**Authorization And Authentication**

**Authentication -**  Authentication is the process of verifying the identity of a user. It ensures that the user requesting access to a resource is who. Authentication involves credentials such as username and password, tokens. for ex - logging into a system with a username and password.

**Authorization** - Authorization in postman is ensuring that the requests we send to an API are permitted by the server. Different types of authorization methods are required.

**Common Authorization**

**Ty pes in Postman :**

1.API Key

2.Bearer Token

3. Basic Auth

4.OAuth 1.0

5. OAuth 2.0

6. AWS Signature

7.Hawk Authentication

8. NTLM Authentication

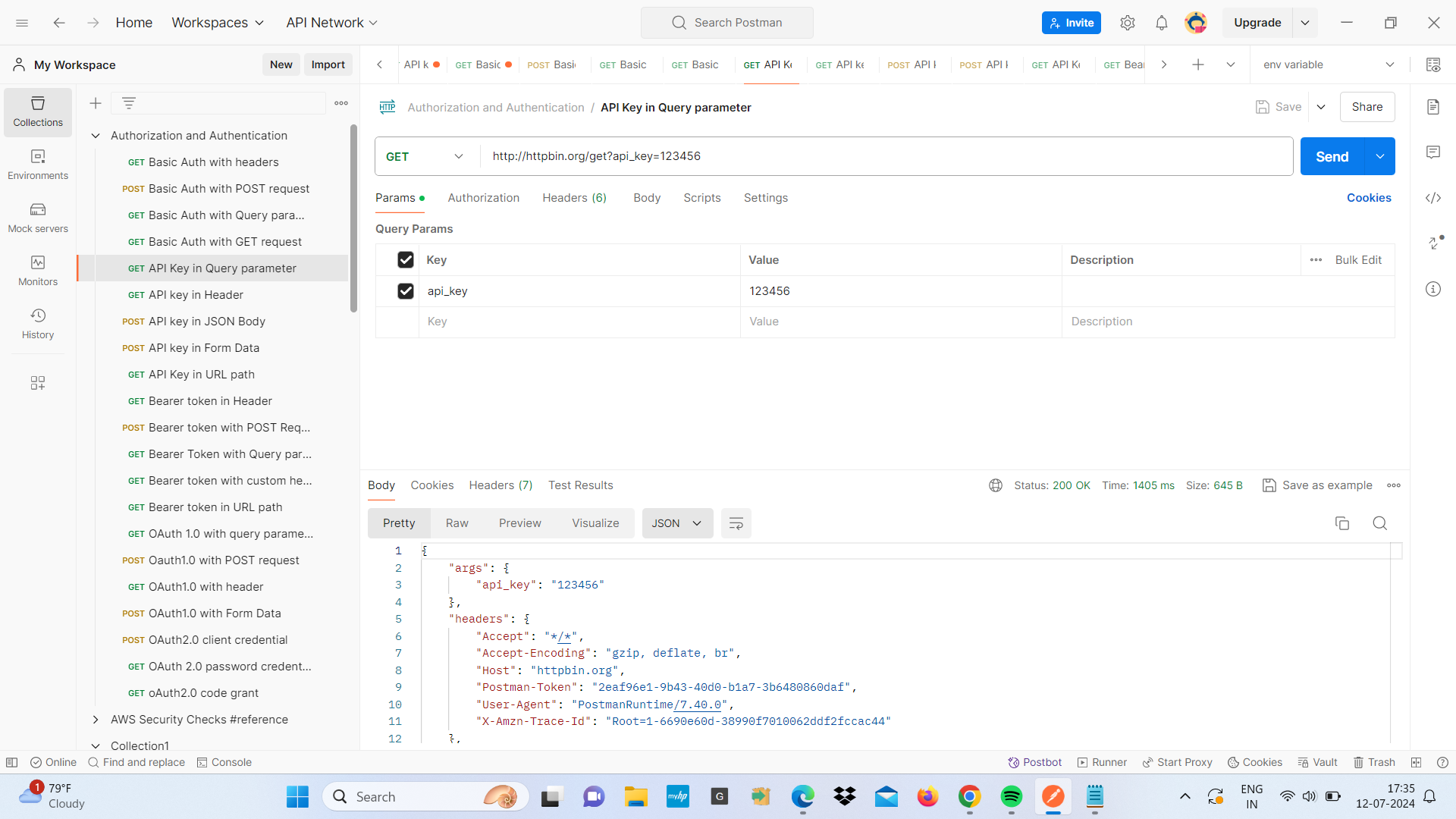
1. **API Key -**

* API key is a way to secure an API by requiring a unique key from anyone who wants to access the API. The key acts like a password and is used to identify.
* When we sign up for access to an API the service provides a unique API key This key is usually along letters.
* we include API key in our request
* The API key can be sent the query parameter in th URL
* If the key is invalid the server rejects the request and returns an error message like “401 unauthorized”

**Examples -**

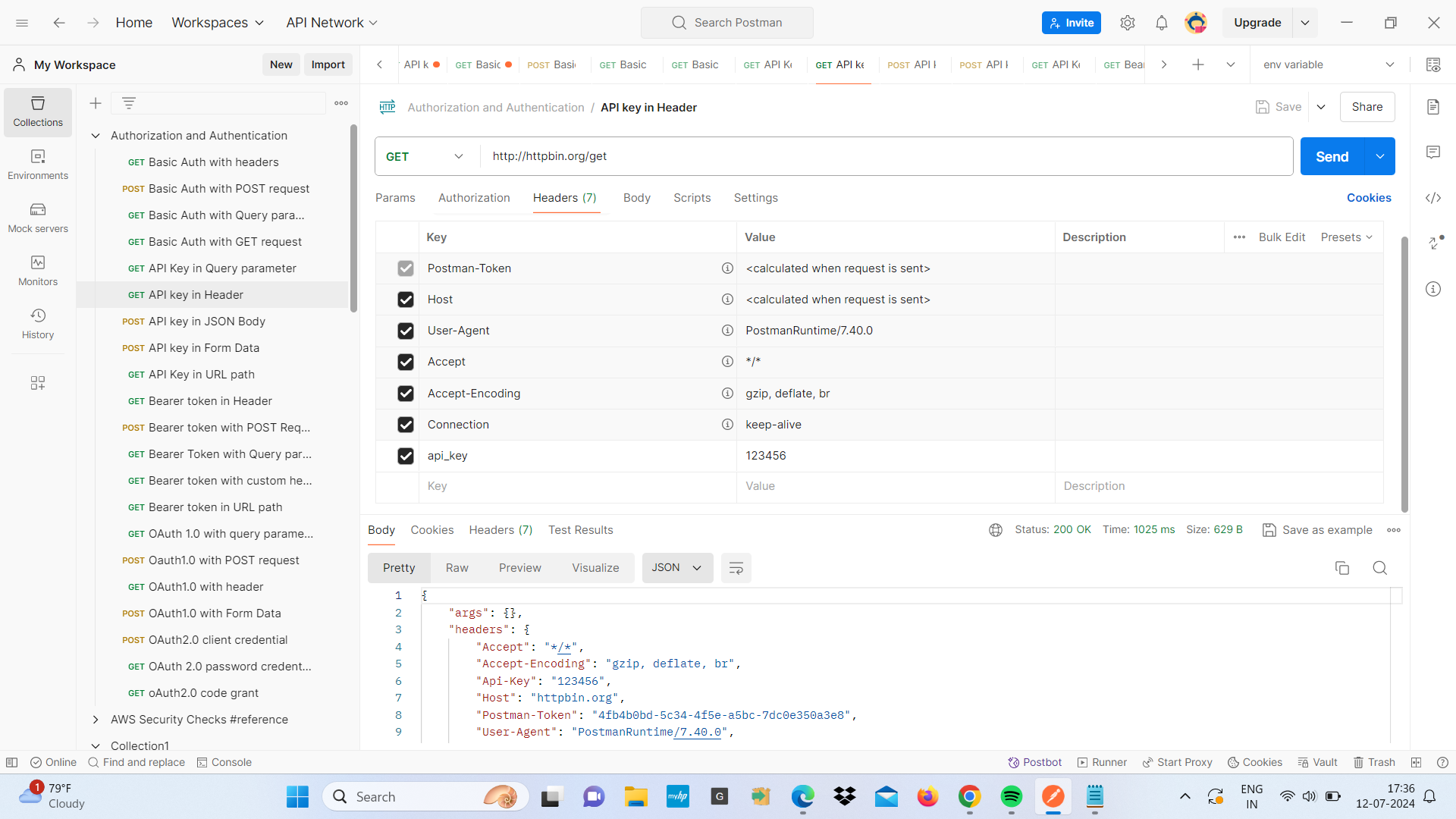
1. **API key in Query Parameter**

* Show authentication using an API key passed as a query parameter with “<https://httpbin.org/get>” API endpoint.
* we enter <https://httpbin.org/get> as the request URL.
* And add API key as query parameter “?api\_key=123456 to the end of the request
* Send the request and check the response body to confirm that the API key was received a query parameter
* The response should include the query parameter , authentication with API key

