# 0403 - using csv2json and curl to post to an endpoint

Monday, 3 April 2023 10:21 PM

## Purpose

A very convenient way to push csv into CluedIn but via json arrays and an endpoint

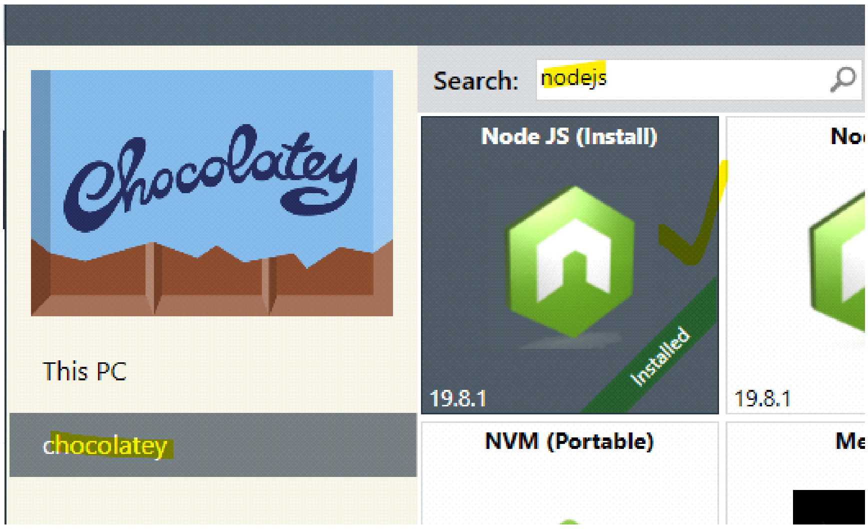
## Journal

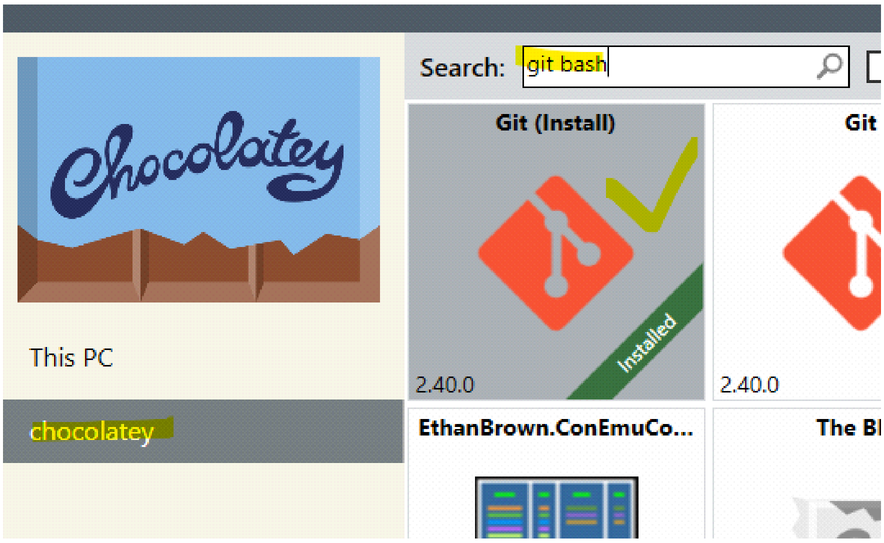
Install csv2json

It's a nodejs command line and library

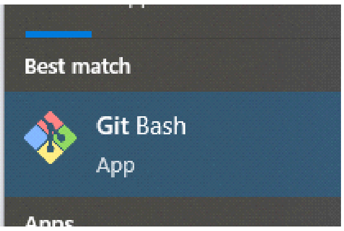
See
https://www.npmjs.com/package/csv2json
https://stackoverflow.com/questions/16831250/how-to-convert-csv-to-json-in-node-js#17190385

Install nodejs and git bash





Check install - run in git bash



```
rudi@PIP MINGW64 ~
$ which npm
/c/Program Files/nodejs/npm

rudi@PIP MINGW64 ~
$ which node
/c/Program Files/nodejs/node

rudi@PIP MINGW64 ~
$ npm --version
9.5.1

rudi@PIP MINGW64 ~
$ node --version
v19.8.1
```

Install csv2json

```
rudi@PIP MINGW64 ~
$ npm install --global csv2json

added 31 packages in 5s

2 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New minor version of npm available! 9.5.1 -> 9.6.3
npm notice Changelog: <https://github.com/npm/cli/releases/tag/v9.6.3>
npm notice Run `npm install -g npm@9.6.3` to update!
npm notice
```

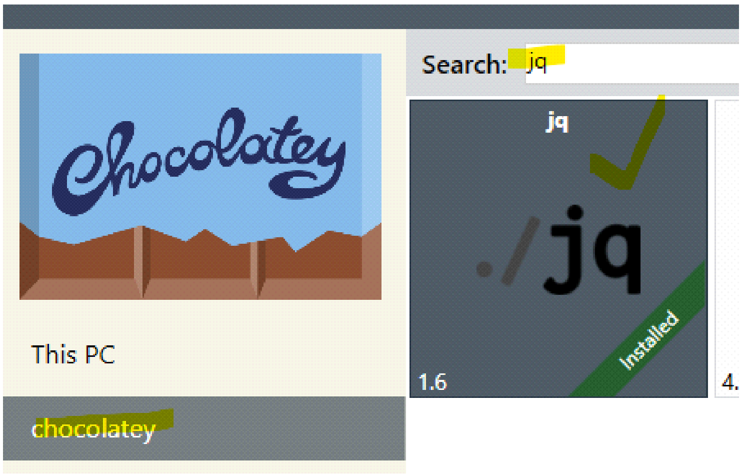
Check installed

```
rudi@PIP MINGW64 ~
$ csv2json -h
Usage: csv2json [OPTIONS] [<input file> [<output file>]]

  -d, --dynamic-typing
    Convert booleans and numeric to their type instead of strings.

  -s <separator>, --separator=<separator>
    Field separator to use (default to comma “,”).

  -t, --tsv
    Use tab as separator, overrides separator flag.

  <input file>
    CSV file to read data from.
    If unspecified or a dash (“-”), use the standard input.

  <output file>
    JSON file to write data to.
    If unspecified or a dash (“-”), use the standard output.

csv2json v2.0.2
```

Test with some sample data

```
rudi@PIP MINGW64 ~
$ cat sample_data.csv
id,first_name,last_name,email,gender
1,Terry,Brodley,tbrodley0@barnesandnoble.com,Male
2,Felicle,Wearne,fwearne1@cloudflare.com,Agender
3,Agata,Petyankin,apetyankin2@ca.gov,Female
4,Bird,O'Driscole,bodriscole3@dion.ne.jp,Female
5,Aleta,Overell,aoverell4@rambler.ru,Female
6,Philippa,Hazelby,phazelby5@sun.com,Female
7,Danie,Brecknock,dbrecknock6@stanford.edu,Male
8,Isador,Rowlands,irowlands7@gov.uk,Male
9,Tiler,Clewlowe,tclewlowe8@globo.com,Male
10,Obidiah,Dornin,odornin9@ucsd.edu,Male

rudi@PIP MINGW64 ~
$ head -2 sample_data.csv | csv2json
[
{"id":"1","first_name":"Terry","last_name":"Brodley","email":"tbrodley0
@barnesandnoble.com","gender":"Male"}
]
```

(optional) install jq and test resultant json array against it



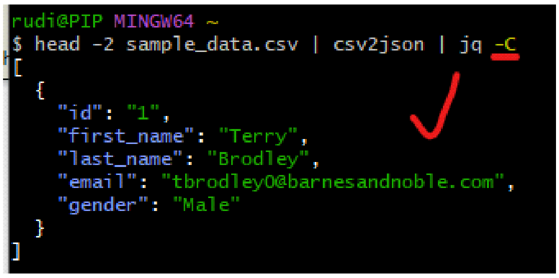
Test installed jq

```
rudi@PIP MINGW64 ~
$ which jq
/c/ProgramData/chocolatey/bin/jq

rudi@PIP MINGW64 ~
$ jq -h
jq - commandline JSON processor [version 1.6]

Usage:  C:\ProgramData\chocolatey\lib\jq\tools\jq.exe [options] <jq filter>
[file...]
        C:\ProgramData\chocolatey\lib\jq\tools\jq.exe [options] --args <jq filter>
[strings...]
        C:\ProgramData\chocolatey\lib\jq\tools\jq.exe [options] --jsonargs <jq
filter> [JSON_TEXTS...]

jq is a tool for processing JSON inputs, applying the given filter to
its JSON text inputs and producing the filter's results as JSON on
standard output.

The simplest filter is ., which copies jq's input to its output
unmodified (except for formatting, but note that IEEE754 is used
for number representation internally, with all that that implies).

For more advanced filters see the jq(1) manpage ("man jq")
and/or https://stedolan.github.io/jq

Example:

        $ echo '{"foo": 0}' | jq .
        {
                "foo": 0
        }

Some of the options include:
  -c               compact instead of pretty-printed output;
  -n               use `null` as the single input value;
  -e               set the exit status code based on the output;
  -s               read (slurp) all inputs into an array; apply filter to it;
  -r               output raw strings, not JSON texts;
  -R               read raw strings, not JSON texts;
  -C               colorize JSON;
  -M               monochrome (don't colorize JSON);
  -S               sort keys of objects on output;
  --tab            use tabs for indentation;
  --arg a v        set variable $a to value <v>;
  --argjson a v    set variable $a to JSON value <v>;
  --slurpfile a f  set variable $a to an array of JSON texts read from <f>;
  --rawfile a f    set variable $a to a string consisting of the contents of <f>;
  --args           remaining arguments are string arguments, not files;
  --jsonargs       remaining arguments are JSON arguments, not files;
  --               terminates argument processing;

Named arguments are also available as $ARGS.named[], while
positional arguments are available as $ARGS.positional[].

See the manpage for more options.
```

