellamt":"00",
"upload_prc":"00"
}
]
}
]

Sample Failure Response :

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

Product Conversion

Request to be POSTed to uri : **/NorenWClientTP/ProductConversion**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json 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 :

Response data will be in json format with below fields.

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.

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"
}

Get Order Report

Request to be POSTed to uri : **/NorenWClientTP/GetOrderReport**

Request Details :

Parameter Name	**Possible value**	**Description**

jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
actid		Account Id
from_date	13-06-2022	From date [dd-mm-yyyy]
to_date	18-06-2022	To date [dd-mm-yyyy]
brkname		Login users broker ID

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	Order Report success or failure indication.
norenordno		Noren Order Number
kidid		Kid Id
status		Order status (New, Replaced, Complete, Rejected etc)
rpt		Report Type (fill/complete etc)
uid		Logged in User Id
actid		Account Id of logged in user
token		Contract token
cname		Company Name
symname		Symbol Name
instname		Instrument Name
exch	NSE, BSE,	Exchange

	NFO ...	
seg		Segment
optt		Option Type
isin		ISIN
ls		Lot Size
tsym		Trading symbol
ordersource	MOB / WEB / TT	Used to generate exchange info fields.
trantype		Transaction type
prctyp		Price Type
ret		Order retention type (DAY, EOS, IOC,...)
prd	H / M / ...	Product name
s_prdt_ali		Product display name
qty		Order Quantity
prc		Order Price
norentm		Noren time stamp
exchordid		Exchange Order Number
rejreason		Order rejection reason, if rejected
cancelqty		Canceled quantity for order which is in status cancelled.
brkname		Login users broker ID
brnchid		Branch Id
fillshares		Total Traded Quantity of this order
flprc		Fill Price
trgprc		Order trigger price
avgprc		Average trade price of total traded quantity

strprc		Strike Price
ti		Tick Size
prcftr_d		((GN / GD) * (PN/PD)) (actual value for calculations)
tsym2		Trading symbol of second leg
trantype2		Transaction type of second leg
tsym3		Trading symbol of third leg
trantype3		Transaction type of third leg

Sample Success Response :


```
[
	{
		"stat":"Ok",
		"norenordno":"22061700002476",
		"kidid":"1",
		"status":"COMPLETE",
		"rpt":"Fill",
		"uid":"NARESHDEL",
		"actid":"NTEST2",
		"token":"3435",
		"cname":"NULL",
		"symname":"BBOX",
		"instname":"EQ",
		"exch":"NSE",
		"seg":"EQT",
		"optt":"NULL",
		"isin":"NULL",
		"ls":"1",
		"tsym":"BBOX-EQ",
		"ordersource":"TT",
		"trantype":"B",
		"prctyp":"LMT",
		"ret":"DAY",
		"prd":"I",
		"qty":"10",
		"prc":"inf",
		"norentm":"17:08:09 17-06-2022",
		"exchordid":"81",
		"rejreason":"NULL",
		"cancelqty":"0",
		"brkname":"KSPL",
		"brnchid":"MANGALORE",
```

"fillshares":"10",
"flprc":"inf",
"trgprc":"-nan",
"avgprc":"inf",
"strprc":"-nan",
"ti":"inf",
"prcftr_d":"(0 / 0) * (0 / 0)",
"tsym2":"BBOX-EQ",
"trantype2":"NULL",
"tsym3":"BBOX-EQ",
"trantype3":"NULL"
}
]

Sample Failure Response :

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

Get Trade Report

Request to be POSTed to uri : **/NorenWClientTP/GetTradeReport**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
actid		Account Id
from_date	13-06-2022	From date [dd-mm-yyyy]
to_date	18-06-2022	To date [dd-mm-yyyy]
brkname		Login users broker ID