****

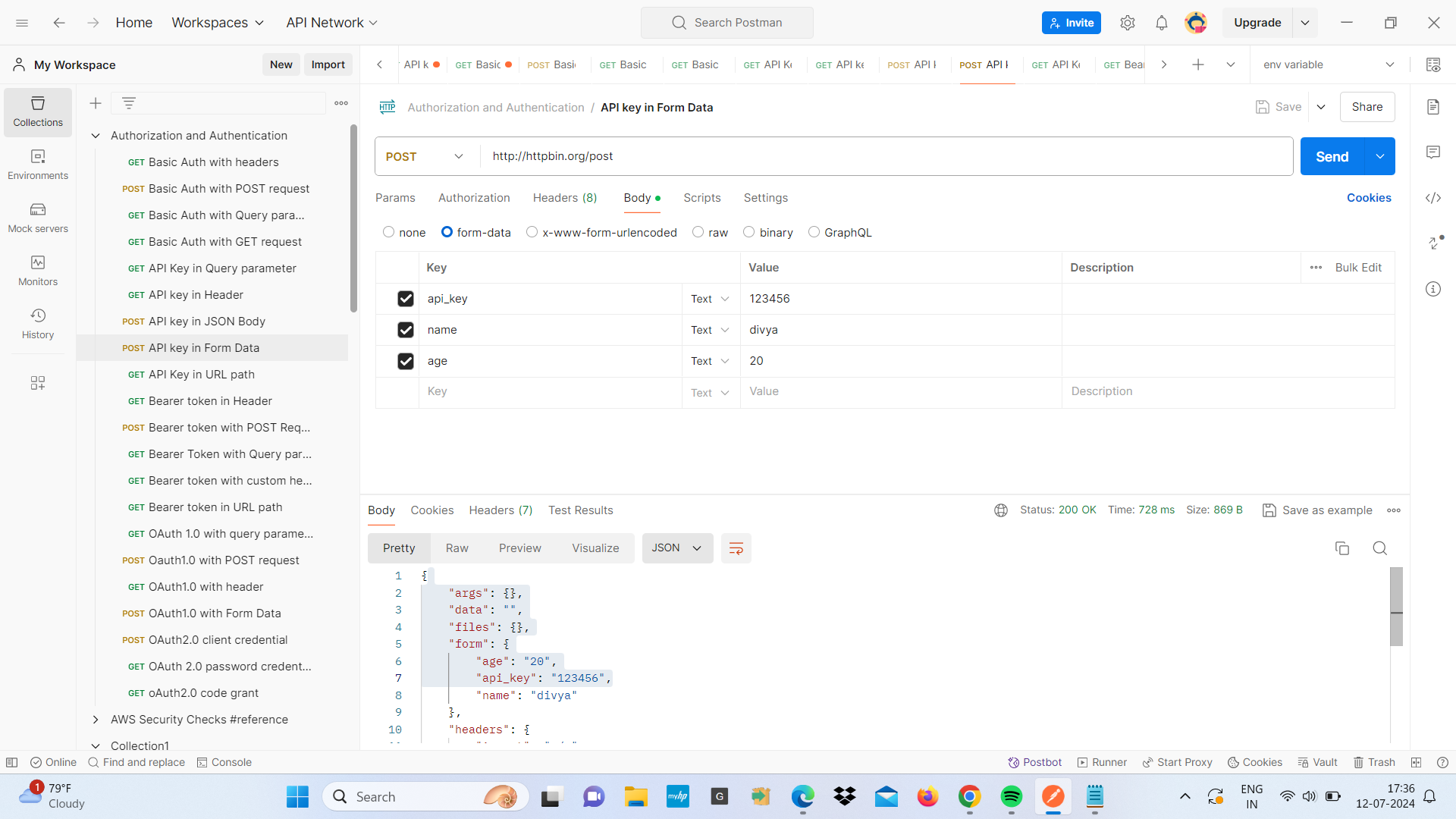
1. **API key in headers**

* show how to authentication requests to httpbin.org using an API key in the headers
* we enter <https://httpbin.org/get> as the request URL.
* click on the header and add header api\_key and key value 1 23456
* send the request and check the response body and the response should include the headers and API key

****

1. **API Key in Form Data**

* the purpose is how to authenticate requests to httpbin.org using an API key sent in form data
* we enter http://httpbin.org/post as the request URL.
* select the post method from the dropdown menu and click on the body and in body select form-data
* Add a key-value pair with api\_key api key value 123456 and name :divya and age:20
* Send the request and check the response.
* the response should include the form data we sent with API key

****

1. **API key in JSON**

* purpose is how to authenticate request to httpbin.org using an API key sent in jSON format
* we enter http://httpbin.org/post as the request URL.
* select the post method from the dropdown menu and click on the body and in body select raw and json as the body type
* Add API key to JSON body

{

"api\_key":"123456",

"name": "Divya",

"age":20

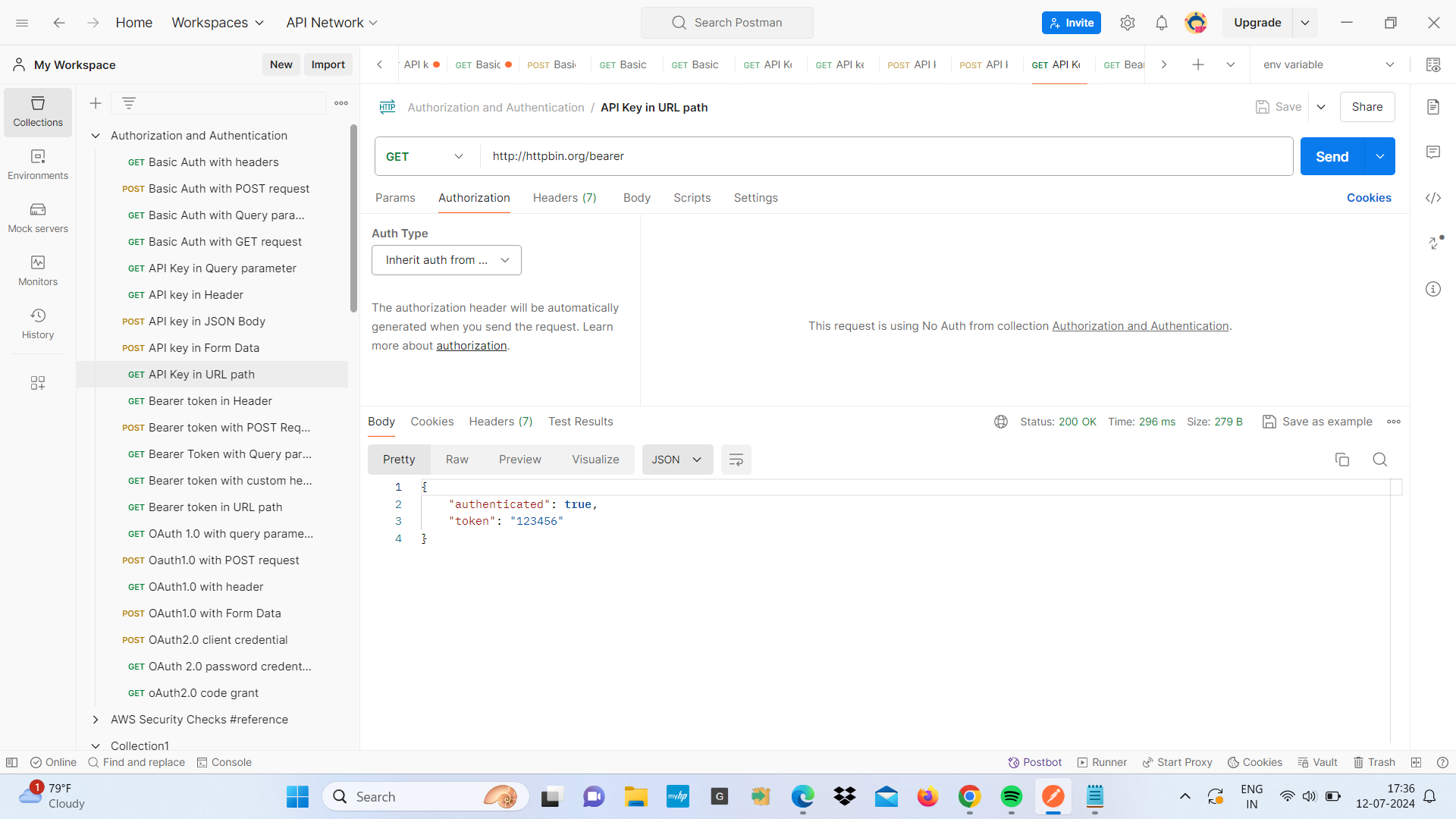
}

* send the request and check the response body and the response should include the header with API key

