GOVERNMENT OF INDIA
MINISTRY OF STATISTICS AND
PROGRAMME IMPLEMENTATION

API USER MANUAL FOR CONSUMER PRICE INDEX (CPI)

Disclaimer

This is an API Specifications created to facilitate the integraton of various Ministries/ Departments to NMDS Repository through APIs and manage Metadata in Standard Format. However, utmost care was taken to ensure all steps are explained to make it easier to understand the metadata management process, if any issues are found, same may be communicated with ASPD, MoSPI, and next version shall be released after incorporating the feedbacks.

Document History

Status	Version	Author	Change description	Date
Initial	Rev 1	DIID	Initial Document	22.10.2025
Approved	Rev 2	DIID	Approved Document	07.11.2025

Abbreviations

MoSPI – Ministry of Statistics and Programme Implementation
DIID – Data Informatics and Innovation Division
ASPD – Administrative Statistics and Policy Division
MWP – Metadata Web Portal
API – Application Programming Interface
JSON – JavaScript Object Notation
CPI – Consumer Price Index
UI – User Interface
XLS – Excel Spreadsheet Format
CSV – Comma Separated Values
URL – Uniform Resource Locator
GitHub – Code Repository Platform

Table of Contents

1. Introduction

The CPI API platform provides structured access to Consumer Price Index datasets released by the Ministry of Statistics and Programme Implementation. The platform supports authenticated data retrieval and delivers CPI Index and Inflation records in standard formats.

The Administrative Statistics and Policy Division manages the platform and ensures that users receive secure and stable API services for CPI data. The platform offers Group, Subgroup and Item level CPI data from January 2011 onward. Users follow a simple registration and login process to obtain an access token required for full data access.

New users must register once with a unique email ID on the API portal. After registration, the system issues an access token valid for 30 minutes. This token allows users to fetch complete datasets. Without a token, only the first ten records are returned.

The platform provides two tools for users. Swagger gives domain users a view of parameters and response structures. Postman helps technical users test and execute API requests with custom parameters. Both tools support JSON and CSV responses.

Users can download metadata sheets in XLS format from the API homepage. These sheets contain the codes needed for parameters like Year, Month, State, Group, Subgroup and Item.

Developers can also run the APIs using Python or Curl scripts available in the official GitHub repository. For further support or integration details, users may contact the ASPD Division of MoSPI.

2. Purpose of Document

This document guides users who work with the CPI API platform. It explains the process for registration, authentication and data access through the APIs hosted by the Ministry of Statistics and Programme Implementation. The document describes each API used to fetch CPI Index and Inflation data in standard formats.

The Administrative Statistics and Policy Division maintains the platform and provides technical support for its usage. The document helps users understand the parameters, response structures and execution steps through Swagger, Postman and script-based methods. Users are responsible for using valid credentials, applying correct parameters and ensuring proper handling of CPI data downloaded through the APIs.

2. User Registration

2.1. Using Swagger, the steps for executing the APIs are mentioned below:

2.1.1. The users of the platform are required to sign up for using this API platform by accessing the **Base URL/api/users/usersignup**.

2.1.2. The below screen will open on accessing this URL

2.1.3. Click anywhere on the green tile shown above and then click on the “Try it out” button on the right as indicated below. The fields in the Request Body section will become editable.

2.1.4. Enter the details in the Request Body as below:

```
{
 "username": "test@test.com",
  "password": "Test@12345",
"organization": "Ministry of Statistics and Programme Implementation",
 "purpose": "View/Download the Data",
 "gender": "Male"
}
```

2.1.4.1. A valid email is required in the username field. Only ‘@’ special character is allowed in this field.

2.1.4.2. One email can be used for registration only once.

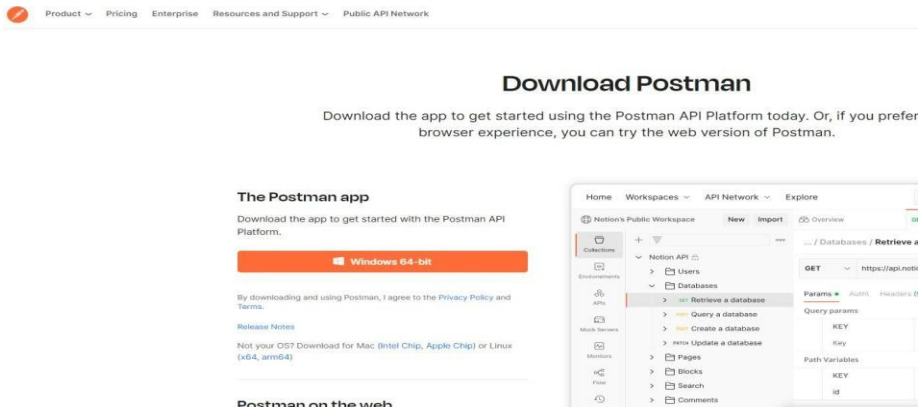
2.1.4.3. No special characters are allowed in the organization, purpose fields and gender.

2.1.4.4. Click on the Execute button. The API response code and description will be returned. response code of 200 indicates the execution was successful.

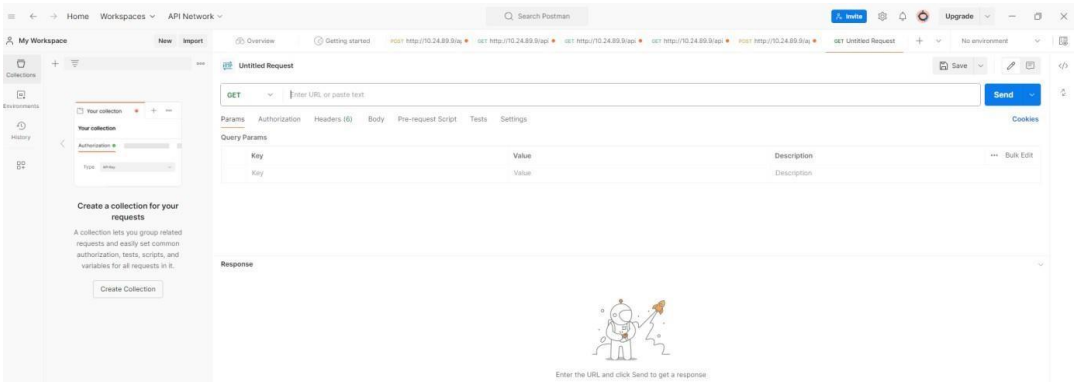
```
{
  "msg": "User registered successfully.",
  "statusCode": true,
  "response": {
    "id": 17,
    "gender": "Male",
    "username": "test7834@test.com",
    "organization": "Ministry of Statistics and Programme Implementation",
    "purpose": "View/Download the Data",
    "password": "VGVzdEAxMjM0NQ==",
    "role": "users",
    "createdBy": "",
    "isActive": "1",
    "updatedAt": "2024-04-15T21:50:14.098Z",
    "createdAt": "2024-04-15T21:50:14.098Z"
  }
}
```


2.2. Using Postman, the steps are mentioned below:

2.2.1.Download Postman from any web browser by visiting the link https://www.postman.com/downloads/

