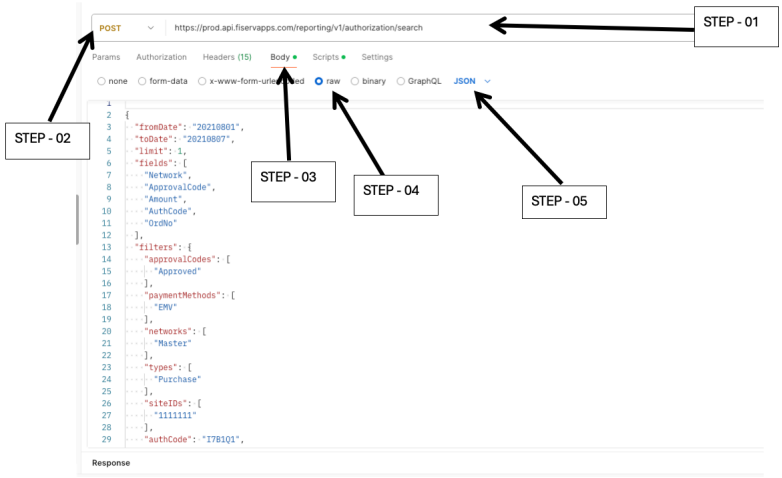
Following are the steps needed to use our Reporting APIs from Postman using HMAC Authentication Signature

01: Add the URL
Example: https://prod.api.fiservapps.com/reporting/v1/authorization/search

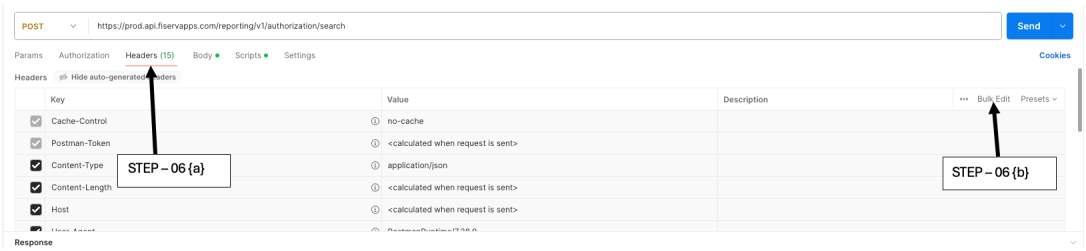
02: Select the HTTP Methods used in the restful webservices like **GET**, **POST**



03: In the Body section, enter the request Body used to make an API Call
04: The request body format should be **raw**

05: The request body format should be JSON as well.

06: Go to **headers {a}** section and click on **Bulk Edit {b}** as shown and enter the headers in **step 07**



07: Copy and paste the below headers as shown

Content-Type:Application/JsOn

Client-Request-Id:{{ClientRequestId}}

Api-Key:{{key}}

Timestamp:{{time}}

Authorization:{{signature}}

Auth-Token-Type:HMAC

08 : Go to **Scripts {a}** and select **Pre-request {b}** to Enter the **HMAC signature{c}** as follows:

```
var key = '{{YOUR KEY}}';

var secret = '{{YOUR SECRET}}';

var ClientRequestId = Math.floor((Math.random() * 10000000) + 1);

var time = new Date().getTime();

var method = request.method;

var requestBody = request.data;

var rawSignature = key + ClientRequestId + time + requestBody;

var computedHash = CryptoJS.algo.HMAC.create(CryptoJS.algo.SHA256,
secret.toString());

computedHash.update(rawSignature);

computedHash = computedHash.finalize();

var computedHmac = b64encode(computedHash.toString());


postman.setEnvironmentVariable('key', key);
```

```
postman.setEnvironmentVariable('time', time);

postman.setEnvironmentVariable('signature', computedHmac);

postman.setEnvironmentVariable('ClientRequestId', ClientRequestId)

function b64encode (input) {

  var swaps = ["A","B","C","D","E","F","G","H","I","J","K","L","M",

               "N","O","P","Q","R","S","T","U","V","W","X","Y","Z",

               "a","b","c","d","e","f","g","h","i","j","k","l","m",

               "n","o","p","q","r","s","t","u","v","w","x","y","z",

               "0","1","2","3","4","5","6","7","8","9","+","/"],

     tb, ib = "",

      output = "",

      i, L;

  for (i=0, L = input.length; i < L; i++) {

    tb = input.charCodeAt(i).toString(2);

    while (tb.length < 8) {

      tb = "0"+tb;

    }

    ib = ib + tb;

    while (ib.length >= 6) {

      output = output + swaps[parseInt(ib.substring(0,6),2)];

      ib = ib.substring(6);

    }

  }

  if (ib.length == 4) {

    tb = ib + "00";

    output += swaps[parseInt(tb,2)] + "=";

  }
```

```
  if (ib.length == 2) {
    tb = ib + "0000";
    output += swaps[parseInt(tb,2)] + "==";
  }
  return output;
}
```

09 : Now Click on **Send {a}** button to see the **desired response {b}**: