

Use of OpenRefine

Biodiversity Data Mobilization - Data Cleaning - OpenRefine Exercise



CONTENTS

[CONTENTS](#)

[CONVENTIONS](#)

[2. BASIC USE](#)

[2.1. FILE LOADING AND PROJECTS](#)

[2.1.1. Before starting](#)

[2.1.2. Exercise 1. Create a project](#)

[2.2. FACETING](#)

[2.2.1. Before starting](#)

[2.2.2. Exercise 2. Faceting and mass editing](#)

[2.2.3. Exercise 3. Faceting and white spaces I](#)

[2.2.4. Exercise 4. Faceting and white spaces II](#)

[2.2.5. Exercise 5. Faceting and duplicates](#)

[2.3. FILTERING](#)

[2.3.1. Exercise 6. Basic filter](#)

[2.3.2. Exercise 7. Advanced filter I](#)

[2.3.3. Exercise 8. Advanced filter II](#)

[2.4. CLUSTERING](#)

[2.4.1. Exercise 9. Basic clustering](#)

[2.5. EXPORTING](#)

[You will have several options for exporting your cleaned data, but the following option is useful in most cases.](#)

[3. BASIC API USE](#)

[3.1. Before starting](#)

[3.2. Exercise 1. Higher taxonomy](#)

[4. USEFUL LINKS AND REFERENCES](#)

CONVENTIONS

Formulas (copy-paste)

Text in blue

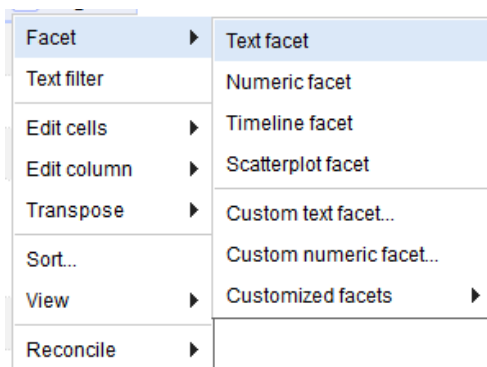
Example: ...then paste the expression `^[a-z]`



Commands in Refine

Text in red





Example: ...and follow the route to **Text facet**



Column names

Text in green

Example: ...go to column **Cat. Numb**

Show as: rows records		Show: 5 10 25 50 rows		
▼ All	▼ Cat. Numb.	▼ University	▼ Collector	
☆ 	7.	UWP:157339	University of Guatemala	Betancur J
☆ 	8.	UWP:157339	University of Guatemala	Betancur H
☆ 	224.	UWP:122471	University of Guatemala	Vargas P
☆ 	225.	UWP:122471	University of Guatemala	Vargas I