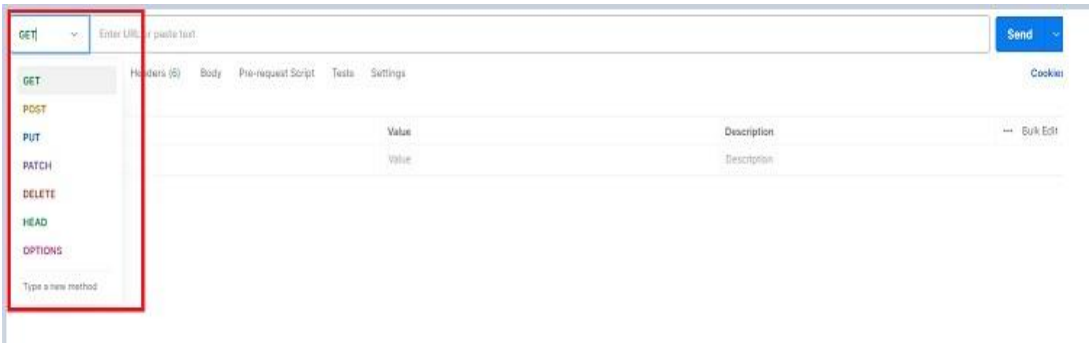


2.2.2. After downloading Postman app, open the app sign in or use the “lightweight API Client” and you will see the following menu:

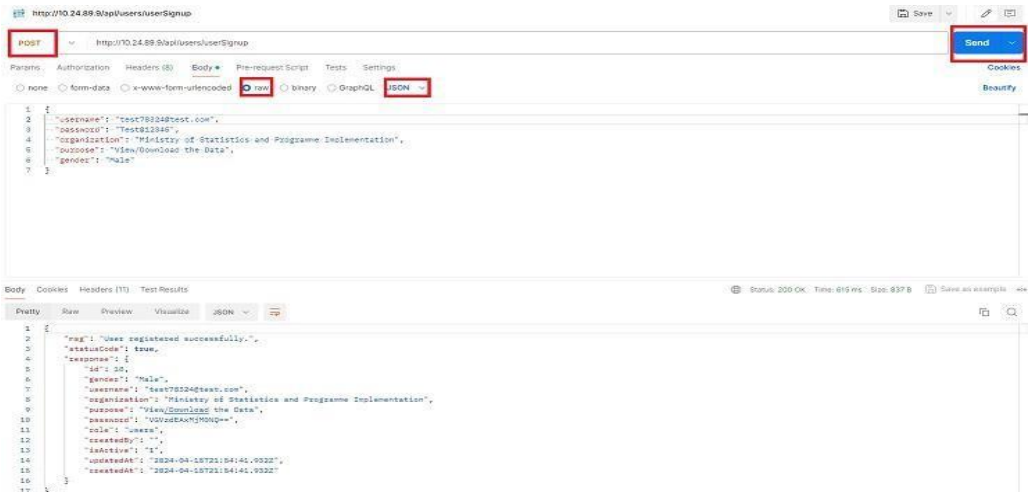


3.2.3 Click the “+” button to add a request.

3.2.4 Expand the method button.



2.2.3. Use the signup API to register user for using the API platform, as described below.
2.2.3.1. Change the method to POST
2.2.3.2. Enter the **API *BASE URL*/api/users/usersignup** in the textbox
2.2.3.3. Select Body, raw radio button and select JSON from the dropdown where JSON is shown by default as highlighted in Yellow below
2.2.3.4. Enter the data for the signup user details as in step#2.1.4 but make sure the username is unique as shown below and click Send.



2.2.3.5. A success message will be sent as the response of the API indicating user signup is successful as shown above.

3. Login Process

3.1. The below steps describe the process to get the access token for the API through POSTMAN

3.2. Select “POST” and add the URL http://api.mospi.gov/api/login

3.3. Select Body, raw radio button and select JSON from the dropdown where JSON is shown by default as highlighted in Yellow below

3.4. Enter username and password as shown below using the same credentials used during signup process:

{

“username”: “test78324@test.com”,

“password”: “Test@123434435”