****

1. **API key in URL path**

* purpose how to authenticate the request to httpbin.org the API key in the URL path
* we enter <http://httpbin.org/bearer> as the request URL.
* select the get method from the dropdown menu
* send the request and check the response body to confirm that the API key was correctly processed
* The response should indicate that AP key was successful

****

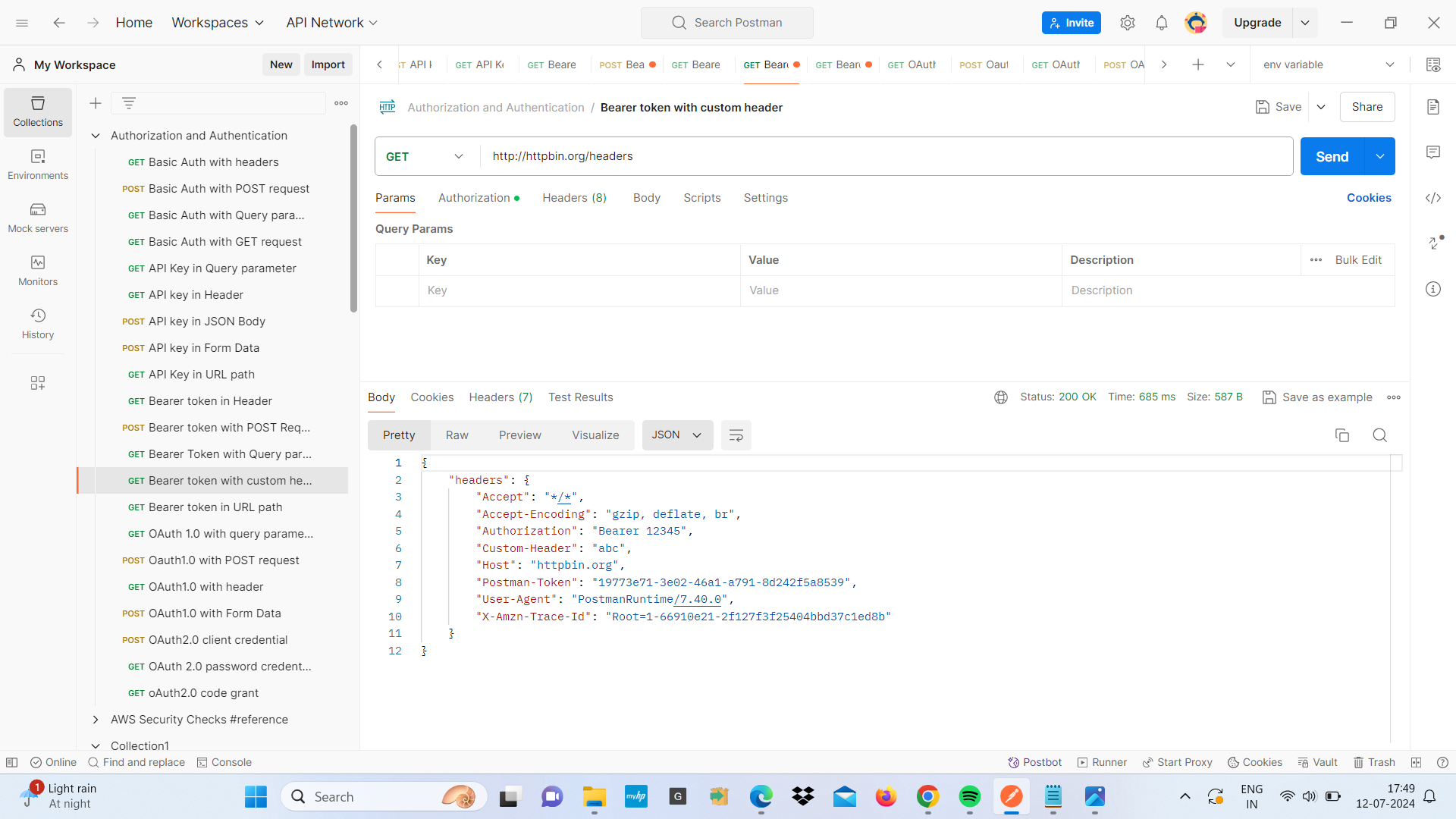
1. **Bearer Token -**

* In postman a bearer Token is a security token that allows the user to access an API endpoint. It is used to authenticate API requests and ensure that only authorized users can access resources.
* The Token is usually generated by the server and provided to the client
* we enter the URL of the API endpoints we want to access and choose the HTTP method
* And select the Bearer token and paste our bearer token in the provided field
* The token is typically a long string

**Examples -**

1. **Bearer token with custom -**

* the purpose is show how to authenticate requests using bearer token with a custom header
* We use <http://httpbin.org/headers> as a URL
* Select the HTTP method GET
* And click on the Header tab and add a custom key “abc”
* Send the request and check the response and the response should include that the custom header with the bearer token