Hyperlinks

www.gbif.org

Column menu



2. BASIC USE

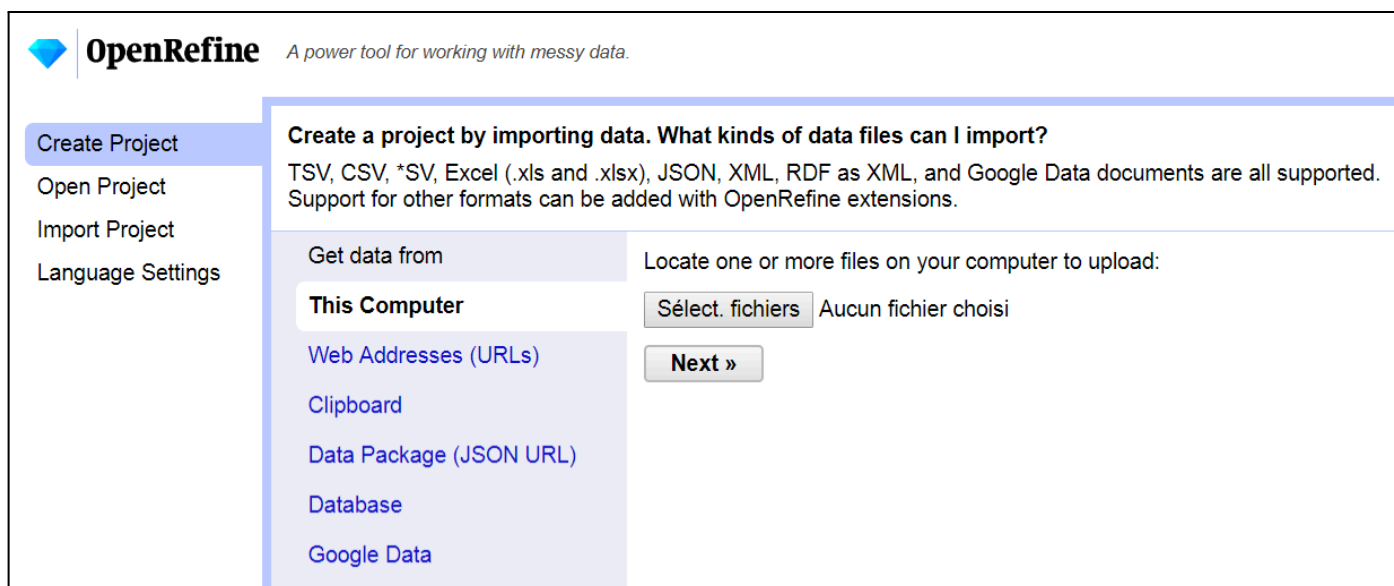
2.1. FILE LOADING AND PROJECTS

2.1.1. Before starting

Data loading can be done from various data sources: TSV, CSV, SV, Excel (.xls and .xlsx), JSON, XML, RDF and XML data as Google Docs. Loading data involves two stages: the first is the file load and the second is the creation of the project.

2.1.2. Exercise 1. Create a project

1. Load the data file from the indicated link one the training site.
2. Open *OpenRefine* (in the latest version of OpenRefine, you can select your preferred language in [Language Settings](#)), click on [Create Project](#), and follow the route [Get data from](#) > [This Computer](#), then click on [Choose Files](#). Select the file. Click on [Next](#).



OpenRefine *A power tool for working with messy data.*

Create Project

Open Project

Import Project

Language Settings

Create a project by importing data. What kinds of data files can I import?
 TSV, CSV, *SV, Excel (.xls and .xlsx), JSON, XML, RDF as XML, and Google Data documents are all supported. Support for other formats can be added with OpenRefine extensions.

Get data from

This Computer Locate one or more files on your computer to upload:

Select. fichiers Aucun fichier choisi

Next »

Web Addresses (URLs)

Clipboard

Data Package (JSON URL)

Database

Google Data

- OpenRefine will automatically detect the format of your file as csv; a parsing options menu will appear. Choose the options as shown in the picture:

Parse data as

CSV / TSV / separator-based files

- Line-based text files
- Fixed-width field text files
- PC-Axis text files
- JSON files
- MARC files
- JSON-LD files
- RDF/N3 files
- RDF/N-Triples files

Character encoding

Columns are separated by

commas (CSV)

tabs (TSV)

custom , _____

Use character " " to enclose cells containing column separators

Trim leading & trailing whitespace from strings

Escape special characters with \

Ignore first 0 line(s) at beginning of file

Parse next 1 line(s) as column headers

Column names (comma separated)

Discard initial 0 row(s) of data

Load at most 0 row(s) of data

Attempt to parse cell text into numbers

Store blank rows

Store blank cells as nulls

Store file source

Store archive file

Disable auto preview


- On the top right in the Project Name field, rename your file as [YourName]UC-Practice-OpenRefine and click **Create Project** and you will be ready to work!

