

National Stock Exchange of India Limited

Circular

DEPARTMENT: INSPECTION	
Download Ref No: NSE/INSP/55380	Date: January 25, 2023
Circular Ref. No: 08/2023	

To All Trading Members,

Sub: Application Programming Interface (API) for daily submission of Holding Statement and Bank Balances

This has reference to Exchange Circular Ref Nos. NSE/INSP/55039 dated December 28, 2022 & NSE/INSP/55250 dated January 13, 2023 regarding revision in frequency of submission of Holding Statement and Bank Balances from weekly to daily through Application Programming Interface (API), which shall be made applicable from day ended January 30, 2023 and onwards, the due date of submission will be January 31, 2023, and for each day thereafter.

Accordingly, please take note that Members shall be mandatorily required to report daily submissions of Holding Statement and Bank Balances through API from day ended January 30, 2023 and onwards, on T+1 basis.

The current functionality of weekly submission of Holding Statement through member portal and Bank Balances through API shall be available only to make the submission till week ending January 28, 2023 (due date for which is February 02, 2023), and the said weekly functionality will not be available subsequently.

Members are requested to find API specification document enclosed as **Annexure A** and user manual for daily submissions of Holding Statement and Bank Balances through API system enclosed as **Annexure B**.

All Members are advised to take note of the above and comply.

For any technical assistance, please reach at DL-IT-API@nse.co.in

For and on behalf of
National Stock Exchange of India Limited

Ajinkya Nikam
Senior Manager

National Stock Exchange of India Limited

In case of any clarifications, Members may contact our below offices:

Regional Office	E MAIL ID	CONTACT NO.
Ahmedabad (ARO)	inspectionahm@nse.co.in	079- 49008632
Chennai (CRO)	inspection_cro@nse.co.in	044- 66309915 / 17
Delhi (DRO)	delhi_inspection@nse.co.in	011- 23459127 / 38 / 46
Kolkata (KRO)	inspection_kolkata@nse.co.in	033-4040 0412/59
Mumbai (WRO)	compliance_wro@nse.co.in	Board Line: 022-25045000 / 022-61928200 Direct Line: 022-25045138 / 022-25045144 Extn: 28144/28138
Central Help Desk	compliance_assistance@nse.co.in	

API Specifications

Member Holding Statement and Bank Account Balances Daily Submissions

Version 2.0



National Stock Exchange of India Ltd
Exchange Plaza, Plot No. C/1, G Block,
Bandra-Kurla Complex, Bandra (E), Mumbai
- 400 051.

Table of Contents

Contents

Contents

Table of Contents.....	1
Preface	2
<i>Purpose</i>	2
<i>Target Audience</i>	2
<i>Organization of This Document</i>	2
Introduction.....	3
<i>Authorization</i>	3
<i>Message Format</i>	3
<i>Exchange APIs:</i>	4
Loginssss	4
Logout.....	5
Process for Submission of Holding Statements	6
Process for Submission of Bank Account Balances.....	12
Standard HTTP response codes	15
Master codes	15

Preface

Purpose

This document describes the Web API based communication and messaging format between the Exchange (NSE) and Trading Members.

Target Audience

- Business and Tech teams at Trading Member
- Business and Tech teams at Exchange (NSE)

Organization of This Document

This document is organized as follows:

Chapters	Description
Introduction	Overview of the document covering security features of the API and online messaging guidelines.
Exchange API	This chapter gives details of the API's exposed by exchanges. These API's will be consumed by Trading Members.

Introduction

REST API endpoints accessible over internet using HTTPS protocol will be provided to Trading Members to push the required Holding Statement and Bank Account Balance data on daily basis.

Authorization

- All the TM's will process user registration on existing inspection system.
- Member Portal Registration:
 - <https://inspection.nseindia.com/MemberPortal/>
 - User will sign-up and receive registration confirmation.
 - [Please refer user User manual for API registration and data submission through API on Production](#)
- On Successful login, Token will be provided to Consumers.
- The API consumer application is expected to send Token in header in every API HTTP request.
- **Token:** A unique identifier generated for all successful login.

For e.g.,

Header Key	Header Value
Authorization	Bearer <Token received from Login API>

- Token will have expiry of 30 minutes.
- JWT mechanism will be used to generate tokens as it is widely used industry standard.

Message Format

- Messages will be in JSON.
- Following headers need to be provided in all API calls.
 - **Content-Type:** Header value should be "application/json" or "multipart/form-data" as specified in API details.
- All request and response messages are in JSON (JavaScript Object Notation) format. (Refer <http://www.json.org/> for details).
- Some of the key specifications related to JSON and standards followed for the API's are as follows
 - JSON is built on 2 structures. Object or Map containing key value pairs and an ordered list of values
 - A value could be Boolean (true/false), number, decimal, String or a structure (List or Object)
 - Object or key value pair structure consists of keys which are strings and values of any of the above types. E.g. {"ClientCode": "Test1", "PanNo": "ABCDE1234F"}
 - List contains list of Clients. E.g. ["Test1", "Test2", "Test3"]
 - A Boolean has only 2 values true or false
 - String values are enclosed in single quote or double quotes. e.g. "ClientCode", "PanNo"
 - Numbers and decimals are represented without any thousand - separator character
Decimal indicator is dot (".")
 - All dates, times and date times are represented as strings and in Indian standard time. Dates are formatted using format "DD-MM-YYYY". Time are formatted as "hh24:mm:ss". Date times are formatted as "DD-MM-YYYY hh24:mm:ss"
- Request Payload will be compressed by gzip

Exchange APIs:

Following are the API endpoints for integrations between Exchange and Trading Members:

Method	API EndPoint	Remarks
POST	https://api.nseindia.com/inspectionapi/login	To Login and generate Session Token
POST	https://api.nseindia.com/inspectionapi/logout	To logout
POST	https://api.nseindia.com/inspectionapi/tradingHoldingUpload/{version}	To submit-holding-statement
POST	https://api.nseindia.com /inspectionapi/tradingEodBalanceUpload/{version}	Member to submit their Bank account balance

Login

This API validates the user credentials. On successful login, a unique token will be provided per request and session will be established with user's system. In subsequent API calls, this token should be passed in HTTP header "Authorization" as mentioned in API Help Document.

Note:

- The password needs to be sent in encrypted format using AES256. The encryption secret key will be generated on user registration.
- Login session validity will be for 30 mins.

Method	API EndPoint	Header	Content-Type
POST	https://api.nseindia.com /inspectionapi/login	NA	application/json

Request JSON values

Sr. No	Field Description	Field Name	Data Type	Description	Expected Values & Validations	Optional /Mandatory
1	Member Code	memberCode	Char(5)	Member Code	Member Code	M
2	Login Id	loginId	Char(16)	Login ID	Login Id	M
3	Password	password	Char(50)	Password	AES encrypted password	M

Request Object Structure (JSON)

```
{
  "memberCode": "00000",
  "loginId": "ApiUser00000",
  "password": "0CxUgThnqlK6gaKU2HnL2A=="
}
```

Password in example is AES encrypted text

Response JSON values

Sr. No.	Field Description	Field Name	Data Type	Description
1	Member Code	memberCode	Char(5)	Member Code
2	Response Code	code	Char(3)	Response Code
3	Login Id	loginId	Char(16)	Login ID
4	Token	token	String	Unique Token for further API's call
5	Status	status	String	Request Status

Response Object Structure (JSON)

```
{
  "memberCode": "00000",
  "code": ["601"],
  "loginId": "ApiUser00000",
  "token": "eyJhbGciOiJIUzU1NiJ9.eyJzdWIiOiIwOTFwMCIsImV4cCI6MTYzZmJwNzYyOCwiaWF0IjoxNjMyMjA0MDI4LCJqdGkiOiJhNWl0YjZiZi1lZTQwLTQ4NzItOGM4ZC0xNjE5ZjA2Nzk2NTEifQ.gGmBfpIbtP59X6gK28z4l6l10kFgOBOIGjAX-WhjkY5DPJRxHJWtc-TOcDyPjLGAvnHNIvh-SZRRawkxFSNgYw",
  "status": "success"
}
```

Logout

This API closes the current session and gets the user logged off.

Method	API EndPoint	Header	Content-Type
POST	https://api.nseindia.com /inspectionapi/logout	Authorization	application/json

Request JSON values

Sr. No.	Field Description	Field Name	Data Type	Description
1	Member Code	memberCode	Char(5)	Member Code
2	Login Id	loginId	Char(16)	Login ID

Request Object Structure (JSON)

```
{
  "memberCode": "00000",
  "loginId": "ApiUser00000"
}
```

Response JSON values

Sr. No.	Field Description	Field Name	Data Type	Description
1	Response Code	code	Char(3)	Response Code
2	Status	status	String	Request Status

Response Object Structure (JSON)

```
{
  "code": ["601"],
  "status": "success"
}
```

Process for Submission of Holding Statements

1. TM has to submit the Holding Statement on Daily basis.
2. Nomenclature should be in this format PAN_HS_DDMMYYYY_Seqno.gz
3. Date in file name should be the date for which member is submitting data. for e.g., if member is submitting data for 01-01-2023 then file name will be PAN_HS_01012022_01.gz
4. In REST API ,file will be submitted in multipart/form-data.
5. System will unzip the gzip file received and process data validations.
6. Member will specify filePeriodDate and date in dd-mm-yyyy which will be the date for which member wants to upload Holding statements in JSON Body Parameter i.e date in “filePeriodDate”, ”date” json keys and in file name must be same.
7. System will not accept data for T Day if TM has not submitted the data for T-1 day. TM needs to first submit T-1 day data and then can submit data for T Day i.e. Data for Wednesday cannot be submitted until Monday and Tuesday's data is submitted.
8. If there is any error then error file will generate else success file will be generated.
9. System will provide response in JSON with gZip having record status with code in case of errors or incorrect: date, data type (Member PAN, Date, Member's Demat Account No., Client PAN, ISIN Code), blank fields and duplicate records and store data in csv format in Member folder on server.
10. The submission for Friday and Saturday has to be made on Monday. Further, incase of holiday the submission has to be made on next trading day, e.g. if there is holiday on Monday, the submission has to be made for Friday; Saturday and Monday on Tuesday.
11. TM has to correct and resubmit entire data to Exchange.

Method	API EndPoint	Header	Content-Type
POST	https://api.nseindia.com /inspectionapi/tradingHoldingUpload/{version}	Authorization	multipart/form-data

Request JSON values

Sr. No	Field Description	Field Name	Data Type	Description	Expected Values and Validation	Type of Field
1	Member PAN	memberPan	Char(10)	Member PAN no	Basic PAN level Validations	Header field. Only once per request
2	File Period Date	filePeriodDate	Char(10)	Date for which data is being submitted	DD-MM-YYYY	Header field. Only once per request
3	Date	date	Char(10)	Date for which data is being submitted	DD-MM-YYYY	Header field. Only once per request
4	Holding Data	holdingData	String	Request Holding Data Array		Holding Data Array
4.1	Record Id	recId	String	Record id of referred data	Record Id Provided in request	Details repeated for every row
4.2	Member's Demat Account No.	dmat	Char(16)	Member DMAT Account Number	Max length 16	DMAT number assigned for every row of respective client
4.3	Member Account Type	accountType	Char(12)	Member Account Type	POOL, CM COLL, OWN, EPI, CLNTBENE-MTF, TMCOLL, TMCOLL-MTF, CLNT UNPAID, CLNT COLL, MRGN PLEDGE, MTF PLEDGE, CLNTBENE	Details repeated for every row
4.4	Unique Client Code (UCC)	ucc	Char(20)	Unique Client Code	Special characters other than underscore should not be accept. Accept underscore at the 3rd position	Details repeated for every row
4.5	Client Name	clientName	Char(100)	Client Name	Max length accepted will be 100 characters	Details repeated for every row
4.6	Client PAN	pan	Char(10)	Member PAN	Basic PAN level Validations	Details repeated for every row

4.7	ISIN Code	isin	Char(12)	Security ISIN	Alpha Numeric ISIN Code NA	Details repeated for every row
4.8	Security / Commodity Type	securityType	Char(4)	Security Type	EQ, MF, BOND, DEBT,PREF, IDR, WT,COMM	Details repeated for every row
4.9	Name of Commodities	nameOfCommodity	Char(100)	Name of Security / Commodities	Alphanumeric values, special characters	Details repeated for every row
4.10	Unit Type	unitType	Char(20)	Unit Type	GMS,KG,QUINT AL, MT,BARRELS,M MBTU BALES,DRUMS, OTHERS.NA	Details repeated for every row
4.11	Pledged Balance (Qty.)	totalPldgQty	Char(20)	Total Pledged Qty	Quantity of securities in Numeric form. Decimals shall be allowed upto 3 digits	Details repeated for every row
4.12	Free Balance (Qty.)	freeBalQty	Char(20)	Free Balance Qty	Quantity of securities in Numeric form. Decimals shall be allowed upto 3 digits	Details repeated for every row
4.13	Total (Qty.)	totalQty	Char(20)	Total Qty	Quantity of securities in Numeric form. Decimals shall be allowed upto 3 digits	Details repeated for every row

```
{
  "memberPan": " XXXXX0782X ",
  "filePeriodDate": "26-03-2022",
  "date": "26-03-2022",
  "holdingData": [
    {
      "recId": "1",
      "dmat": "0000000000000000",
      "accountType": "POOL",
      "ucc": "271847",
      "clientName": "PRAVIN CHANDRA RATILAL MEHTA",
      "pan": " XXXXX0782X ",
      "isin": "XX0020200286",
      "securityType": "BOND",
      "nameOfCommodity": "NA",
      "unitType": "NA",
      "totalPldgQty": "0",
      "freeBalQty": "1",
      "totalQty": "1"
    },
    {
      "recId": "2",
      "dmat": "0000000000000000",
      "accountType": "POOL",
      "ucc": "271848",
      "clientName": "NAVIN CHANDRA",
      "pan": " XXXXX0782X ",
      "isin": "IN0020200286",
      "securityType": "BOND",
      "nameOfCommodity": "NA",
      "unitType": "NA",
      "totalPldgQty": "0",
      "freeBalQty": "1",
      "totalQty": "1"
    }
  ]
}
```

Response JSON values

Sr.No	Field Description	Field Name	Data Type	Description	Expected Values and Validation	Type of Field
1	Header Code	hdrCode	Num(4)	Header Code	Overall Response Code	Header field. Only once per request
2	Acknowledge ID	ackId	Num(15)	Acknowledge ID	Unique ID	Header field. Only once per request
3	Member PAN	memberPan	Char(10)	Member PAN no	Basic PAN level Validations	Header field. Only once per request
4	Header Status	hdrStatus	String	Request Status	Response Status	Header field. Only once per request
5	Date	date	Char(10)	Date for which data is being submitted	DD-MM-YYYY	Header field. Only once per request
6	Holding Data	holdingData	String	Response Holding Data Array		Response Array
6.1	Record Id	recId	String	Record id of referred data	Record Id Provided in request	Details repeated for every row
6.2	Response Code	code	String	Master code for referred data	Record Specific Response Code	Details repeated for every row
6.3	Status	status	String	Success or failure of referred data	Record Specific Response Status	Details repeated for every row
7	File Period Date	filePeriodDate	Char(10)	Date for which data is being submitted	DD-MM-YYYY	Only once per request

JSON Failure Response:

Below will be the response in case of file/date related validations fails. (Ex: Invalid file name or format OR previous period data not found)

```
{
  "hdrCode": [ "805" ],
  "ackId": "1133525032022213862333",
  "memberPan": " XXXXX0782X ",
  "hdrStatus": "Kindly first submit data for 31-Mar-2022",
  "date": "01-04-2022",
  "holdingData": [
    {
      "unitType": "NA",
      "dmat": "0000000000000000",
      "ucc": "251868",
      "freeBalQty": "1",
      "clientName": "XYZ",
      "totalQty": "1",
      "securityType": "PREF",
      "accountType": " POOL",
      "totalPldgQty": "0",
      "pan": "XXXXXX0782X",
      "isin": "XX0000000000",
      "nameOfCommodity": "NA"
    }
  ],
  "filePeriodDate": "01-04-2022"
}
```

JSON Success Response:

Below will be the response in case of successful submission.

```
{
  "hdrCode": [ "601" ],
  "ackId": "1133525032022213862333",
  "memberPan": " XXXXX0782X ",
  "hdrStatus": "success",
  "date": "25-03-2022",
  "holdingData": [
    {
      "recId": "1",
      "code": [ "601" ],
      "status": "success"
    },
    {
      "recId": "2",
      "code": [ "601" ],
      "status": "success"
    }
  ],
  "filePeriodDate": "25-03-2022"
}
```

JSON Failure Response:

Below will be the response in case of record validation failure.

```
{
  "hdrCode": [ "603" ],
  "ackId": "C",
  "memberPan": "XXXXXX3359X",
  "hdrStatus": "partial",
  "date": "01-04-2022",
  "holdingData": [
    {
      "code": ["601"],
      "status": "success"
    },
    {
      "code": [ "906", "902", "907"],
      "status": "error"
    },
    {
      "code": ["902"],
      "status": "error"
    }
  ],
  "filePeriodDate": "01-04-2022"
}
```

Process for Submission of Bank Account Balances

1. TM has to submit Bank Account Balances on Daily basis.
2. In REST API ,file will be submitted in multipart/form-data .
3. TM will submit Bank Account Balances as prescribed by Exchange using API.
4. Nomenclature must include PAN_BA_DDMMYYYY.gz.
5. Date in file name should be the date for which member is submitting data. for eg., if member is submitting data for 01-01-2023 then file name will be PAN_BA_01012023.gz and Balance Date in json must be same as date in file name
6. System will not accept data for T Day if TM has not submitted the data for T-1 day. TM needs to first submit T-1 day data and then can submit data for T Day i.e. Data for Wednesday cannot be submitted until Monday and Tuesday's data is submitted.
7. System will reject pushed data and provide response with errors in case of incorrect date, data type, blank fields or duplicate records.
8. On submission, system will provide errors in response in-case of failure. TM has to correct and resubmit the data to Exchange. In case of success, system will provide success JSON response.
9. The submission for Friday and Saturday has to be made on Monday. Further, incase of holiday the submission has to be made on next trading day, e.g. if there is holiday on Monday, the submission has to be made for Friday; Saturday and Monday on Tuesday.

10. Files will be Store on Server for both success and failure condition.

Method	API EndPoint	Header	Content-Type
POST	https://api.nseindia.com /inspectionapi/tradingEodBalanceUpload/{version}	Authorization	multipart/form-data

Request JSON values

Sr. No.	Field Description	Field Name	Data Type	Description	Expected Values & Validations	Opt/ Mandate
Body Parameters						
1	Bank Balance Data	BankBalanceData	Array	Array of Bank Balance Data		M
1.1	Bank Account No.	Bank Account No	Number	Bank Account Number	Max Length :20	M
1.2	IFSC	IFSC	String	IFSC Code	Basic IFSC level Validations	M
1.3	Bank Account Type	Bank Account Type	String	Bank Account Type	Max Length :20	M
1.4	Balance Date		String	Balance Amount	Amount	M

Request Object Structure (JSON)

```
{
  "BankBalanceData": [{
    "Bank Account No": "00000000000000",
    "IFSC": "XXXX00000000",
    "Bank Account Type": "ANY OTHER",
    "03-05-2021": "102007496"
  },
  {
    "Bank Account No": "00000000000000",
    "IFSC": "XXXX00000000",
    "Bank Account Type": "SETTLEMENT ACCOUNT",
    "03-05-2021": "196163955"
  }
]
```

Response JSON values

Sr. No.	Field Description	Field Name	Data Type	Description	Expected Values & Validations
1.1	Bank Account No.	Bank Account No	Number	Bank Account Number	Max Length :20
1.2	IFSC	IFSC	String	IFSC Code	Basic IFSC level Validations
1.3	Bank Account Type	Bank Account Type	String	Bank Account Type	Max Length :20
1.4	Balance Date		String	Balance Amount	Amount
1.7	Status	Status	String	Success or failure with reason	

Response Object Structure (JSON)

JSON Failure Response

Below will be the response in case if file/date related validations fails. (Ex: Invalid file name or format OR previous period data not found)

```
{"Status":"error","Code":"Kindly first submit data for earlier period"}
```

JSON Success Response :

Below will be the response in case of successful submission.

```
[ {
  "Bank Account No" : "0000000000000000", "IFSC" : "XXXXX00000000",
  "Bank Account Type" : "ANY OTHER", "03-05-2021" : "102007496",
  "Status" : "Success"
}, {
  "Bank Account No" : "0000000000000000", "IFSC" : "XXXXX00000000",
  "Bank Account Type" : "SETTLEMENT ACCOUNT", "03-05-2021" : "196163955",
  "04-05-2021" : "275308037",
  "Status" : "Success"
} ]
```

JSON Partial Success Response

Below will be the response in case of record validation failure.

```
[ {
  "Bank Account No" : "0000000000000000",
  "IFSC" : "XXXXX000000060",
  "Bank Account Type" : "CLIENT BANK ACCOUNT",
  "26-04-2022" : "1.8133711E7",
  "Status" : "Invalid amount at column :4; "
}, {
  "Bank Account No" : "0000000000000000",
  "IFSC" : "XXXXX00000006",
  "Bank Account Type" : "CLIENT BANK ACCOUNT",
  "26-04-2022" : "8133711.0",
  "Status" : "Success"
} ]
```

Standard HTTP response codes

Following are the standard HTTP Response codes

HTTP Code	Description
200	Success
400	Bad Request
401	Unauthorized
404	Not Found
409	Conflict
412	Precondition Failed
413	File size greater than 200 MB
500	Internal Server Error
501	Not Implemented
503	Service Unavailable

Master codes

Code	Description
601	SUCCESS
602	GENERAL ERROR
603	PARTIAL_SUBMISSION
604	ENTIRE_HOLDINGDATA_IS_INCORRECT
605	SOMETHING_WENT_WRONG
606	METHOD_UNKOWN_CD
608	PREVIOUS_REQUEST_IS_IN_PROGRESS
701	INVALID_MEMBER_CODE_OR_LOGIN_ID_OR_PASSWORD
702	USER_IS_DISABLED
703	INVALID_MEMBER_CODE_OR_LOGIN_ID
704	INVALID_DECLARATION
705	JSON_KEY_NOT_PROPER
706	ERROR
803	KINDLY_CHECK_AUTH_TOKEN_TOKEN_WRONG_OR_TOKEN_EXPIRED
804	REQUEST_EXCEEDS_MAX_NUMBER_OF_RECORDS
805	KINDLY_FIRST_SUBMIT_DATA_FOR_PREVIOUS_DAYS
806	MEMBER_PAN_NO_NOT_PROPER
807	DATE_DOES_NOT_BELONG_TO_SELECTED_HOLDING_PERIOD
808	HOLDING_DATE_FORMAT_NOT_PROPER
810	HOLDING_FILEPERIODDATE_DATE_FORMAT_NOT_PROPER
901	DEMAT_ACCOUNT_NO_NOT_PROPER
902	INVALID_MEMBER_ACCOUNT_TYPE
903	UCC_NOT_PROPER
904	CLIENT_NAME_NOT_PROPER
905	CLIENT_PAN_NOT_PROPER
906	ISIN_CODE_NOT_PROPER
907	INVALID_SECURITY_TYPE
908	PLEDGE_BALANCE_NOT_PROPER