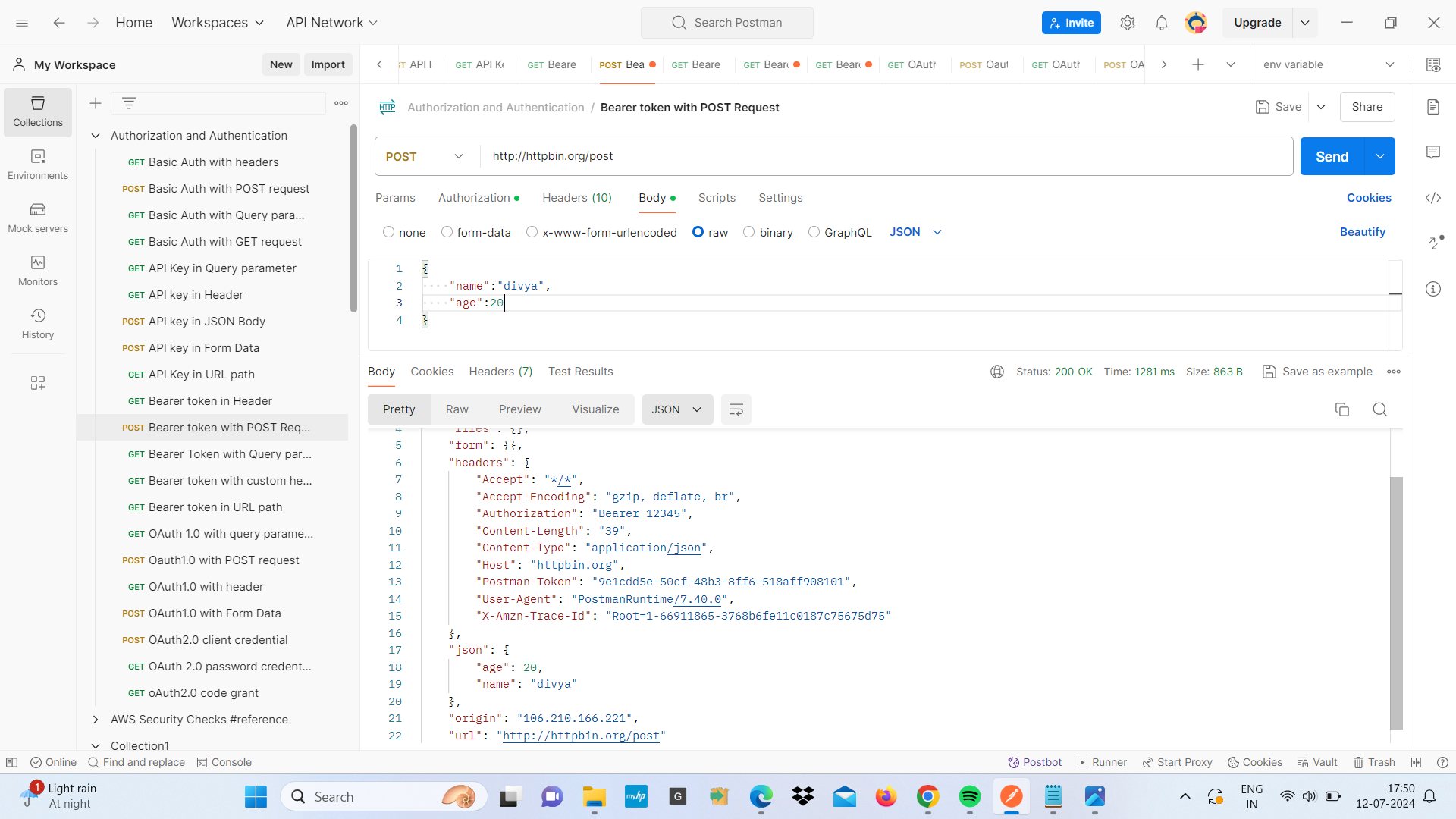


1. **bearer token with Query parameter -**

* The purpose is show how to include a bearer token as query parameter in request URL
* we use <http://httpbin.org> /get?name=divya & age=20 as URL
* choose the HTTP method GET
* Send the request and check the response and the response should show the token in the query parameter and header

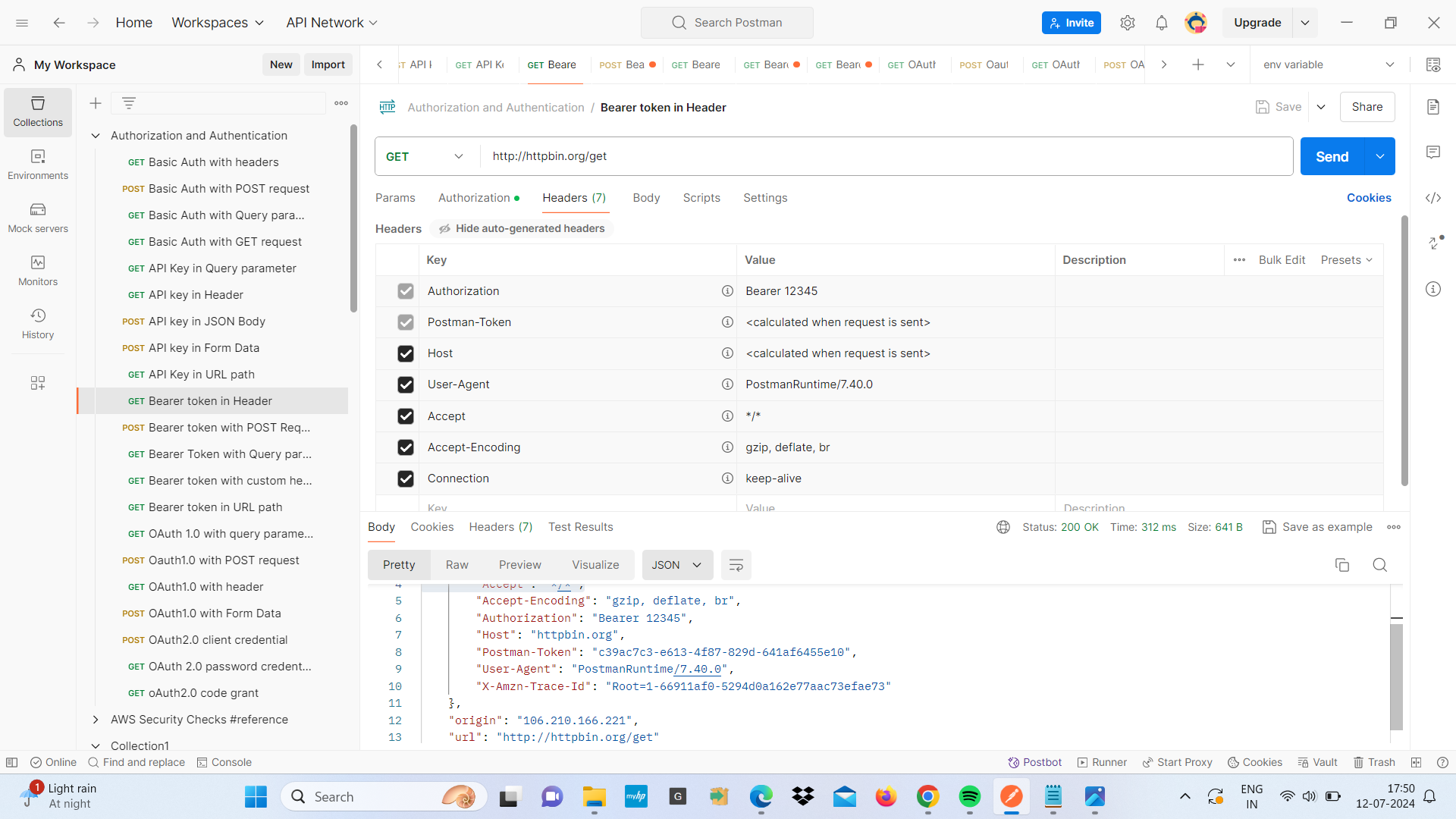


1. **bearer token with POST request -**

* The purpose is show how to authenticate a post request using bearer token
* We use <http://httpbin.org/post> as the request URL
* choose POST from the HTTP method from dropdown menu
* Add header with key Authorization and value
* In Body select raw and JSON and enter the data for your request payload
* Send the request and check the response and the response should confirm that request was successful

1. **bearer token in header -**

* the purpose is to authenticate request using a bearer token in header
* We use <http://httpbin.org/get> URL the request URL
* Select the HTTp method GET
* Add a header with key Authorization and value bearer12345
* send the request and check the response and the response should indicate that the bearer token was successful processed