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	Order Report success or failure indication.
norenordno		Noren Order Number
kidid		Kid Id
status		Order status (New, Replaced, Complete, Rejected etc)
rpt		Report Type (fill/complete etc)
uid		Logged in User Id
actid		Account Id of logged in user
token		Contract token
cname		Company Name
symname		Symbol Name
instname		Instrument Name
exch	NSE, BSE, NFO ...	Exchange
seg		Segment
optt		Option Type
isin		ISIN
ls		Lot Size
tsym		Trading symbol
ordersource	MOB / WEB / TT	Used to generate exchange info fields.
trantype		Transaction type
prctyp		Price Type
ret		Order retention type (DAY, EOS, IOC,...)

prd	H / M / ...	Product name
s_prdt_ali		Product display name
qty		Order Quantity
prc		Order Price
norentm		Noren time stamp
exchordid		Exchange Order Number
rejreason		Order rejection reason, if rejected
cancelqty		Canceled quantity for order which is in status cancelled.
brkname		Login users broker ID
brnchid		Branch Id
fillshares		Total Traded Quantity of this order
flprc		Fill Price
trgprc		Order trigger price
avgprc		Average trade price of total traded quantity
strprc		Strike Price
ti		Tick Size
flqty		Fill Quantity
prcftr_d		((GN / GD) * (PN/PD)) (actual value for calculations)
tsym2		Trading symbol of second leg
trantype2		Transaction type of second leg
tsym3		Trading symbol of third leg
trantype3		Transaction type of third leg

Sample Success Response :

```
[
    {
```

"stat":"Ok",
"norenordno":"22061700002476",
"kidid":"1",
"status":"COMPLETE",
"rpt":"Fill",
"uid":"NARESHDEL",
"actid":"NTEST2",
"token":"3435",
"cname":"NULL",
"symname":"BBOX",
"instname":"EQ",
"exch":"NSE",
"seg":"EQT",
"optt":"NULL",
"isin":"NULL",
"ls":"1",
"tsym":"BBOX-EQ",
"ordersource":"TT",
"trantype":"B",
"prctyp":"LMT",
"ret":"DAY",
"prd":"I",
"qty":"10",
"prc":"inf",
"norentm":"17:08:09 17-06-2022",
"exchordid":"81",
"rejreason":"NULL",
"cancelqty":"0",
"brkname":"KSPL",
"brnchid":"MANGALORE",
"fillshares":"10",
"flprc":"inf",
"trgprc":"-nan",
"flqty":"10",
"avgprc":"inf",
"strprc":"-nan",
"ti":"inf",
"prcftr_d":"(0 / 0) * (0 / 0)",
"tsym2":"BBOX-EQ",
"trantype2":"NULL",
"tsym3":"BBOX-EQ",
"trantype3":"NULL"
}
]

Sample Failure Response :

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

Holdings and Limits

Holdings

Request to be POSTed to uri : **/NorenWClientTP/Holdings**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

Response Details :

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

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
hair_cut		Hair Cut
prd		Product
s_prdt_ali		Product display name

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

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

Request to be POSTed to uri : **/NorenWClientTP/Limits**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
actid*		Account id of the logged in user.
prd		Product name
s_prdt_ali		Product display name
seg	EQT/DER/FX/COM	Segment
exch		Exchange

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description

stat	Ok or Not_Ok	Limits request success or failure indication.
actid		Account id
prd		Product name
seg	EQT/DER/FX/ COM	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)
peak_mar		Peak margin used by the client
mr_eqt_u		MR equity used
mr_der_u		MR derivatives used
mr_fx_u		MR fx used
mr_com_u		MR commodity used
mr_sell		MR sell credit