909	PLEDGE_BALANCE_NOT_PROPER_CHECK_IF_THIS_SPLIT_CHARACTER_IS_REQUIRED_HERE_SHOULD_CONTAIN_MAX_3_DIGIT_AFTER_DECIMAL
910	FREE_BALANCE_NOT_PROPER
911	FREE_BALANCE_NOT_PROPER_CHECK_IF_THIS_SPLIT_CHARACTER_IS_REQUIRED_HERE_SHOULD_CONTAIN_MAX_3_DIGIT_AFTER_DECIMAL
912	TOTAL_NOT_PROPER
913	TOTAL_NOT_PROPER_CHECK_IF_THIS_SPLIT_CHARACTER_IS_REQUIRED_HERE_SHOULD_CONTAIN_MAX_3_DIGIT_AFTER_DECIMAL
914	TOTAL_SHOULD_BE_SUM_OF_PLEDGED_AND_FREE_BALANCE
916	ACTION_NOT_PROPER
917	NOT_VALID_RECORD_BLANKNULL_VALUES_NOT_ALLOWED
918	DUPLICATE_ROW
919	INVALID_REQUEST_ID
920	INVALID_VERSION_NO
921	INVALID_SYNTAX
922	INVALID_FILE_FORMAT
923	INVALID_URL
924	INVALID_CONTENT_TYPE
925	AUTH_SERVER_DOWN
926	HS_SERVER_DOWN
927	EODBB_SERVER_DOWN
928	ACCOUNT_INACTIVE
929	API_REGISTRATION_IS_NOT_DONE
930	NOT_ALLOWED_BECAUSE_HOLDING_PERIOD_IS_NOT_YET_STARTED
931	NOT_ALLOWED_TO_UPLOAD_DATA_BECAUSE_VALID_ACCESSIBLE_DAYS_ARE_EXPIRED
932	INVALID_NAME_OF_SEC
933	INVALID_UNIT_TYPE
956	SYSTEM_BUSY_TRY_AFTER_SOME_TIME
510	INVALID_FILE_NAME
511	INVALID_FILESEQUENCE
512	WRONG_DATE
513	INVALID_FILE_TYPE
514	INVALID_MEMBER_PAN
515	WRONG_FILE_FOMAT
516	DUPLICATE_FILE
517	WRONG_FILE_HEADER
518	INTERNAL_SERVER_EXCEPTION
519	FILE_MISSING
520	FILE_SIZE SHOULD BE LESS THAN 200MB KINDLY REDUCE RECORDS IN FILE.
521	MEMBER_CODE_NOT_FOUND
522	ERROR_WHILE_COPYING_FILE
524	ERROR_WHILE_UNCOMPRESSING_GZIP_FILE
526	ERROR_WHILE_READING_CSV_FILE
527	PLEASE_UPLOAD_MULTIPART_FILE
530	PAN_NOT_MATCHED
959	API_CALL_NOT_ALLOWED
958	CACE_SERVER_DOWN
809	HOLDING_PERIOD_NOT_PROPER
934	INVALID_COMM_NAME
533	FILE_IS_UNDER_PROCESS
531	DELETED_PERIOD

File Ended

User manual for API registration and daily data submission through API on Production

Key Points:

1. Member have to first register for API on Member Portal if not done previously
<https://inspection.nseindia.com/MemberPortal/>
2. Password must be encrypted with secret key. Secret key will be received in mail after successful registration.
3. The request and response will be in GZIP format.

Part A: Registration for API

1. Details of member registration are as under:

<https://inspection.nseindia.com/MemberPortal/>

Please use the existing credentials used before for ENIT.

If first time login, members will have to change the password and re-login with new password.
For first time login on member portal, OTP will be sent on registered mobile number or email id.
Once correct OTP is entered user will be logged in successfully.



Member TrainingExisting MembershipNSCCLProductsInvestorServices & Circulars



Sign In

User Id

Member Code

Password

Enter Captcha - Case Sensitive

skVgkn

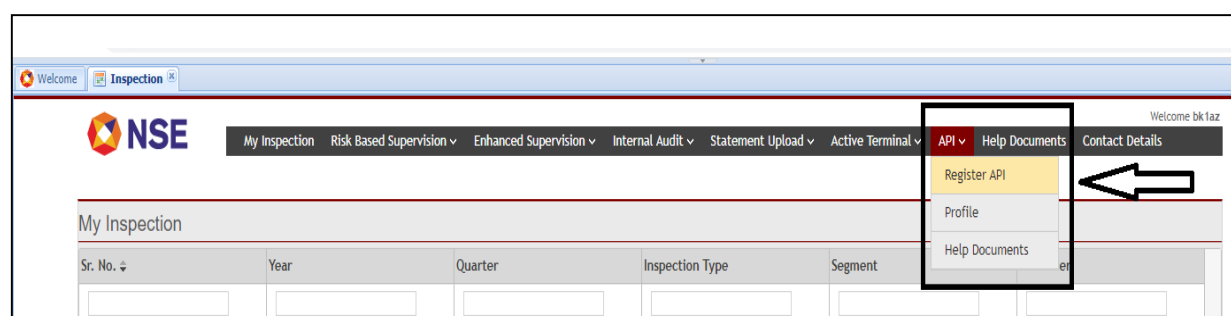
Forgot Password?

LoginCancel

- After successfully login, on left hand corner click on “Inspection” tab and proceed for API registration.



- Navigation: Inspection---> API---> Register API. Click “Register API”



- After clicking the “Register API” button and member will have to submit the following information in Registration form.

Sr. No.	Field	Validation points
1	Login Id	Login Id should be alphanumeric and 6 to 15 characters long. Special characters are not allowed.
2	Member PAN	Alpha-numeric Trading member PAN Character (10)
3	Email Id	Trading member’s Email Id
4	Phone No.	Mobile number Character (10) and should start with 9,8 or 7 only.
5	Password	Characters (12) (At least one capital character, one small character, one numeric and one special character from @, #, \$, %, ^, &, *, =)

After providing inputs for the information sought and accepting the declaration, click on “Register” button.

If all details provided are valid then Registration will be successful.



My Inspection ▾ Risk Based Supervision ▾ Enhanced Supervision ▾ Internal Audit ▾ Statement Upload ▾ Active Terminal ▾ API ▾ Help Documents Contact Details Test Digital Signature

Welcome 06

Register API

Login Id*		Member PAN*		Member's Email Id*		Mobile No.*	
Password*		Confirm Password*					
Declaration*	☐ I/We hereby confirm that the information submitted to the Exchange will be accurate, correct and complete. I/we will not tamper / modify the data before submission in any manner. In case, if the same is found to be tampered / modified, I / we understand that my / our submission will be considered as rejected.						
				Register			Reset

5. Member can view the profile by clicking “Profile” button.

Further, member can reset the Password, Email id and Phone no. by

The screenshot shows the NSE API dashboard. The top navigation bar includes the NSE logo and a menu with items: My Inspection, Risk Based Supervision, Enhanced Supervision, Internal Audit, Statement Upload, Active Terminal, API, Help Documents, Contact Details, and Test Digital Signature. The 'API' dropdown menu is open, showing options: Register API, Profile (highlighted with a yellow background), and Help Documents. A white arrow points to the 'Profile' button. Below the navigation bar, there is a section titled 'My Inspection' with a table containing columns: Sr. No., Year, Quarter, Inspection Type, and Segment.

clicking “Reset” button.

The screenshot shows the NSE API Profile page. It displays the following information: Login Id - SubUser90070, Member PAN - UIYTR4567Y, Member's Email Id - nseit_pbhol@vendor.nse.co.in, Mobile No. - 9004919890, Member Code - 90070, and Key - Vv1ddPpNsz9d/xqsvXahpw==. Below this information, there are two buttons: 'Reset Password' and 'Edit Profile'. A white arrow points to the 'Reset Password' button. At the bottom, there are input fields for 'New Password' and 'Confirm Password', followed by a 'Submit' button. A white arrow points to the 'Submit' button.

6. Member can change “Member PAN”, “Member’s Email ID”, and “Mobile No.” on click of “Edit Profile” button.

Profile					
Login Id -	SubUser90070	Member PAN -	UIYTR4567Y	Member's Email Id -	nseit_pbhoi@vendor.nse.co.in
Mobile No. -	9004919890	Member Code -	90070	Key -	Vv1ddPpNsz9d/xqsvXahpw==
		Reset Password Edit Profile			
Member PAN*	UIYTR4567Y	Member's Email Id*	nseit_pbhoi@vendor.nse.co.in	Mobile No. *	9004919890
Submit					

After successful registration, member will receive an e-mail, on the e-mail address provided by member while registration, which contains Login Id, Password, and Secret key.

Note: Secret key will change on every password reset.

7. Member must encrypt its raw password by using secret key. This encrypted password need to be used while calling login API. Java and .Net Code Snippet of AES encryption is added at the end of the document.

Part B: Steps to consume API's.

1. API endpoints:

Login:

<https://api.nseindia.com/inspectionapi/login>

Holding Upload:

<https://api.nseindia.com/inspectionapi/tradingHoldingUpload/{version}>

EOD Bank Balance Upload:

<https://api.nseindia.com/inspectionapi/tradingEodBalanceUpload/{version}>

Logout:

<https://api.nseindia.com/inspectionapi/logout>

Note:

1. Version is 1.0
2. API endpoint cannot be accessed from the browser (like google chrome, internet explorer, etc.).
3. Members may have to develop application to consume the API's.
4. Use 7Zip tool to generate .gz.