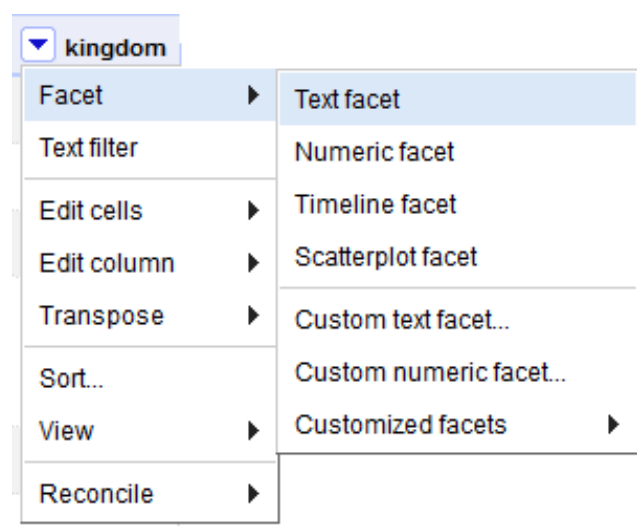
2.2. FACETING

2.2.1. Before starting

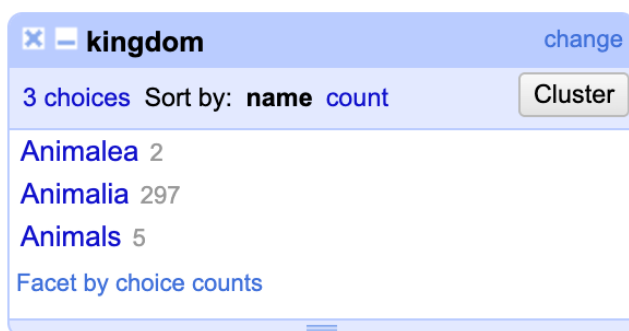
Faceting is a feature that will allow us to get a big picture overview of the data, and to filter down to just the subset of rows that we want to change or view in bulk. It facilitates the use and analysis of data and can be done with cells containing any kind of text, numbers and dates.

2.2.2. EXERCISE 2. Faceting and mass editing

- Go to column **kingdom**, and then click on the column menu  and follow the route to **Text facet** as shown below:

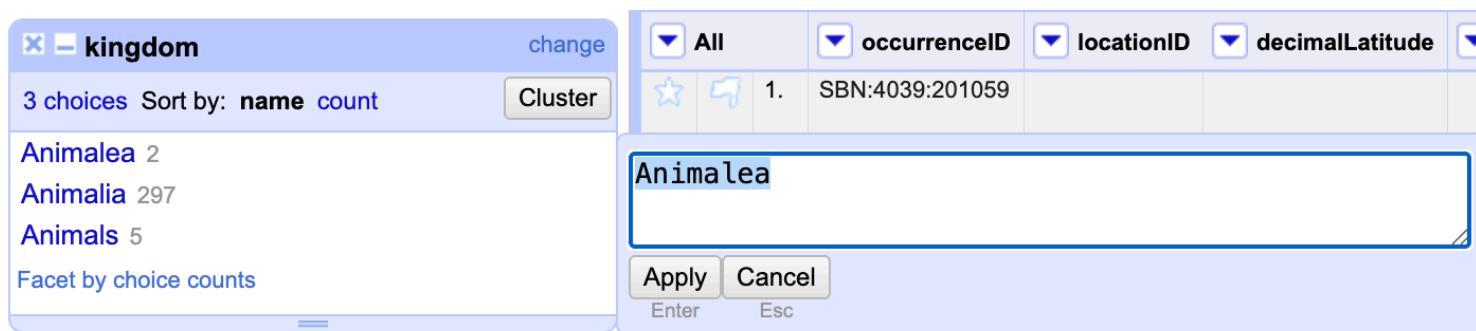


2. On the left a window with the name of the column will appear, that is the facet:




Click on **count** to sort by count, then click on **name** to sort alphabetically.

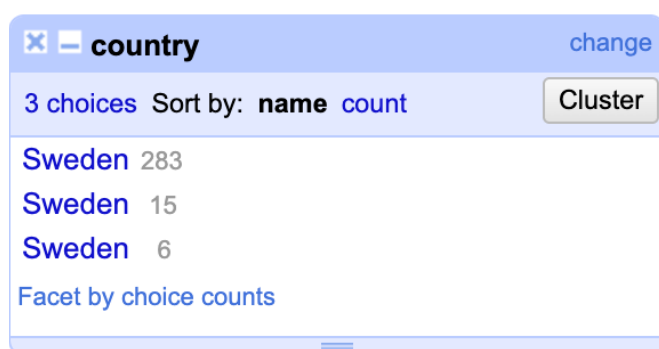
3. Fix the spelling mistakes. Place the cursor over the text in the window and click on **edit**, then fix the error in the text box, and to save click on **apply**.