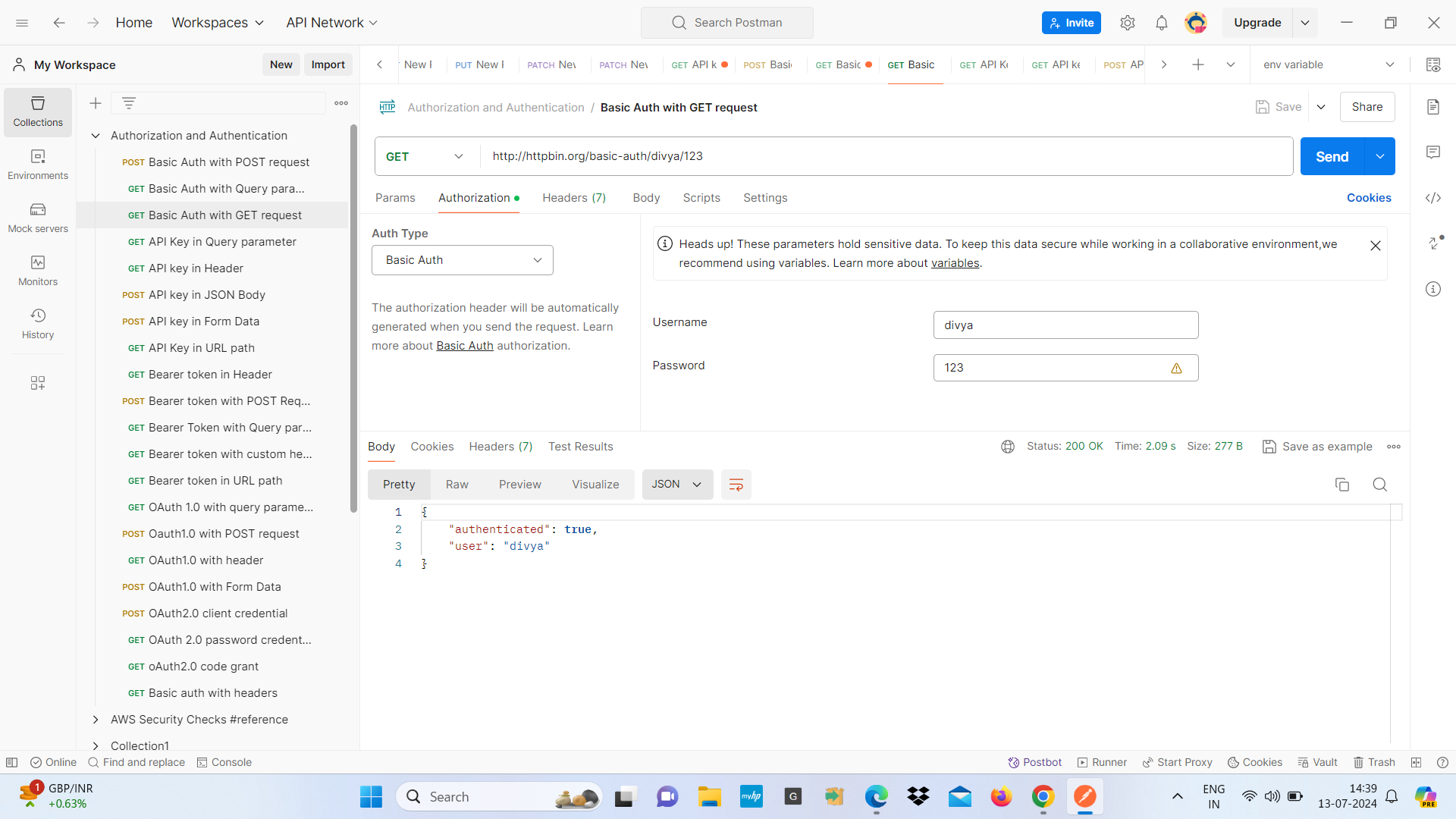
1. **Basic Authentication**

* Basic authentication is a simple method for sending username and password credentials with each request
* It’s a standard HTTP authentication method that involves base 64 encoding
* It’s widely supported
* It's easy to setup and use
* First we select BAsic Auth in the authorization tab
* Then we enter our username and password
* Send the request to retrieve user information

**Examples-**

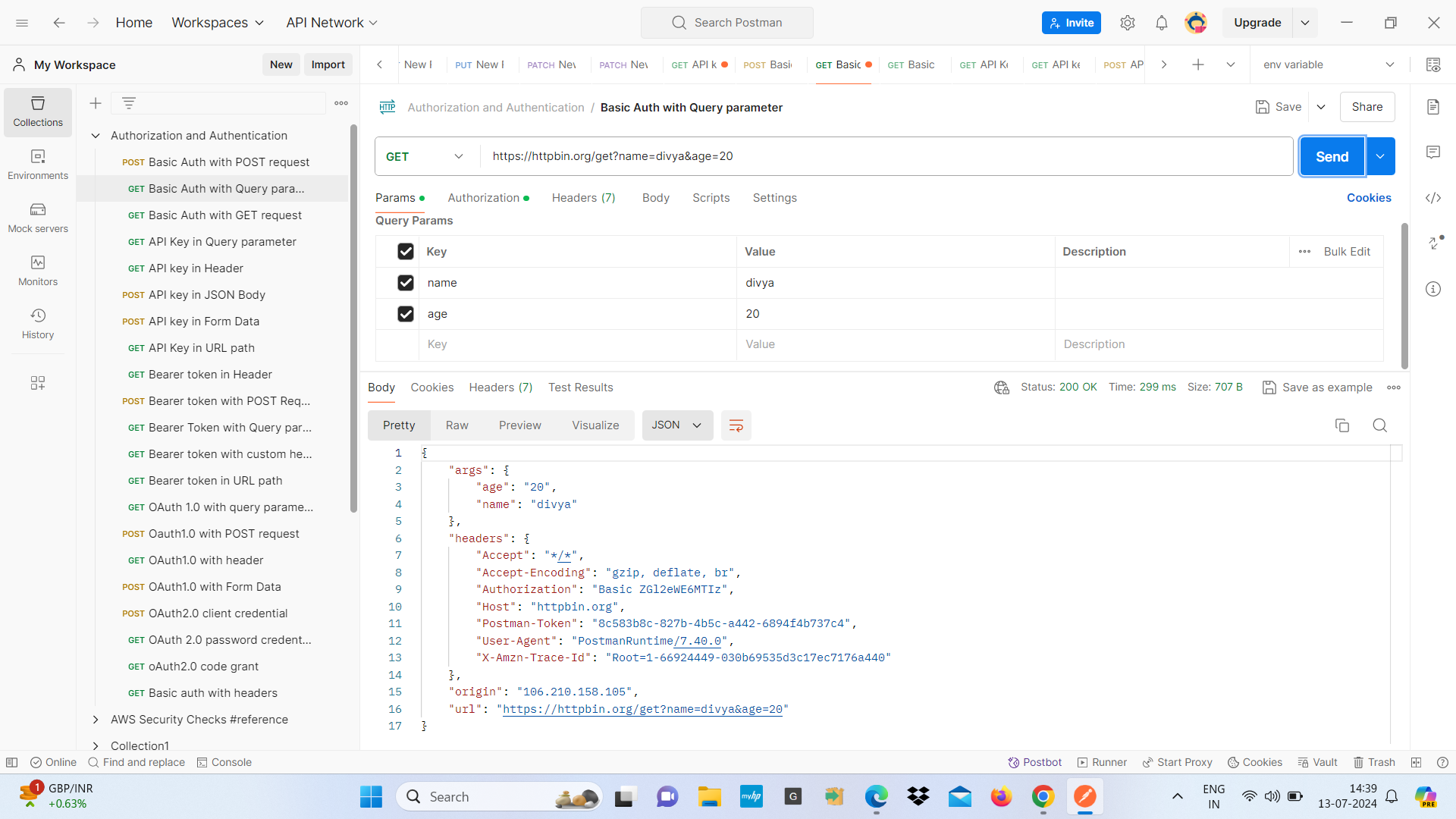
**1.Basic Auth with GET Method -**

* We use Basic Authentication with GET method for th URL “<http://httpbin.org/basic-auth/divya/123>”
* In the Request we select GET method and enter the URL “<http://httpbin.org/basic-auth/divya/123>”
* We setup Authentication so in the authorization tab select basic auth two fields will appear for username and password
* Enter “divya” as the “username” and “123” as “password”
* send the request and check the response section
* The server checks the credential and if they are correct it returns a response “authenticate”:true



**2. Basic Auth with Query parameter -**

* We will see how to include query parameter in the request and use basic authentication
* We set the request and select the GET method
* <https://httpbin.org/get> in the request URL
* Add query parameter on the “params” tab and add key value pair as query parameter
* key: name , value : divya
* key: age, value: 20
* our URL should now look like this : <https://httpbin.org/get?name=divya&age=20>
* Set up Authentication in the dropdown select “Basic Auth” enter our username and password
* Send the request and check the response