2. How to login with /login API endpoint?

- Create **login text file** with below JSON code having details **Member Code, Login ID, and Password**.

Login Request Object Structure (JSON)

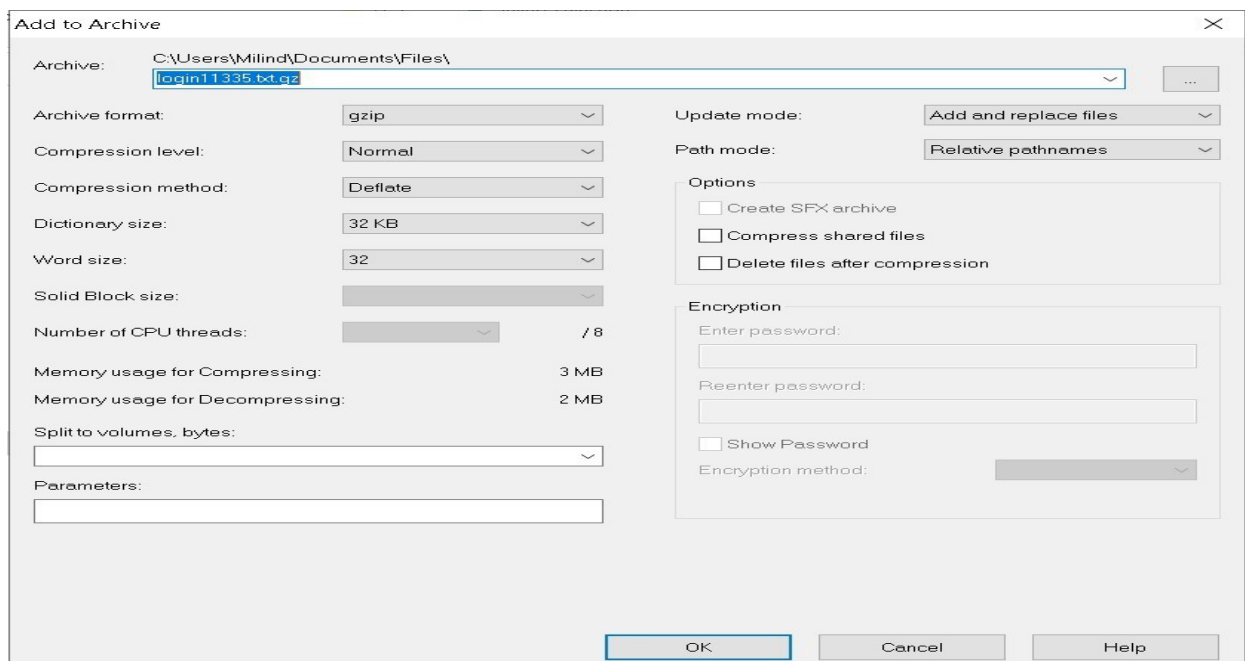
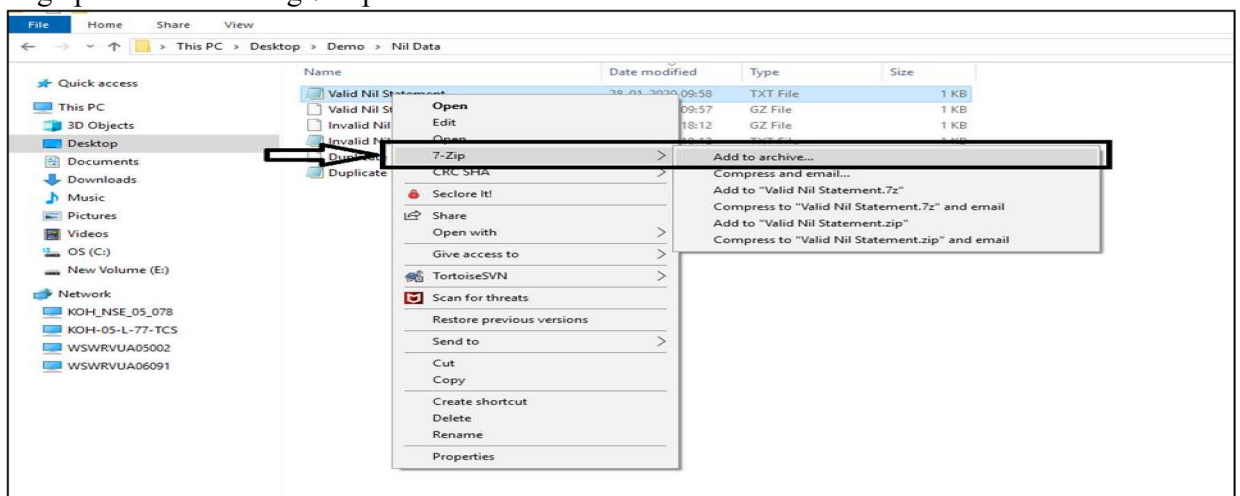
```
{
  "memberCode": "00000",
  "loginId": "ApiUser00000",
  "password": "qi8H8R7OM4xMUNMPuRAZxIY==" }

```

Note: Use login Id and encrypted password.

- **How to convert above text file into gzip File format?**

Following is the diagrammatical representation of conversion of text file into .gzip file format using 7-Zip tool.



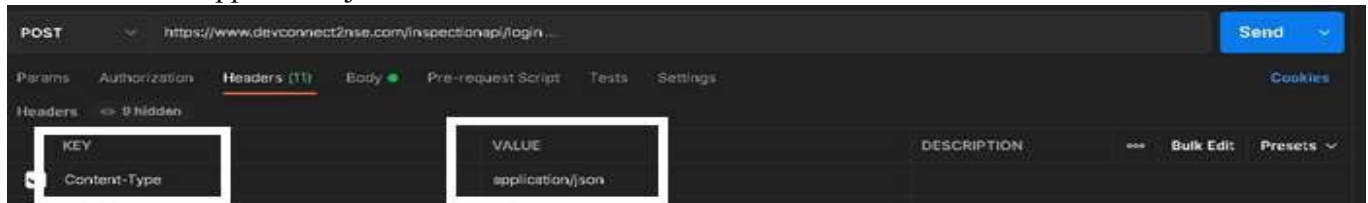
- **Steps to access API using postman tool:**

After opening the Postman tool/software, please select “POST” method & enter login URL in address bar.

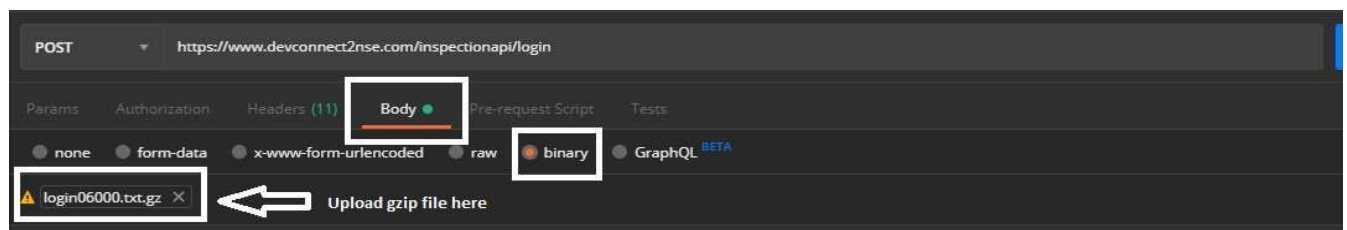
LOGIN: <https://api.nseindia.com/inspectionapi/login>

Click on “Headers” tab [Check below image]

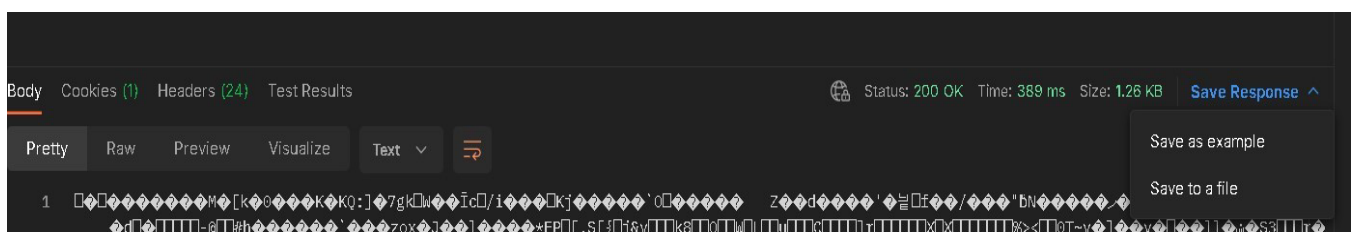
- In “Key” column enter Content-Type and in “VALUE” column enter ‘application/json’.