All the values will be fixed automatically.

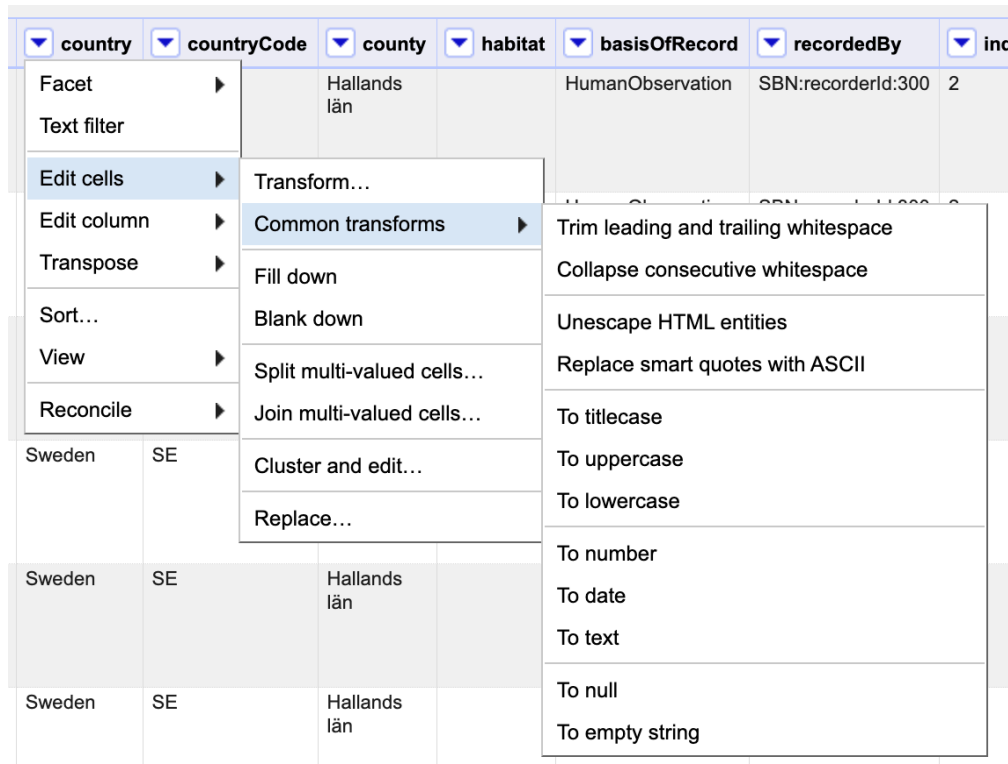
2.2.3. EXERCISE 3. Faceting and white spaces I

1. Go to **country** and click on column menu  and perform a **Text Facet**.



On a quick view, the country appears to be spelled correctly, but the facet shows three different values due to the extra spaces at the end of the text.

- Fix the error from the column menu on `country.`, following the route **Edit Cells** > **Common transforms** > **Trim leading and trailing whitespace**.



You will see a notification message:

Text transform on 21 cells in column country: value.trim()
[Undo](#)

- Now check the facet window; only one value will remain.