**3.Basic Auth with POST request-**

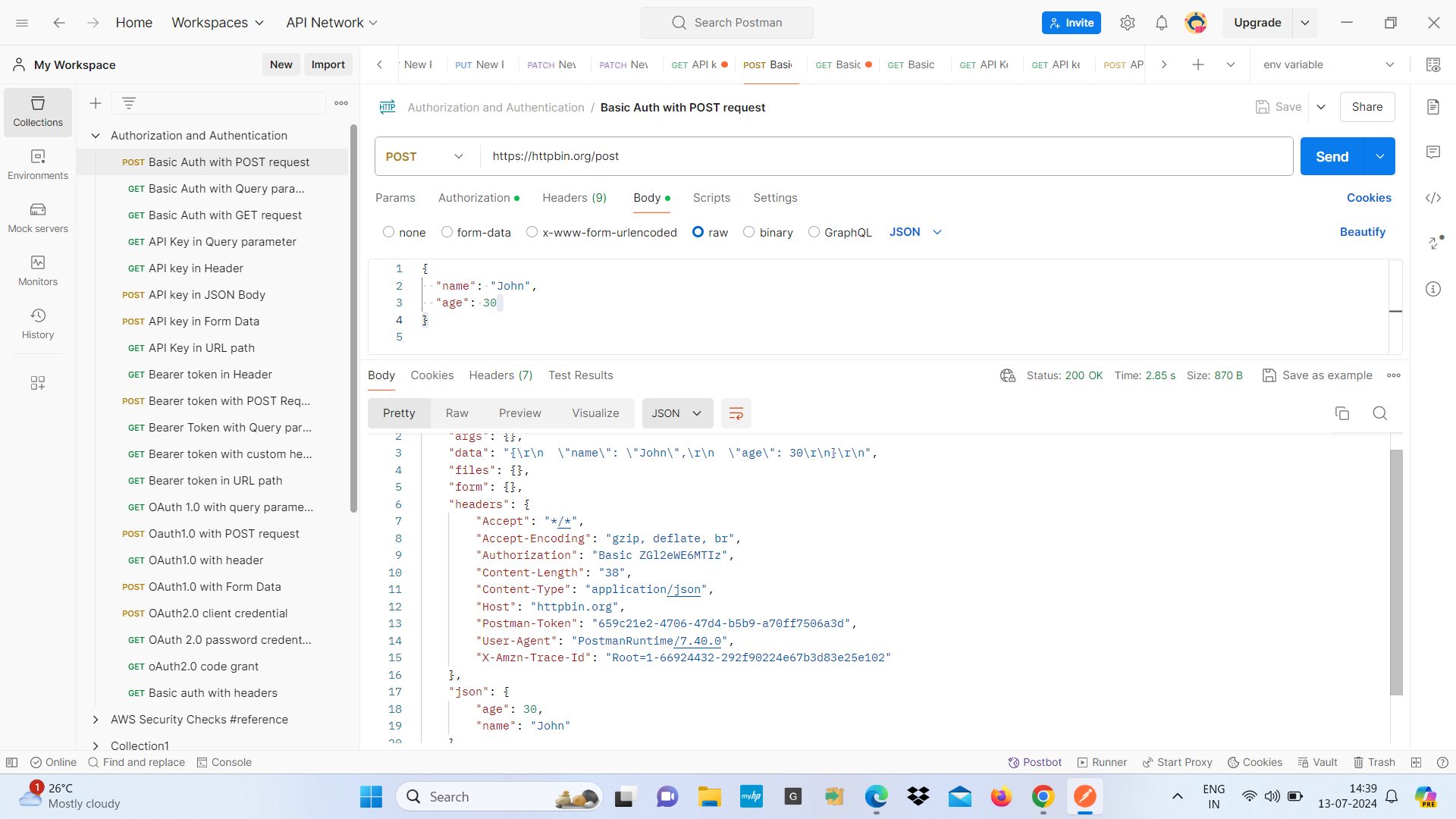
* By using Basic Authentication with POST method
* We select POST method and enter the URL “<https://httpbin.org/post>” in the request URL
* Set up Basic Authentication on the Authorization tab and select basic Auth enter our username and password
* Add request body select raw and choose JSON from the dropdown
* enter a JSON {

“name” : john,

“age”:30

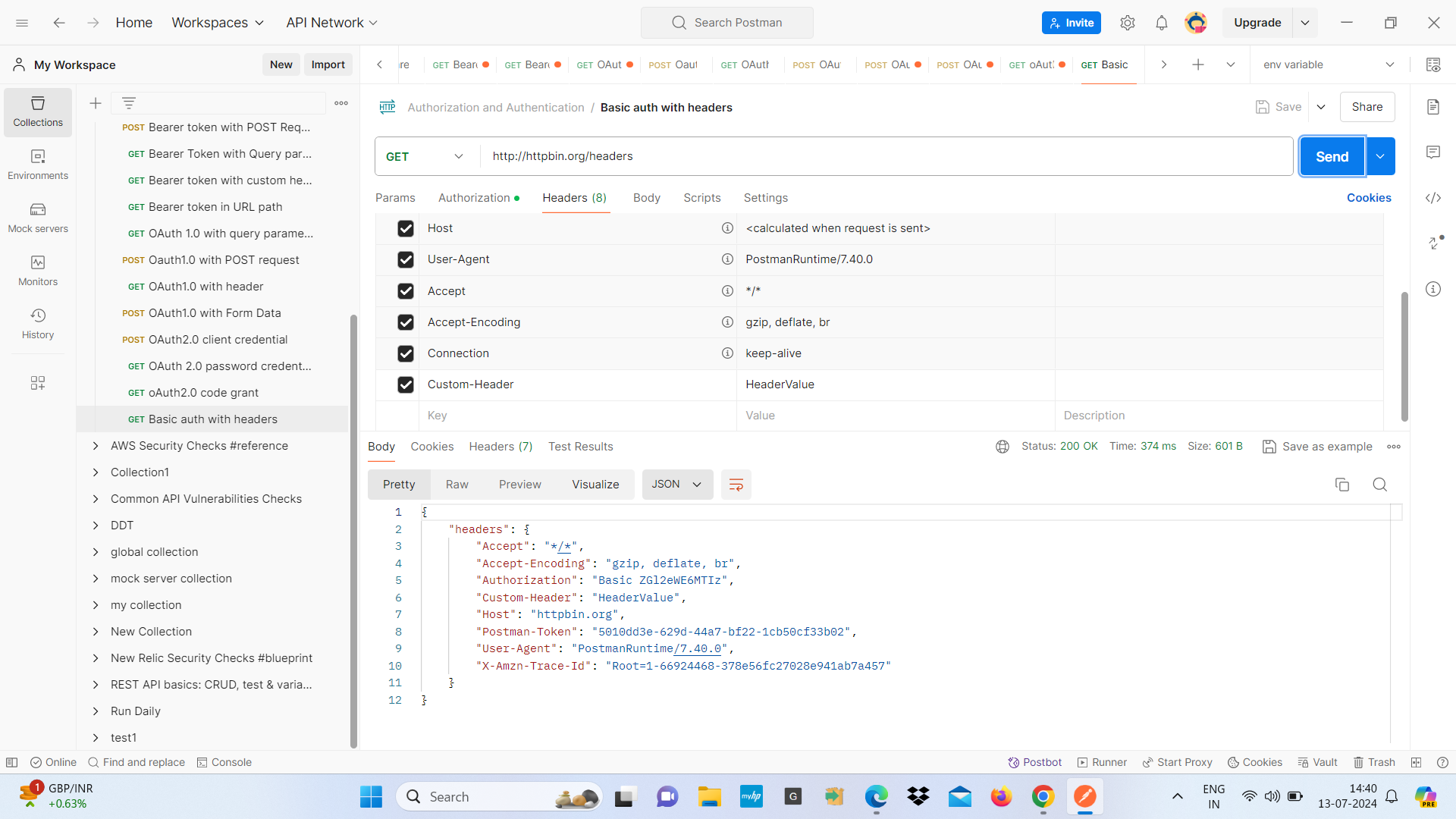
}

* Send the request and check the response
* response include the request body and headers sent in the request



**4.Basic Auth with headers -**

* Using basic Auth with headers
* Set the request GET method and enter the URL “<http://httpbin.org/headers> in the request URL filed
* Set up Basic Authentication on the Authorization tab and select basic Auth enter our username and password
* If we want to add custom header click on the header and add key and value
* key : Custom-header, value : HeaderValue
* Send the request and check the response
* response include the headers sent and custom header added