mr_t1sell		MR t1 sell credit
mr_eqt_a		MR equity allocated
mr_der_a		MR derivatives allocated
mr_fx_a		MR fx allocated
mr_com_a		MR commodity allocated
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 :  "
}
```


Get Sub Limits

Request to be POSTed to uri : **/NorenWClientTP/GetSubLimits**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
actid*		Account id of the logged in user.

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

Json Fields	Possible value	Description
stat	Ok or Not_Ok	Limits request success or failure indication.
actid		Account id
prd		Product name
s_prdt_ali		Product display name
seg	EQT/DER/FX/ COM	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
------------------------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_		Additional scrip basket margin (Derivative Intraday)

d_i		
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)
peak_mar		Peak margin used by the client
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":"17:03:37 04-07-2022",
		"stat":"Ok",
		"prfname":"KAMBALA",
		"cash":"50000000.00",
		"daycash":"0.00",
		"blk_amt":"0.00",
		"unclearedcash":"0.00",
		"brkcollamt":"0.00",
		"payin":"320.00",
		"payout":"-24.00"
	},
	{
		"request_time":"17:03:37 04-07-2022",
		"stat":"Ok",
		"prfname":"DEFAULT",
		"cash":"213145.00",
		"daycash":"0.00",
		"blk_amt":"0.00",
		"unclearedcash":"0.00",
		"brkcollamt":"0.00",
		"payin":"320.00",
		"payout":"-24.00"
	},
	{
		"request_time":"17:03:37 04-07-2022",
		"stat":"Ok",
		"prfname":"DEFAULT",
		"cash":"12345.00",
		"daycash":"0.00",
		"blk_amt":"0.00",
		"unclearedcash":"0.00",
		"brkcollamt":"0.00",
		"payin":"320.00",
		"payout":"-24.00"

}
]

Sample Failure Response :

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

Get Max Payout Amount:

Request to be POSTed to uri : **/NorenWClientTP/GetMaxPayoutAmount**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User id of the logged in user.
actid*		Login users account ID
seg		EQT/FX/DER/COM can be sent if limits are managed at segregated level
exch		NSE/BSE/NFO… can be sent if limits are managed at exchange level
prd		C/I/M/F/H/B can be sent if limits are set at product level

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description

stat		success or failure indication.
request_time		This will be present only in a successful response.
actid		Account id
payout		Maximum payout amount

Sample Success Response :

{
"request_time":"15:52:26 10-05-2021",
"stat":"Ok",
"actid":"C-GURURAJ",
"payout":"21200.20"
}

Sample Failure Response :

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

Market Info

Get Index List

Request to be POSTed to uri : **/NorenWClientTP/GetIndexList**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id

exch*		Exchange

Response Details :

Response data will be in json format with below fields.

Json 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 :

Json 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"
```

```
        },
        {
            "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 Option Chain

Request to be POSTed to uri : **/NorenWClientTP/GetOptionChain**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json 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 :

Response data will be in json format with below fields.

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. That is : 1) Invalid Input 2) Session Expired

Json Fields of object in values	Possible value	Description

Array		
exch	CDS, 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

Get Linked Scrips

Request to be POSTed to uri : **/NorenWClientTP/GetLinkedScrips**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
token*		Trading symbol of any of the options 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)

Response Details :

Response data will be in json format with below fields.

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

equls **Json Fields of object in values Array**	**Possible value**	**Description**
exch	NSE, BSE ...	Exchange
tsym		Trading symbol of the scrip (contract)
token		Token of the scrip (contract)
pp		Price precision
ti		Tick size
ls		Lot size
mult		Contract price multiplier, (used for order value calculation)

fut **Json Fields of object in values Array**	**Possible value**	**Description**
exch	NFO, MCX ...	Exchange

tsym		Trading symbol of the scrip (contract)
token		Token of the scrip (contract)
exd		Expiry date
pp		Price precision
ti		Tick size
ls		Lot size
mult		Contract price multiplier, (used for order value calculation)

opt_exp **Json Fields of object in values Array**	**Possible value**	**Description (Used to show dropdown in option chain)**
exd		Used for calling option chain api
exch	NFO, MCX ...	Exchange
tsym		One of the random Trading symbol on given expiry, useful in calling option chain

Exch Msg

Request to be POSTed to uri : **/NorenWClientTP/ExchMsg**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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

		response)

Example:
curl https://apitest.kambala.co.in/NorenWClientTP/ExchMsg \
  -d "jData={\"uid\":\"VIDYA\"}“ \
  -d “jKey=NSE”

Response Details :

Response data will be in json format with below fields in case of success.

Json 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 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

Get Broker Msg

Request to be POSTed to uri : **/NorenWClientTP/GetBrokerMsg**
Request Details :

Parameter Name	**Possible value**	**Description**

jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id

Response Details :

Response data will be in json format with below fields in case of success.

Json Fields	Possible value	Description
stat	Ok	Broker Msg success or failure indication.
dmsg		This will be present only in case of success. Number of days to expiry will be present in same.
norentm		Noren Time

Sample Success Response :