}

3.5. Click “Send”.

3.6. Copy the “Token” value by clicking the “small double square” icon seen in the snapshot above.

3.7. Open a notepad and paste the value here. Copy only the value from the “token” element without the Quotes.

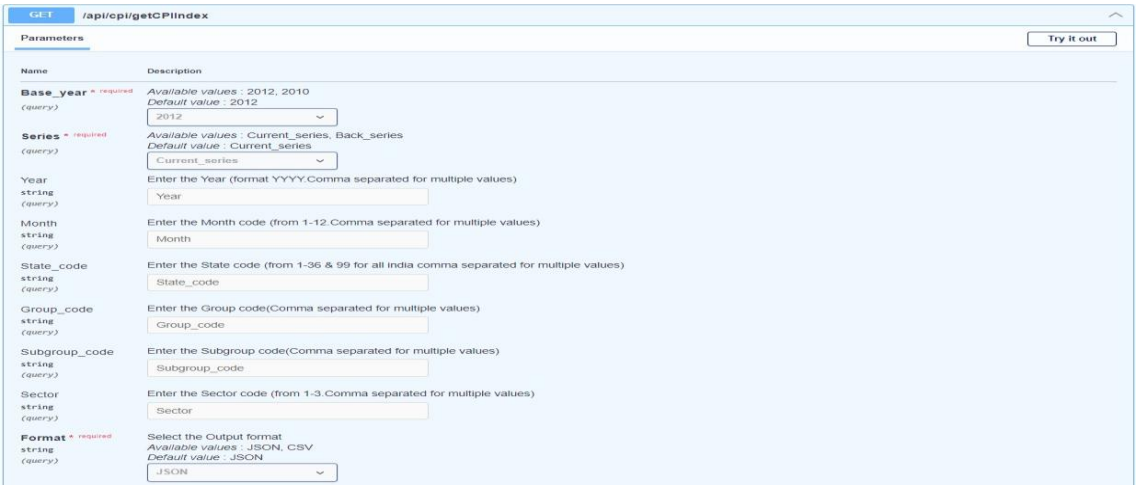
3.8. Use this token in the next sections for fetching the complete data from Postman only and NOT Swagger.

4. Accessing Inflation and Index data from Swagger

4.1. **Swagger UI**: Following are the steps for the API execution through the Swagger UI tool:

4.1.1. Access the API **Base URL/cpi** on the web browser. The following page will load.

4.1.2. Click the tile in BLUE as shown above to expand the API **/api/cpi/getCPIIndex**.



4.1.3. Click “Try it out” to enter the following parameters of the API to query for the required data:

4.1.3.1. Base year: 2012,2010. Only a single value can be select for this parameter.

4.1.3.2. Series: Current series, Back series. Only a single value can be sent for this parameter.

4.1.3.3. Year: Comma separated multiple values in the Format YYYY is accepted in this parameter.

4.1.3.4. Month: Comma separated multiple values in the Format ‘1,4’ is accepted in this parameter. Refer the metadata sheet for the codes corresponding to the Month names.

4.1.3.5. State code: Comma separated multiple values Format ‘1,2’ etc, is accepted in this parameter. Refer the metadata sheet for the State names.

4.1.3.6. Group code: Comma separated multiple values in the Format ‘5,7’ etc. is accepted in this parameter. Refer the metadata sheet for the codes corresponding to the Group names.

4.1.3.7. Subgroup _code: Comma separated multiple values in the format ‘1.1.01, 3.1.01’ etc. is accepted in this parameter. Refer the metadata sheet for the codes corresponding to the Subgroup names.

4.1.3.8. Sector: Comma separated multiple values in the Format ‘1,2,3’ is accepted in this parameter. Refer the metadata sheet for the codes corresponding to the Subgroup names.

4.1.3.9. Format: CSV or JSON option can be mentioned for the data format returned from the API.

4.1.4. If no value is supplied for a particular parameter, then the API will not apply a filter for that parameter.

4.1.5. Click on ‘Execute’ for fetching the data with the applied parameter values.

4.1.6. The below is the response of the API in JSON and CSV formats. Click on the Download buttoto download the data in a file:

Server response

Code Details

200

Response body


```
{
  "data": [
    {
      "baseyear": "2010",
      "year": 2011,
      "month": "April",
      "state": "All India",
      "sector": "Combined",
      "group": "General",
      "subgroup": "",
      "index": "106.2",
      "inflation": null,
      "status": "F"
    },
    {
      "baseyear": "2010",
      "year": 2011,
      "month": "April",
      "state": "All India",
      "sector": "Rural",
      "group": "General",
      "subgroup": "",
      "index": "107.5",
```