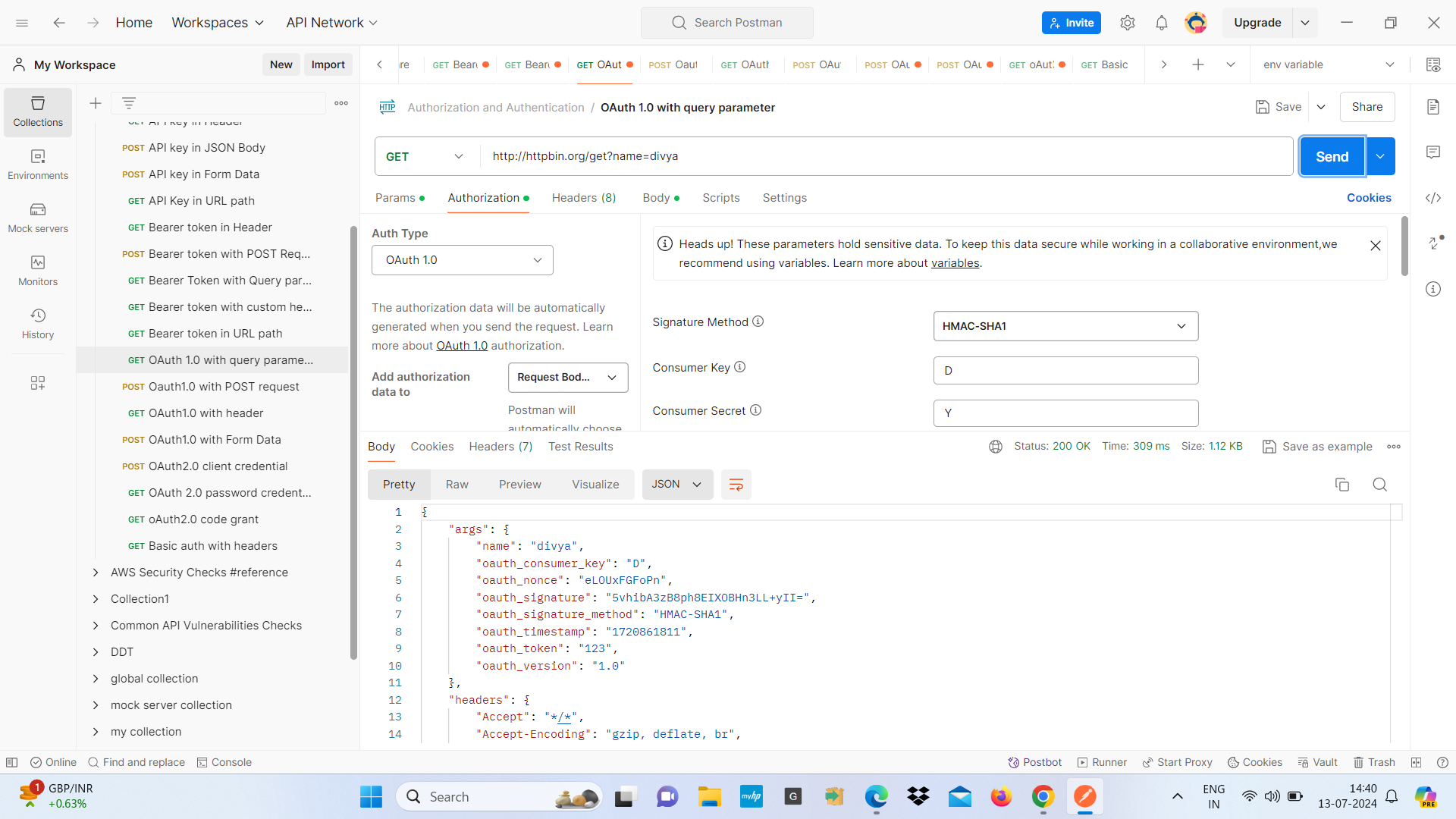


1. **OAuth 1.0**

* OAuth 1.0 is an authentication protocol that allows third party applications to access a user resources without sharing credentials
* It provide a secure way to grant access to an HTTP without exposing username and password

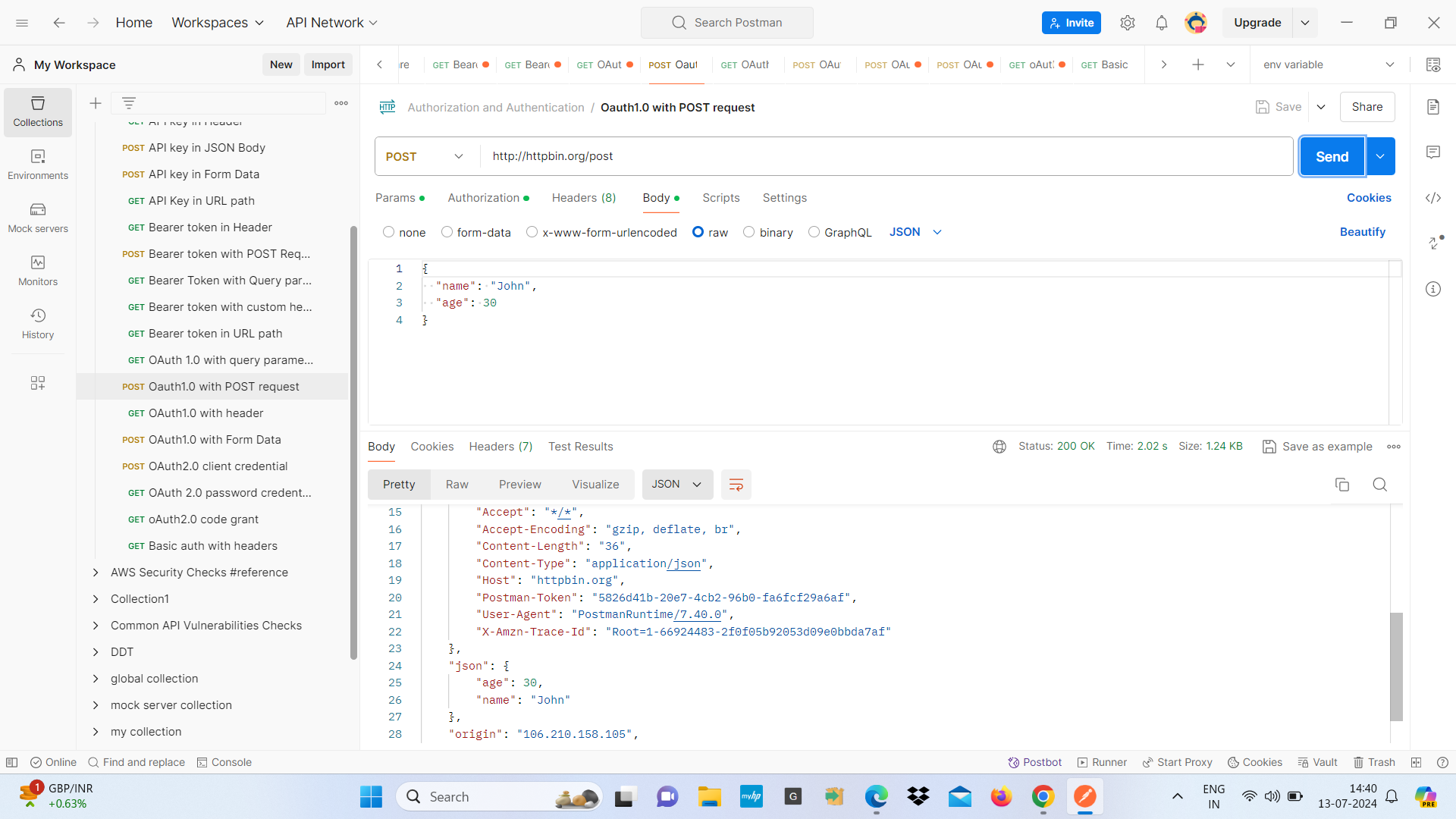
1. **OAuth 1.0 with Query Parameter**

* Using OAuth1.0 with Query parameter
* Set the request GET method and enter URL “<http://httpbin.org/get?name=divya>
* Set up Oauth1.0 on the authorization tab and select “OAUth 1.0 “ and fill required details
* Consume key : enter our consume key
* Consume secret : enter our consume secret
* Token : enter our token
* Token Secret : enter our token secret
* Send the request and check the response
* response include the headers sent and query parameter “name”