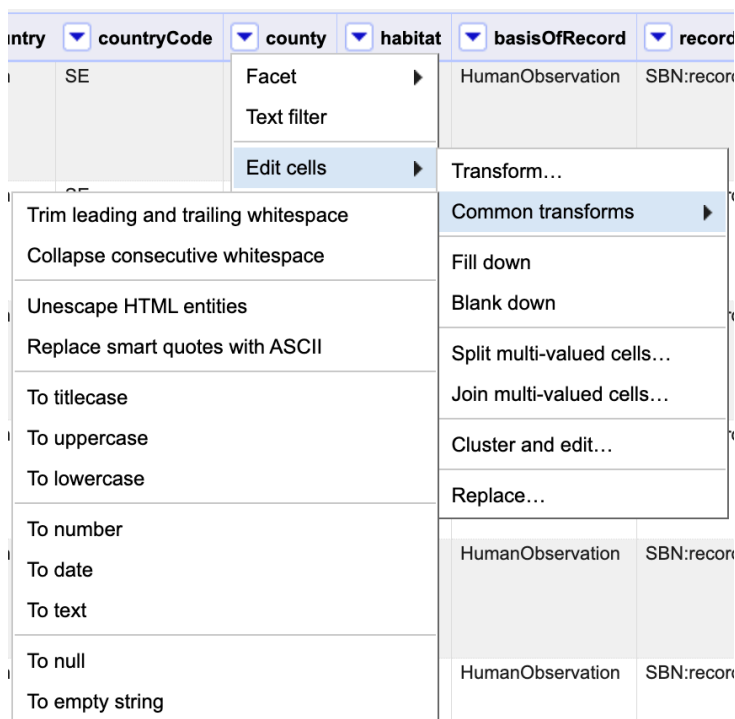


2.2.4. EXERCISE 4. Faceting and white spaces II

- Go to column **county** and click on  then go to **Text facet**. Then click on **count**. The facet will show:



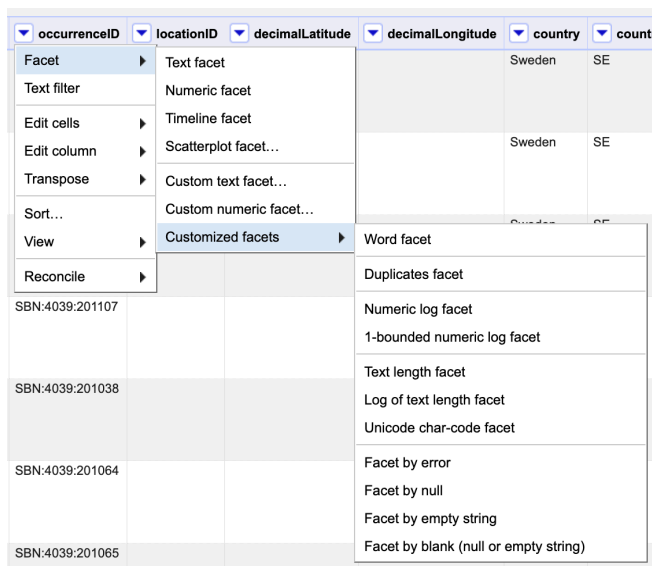
As seen above, *Skåne län* is the first item in the list with 57 specimens, but it is also present in the 3rd place with 37 specimens.



2. Fix the error from the `county` column menu, **Edit Cells** > **Common transforms** > **Collapse consecutive whitespaces**.
3. Once the white-spaces are removed, *Skåne län* should only appear in the list with 94 records.

2.2.5. EXERCISE 5. Faceting and duplicates

1. Go to column catalog in `occurrenceID`, and follow the route **Facet** > **Customized facets** > **Duplicates facet**. The facet will show 4 duplicates



2. Click on **true**, and you'll see the values in the main window:


4 matching rows (305 total)						
Show as: rows		records		Show: 5 10 25 50 100 500 1000 rows		
All	occurrenceID					scientificName
☆	39.	SBN:11280:201064				Araschnia levana
☆	40.	SBN:11280:201064				Argynnis paphia
☆	208.	SBN:104785:201040				Pieris napi
☆	209.	SBN:104785:201040				Pieris rapae

After a check with the transcribers, fix the values clicking edit directly on the cell with the incorrect occurrenceIDs:

SBN:104785:201041 Araschnia levana
 SBN:104785:201040 Argynnis paphia
 SBN:11280:201064 Pieris napi
 SBN:11280:201065 Pieris rapae