Download

Server response

Code Details

200

Response body


```
baseyear,year,month,state,sector,group,subgroup,index,inflation,status
2012,2013,April,All India,Urban,General,,105.7,,F
2012,2013,April,All India,Rural,General,,106.4,,F
2012,2013,April,All India,Combined,General,,106.1,,F
2012,2013,April,All India,Urban,Food and Beverages,Meat and Fish,113.4,,F
2012,2013,April,All India,Urban,Food and Beverages,Fruits,109.5,,F
2012,2013,April,All India,Urban,Food and Beverages,Milk and Products,104.7,,F
2012,2013,April,All India,Urban,Food and Beverages,Cereals and Products,114.6,,F
2012,2013,April,All India,Urban,Food and Beverages,"Prepared Meals, Snacks, Sweets etc.",110.6,,F
2012,2013,April,All India,Urban,Food and Beverages,Spices,103.5,,F
2012,2013,April,All India,Urban,Food and Beverages,Oils and Fats,102.1,,F
2012,2013,April,All India,Urban,Food and Beverages,Sugar and Confectionery,102.0,,F
2012,2013,April,All India,Urban,Food and Beverages,Pulses and Products,104.6,,F
2012,2013,April,All India,Urban,Food and Beverages,Vegetables,109.7,,F
2012,2013,April,All India,Urban,Food and Beverages,,108.8,,F
2012,2013,April,All India,Urban,Food and Beverages,Egg,106.0,,F
2012,2013,April,All India,Urban,Food and Beverages,Non-alcoholic Beverages,108.2,,F
2012,2013,April,All India,Combined,Food and Beverages,Fruits,108.9,,F
2012,2013,April,All India,Combined,Food and Beverages,Oils and Fats,104.4,,F
2012,2013,April,All India,Rural,Food and Beverages,Sugar and Confectionery,104.2,,F
2012,2013,April,All India,Combined,Food and Beverages,Milk and Products,105.7,,F
```

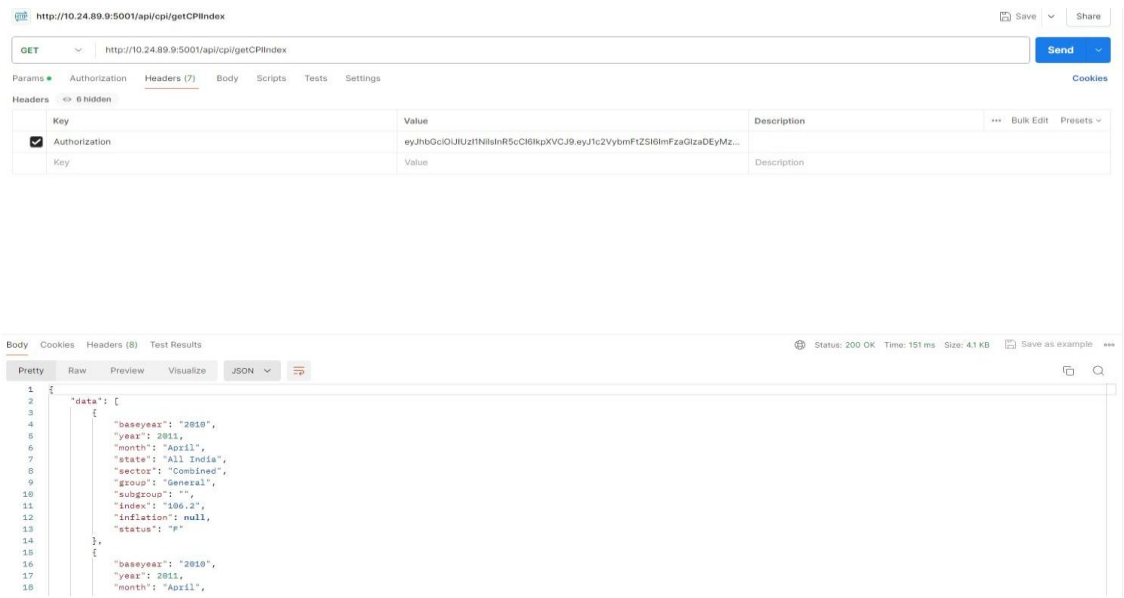

Download

4.2. **Postman**: Following are the steps for the API execution through Postman tool:

4.2.1. The below steps describe the process to get the access token for the API through POSTMAN

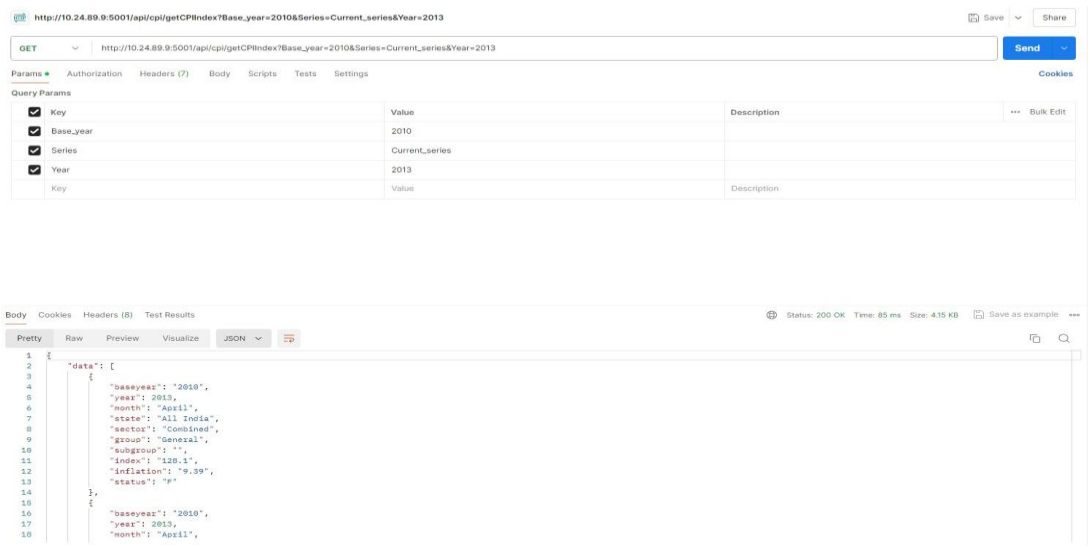
4.2.1.1. Select “GET”, enter the API **Base URL/api/cpi/getCPIIndex** in the URL field.

4.2.1.2. Fill the Authorization access token (refer Section 3.4 above to generate the token) and add it in the Headers tab as shown below and click on Send:



4.2.1.3. If no parameters are given then the complete data is fetched for the API as shown in the snapshot above.

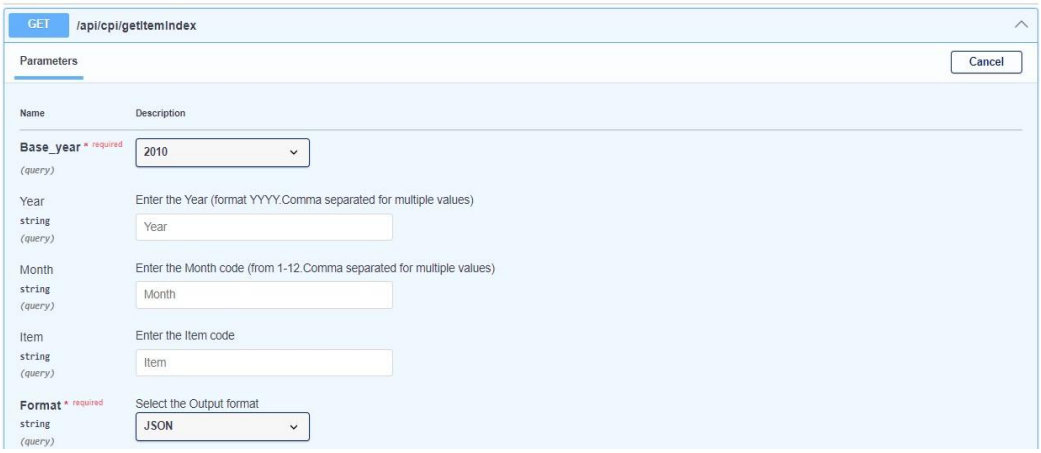
4.2.1.4. Now enter the required parameters in the Params tab along with Authorization token under Header's tab and click Send as per the below screenshot. The data as per the query parameters is fetched in the JSON/CSV format.