Run jq at the end of the pipe

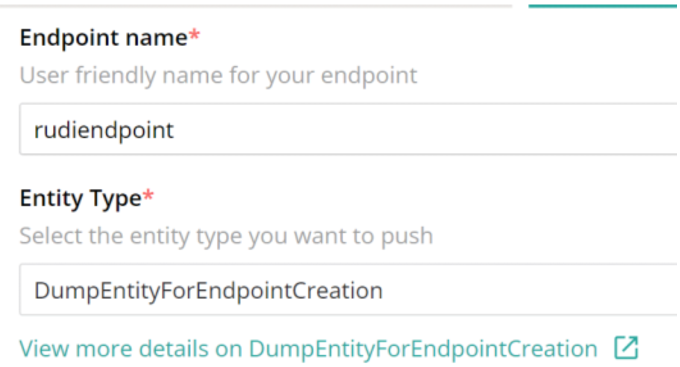
```
rudi@PIP MINGW64 ~
$ head -2 sample_data.csv | csv2json | jq
[
  {
    "id": "1",
    "first_name": "Terry",
    "last_name": "Brodley",
    "email": "tbrodley0@barnesandnoble.com",
    "gender": "Male"
  }
]
```

With colour!



```
rudi@PIP MINGW64 ~
$ head -2 sample_data.csv | csv2json | jq -C
[
  {
    "id": "1",
    "first_name": "Terry",
    "last_name": "Brodley",
    "email": "tbrodley0@barnesandnoble.com",
    "gender": "Male"
  }
]
```

Create endpoint



**Endpoint name***

User friendly name for your endpoint

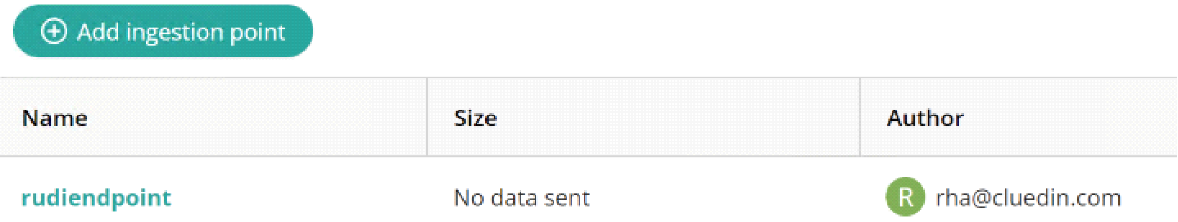
rudiendpoint

**Entity Type***

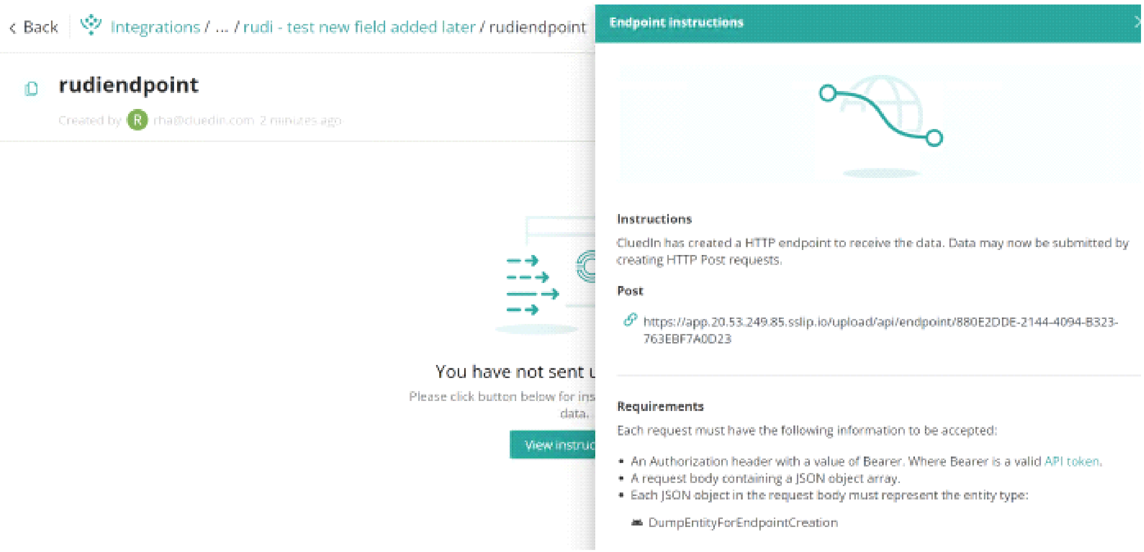
Select the entity type you want to push

DumpEntityForEndpointCreation

View more details on DumpEntityForEndpointCreation



| Name | Size | Author |
|---|---|---|
| rudiendpoint | No data sent | R rha@cluedin.com |



‹ Back
Integrations / ... / rudi - test new field added later / rudiendpoint

**rudiendpoint**

Created by R rha@cluedin.com 2 minutes ago

You have not sent u

Please click button below for ins data.

**Endpoint instructions**

**Instructions**

CluedIn has created a HTTP endpoint to receive the data. Data may now be submitted by creating HTTP Post requests.

**Post**

https://app.20.53.249.85.sslip.io/upload/api/endpoint/880E2DDE-2144-4094-B323-763EBF7A0D23

**Requirements**

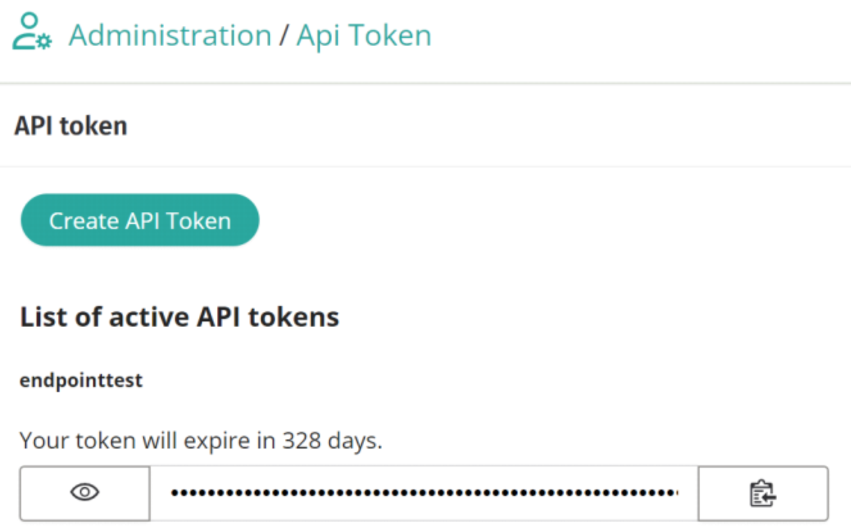
Each request must have the following information to be accepted:

- An Authorization header with a value of Bearer. Where Bearer is a valid API token.
- A request body containing a JSON object array.
- Each JSON object in the request body must represent the entity type:

DumpEntityForEndpointCreation

https://app.20.53.249.85.sslip.io/upload/api/endpoint/880E2DDE-2144-4094-B323-763EBF7A0D23

Create API token



Administration / Api Token

API token

List of active API tokens

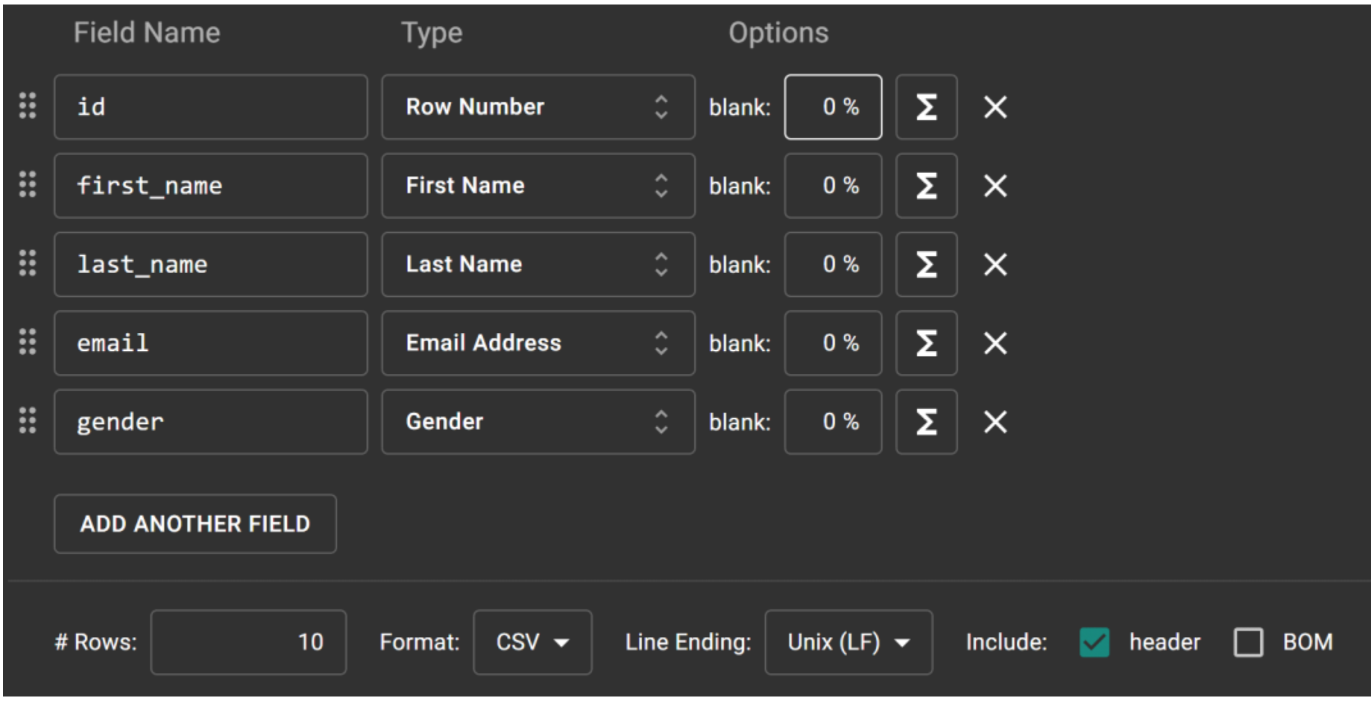
endpointtest

Your token will expire in 328 days.

```
eyJhbGciOiJSUzI1NiIsImtpZCI6Ijk2QjBEMjIzQ0E1ODNDNEIzRTE2RjZBRTFDQ0VBNThDMDJDMz
BFOUFSUzI1NiIsInR5cCI6ImF0K2p3dCIsIng1dCI6ImxyRFNJOHBZUEVzLUZ2YXVITTZsakFMRERwb
yJ9.eyJuYmYiOjE2NjUzNzMyMjIsImV4cCI6MTY5Njk1MjQyMSwiaXNzIjoiaHR0cHM6Ly9hcHAuM
jAuNTMuMjQ5Ljg1LnNzbGlwLmlvL2F1dGgvIiwiYXVkIjpbIlNlcnZlckFwaUZvclVJIiwiUHVibGljQXB
pIl0sImNsaWVudF9pZCI6IlB1YmxpY0FwaUNsaWVudCIsInJvbGUiOiJBUEkiLCJDbGllbnRJZCI6ImN
sdWVkaW5hcGFjZGVtbyIsIk9yZ2FuaXphdGlvbklkIjoiYjgxMTM0ZTMtMjdkOC00YmI1LTg0MzctYj
U4OTk1ODgxODRmIiwiT3JnYW5pemF0aW9uTmFtZSI6ImNsdWVkaW5hcGFjZGVtbyIsIk9yZ2Fu
aXphdGlvbkNsaWVudFJlZmVyZW5jZSI6ImNsdWVkaW5hcGFjZGVtbyIsIklkIjoiMzYzZGY5NzUtZTc
3OC00MTQ3LTk1MjItZDlmZGIwMWQzNTI4IiwiYXBpIjoiMjk1ZmE3OTItY2M5Yy00ZTNiLWJiM2
MtMTVlYWMyYTA2Y2ExIiwianRpIjoiQUUyNDY0MDEyQUZENTI1NjdDRTVEOThBNUI1NDc3QjYi
LCJpYXQiOjE2NjUzNzMyMjIsInNjb3BlIjpbIlB1YmxpY0FwaSIsIlNlcnZlckFwaUZvclVJIl19.R8AlKBm
vy3oysN9HCUXy9JPvN4xtM13EbMXGwxo4agCcD7dYTNjdsbY7YarHiv9SZvLpZGk-bDtl6h-
CnV1cmrZNQXib9HxwmeCl9pj-TTGq7o5l7GnqGVQrB8bJneVUGK_c1IHn5dMWpaYr4Nar7Z-
e1jTUcK4uwtfrZepBFhepb4wfnEJ9N7NlMx9OVTBhotjNBhFyQIq3CoBJYbF9K5-fn_al-
w6IJsmkTMM5m-ZtzXN23BL98XM8R9qDfdnYc3tW-15FIEek7jHc-
V7Yqoec7eLwgAqMtiOpVnxU5aEWT90NsZ03MUcwJ_9dubH6vHTTz84esaRAd2cYCcoO-A
```

Create 10 rows of sample data - referred to as sample data B

https://mockaroo.com/



| Field Name | Type | Options |
|---|---|---|
| id | Row Number | blank: 0 % |
| first_name | First Name | blank: 0 % |
| last_name | Last Name | blank: 0 % |
| email | Email Address | blank: 0 % |
| gender | Gender | blank: 0 % |

# Rows: 10 Format: CSV Line Ending: Unix (LF) Include: ☑ header □ BOM

```
$ cat sample_data_A.csv
id,first_name,last_name,email
1,Terry,Brodley,tbrodley0@barnesandnoble.com
```

```
2,Felicle,Wearne,fwearne1@cloudflare.com
3,Agata,Petyankin,apetyankin2@ca.gov
4,Bird,O'Driscole,bodriscole3@dion.ne.jp
5,Aleta,Overell,aoverell4@rambler.ru
6,Philippa,Hazelby,phazelby5@sun.com
7,Danie,Brecknock,dbrecknock6@stanford.edu
8,Isador,Rowlands,irowlands7@gov.uk
9,Tiler,Clewlowe,tclewlowe8@globo.com
10,Obidiah,Dornin,odornin9@ucsd.edu
```

Push sample data A to endpoint using csv2json and curl in a bash script

create script using endpoint URL and token details from above

post_to_endpoint.sh

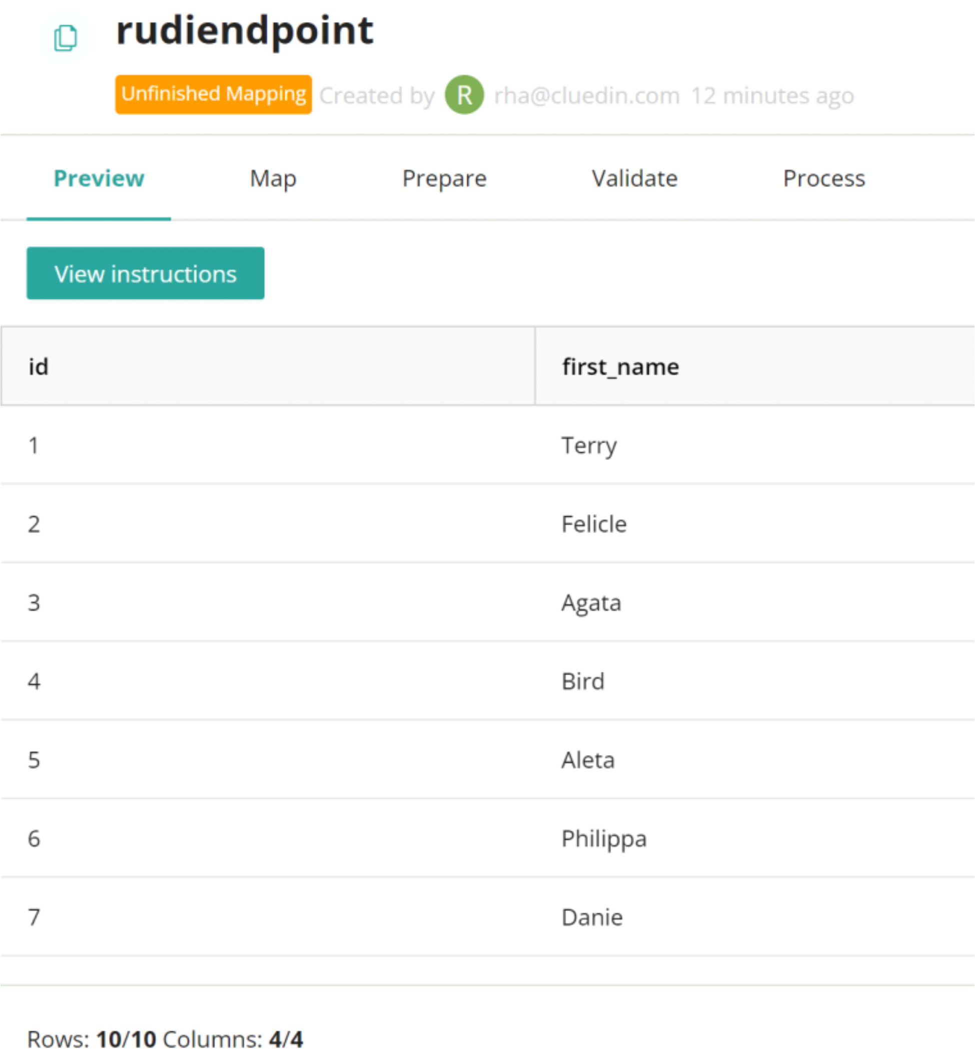
```
#!/bin/sh

URL=https://app.20.53.249.85.sslip.io/upload/api/endpoint/880E2DDE-2144-4094-B323-763EBF7A0D23

access_token="eyJhbGciOiJSUzI1NiIsImtpZCI6Ijk2QjBEMjIzQ0E1ODNDNEIzRTE2RjZBRTFDQ0VBNThDMDJDMzBFOUFSUzI1NiIsInR5cCI6ImF0K2p3dCIsIng1dCI6ImxyRFNJOHBZUEVzLUZ2YXVITTZsakFMRERwbyJ9.eyJuYmYiOjE2NjUzNzMyMjIsImV4cCI6MTY5Njk1MjQyMSwiaXNzIjoiaHR0cHM6Ly9hcHAuMjAuNTMuMjQ5Ljg1LnNzbGlwLmlvL2F1dGgvIiwiYXVkIjpbIlNlcnZlckFwaUZvclVJIiwiUHVibGljQXBpIl0sImNsaWVudF9pZCI6IlB1YmxpY0FwaUNsaWVudCIsInJvbGUiOiJBUEkiLCJDbGllbnRJZCI6ImNsdWVkaW5hcGFjZGVtbyIsIk9yZ2FuaXphdGlvbklkIjoiYjgxMTM0ZTMtMjdkOC00YmI1LTg0MzctYjU4OTk1ODgxODRmIiwiT3JnYW5pemF0aW9uTmFtZSI6ImNsdWVkaW5hcGFjZGVtbyIsIk9yZ2FuaXphdGlvbkNsaWVudFJlZmVyZW5jZSI6ImNsdWVkaW5hcGFjZGVtbyIsIklkIjoiMzYzZGY5NzUtZTc3OC00MTQ3LTk1MjItZDlmZGIwMWQzNTI4IiwiYXBpIjoiMjk1ZmE3OTItY2M5Yy00ZTNiLWJiM2MtMTVlYWMyYTA2Y2ExIiwianRpIjoiQUUyNDY0MDEyQUZENTI1NjdDRTVEOThBNUI1NDc3QjYiLCJpYXQiOjE2NjUzNzMyMjIsInNjb3BlIjpbIlB1YmxpY0FwaSIsIlNlcnZlckFwaUZvclVJIl19.R8AlKBmvy3oysN9HCUXy9JPvN4xtM13EbMXGwxo4agCcD7dYTNjdsbY7YarHiv9SZvLpZGk-bDtl6h-CnV1cmrZNQXib9HxwmeCl9pj-TTGq7o5l7GnqGVQrB8bJneVUGK_c1IHn5dMWpaYr4Nar7Z-e1jTUcK4uwtfrZepBFhepb4wfnEJ9N7NlMx9OVTBhotjNBhFyQIq3CoBJYbF9K5-fn_aI-w6IJsmkTMM5m-ZtzXN23BL98XM8R9qDfdnYc3tW-15FIEek7jHc-V7Yqoec7eLwgAqMtiOpVnxU5aEWT90NsZ03MUcwJ_9dubH6vHTTz84esaRAd2cYCcoO-A"

echo $URL
curl -v --location -g --request POST --no-progress-meter $URL \
      --header 'Content-Type: application/json' \
      --header "Authorization: Bearer $access_token" \
      --data-binary @-
```

run it

```
rudi@RudiWin10MBP MINGW64 ~/Documents/testdata/
$ csvtojson sample_data_A.csv
[
,"id":"1","first_name":"Terry","last_name":"Brodley","email":"tbrodley0
@barnesandnoble.com"}
,"id":"2","first_name":"Felicle","last_name":"Wearne","email":"fwearne1@cloudflare.com"}
,"id":"3","first_name":"Agata","last_name":"Petyankin","email":"apetyankin2@ca.gov"}
,"id":"4","first_name":"Bird","last_name":"O'Driscole","email":"bodriscole3@dion.ne.jp"}
,"id":"5","first_name":"Aleta","last_name":"Overell","email":"aoverell4@rambler.ru"}
,"id":"6","first_name":"Philippa","last_name":"Hazelby","email":"phazelby5@sun.com"}
,"id":"7","first_name":"Danie","last_name":"Brecknock","email":"dbrecknock6@stanford.edu"}
,"id":"8","first_name":"Isador","last_name":"Rowlands","email":"irowlands7@gov.uk"}
,"id":"9","first_name":"Tiler","last_name":"Clewlowe","email":"tclewlowe8@globo.com"}
{"id":"10","first_name":"Obidiah","last_name":"Dornin","email":"odornin9@ucsd.edu"}
]

rudi@RudiWin10MBP MINGW64 ~/Documents/testdata/
$ csvtojson sample_data_A.csv | ./post_to_endpoint.sh
https://app.20.53.249.85.sslip.io/upload/api/endpoint/880E2DDE-2144-4094-B323-763EBF7A0D23
{"success":true,"received":10,"warning":false,"error":false,"message":{}}Note: Unnecessary
use of -X or --request, POST is already inferred.
*   Trying 20.53.249.85:443...
* Connected to app.20.53.249.85.sslip.io (20.53.249.85) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* successfully set certificate verify locations:
*  CAfile: C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
*  CApath: none
} [5 bytes data]
* TLSv1.3 (OUT), TLS handshake, Client hello (1):
} [512 bytes data]
* TLSv1.3 (IN), TLS handshake, Server hello (2):
{ [122 bytes data]
* TLSv1.3 (IN), TLS handshake, Encrypted Extensions (8):
{ [19 bytes data]
* TLSv1.3 (IN), TLS handshake, Certificate (11):
{ [4233 bytes data]
* TLSv1.3 (IN), TLS handshake, CERT verify (15):
{ [264 bytes data]
* TLSv1.3 (IN), TLS handshake, Finished (20):
{ [36 bytes data]
* TLSv1.3 (OUT), TLS change cipher, Change cipher spec (1):
} [1 bytes data]
* TLSv1.3 (OUT), TLS handshake, Finished (20):
} [36 bytes data]
```

```
* SSL connection using TLSv1.3 / TLS_AES_128_GCM_SHA256
* ALPN, server accepted to use h2
* Server certificate:
*  subject: CN=cluedinapacdemo.20.53.249.85.sslip.io
*  start date: Nov 15 21:03:05 2022 GMT
*  expire date: Feb 13 21:03:04 2023 GMT
*  subjectAltName: host "app.20.53.249.85.sslip.io" matched cert's
"app.20.53.249.85.sslip.io"
*  issuer: C=US; O=Let's Encrypt; CN=R3
*  SSL certificate verify ok.
