

Example THREE - Using APIs to enrich your data

Introduction

On RaidForums, the famous cybercrime platform, you collect an interesting dataset: a list of names, credit card numbers, emails. You want to check with bank needs to be alerted by identifying the credit card issuer.

The dataset is `allworldcards.csv`.

This can be done by using the API of [BinList](#).

Using the command line tool `curl`, you can request the website through its API as shown below:

```
curl -H "Accept-Version: 3" "https://lookup.binlist.net/45717360"
```

So, just by using the first eight digits of a credit card, <https://lookup.binlist.net/45717360> will give a json file, quite handy to manipulate with OpenRefine.

Please note that this service is not updated anymore and will be used for demo purposes

Another way to do it is to use this website : <https://bincheck.io/fr/>

First : do a manual request to get the URL and the token.

Let's refine!

Create the project

- Import the data in openRefine. Be sure that the first line of the dataset is not considered as a header.
- Rename the first column as "Number".
- Create a new column based on this column, that you name "url":

The expression will be

```
"https://lookup.binlist.net/"+value.slice(0,8)
```

It means **"Concatenate the url "https://lookup.binlist.net/" and the value of the number, by only taking the first 8 digits"**.

5037 rows will get: https://lookup.binlist.net/
l'+value.slice(0,8) Undo

5037 rows												
Show as: rows records Show: 5 10 25 50 rows												
All	Number	url	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10	Column 11
1	370784037131001	https://lookup.binlist.net/37078403	12/23	4265	NAME	MEXICO	STATE	CITY	ADD	ZIP	Column	Column
2	374950102043019	https://lookup.binlist.net/37495010	04/24	5785	catherine calvo	FRANCE		Lamorlaye	4 allÃ©e du grand jardin	60260	0661193067 EMAIL	
3	4163570120654535	https://lookup.binlist.net/41635701	08/22	678	Jozef Michels	NETHERLANDS	STATE	CITY	ADDRESS	5625	jrn-michels@hotmail.com	
4	4163570154289398	https://lookup.binlist.net/41635701	05/22	102	N.J. Reynoso Perez	NETHERLANDS	Eindhoven	Noord brabant	Sonseweg	5632 NL	0622829044	
5	4350441573270471	https://lookup.binlist.net/43504415	12/24	687	Felipe Carrion	FRANCE	STATE	CITY	ADDRESS	73200	avervara@gmail.com	
6	4372033706505420	https://lookup.binlist.net/43720337	11/22	187	Stephane Huet	FRANCE	STATE	CITY	ADDRESS	49270	stefanhuet49@gmail.com	
7	4875011101376870	https://lookup.binlist.net/48750111	12/22	907	ali Saeedi	NETHERLANDS	STATE	CITY	ADDRESS	1067sl	alisaeedi652@gmail.com	
8	4875128642200003	https://lookup.binlist.net/48751286	08/24	257	fania avrit	NETHERLANDS	STATE	CITY	ADDRESS	ZIP	phone	
9	4875128642200086	https://lookup.binlist.net/48751286	01/23	671	myranda oralia	NETHERLANDS	STATE	CITY	ADDRESS	ZIP	phone	
10	4875128642200250	https://lookup.binlist.net/48751286	02/23	066	ranna imogene	NETHERLANDS	STATE	CITY	ADDRESS	ZIP	phone	

Fetch the data

BE NICE TO APIs!!...

Important :

The webservice indicates that it allows 10 requests per minute (one request every 6000ms). It's mandatory to respect this. This dataset contains 5037 entries, so the full action will take approximately 9 hours... Thus we will only work a sample of 5 entries.

- on the column "url", select **Edit column/Add column by fetching URLs...**

5037 rows

Show as: rows records Show: 5 10 25 50 rows

▼ All	▼ Number	▼ url	▼ Column 2	▼ Column 3	▼ Column 4	▼ Column			
☆	1.	370784037131001	Facet	7078403	12/23	4265	NAME	MEXICO	
☆	2.	374950102043019	Text filter	7495010	04/24	5785	catherine calvo	FRANCE	
☆	3.	4163570120654535	Edit cells	1635701	08/22	678	Jozef Michels	NETHERLAN	
☆	4.	4163570154289398	Edit column	Split into several columns...				eynoso	NETHERLAN
☆	5.	4350441573270471	Transpose	Join columns...				Carrion	FRANCE
☆	6.	4372033706505420	Sort...	Add column based on this column...				ne Huet	FRANCE
☆	7.	4875011101376870	View	Add column(s) from other projects...				edi	NETHERLAN
☆	8.	4875128642200003	Reconcile	Add column by fetching URLs...				vrit	NETHERLAN
☆	9.	4875128642200086		Add field index...				la oralia	NETHERLAN
☆	10.	4875128642200250	https://lookup.binlist.net/	Add columns from HGNC				mogene	NETHERLAN
				Add columns from reconciled values...					
				Rename this column					
				Remove this column					
				Move column to beginning					
				Move column to end					
				Move column left					
				Move column right					

- name the column results, select 6000ms, store errors (just in case), click on show, remove the openrefine user-agent and click ok.

Usage

Limits

Requests are throttled at 10 per minute with a burst allowance of 10. If you hit the speed limit the service will return a [429](#) http status code.

Need more? Get unlimited access at \$ 0.001/request by reaching out to contact@binlist.net.

Uptime

See our [status page](#) (third party operated) for live uptime stats.

Related projects and resources

- [LUHN check/calculate](#)
- [ISO 3166-1 codes](#)
- [Country currencies](#)
- [Country coordinates](#)
- [Country emoji flags](#)

- wait for the answer.

We'll see how to parse the result at the end of this example! __

But let's try something else, to unleash the power of Openrefine!!!

Usint Jython and the command line

The global concept behind this is to execute a command in a hidden terminal and collect the result in OpenRefine.

Long time ago, Jython was the Python interpreter for Java. The problem is that its development is stalled to the 2.7 version of Python, which is, as you probably know, deprecated. What we will still use, is its ability to use core libraries from python2.7 that allow us to execute a bash command and retrieve its result. We will use three python modules :

- `command` (to execute a shell command)
- `time` (to be able to generate pauses between two commands)
- `random` (to add randomization to the pause)

This will give us access to common commands such as `dig`, `whois`, but also other CLI tools such as `googler` or other. Virtually any python script or tools can be installed and executed in command line!

The jython script that we use is the following :

```
import time
import commands
import random
# get status and output of the command
status, output = commands.getstatusoutput(value)
# add a random between 2 and 5s pause to avoid ddos on servers... Be kind
```

```
to APIs!  
time.sleep(random.randint(2, 5))  
# returns the result of the command  
return output.decode("utf-8")
```

Modifying the url column by adding curl

In the url column, we have... a url. what we want to do is to **curl** this url.

We will simply **transform** the content of these cells by adding "curl " before the existing content :

The screenshot shows a data table with 4 matching rows (5036 total). The table has columns: All, 37078403713100, url, 12/23, 4265, NAME, and MEXICO. The 'url' column is selected, and a context menu is open. The 'Edit cells' option is highlighted, and the 'Transform...' option is also highlighted. The table data is as follows:

	All	37078403713100	url	12/23	4265	NAME	MEXICO
6.	4875011101376870	8750111	12/22	907	ali Saeedi	NETHERLAND	
18.	4875128642200920	8751286	11/23	727	janey maddie	NETHERLAND	
19.	4875128642200953	8751286	06/23	187	floria indira	NETHERLAND	
26.	4875128642201589				amalea shirleen	NETHERLAND	

The screenshot shows the 'Custom text transform on column url' dialog box. The 'Expression' field contains the text '*curl *+value'. The 'Language' dropdown is set to 'General Refine Expression Language (GREL)'. The 'Preview' tab is selected, showing the transformed data:

row	value	"curl "+value
6.	https://lookup.binlist.net/48750111	curl https://lookup.binlist.net/48750111
18.	https://lookup.binlist.net/48751286	curl https://lookup.binlist.net/48751286
19.	https://lookup.binlist.net/48751286	curl https://lookup.binlist.net/48751286
26.	https://lookup.binlist.net/48751286	curl https://lookup.binlist.net/48751286

The 'OK' button is highlighted.

Don't forget the space character after curl

The command lines are now ready to be executed...

4 matching rows (5036 total)

Show as: **rows** records Show: 5 10 25 50 rows

▼ All	▼ 37078403713100	▼ url	▼ 12/23
★	6.	4875011101376870	curl https://lookup.binlist.net/48750111 12/22
★	18.	4875128642200920	curl https://lookup.binlist.net/48751286 11/23
★	19.	4875128642200953	curl https://lookup.binlist.net/48751286 06/23
★	26.	4875128642201589	curl https://lookup.binlist.net/48751286 10/22

Execute the script...

4 matching rows (5036 total)

Show as: rows records Show: 5 10 25 50 rows

▼ All	▼ 37078403713100	▼ url	▼ 12/23	▼ 4265	▼ NAME	▼ MEXICO		
★	6.	4875011101376870	Facet	net/48750111	12/22	907	ali Saeedi	NETHERLA
★	18.	4875128642200920	Text filter	net/48751286	11/23	727	janey maddie	NETHERLA
★	19.	4875128642200953	Edit cells	net/48751286	06/23	187	floria indira	NETHERLA
★	26.	4875128642201589	Edit column	net/48751286	10/22	096	amalea shirleen	NETHERLA

Split into several columns...

Join columns...

Add column based on this column...

Add column(s) from other projects...

Add column by fetching URLs...

Add field index...

Add columns from HGNC

Add columns from reconciled values...

Rename this column

Remove this column

Move column to beginning

Move column to end

Move column left

Move column right

- Create a new column named *results_*.
- Select the jython language (not GREL)
- Copy/Paste the script in the field expression
- Wait a few seconds to see the preview...

Add column based on column url

New column name

On error ☒ set to blank ☐ store error ☐ copy value from original column

Expression Language Python / Jython

```
import time
import commands
import random
# get status and output of the command
status, output = commands.getstatusoutput(value)
# add a random between 2 and 5s pause to avoid ddos on servers... Be
kind to APIs!
time.sleep(random.randint(2, 5))
# returns the result of the command
return output.decode("utf-8")
```

No syntax error.

Preview History Starred Help

row	value	import time import commands im ...
6.	curl https://lookup.binlist.net/48750111	% Total % Received % Xferd Average Speed Time Time Time Current Dload Upload Total Spent Left Speed 0 0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0 0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- --:--:-- 0100 241 0 241 0 0 602 0 --:--:-- --:--:-- 600 {"number": {"length":16,"luhn":true},"scheme":"visa","type":"debit",

OK Cancel

Be careful...

- Be sure that your script has the correct random pause to match the API requirements.
- The preview can take long to appear

Parsing the content

- create a new column based on this column (you know that, now...)
- We use `value.partition('brand':"')[2].partition('')[0]` to extract the data.

```
value.partition('brand':"')[2].partition('')[0]
```

Add column based on column result

New column name

Bank

On error

☒ set to blank

☐ store error

☐ copy value from original column

Expression

Language

General Refine Expression Language (GREL) ▾

value.partition('brand':"')[2].partition('')[0]

No syntax error.

Preview

History

Starred

Help

row	value	value.partition('brand':"')[2] ...
6.	<div><div>% Total % Received % Xferd Average Speed Time</div><div>Time Time Current</div><div>Dload Upload Total Spent Left</div><div>Speed</div><div>0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:--</div><div>0100 241 0 241 0 0 4303 0 --:--:-- --:--:--</div><div>--:--:-- 4303</div><div>{"number":</div><div>{"length":16,"luhn":true},"scheme":"visa","type":"debit","brand":"V</div><div>PAY","prepaid":true,"country":</div></div>	V PAY

OK

Cancel