- Click on “Body” tab ==> Select “binary” radio button [Please check below image]
- Click on “Select File” button and upload your gzip File. [Please check below image]



- Click on “Send” button.
- The response will be in GZIP format. In postman tool click on Save Response and then click on Save to a file



- The saved response file need to be extracted with the 7zip tool and the below response file will be available.

Login Response Object Structure (JSON)

❑ **Convert above text file into gzip File format. [Check procedure shown for /login endpoint.]**

❑ **Open Postmen select “POST” method & enter holding upload URL in address bar.**

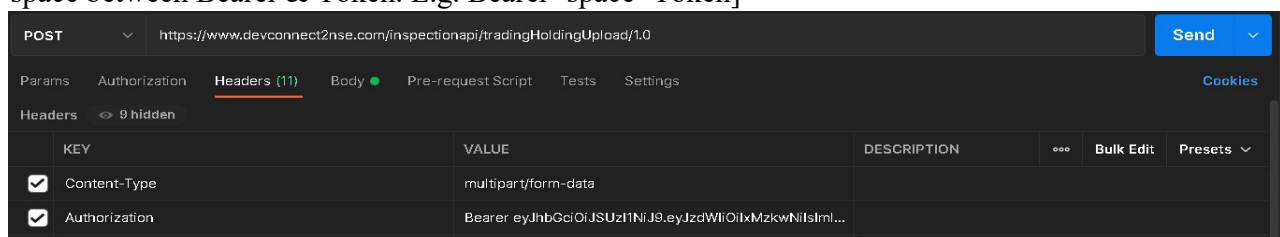
❑ **[https:// api.nseindia.com/inspectionapi/tradingHoldingUpload/{version}](https://api.nseindia.com/inspectionapi/tradingHoldingUpload/{version})**

- Click on “Headers” tab [Check bellow image]

In “Key” column enter Content-Type and in “VALUE” column enter multipart/form-data.

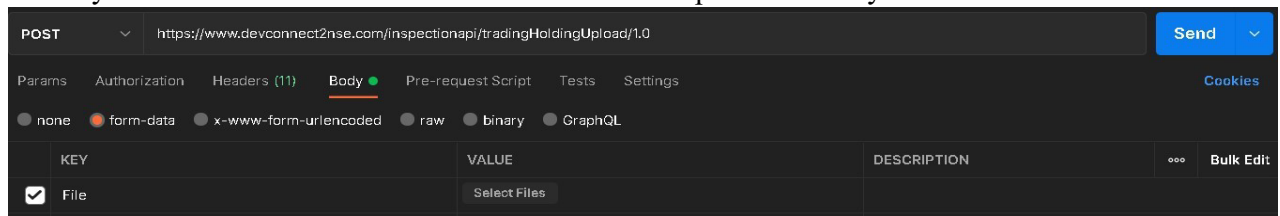
In “Key” column enter Authorization and in “VALUE” column enter token which is generate in

Login api.[Note: - You need to add a “Bearer” before token. It should have space between Bearer & Token. E.g. Bearer<space>Token]



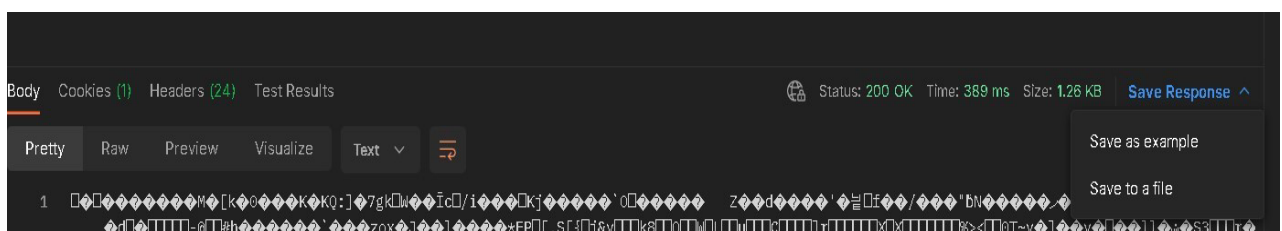
- Click on “Body”_tab ==>> Select “form-data” radio button [Please check below image]

In “Key” column enter File and also select file from dropdown in “Key”



- In “VALUE” column Click on “Select Files” button and upload your .gzip File.

- The response will be in GZIP format. In postman tool click on Save Response and then click on Save to a file



- The saved response file need to be extracted with the 7zip tool and the below response file will be available.
- Note: File Nomenclature should be in this format
PAN_HS_DDMMYYYY_Seqno.gz

Holding Statement Response Object Structure (JSON)

```
{
  "hdrCode": [
    "601"
  ],
  "ackId": "0000000000000000",
  "memberPan": "XXXXXX0000X",
  "hdrStatus": "success",
  "date": "26-03-2022",
  "holdingData": [
    {
      "recId": "1",
      "code": [
        "601"
      ],
      "status": "success"
    },
    {
      "recId": "2",
      "code": [
        "601"
      ],
      "status": "success"
    }
  ],
  "filePeriodDate": "26-03-2022"
}
```

Note:-

- “date” should have date for which data is submitted. If data is submitted for 25-03-2022 then it should be the date and if data is submitted for 10-12-2019 then it should be the date
- filePeriodDate and date will be the same date i.e the date for which data is submitted

4. How to upload EOD bank statement data with /tradingEodBalanceUpload/{version} API endpoint?

- Create holding statement text file in JSON format.
EOD Bank Balance Request Object Structure (JSON)

```
"BankBalanceData": [  
  {  
    "Bank Account No": "00000000000000",  
    "IFSC": "XXXXX0000000",  
    "Bank Account Type": " XYZ ",  
    "03-05-2021 ": 00000000  
  },  
  {  
    "Bank Account No": "00000000000000",  
    "IFSC": "XXXXX0000000",  
    "Bank Account Type": " XYZ ",  
    "03-05-2021": "00000000"  
  }  
]  
}
```

- Convert above text file into gzip File format. [Check procedure shown for /login endpoint.]
- Open Postmen select “POST” method & enter holding upload URL in address bar.

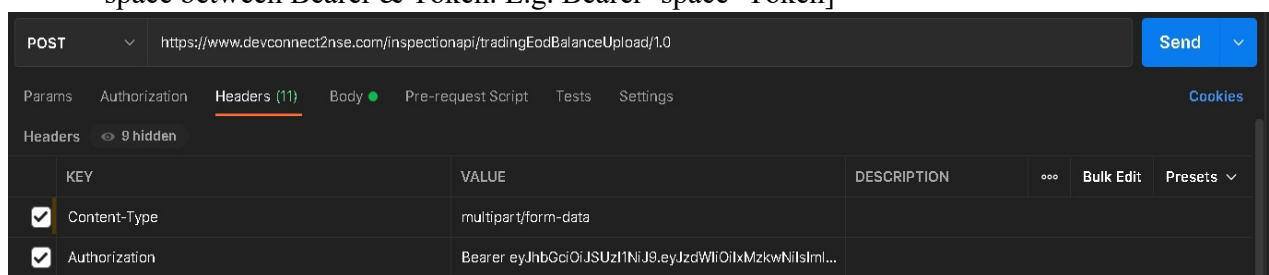
<https://api.nseindia.com/inspectionapi/tradingEodBalanceUpload/{version}>
}

- Click on “Headers” tab [Check bellow image]

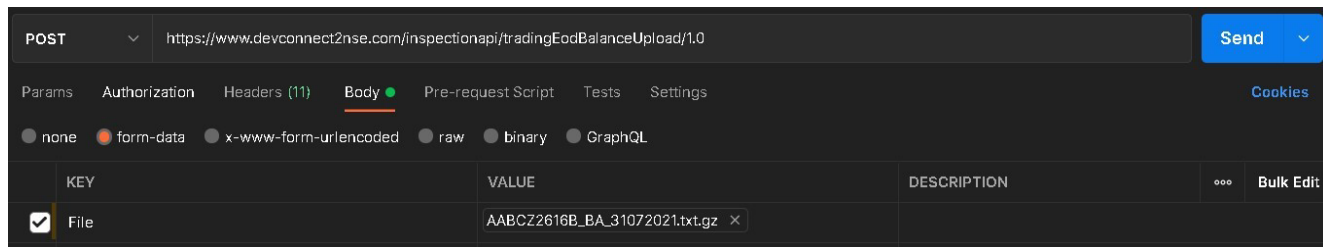
In “Key” column enter Content-Type and in “VALUE” column enter multipart/form-data.