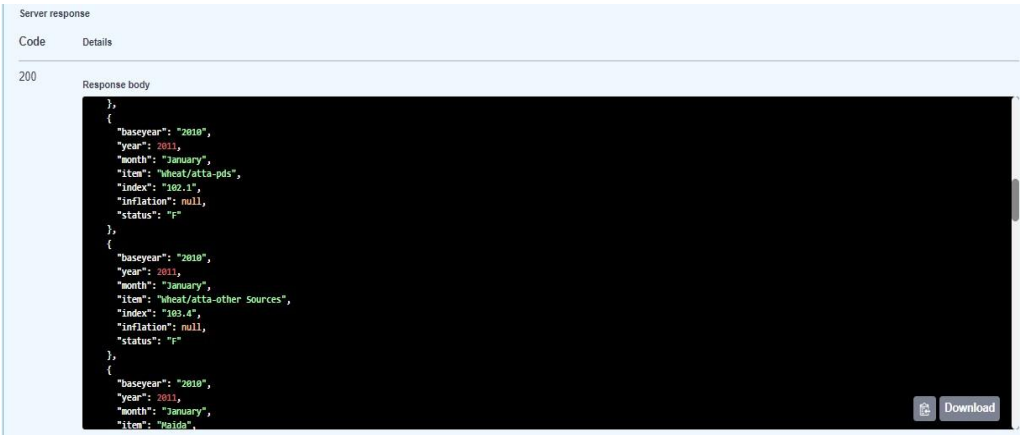
5. Accessing Item level Inflation and Index data from Swagger

5.1. **Swagger UI**: Following are the steps for the API execution through the Swagger UI tool:
5.1.1. Access the API on the web browser: **Base URL/cpi**. The following page will load

5.1.2. Click the tile in BLUE as shown above to expand the API **/api/cpi/getItemIndex**



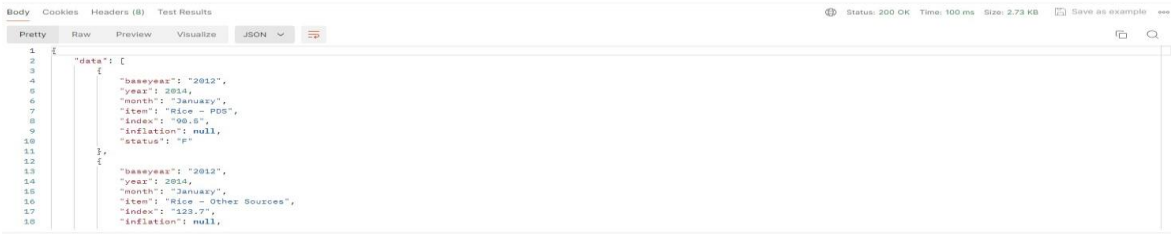
5.1.2.1. Click “Try it out” to enter the following parameters of the API to query for the required data:
5.1.2.2. Base_year:2010,2012, Only single value can be select for this parameter.
5.1.2.3. Year: Comma separated multiple values in the Format YYYY is accepted in this parameter.
5.1.2.4. Month: Comma separated multiple values in the Format ‘1,4’ is accepted in this parameter. Refer the metadata sheet for the codes corresponding to the Month names.
5.1.2.5. Item code: Comma Separated multiple values in the Format ‘1.1.01.1.1.02.X, 1.1.01.1.1.01.P’ is accepted in this parameter. Refer to the metadata sheet for the codes corresponding to the Item codes.
5.1.2.6. Format: CSV or JSON option can be mentioned for the data format returned from the API.
5.1.2.7. Click on ‘Execute’ for fetching the data with the applied parameter values.
5.1.2.8. The below is the response of the API in JSON and CSV formats respectively. Click on the Download button to download the data in a file:



5.2. **Postman**: Following are the steps for the API execution through the Postman tool:

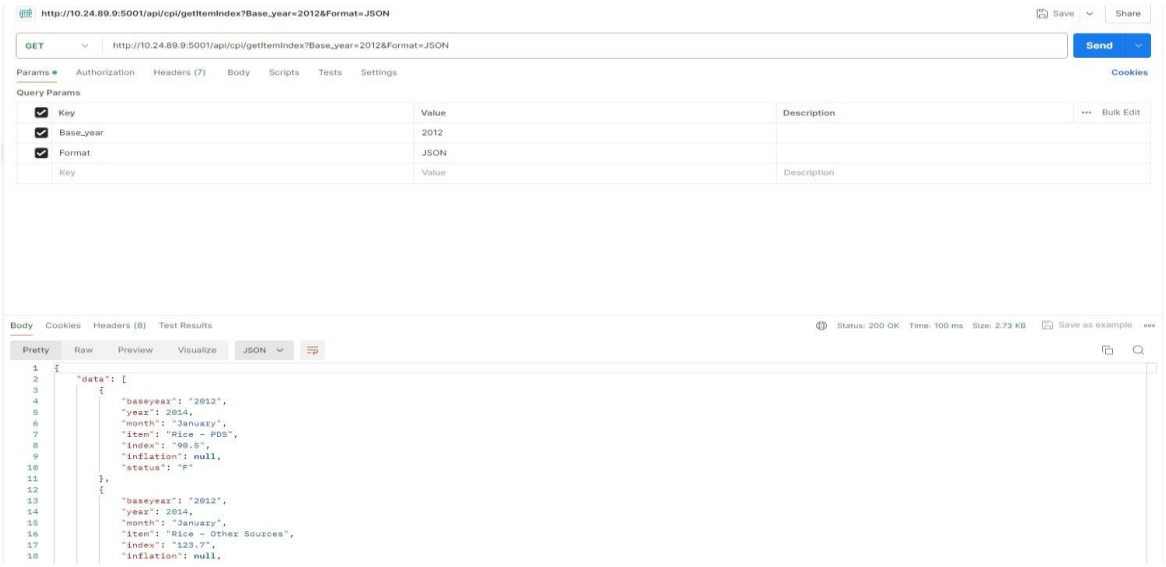
5.2.1. Select “GET”, enter the API **Base URL/api/cpi/getItemIndex** in the URL field.

5.2.2. Fill the Authorization access token (refer Section 3.4 above for the steps to generate the token) and add it in the Headers tab shown below and click Send:



5.2.3. If no parameters are given then the complete data is fetched for the API shown above.

5.2.4. Enter the required parameters in the Params tab along with the authorization token under Headers tab and click Send. The data as per the query parameters is fetched in the JSON/CSV format shown below.




```
{
    "data": [
        {
            "baseyeaz": "2012",
            "yeaz": 2014,
            "month": "January",
            "item": "Rice - PDS",
            "index": "90.5",
            "inflation": null,
            "status": "F"
        },
        {
            "baseyeaz": "2012",
            "yeaz": 2014,
            "month": "January",
            "item": "Rice - Other Sources",
            "index": "123.7",
            "inflation": null,
```

6. Accessing the API through Python and Curl

6.1. The Python and CURL scripts for executing the API is available in the below GitHub repo:
`https://github.com/CCSPIDev/mospi_api_platform.git`

****