* Using HTTP2, server supports multiplexing
* Connection state changed (HTTP/2 confirmed)
* Copying HTTP/2 data in stream buffer to connection buffer after upgrade: len=0
* Using Stream ID: 1 (easy handle 0x22a113d20b0)
} [5 bytes data]
> POST /upload/api/endpoint/880E2DDE-2144-4094-B323-763EBF7A0D23 HTTP/2
> Host: app.20.53.249.85.sslip.io
> user-agent: curl/7.78.0
> accept: */*
> content-type: application/json
> authorization: Bearer
eyJhbGciOiJSUzI1NiIsImtpZCI6Ijk2QjBEMjIzQ0E1ODNDNEIzRTE2RjZBRTFDQ0VBNThDMDJDMzBFOUFSUzI1Ni
IsInR5cCI6ImF0K2p3dCIsIng1dCI6ImxyRFNJOHBZUEVzLUZ2YXVITTZsakFMRERwbyJ9.eyJuYmYiOjE2NjUzNzM
yMjIsImV4cCI6MTY5Njk1MjQyMSwiaXNzIjoiaHR0cHM6Ly9hcHAuMjAuNTMuMjQ5Ljg1LnNzbGlwLmlvL2F1dGgvI
iwiYXVkIjpbIlNlcnZlckFwaUZvclVJIiwiUHVibGljQXBpIl0sImNsaWVudF9pZCI6IlB1YmxpY0FwaUNsaWVudCI
sInJvbGUiOiJBUEkiLCJDbGllbnRJZCI6ImNsdWVkaW5hcGFjZGVtbyIsIk9yZ2FuaXphdGlvbklkIjoiYjgxMTM0Z
TMtMjdkOC00YmI1LTg0MzctYjU4OTk1ODgxODRmIiwiT3JnYW5pemF0aW9uTmFtZSI6ImNsdWVkaW5hcGFjZGVtbyI
sIk9yZ2FuaXphdGlvbkNsaWVudFJlZmVyZW5jZSI6ImNsdWVkaW5hcGFjZGVtbyIsIklkIjoiMzYzZGY5NzUtZTc3O
C00MTQ3LTk1MjItZDlmZGIwMWQzNTI4IiwiYXBpIjoiMjk1ZmE3OTItY2M5Yy00ZTNiLWJiM2MtMTVlYWMyYTA2Y2E
xIiwianRpIjoiQUUyNDY0MDEyQUZENTI1NjdDRTVEOThBNUI1NDc3QjYiLCJpYXQiOjE2NjUzNzMyMjIsInNjb3BlI
jpbIlB1YmxpY0FwaSIsIlNlcnZlckFwaUZvclVJIl19.R8AlKBmvy3oysN9HCUXy9JPvN4xtM13EbMXGwxo4agCcD7
dYTNjdsbY7YarHiv9SZvLpZGk-bDtl6h-CnV1cmrZNQXib9HxwmeCl9pj-
TTGq7o5l7GnqGVQrB8bJneVUGK_c1IHn5dMWpaYr4Nar7Z-
e1jTUcK4uwtfrZepBFhepb4wfnEJ9N7NlMx9OVTBhotjNBhFyQIq3CoBJYbF9K5-fn_aI-w6IJsmkTMM5m-
ZtzXN23BL98XM8R9qDfdnYc3tW-15FIEek7jHc-V7Yqoec7eLwgAqMtiOpVnxU5aEWT90NsZ03MUcwJ_
9dubH6vHTTz84esaRAd2cYCcoO-A
> content-length: 904
>
} [5 bytes data]
* We are completely uploaded and fine
{ [5 bytes data]
* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
{ [57 bytes data]
* TLSv1.3 (IN), TLS handshake, Newsession Ticket (4):
{ [57 bytes data]
* old SSL session ID is stale, removing
{ [5 bytes data]
< HTTP/2 200
< x-powered-by: CluedIn
< access-control-allow-origin: *
< content-type: application/json; charset=utf-8
< content-length: 73
< etag: W/"49-gESlYemHHTcvgFUr56NgwSIXHx4"
< date: Tue, 15 Nov 2022 22:47:06 GMT
< strict-transport-security: max-age=15768000
<
{ [73 bytes data]
* Connection #0 to host app.20.53.249.85.sslip.io left intact
```

Observe endpoint in cluedin UI



rudiendpoint

Unfinished Mapping Created by R rha@cluedin.com 12 minutes ago

| id | first_name |
|---|---|
| 1 | Terry |
| 2 | Felicle |
| 3 | Agata |
| 4 | Bird |
| 5 | Aleta |
| 6 | Philippa |
| 7 | Danie |

Rows: **10/10** Columns: **4/4**