```
[
    {
        "stat": "Ok",
        "norentm": "02-05-1975 08:48:52",
        "msgtyp": "Admin Message",
        "dmsg": "Test Msg All Message Recovery2"
    },
    {
        "stat": "Ok",
        "norentm": "02-05-1975 08:48:52",
        "msgtyp": "Admin Message",
        "dmsg": "Test Msg All Message Recovery2"
    }
]
```

Get Underlying Exch Token

Request to be POSTed to uri : **/NorenWClientTP/GetUnderlyingExchToken**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey		Key Obtained on login success.

Json Fields	Possible value	Description
uid		User id of the logged in user.
exch*		Exchange name
token*		Token

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		success or failure indication.
request_time		This will be present only in a successful response.
und_exch		Underlying Exch seg
und_tk		Underlying Token
emsg		This will be present only in case of errors.

Sample Success Response :

{
  "stat":"Ok",
  "request_time":"01122021110443",
  "und_exch":"NSE",
  "und_tk":"26009"
}

Sample Failure Response :

{
  "stat":"Not_Ok",
  "emsg":"Invalid Input : Invalid Scrip token/exch"
}

Exchange Status

Request to be POSTed to uri : **/NorenWClientTP/ExchStatus**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User id of the logged in user.

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		success or failure indication.
exch		Exchange Segment

exchstat		Exchange status
exchtype		Exch Type

Sample Success Response :