2.3. FILTERING

2.3.1. EXERCISE 6. Basic filter

- Go again to **scientificName** column menu and perform a **Text facet** to visualize the values, then go again to  and click on **Text filter**, perform the following filters and fix them as shown below:

Filter	How to fix	Correct value
	Edit directly in the cell	Hesperidae Pieridae Polyommatini
	Edit directly in the cell, check case sensitive	Polyommatini
	<ol style="list-style-type: none"> Go to  on Full name, then click Edit cells > Transform... In the text box paste the formula <code>value.replace(" spp.", "")</code> Click OK 	Heliconiinae Zygaenidae Pieridae

2.3.2. EXERCISE 7. Advanced filter I

1. Go to column **genus** and perform a **Text filter**.
2. Check **regular expression** and **case sensitive**, then paste the expression `^[a-z]`



This regular expression filters the strings in which the first letter is lowercase.

3. Perform a correction since the genus should be capitalized.

Note: If you want to know more about regular expressions click [here](#).

2.3.3. EXERCISE 8. Advanced filter II

1. Go to column **scientificName** and perform a **Text filter**.
2. Check **regular expression** and **case sensitive**, then paste the expression `^[A-Z].*\s[A-Z]`



This regular expression filters the strings that start with a capital letter followed by 0 or more characters, then a space, then a capital letter.

3. Perform a correction since the second word of the name should be lowercase.

Note: If you want to know more about regular expressions click [here](#).