In “Key” column enter Authorization and in “VALUE” column enter token which is generate in

Login api.[Note: - You need to add a “Bearer” before token. It should have space between Bearer & Token. E.g. Bearer<space>Token]



- Click on “Body”_tab ==>> Select “form-data” radio button [Please check below image] In “Key” column enter File and also select file from dropdown in “Key”
In “VALUE” column Click on “Select Files” button and upload your .gzip File.



- Click on “Send” button.
- The response will be in GZIP format. In postman tool click on Save Response and then click on Save to a file
- The saved response file need to be extracted with the 7zip tool and the below response file will be available.
Note: file Nomenclature should be in this format
PAN_BA_DDMMYYYY.gz Eod Bank Balance Response Object Structure (JSON)

```

"BankBalanceData": [
{
"Bank Account No": "0000000000000000",
"IFSC": "XXXX00000000",
"Bank Account Type": " XYZ ",
"03-05-2021 ": 00000000
},
{
"Bank Account No": "0000000000000000",
"IFSC": "XXXX00000000",
"Bank Account Type": " XYZ ",
"03-05-2021": "00000000"
}
]
}

```

5.How to Logout with /logout API endpoint?

- Create logout text file with below JSON code such as,
Logout Request Object Structure (JSON)

```

{
  "memberCode": "00000",
  "loginId": "test"
}

```

- Convert above text file into gzip File format [Check procedure shown for /login endpoint.]
- Open Postmen select “POST” method & enter holding upload URL in address bar.
- **https:// api.nseindia.com/inspectionapi/logout**

- Click on “Headers” tab [Check bellow image]
In “Key” column enter Content-Type and in “VALUE” column enter application/json.
In “Key” column enter Authorization and in “VALUE” column enter token which is generate in
Login api.[Note: - You need to add a “Bearer” before token. It should have space between
Bearer & Token. E.g. Bearer<space>Token]

- Click on “Body” tab ==> Select “binary” radio button [Please check below image]
- Click on “Select File” button and upload your gzip File. [Please check below image]

Click on “Send” button. Following response will be display in JSON format.

```
{
  "code": [
    "601"
  ],
  "status": "success"
}
```

- 1) If account is inactive due to 5 incorrect password attempts, on Member Portal reset the password to reactivate the account. A new key will be generated on reset.
- 2) Password encrypted using online site may be incorrect, develop own code snippet for encrypting password given below on point 8 and 10. Kindly don't use online sites for encrypting password.

- 3) JSON text file for testing generated online may be incorrect. Kindly don't use online sites for generating text files.

5. How to download, delete and reupload?

- To view the statements uploaded for Holding, Cash Equivalent and Eod Bank Balance respectively go to below pages on Member Portal:
 - i. Inspection->Statement Upload->Holding Statement->View Submission
 - ii. Inspection->Statement Upload->Cash & Cash Equivalent Balances->View Submission
 - iii. Go Inspection->Statement Upload->Bank Account Balances->View Submission

The screenshot shows the NSE Member Portal interface. The top navigation bar includes 'Existing Membership', 'NSCCL', 'Products', 'Investor', 'Services & Circulars', and 'Dashboards'. The left sidebar lists 'SERVICES' and 'Inspection'. The main content area shows the 'Statement Upload' dropdown menu, which includes options like 'Holding Statement', 'Cash & Cash Equivalent Balances', 'Bank Account Balances', 'Bank Statement', 'Collateral Segregation Submission', 'Yes Bank - Client Wise Submission', 'Monthly Regulatory Report', and 'Collateral Shortage Submission'. The 'View Submission' option is highlighted under 'Cash & Cash Equivalent Balances'.

The screenshot shows the NSE Member Portal interface with the 'Showing latest 50 submissions' table. The table has columns for 'Uploaded File', 'Activity', 'Created Time', and 'Delete'. The table lists 8 submissions, each with a file name, activity type, and creation time. The 'Delete' column contains a 'Delete' button for each row. The table is paginated, showing 'Page: 1 of 1' and '50' items.

Uploaded File	Activity	Created Time	Delete
1 CACAB3359A HS 02042022 01 SUCCESS.zip	Trading	Apr 5, 2022 12:44:34 AM	Delete
2 YXXIA3359A HS 02042022 01 FAILURE.zip	Trading	Apr 4, 2022 11:24:08 PM	
3 QWDWQ3359A HS 26032022 01 FAILURE.zip	Trading	Apr 4, 2022 7:13:37 AM	
4 QWDWQ3359A HS 26032022 01 FAILURE.zip	Trading	Apr 4, 2022 7:11:08 AM	
5 QWDWQ3359A HS 26032022 01 FAILURE.zip	Trading	Apr 4, 2022 7:07:11 AM	
6 ARDDQ3359A HS 25122021 01 SUCCESS.zip	Trading	Mar 31, 2022 8:46:38 AM	
7 ABCDQ3359A HS 25122021 01 FAILURE.zip	Trading	Mar 30, 2022 11:37:42 PM	
8 ABCDQ3359A HS 25122021 01 FAILURE.zip	Trading	Mar 30, 2022 11:31:53 PM	

- As shown in images below, by clicking on file name you can download the file. And by clicking on “Delete” button member can delete the file and re-upload the file through API.

MONTH* YEAR*

Showing latest 50 submissions. Kindly select the respective period's month and year to view old submissions.

Uploaded File	Activity	Created Time	Delete
CACAB3359A_HS_02042022_01_SUCCESS.zip	Trading	Apr 5, 2022 12:44:34 AM	Delete
2 XTXA3359A_HS_02042022_01_FAILURE.zip	Trading	Apr 4, 2022 11:24:08 PM	
3 DWDWQ3359A_HS_26032022_01_FAILURE.zip	Trading	Apr 4, 2022 7:13:37 AM	
4 DWDWQ3359A_HS_26032022_01_FAILURE.zip	Trading	Apr 4, 2022 7:11:08 AM	
5 DWDWQ3359A_HS_26032022_01_FAILURE.zip	Trading	Apr 4, 2022 7:07:11 AM	
6 ABDDQ3359A_HS_25122021_01_SUCCESS.zip	Trading	Mar 31, 2022 8:46:38 AM	
7 ABCDQ3359A_HS_25122021_01_FAILURE.zip	Trading	Mar 30, 2022 11:37:42 PM	
8 ABCDQ3359A_HS_25122021_01_FAILURE.zip	Trading	Mar 30, 2022 11:31:53 PM	