```
[
  {
     "stat": "Ok",
     "exch": "NSE",
     "exchstat": "OPEN",
     "exchtype": "NORMAL"
  },
  {
     "stat": "Ok",
     "exch": "NSE",
     "exchstat": "CLOSED",
     "exchtype": "AUCTION"
  }
]
```


Sample Failure Response :


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


System Info

Get AMO Status Flag

Request to be POSTed to uri : **/NorenWClientTP/AMOStatusFlag**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		Logged in User Id
exch*		Exchange

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	AMO status success or failure indication.
exch		Exchange
mode	STOP ACCEPT PUSH	AMO Mode

Sample Output:

{
"stat":"Ok",
"exch":"NSE",
"mode":"STOP"
}

Sample Failure Response :

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

Chart Data

Get Time Price Data

Request to be POSTed to uri : **/NorenWClientTP/TPSeries**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

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	“1”, ”3”, “5”, “10”, “15”, “30”, “60”, “120”, “240”	Candle size in minutes (optional field, if not given assume to be “1”)

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

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"
}

EOD Chart Data

Request to be POSTed to uri : **/NorenWClientTP/EODChartData**
Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey		Key Obtained on login success.

Json Fields	**Possible value**	**Description**
sym*		Symbol name
from*		From date
to*		To date

Example:
curl https://apitest.kambala.co.in/NorenWClientTP/EODChartData \
  -d "jData={"sym":"NSE:RELIANCE-EQ","from":1624838400,"to":1663718400}“

Response Details :

Response data will be in json format with below fields.

Json Fields	**Possible value**	**Description**
time		DD/MM/CCYY hh:mm:ss
into		Interval open
inth		Interval high
intl		Interval low
intc		Interval close
ssboe		Date,Seconds in 1970 format

intv		Interval volume

Sample Success Response :

[
"{
\"time\":\"21-SEP-2022\",
\"into\":\"2496.75\",
\"inth\":\"2533.00\",
\"intl\":\"2495.00\",
\"intc\":\"2509.75\",
\"ssboe\":\"1663718400\",
\"intv\":\"4249172.00\"
}",
"{
\"time\":\"15-SEP-2022\",
\"into\":\"2583.00\",
\"inth\":\"2603.55\",
\"intl\":\"2556.75\",
\"intc\":\"2562.70\",
\"ssboe\":\"1663200000\",
\"intv\":\"4783723.00\"
}",
"{ \"time\":\"28-JUN-2021\",
\"into\":\"2122.00\",
\"inth\":\"2126.50\",
\"intl\":\"2081.00\",
\"intc\":\"2086.00\",
\"ssboe\":\"1624838400\",
\"intv\":\"9357852.00\"
}"
]

Calculators

Span Calculator

Request to be POSTed to uri : **/NorenWClientTP/SpanCalc**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list

Json Fields	Possible value	Description
actid*		Any Account id, preferably actual account id if sending from post login screen
pos*		Array of json objects. (object fields given in below table)

Json Fields of object in values Array	Possible value	Description
prd	M / I	Product name
exch	NFO, CDS, MCX ...	Exchange
instname	FUTSTK, FUTIDX, OPTSTK, FUTCUR...	Instrument name
symname	USDINR, ACC, ABB, NIFTY..	Symbol name
exd	06-DEC-2022	DD-MON-YYYY format
optt	CE, PE	Option Type

strprc	11900.00, 71.0025	Strike price
buyqty		Buy Open Quantity
sellqty		Sell Open Quantity
netqty		Net traded quantity

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
stat	Ok or Not_Ok	Market watch success or failure indication.
span		Span value
expo		Exposure margin
span_trade		Span value ignoring input fields buyqty, sellqty
expo_trade		Exposure margin ignoring input fields buyqty, sellqty

Get Option Greek

Request to be POSTed to uri : **/NorenWClient/GetOptionGreek**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
exd*		Expiry Date
strprc*		Strike Price
sptprc*		Spot Price
int_rate*		Init Rate
volatility*		Volatility
optt		Option Type

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		success or failure indication.
request_time		This will be present only in a successful response.
cal_price		Cal Price
put_price		Put Price
cal_delta		Cal Delta
put_delta		Put Delta
cal_gamma		Cal Gamma
put_gamma		Put Gamma
cal_theta		Cal Theta
put_theta		Put Theta
cal_rho		Cal Rho
put_rho		Put Rho
cal_vego		Cal Vego

put_vego		Put Vego

Sample Success Response :

{
"request_time":"17:22:58 28-07-2021",
"stat":"OK",
"cal_price":"1441",
"put_price":"0.417071",
"cal_delta":"0.997304",
"put_delta":"-0.002696",
"cal_gamma":"0.000001",
"put_gamma":"0.000001",
"cal_theta":"-31.535015",
"put_theta":"-31.401346",
"cal_rho":"0.000119",
"put_rho":"-0.016590",
"cal_vego":"0.006307",
put_vego":"0.006307"
}

Sample Failure Response :

{
"stat":"Not_Ok",
"emsg":"Invalid Input : jData is Missing."
}

Brokerage Calculator

Request to be POSTed to uri : **/NorenWClientTP/`GetBrokerage`**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list

Json Fields	Possible value	Description

uid*		User Id
actid*		Account Id
exch*		Exchange Segment
tsym*		Trading Symbol
qty*		Quantity
prc*		Price
prd*		Product
trantype*		Transaction Type

Response Details :

Response data will be in json format with below fields.

Json Fields	Possible value	Description
request_time		This will be present only in a successful response.
stat	Ok or Not_Ok	success or failure message.
brkage_amt		Brokerage Amount
stt_amt		STT/CTT Amount
exch_chrg		Exchange Charges
sebi_chrg		SEBI Charges
stamp_duty		Stamp Charges
clr_chrg		Clearing Charges
gst		GST
tot_chrg		Total charges
remarks		Remarks

url		Url

Sample Success Response :

{
request_time: 11:36:43 27-12-2022,
stat: Ok,
brkage_amt: 0.00,
stt_amt: 2.00,
exch_chrg: 0.09,
sebi_chrg: 0.01,
stamp_duty: 0.37,
clr_chrg: 0.00,
gst: 0.02,
tot_chrg: 2.49,
remarks: 2,
url: matsya.kambala.co.in
}

Sample Failure Response :

{
"stat":"Not_Ok",
"emsg":"Invalid Input : jData is Missing."
}

Alerts

Set Alert

Request to be POSTed to uri : **/NorenWClientTP/SetAlert**

Request Details :

Parameter Name	**Possible value**	**Description**
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User id of the logged in user.
tsym*		Trading symbol
exch*		Exchange Segment
ai_t*		Alert Type
validity*	DAY or GTT	Validity
d*		Data to be compared with LTP
remarks		Any message Entered during order entry.
sms_flag	true or false	SMS flag
email_flag	true or false	Email flag
push_flag	true or false	Push flag

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		alert success or failure indication.
request_time		This will be present only in a successful response.
al_id		Alert Id
emsg		This will be present only in case of errors. That is : 1) Invalid Input 2) Session Expired

Sample Success Response :

{
"request_time":"11:22:26 08-04-2021",
"stat":"Oi created",
“al_id”:”21040800000004”
}

Sample Failure Response :

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

Cancel Alert

Request to be POSTed to uri : **/NorenWClientTP/CancelAlert**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid		User id of the logged in user.
al_id*		Alert Id

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		alert success or failure indication.
request_time		This will be present only in a successful response.
al_id		Alert Id
emsg		This will be present only in case of errors. That is : 1) Invalid Input

		2) Session Expired

Sample Success Response :

{
"request_time":"15:03:33 08-04-2021",
"stat":"Oi delete success",
“al_id”:”21040800000008”
}

Sample Failure Response :

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

Modify Alert

Request to be POSTed to uri : **/NorenWClientTP/ModifyAlert**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User id of the logged in user.
tsym*		Trading symbol
exch*		Exchange Segment
ai_t*		Alert Type, should be original alert type, can’t be modified.
al_id*		Alert Id
validity*	DAY or GTT	Validity

d		Data to be compared with LTP
remarks		Any message Entered during order entry.
sms_flag	true or false	SMS flag
email_flag	true or false	Email flag
push_flag	true or false	Push flag

Response Details :

Response data will have below fields.

Json Fields	**Possible value**	**Description**
stat		alert success or failure indication.
request_time		This will be present only in a successful response.
al_id		Alert Id
emsg		This will be present only in case of errors. That is : 1) Invalid Input 2) Session Expired

Sample Success Response :

{
"request_time":"16:36:42 08-04-2021",
"stat":"Oi Replaced",
“al_id”:”21040800000013”
}

Sample Failure Response :

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

Get Pending Alert

Request to be POSTed to uri : **/NorenWClientTP/GetPendingAlert**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list
jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User id of the logged in user.

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		alert success or failure indication.
ai_t		Alert type
al_id		Alert Id
tsym		Trading symbol
exch		Exchange Segment
token		Contract token
remarks		Any message Entered during order entry.
validity	DAY or GTT	Validity
d		Data to be compared with LTP

sms_flag	true or false	SMS flag
email_flag	true or false	Email flag
push_flag	true or false	Push flag
emsg		This will be present only in case of errors. That is : 1) Invalid Input 2) Session Expired

Sample Success Response :

[
{
"Stat":"ok",
“ai_t”:”LTP_A”,
“al_id”:”21040800000008”,
“tsym”:”ACC-EQ”,
“exch”:”NSE”
“token”:”22”,
“remarks”:”test”,
“validity”:”DAY”,
“d”:”95000.00”
}
]

Sample Failure Response :

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

Get Enabled Alert Types

Request to be POSTed to uri : **/NorenWClientTP/GetEnabledAlertTypes**

Request Details :

Parameter Name	Possible value	Description
jData*		Should send json object with fields in below list

jKey*		Key Obtained on login success.

Json Fields	Possible value	Description
uid*		User id of the logged in user.

Response Details :

Response data will have below fields.

Json Fields	Possible value	Description
stat		Alert order success or failure indication.
request_time		This will be present only in a successful response.
ai_ts		Array of alert types

Sample Success Response :