****

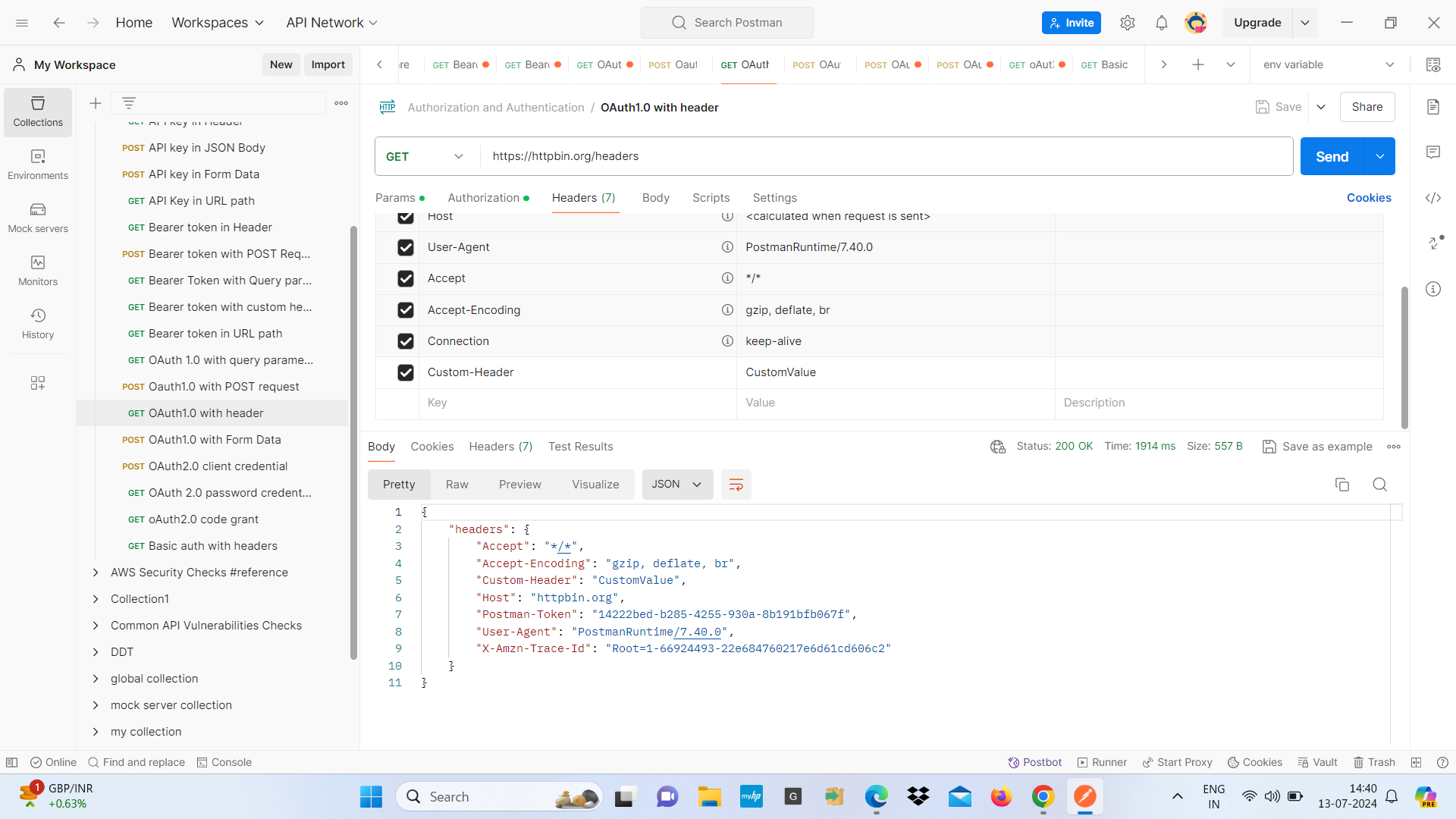
**2.OAuth 1.0 with POST request**

* Using OAuth 1.0 with a POST request for the URl <http://httpbin.org/post>
* we set the request POST method and enter the URL <http://httpbin.org/post> in the request URl field
* Set up OAuth 1.0 Authorization on the Authorization tab
* fill required details
* Consume key : enter our consume key
* Consume secret : enter our consume secret
* Token : enter our token
* Token Secret : enter our token secret
* Add request body we choose the format of our request body e.g (raw)
* Enter the necessary data for our POST request
* Send the request and check the response



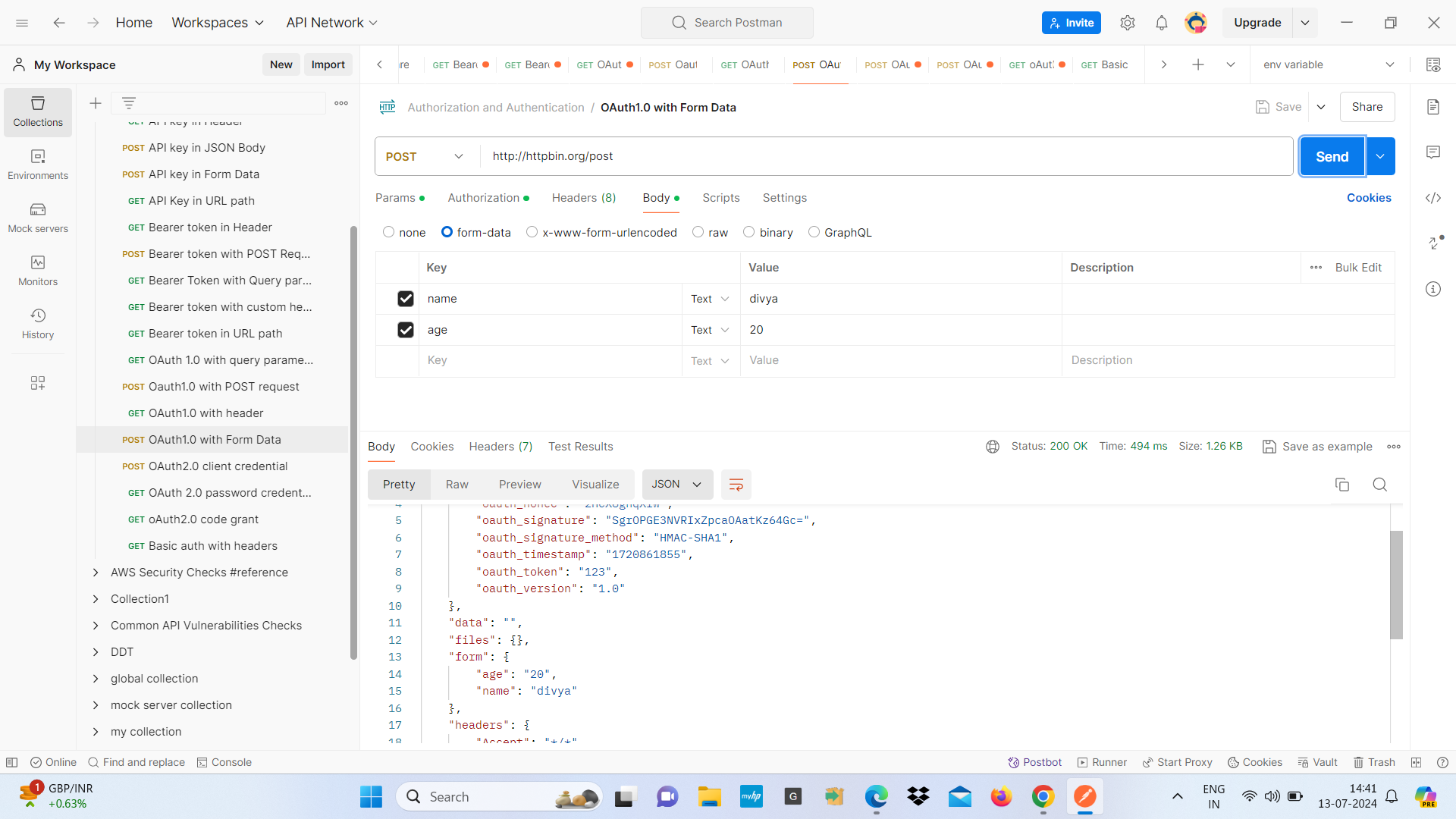
**3.OAuth 1.0 with header -**

* OAuth 1.0 authentication used with GET method to authenticate requests using tokens and secrets
* we set the request GET method and enter the URL <https://httpbin.org/headers> in the request URl field
* Set up OAuth 1.0 Authorization on the Authorization tab
* fill required details
* Consume key : enter our consume key
* Consume secret : enter our consume secret
* Token : enter our token
* Token Secret : enter our token secret
* Add custom Headers by adding key value pairs
* Key : Custom-Header
* Value : HeaderValue
* Send the request and check the response sows authorization header with OAuth parameters and custom headers



**4. OAUth 1.0 with Form Data**

* Using OAuth 1.0 with Form Data
* we set the request POST method and enter the URL <http://httpbin.org/post> in the request URl field
* Set up OAuth 1.0 Authorization on the Authorization tab
* fill required details
* Consume key : enter our consume key
* Consume secret : enter our consume secret
* Token : enter our token
* Token Secret : enter our token secret
* Add request body we choose Form data add key value pair for the form data
* name: divya
* age: 20
* Send the request and check the response
* The response includes the form data sent in the request



**5. OAuth 2.0**

* OAuth 2.0 allows us to share specific data with an application while keeping our login credentials private
* OAuth 2.0 is a secure way to authorize applications to access our data without sharing our login details.
* Our login credentials are not shared with the client
* we can grant and revoke access to our data

**Examples-**

1. **OAuth 2.0 password credentials grant**

* OAuth 2.0 password credentials grant method in which the username and password are exchanged directly for an access token
* We set the URl “<https://authorization-server.com/oauth/token>
* Set up OAuth 2.0 on Authorization tab and get New Access token
* Fill the required details
* token name: tokening
* Grant type : select “password credentials”
* Access token URL : <https://authorization-server.com/oauth/token>
* username: divya
* password : 123
* If the credentials are correct the authorization server will return an access token