Page 1 of 1 50 View 1 - 26 of 50

Download

MONTH* YEAR*

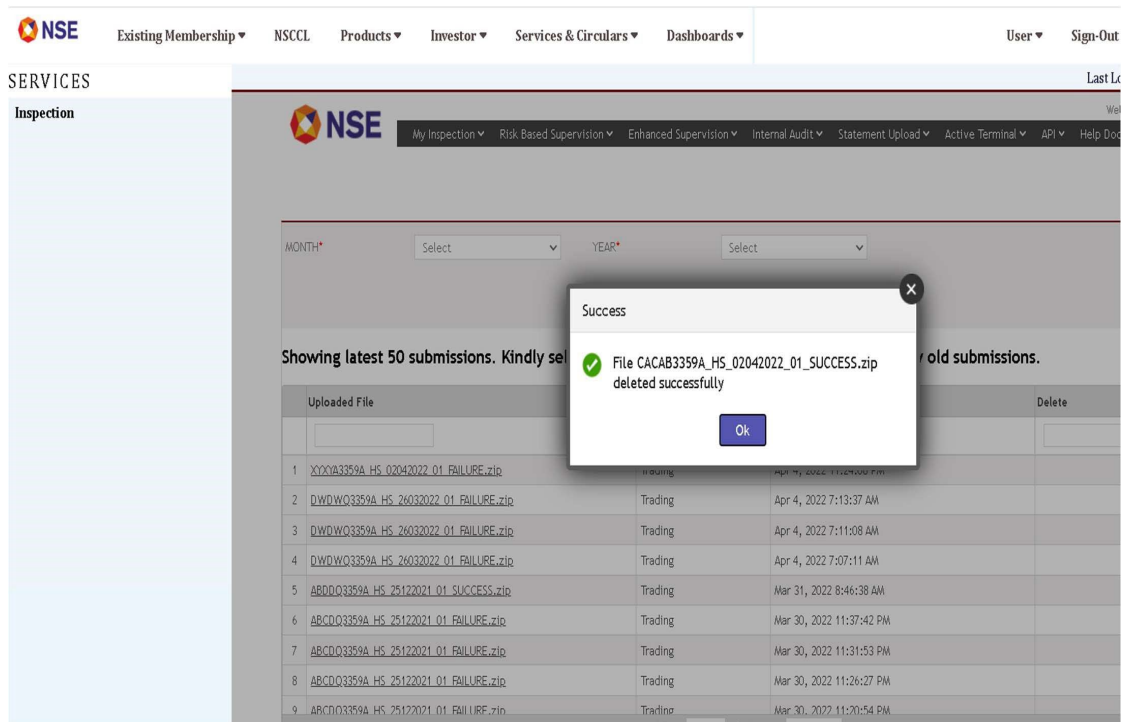
Showing latest 50 submissions. Kindly select the respective period's month and year to view old submissions.

Uploaded File	Activity	Created Time	Delete
1 CACAB3359A_HS_02042022_01_SUCCESS.zip	Trading	Apr 5, 2022 12:44:34 AM	Delete
2 XTXA3359A_HS_02042022_01_FAILURE.zip	Trading	Apr 4, 2022 11:24:08 PM	
3 DWDWQ3359A_HS_26032022_01_FAILURE.zip	Trading	Apr 4, 2022 7:13:37 AM	
4 DWDWQ3359A_HS_26032022_01_FAILURE.zip	Trading	Apr 4, 2022 7:11:08 AM	
5 DWDWQ3359A_HS_26032022_01_FAILURE.zip	Trading	Apr 4, 2022 7:07:11 AM	
6 ABDDQ3359A_HS_25122021_01_SUCCESS.zip	Trading	Mar 31, 2022 8:46:38 AM	
7 ABCDQ3359A_HS_25122021_01_FAILURE.zip	Trading	Mar 30, 2022 11:37:42 PM	
8 ABCDQ3359A_HS_25122021_01_FAILURE.zip	Trading	Mar 30, 2022 11:31:53 PM	

https://uat.connect2nsccl.com/inspection/rhelB/openDocument?fileId=UFT0sg0+nB59CMdl+8/zg==

Show all

Delete



7.AES Code Snippet [Java]

```
public class CodeSnippet
{
    public static void main(String[] args) throws UnsupportedOperationException,
    NoSuchAlgorithmException, NoSuchPaddingException, InvalidKeyException,
    IllegalBlockSizeException, BadPaddingException
    {
        //Plain Text Password
        String password = "Nseitjan@123";

        //sample key is - "XBaNb0xmK2TNRIfcHA3F306Oi14HWaEYmtUd0qRheTc="
        String key = "XBaNb0xmK2TNRIfcHA3F306Oi14HWaEYmtUd0qRheTc=";
        //Key is converted to byte array
        byte[] keyByteArray = new Base64().decode(key.getBytes("UTF-8"));

        //SecretKeySpec is used to construct a SecretKey from a byte array
        SecretKeySpec secretKeySpec = new SecretKeySpec(keyByteArray, "AES");
        Cipher cipher = Cipher.getInstance("aes/ecb/pkcs5padding");
        cipher.init(Cipher.ENCRYPT_MODE, secretKeySpec);

        //pass plain text that is to be encrypt
        String encrypt = (new Base64()).encodeAsString(cipher.doFinal(password.getBytes()));

        //actual key in base64 format
        System.out.println("encrypted string:" + encrypt);
    }
}
```

8.Gzip Code Snippet [Java]

```
public class NetClientGet
{
    public static void main(String[] args) throws IOException
    { byte[] body = readTextFile();
      System.out.println("gzip bytes "+body);
    }
    public static byte[] readTextFile() throws IOException
    {
        //LoginJson.text contains json login Request example listed bellow
        //{
        // "memberCode": "00000",
        // "loginId": "SubUser00000",
        // "password":"sMzvV6fvENaQfRedWYd07w=="
        //}

        InputStream inputStream = new FileInputStream("<path>\\login00000.txt");
        @SuppressWarnings("resource")
        BufferedReader bufferReader = new BufferedReader(new
        InputStreamReader(inputStream));
        String line = bufferReader.readLine();
        StringBuilder stringBuffer = new StringBuilder(); while (line != null)
        { stringBuffer.append(line).append("\n"); line = bufferReader.readLine(); }
        //Creates a gzip file compressString(stringBuffer.toString()); //create gzip compress bytes.
        return compress(stringBuffer.toString()); }

    public static byte[] compress(final String stringToCompress)
    {
        try (final ByteArrayOutputStream baos = new ByteArrayOutputStream(); final
        GZIPOutputStream gzipOutput = new GZIPOutputStream(baos))
        { gzipOutput.write(stringToCompress.getBytes(StandardCharsets.UTF_8));
          gzipOutput.finish(); return baos.toByteArray();
        } catch (IOException e)
        { throw new UncheckedIOException("Error while compression!", e);} }

    public static String compressString(String str) throws IOException
    { if (str == null || str.length() == 0)
      { return str;
      }
      BufferedWriter writer = null;
      try{
          File file = new File("your.gzip");
          GZIPOutputStream zip = new GZIPOutputStream(new FileOutputStream(file)); writer =
          new BufferedWriter(new OutputStreamWriter(zip, "UTF-8")); writer.append(str); }
      Finally{ if(writer != null){ writer.close();
      } } return str;
    }
}
```

9. AES Code Snippet [.Net]

```
using System;
using System.Security.Cryptography; using System.Text;

namespace AES256{ class Program{ private static string GetString(byte[] b)
{
return Encoding.UTF8.GetString(b);
}
static void Main(string[] args){
byte[] data = Encoding.UTF8.GetBytes("NseitJan@201");
byte[]a = Convert.FromBase64String("AAECAwQFBgcICQoLDA0ODw==");
Console.WriteLine("Key : {0}", GetString(a)); byte[] enc = Encrypt(data, a); string result =
Convert.ToBase64String(enc); Console.WriteLine("Encrypted text", result); byte[] dec =
Decrypt(enc, a);
Console.WriteLine("Encrypted : {0}", GetString(enc)); Console.WriteLine("Decrypted :
{0}", GetString(dec));
// Console.ReadKey();
} public static byte[] Encrypt(byte[] data, byte[] key){ using (RijndaelManaged csp = new
RijndaelManaged())
{ csp.KeySize = 256; csp.BlockSize = 128; csp.Key = key; csp.Padding =
PaddingMode.PKCS7; csp.Mode = CipherMode.ECB;
CryptoTransform encrypter = csp.CreateEncryptor(); return
encrypter.TransformFinalBlock(data, 0, data.Length);
} } private static byte[] Decrypt(byte[] data, byte[] key){ using (RijndaelManaged csp =
new RijndaelManaged())
{ csp.KeySize = 256; csp.BlockSize=128; csp.Key = key; csp.Padding =
PaddingMode.PKCS7; csp.Mode = CipherMode.ECB;
CryptoTransform decrypter = csp.CreateDecryptor(); return
decrypter.TransformFinalBlock(data, 0, data.Length);
}
}
}
}
```

Technology Used:

Language: - java version "11.0.12" 2021-07-20 LTS

*****End of document*****