```
{
        "stat":"Ok",
        "request_time":"04062021121503",
        "ai_ts":
        [
         {"ai_t":"ATP"},
         {"ai_t":"LTP"},
         {"ai_t":"Perc. Change"}
        ]
}
```


Sample Failure Response :


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

Web Socket API

Connect to wss://api.broker1.com/NorenStream/NorenWS

General Guidelines

1) As soon as connection is done, a connection request should be sent with User id and login session id.
2) All input and output messages will be in json format.
3) ATO price is sent as "42949672.95"

Connect

Request:

Json Fields	Possible value	Description
t	c	'c' represents connect task
uid		User ID
actid		Account id
source	WEB / MOB	Source should be same as login request.
susertoken		User Session Token

Response :

Json Fields	Possible value	Description
t	ck	'ck' represents connect acknowledgement
uid		User ID
s		Ok or Not_Ok(in case of invalid user id or session id)

Subscribe Touchline

Request :

Json Fields	Possible value	Description
t	t	't' represents touchline task
k		One or more scriplist for subscription. Example NSE|22#BSE|508123#NSE|NIFTY

Subscription Acknowledgement:

Number of Acknowledgements for a single subscription will be the same as the number of scrips mentioned in the key (k) field.

Json Fields	Possible value	Description
t	tk	'tk' represents touchline acknowledgement
e	NSE, BSE, NFO ..	Exchange name
tk	22	Scrip Token
pp	2 for NSE, BSE 4 for CDS USDINR	Price precision
ts		Trading Symbol
ti		Tick size
ls		Lot size
lp		LTP
pc		Percentage change
v		volume
o		Open price
h		High price
l		Low price

c		Close price
ap		Average trade price
oi		Open interest
poi		Previous day closing Open Interest
toi		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
ft		Feed time

TouchLine subscription Updates :
Accept for t, e, and tk other fields may / may not be present.

Json 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
oi		Open interest

poi		Previous day closing Open Interest
toi		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
ft		Feed time

Unsubscribe Touchline

Request:

Json Fields	Possible value	Description
t	u	'u' represents Unsubscribe Touchline
k		One or more scriplist for unsubscription. Example NSE|22#BSE|508123

Response :

Json Fields	Possible value	Description
t	uk	'uk' represents Unsubscribe Touchline acknowledgement
k		One or more scriplist for unsubscription. Example NSE|22#BSE|508123

Subscribe Depth

Request :

Json Fields	Possible value	Description

t	d	‘d’ represents depth subscription
k		One or more scriplist for subscription. Example NSE|22#BSE|508123

Subscription Depth Acknowledgement:

Number of Acknowledgements for a single subscription will be the same as the number of scrips mentioned in the key (k) field.

Json Fields	**Possible value**	**Description**
t	dk	‘dk’ represents depth acknowledgement
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
oi		Open interest
poi		Previous day closing Open Interest
toi		Total open interest for underlying
ft		Feed time

Depth subscription Updates :

Json 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
oi		Open interest
poi		Previous day closing Open Interest
toi		Total open interest for underlying
ft		Feed time

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

Request:

Json Fields	Possible value	Description
t	ud	‘ud’ represents Unsubscribe depth
k		One or more scriplist for unsubscription. Example NSE|22#BSE|508123

Response :

Json Fields	Possible value	Description
t	udk	‘udk’ represents unsubscribe depth acknowledgement
k		One or more scriplist for unsubscription. Example

		NSE|22#BSE|508123

Subscribe Order Update

Request :

Json Fields	Possible value	Description
t	o	‘o’ represents order update subscription task
actid		Account id based on which order updated to be sent.

Subscription Acknowledgement:

Json Fields	Possible value	Description
t	ok	‘ok’ represents order update subscription acknowledgement

Order Update subscription Updates :

Json 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 (New, Replaced, 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		Total Filled shares for this order
avgprc		Average fill price
fltm		Fill Time(present only when reporttype is Fill)
flid		Fill ID (present only when reporttype is Fill)
flqty		Fill Qty(present only when reporttype is Fill)
flprc		Fill Price(present only when reporttype is Fill)
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
snonum		This will be present for child orders in case of cover and bracket orders, if present needs to be sent during exit
snoordt		This will be present for child orders in case of cover and bracket orders, it will indicate whether the order is profit or stoploss
blprc		This will be present for cover and bracket parent order. This is the differential stop loss trigger price to be entered.
bpprc		This will be present for bracket parent order. This is the differential profit price to be entered.
trailprc		This will be present for cover and bracket parent order. This is required if trailing ticks is to be enabled.
exch_tm		This will have the exchange update time Format: dd-mm-YYYY hh:MM:SS

amo		This field will be present if the order is After Market Order. Data will be "Yes"
tm		TimeStamp
kidid		Kid Id
sno_fillid		BO Sequence Id

Unsubscribe Order Update

Request:

Json Fields	Possible value	Description
t	uo	'uo' represents Unsubscribe Order update

Response :

Json Fields	Possible value	Description
t	uok	'uok' represents Unsubscribe Order update acknowledgement

Annexure

Alert Type:

Alert Criteria	Condition	Alert type	Transformation and data validations
LTP	>	LTP_A	depending on scrip 'pp' from search results allow 2/4 precision
LTP	<	LTP_B	depending on scrip 'pp' from search results allow 2/4 precision

Change %	>	CH_PER_A	Upto 2 decimals allowed
Change %	<	CH_PER_B	Upto 2 decimals allowed
Average Trade price of day	>	ATP_A	depending on scrip 'pp' from search results allow 2/4 precision
Average Trade price of day	<	ATP_B	depending on scrip 'pp' from search results allow 2/4 precision
LTP vs 52week high	>	LTP_A_52HIGH	No input data
LTP vs 52week low	<	LTP_B_52LOW	No input data
Volume	>	VOLUME_A	Non decimal number
Open Interest	>	OI_A	Non decimal number, allow only for derivative contracts
Open Interest	<	OI_B	Non decimal number, allow only for derivative contracts
Total Open Interest	>	TOI_A	Non decimal number, this will work only for NSE symbols which are FO listed.
Total Open Interest	<	TOI_B	Non decimal number, this will work only for NSE symbols which are FO listed.
LTP	Both > and <	LMT_BOS_O	depending on scrip 'pp' from search results allow 2/4 precision

Note: All alert types with _O appended will work for GTT order types. Example: to set GTT order when LTP goes above 1,000, set alert type as LTP_A_O

Report Type:

Possible Values
NewAck

ModAck
CanAck
PendingNew
PendingReplace
PendingCancel
New
Replaced
Canceled
Fill
Rejected
ReplaceRejected
CancelRejected
INVALID_REPORT_TYPE

Status Type:

Possible Values
PENDING
CANCELED
OPEN
REJECTED
COMPLETE
TRIGGER_PENDING
INVALID_STATUS_TYPE

Internal Status Type:

Possible Values
COMPLETE
PARTIAL FILL
REJECTED
CANCELED
MODIFY PENDING
CANCEL PENDING
ORDER PENDING
OPEN
ORDER ACK
MODIFY ACK
CANCEL ACK
TRIGGER_PENDING
AMO OPEN
AMO MODIFIED
AMO CANCELED