2.4. CLUSTERING

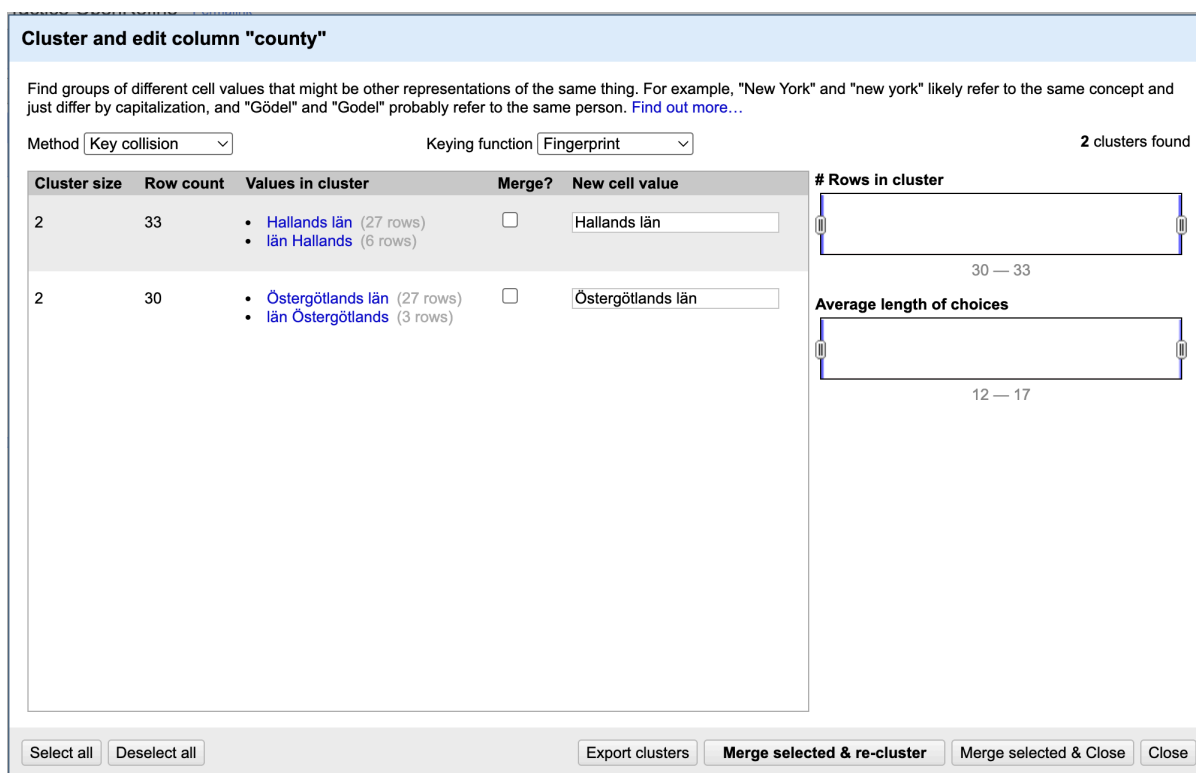
2.4.1. EXERCISE 9. Basic clustering

1. Go to **county**, then in the menu column click **Text facet**.



Keep in mind that the correct counties are: Hallands län, Jönköpings län, Östergötlands län, Skåne län, Stockholms län, Värmlands län, Västmanlands län.

2. On the top right of the facet window click on **Cluster**, a new window will appear:



Cluster size	Row count	Values in cluster	Merge?	New cell value
2	33	<ul style="list-style-type: none"> Hallands län (27 rows) län Hallands (6 rows) 	<input type="checkbox"/>	Hallands län
2	30	<ul style="list-style-type: none"> Östergötlands län (27 rows) län Östergötlands (3 rows) 	<input type="checkbox"/>	Östergötlands län

3. Now you can see information about the clusters:
 - **Cluster size:** the number of different versions that the clustering algorithm believes to be the same.
 - **Row count:** the number of records with any of the cluster values.
 - **Values in cluster:** the actual values that the algorithm believes to be the same. There is also the number of records with each particular value, and the possibility to browse the contents of the cluster in a different tab.
 - **Merge?:** check if values are to be merged into a single standard value.
 - **New cell value:** the value to be applied to every record in the cluster. By default, it is the value with most records. You can also click on any value to apply that to the **New cell value**.

Note: If you want to know more about clustering click [here](#).

4. Click on **Select All** and then on **Merge Selected & close**, you will see a notification message:

Mass edit 63 cells in column county Undo

5. To fix the remaining counties go again to **Cluster** in the facet window of **county**.
6. In the Cluster and edit window, go to **Keying Function**, then select **ngram-fingerprint**, and set **1** as the value in **Ngram Size**. Press the enter key.
7. Click on **Select All** and then on **Merge Selected & close**, you will see a notification message:

Mass edit 88 cells in column county Undo

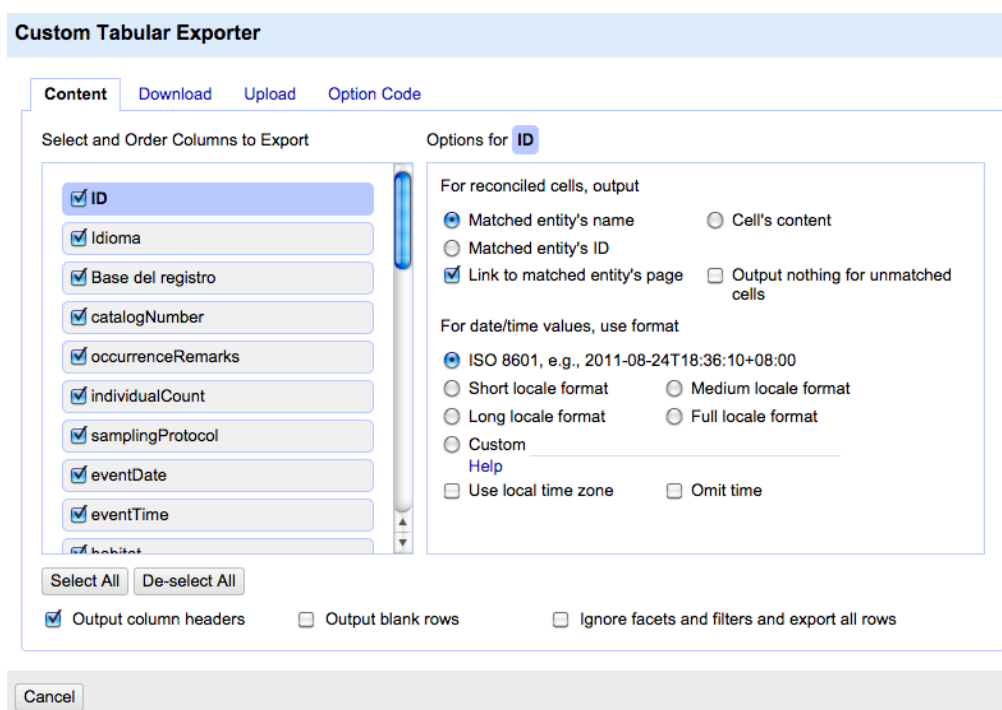
8. Your counties are now fixed and your window should look like the image below:



2.5. EXPORTING

You will have several options for exporting your cleaned data, but the following option is useful in most cases.

1. On the upper right corner click on **Export** and select **Custom tabular exporter...**
2. You will see the exportation window:



Custom Tabular Exporter

Content | Download | Upload | Option Code

Select and Order Columns to Export

- ID
- Idioma
- Base del registro
- catalogNumber
- occurrenceRemarks
- individualCount
- samplingProtocol
- eventDate
- eventTime
- habitat

Options for ID

For reconciled cells, output

- Matched entity's name
- Matched entity's ID
- Link to matched entity's page
- Cell's content
- Output nothing for unmatched cells

For date/time values, use format

- ISO 8601, e.g., 2011-08-24T18:36:10+08:00
- Short locale format
- Long locale format
- Custom
- Medium locale format
- Full locale format
- Use local time zone
- Omit time

Output column headers
 Output blank rows
 Ignore facets and filters and export all rows

3. In the **content** tab you can choose the columns that you want to export, if you select **Ignore facets and filters and export all rows** all facets and filterings will be ignored, this is useful if you forget to clear them before exporting.
4. Go to the **Download** tab and select the separator that you prefer. Don't modify the other options unless you need to.

You can also export the whole project to open it in OpenRefine on another computer by following the route **Export > Export project**. In this case you're not downloading a data file to open in a spreadsheet or text processor, but rather a GZIP file that will only be accessible through OpenRefine.

3. BASIC API USE

3.1. Before starting

Reconciliation matches the information in one of your columns to an outside database. This is particularly helpful when it comes to name validation, as it proves the name you have exists somewhere else. This is a really useful service, but can be time consuming. In this case we will go through the process with only three records using the API from GBIF. Internet connection is required.

3.2. Exercise 1. Higher taxonomy

1. Go to `recordedBy`, then make a **Text facet**. Select `SBN:recorderId:716`



Facet Value	Count
SBN:recorderId:12	29
SBN:recorderId:215	94
SBN:recorderId:266	30
SBN:recorderId:300	33
SBN:recorderId:308 SBN:recorderId:558	59
SBN:recorderId:608	30
SBN:recorderId:716	29

2. Under `scientificName`, click on column menu and then **Edit column** > **Add column by fetching URLs...**, call the new column `Api_name`
3. Change the **Throttle Delay** to `250` and paste the expression:

```
"http://api.gbif.org/v1/species/match?verbose=true&name="+escape(value, 'url')
```

4. Click **ok** and wait, this might take some time depending on your internet connection and the number of taxa.
5. Go to `Api_name`, click on column menu and then **Edit column** > **Add column based on this column....** Call the new column `higherClassification` and paste the expression:

```
value.parseJson().get("kingdom")+
", "+value.parseJson().get("phylum")+
", "+value.parseJson().get("class")+
", "+value.parseJson().get("order")+
", "+value.parseJson().get("family")
```

You will see the Kingdom, Phylum, Class, Order and family of each taxon.

Add column based on column Api_name

New column name

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

Expression Language General Refine Expression Language (GREL) ▾

No syntax error.

Preview History Starred Help

row	value	value.parseJson().get("kingdom ...
247.	{\"usageKey\":1894840,\"scientificName\":\"Aglais urticae (Linnaeus, 1758)\",\"canonicalName\":\"Aglais urticae\",\"rank\":\"SPECIES\",\"status\":\"ACCEPTED\",\"con name=110; authorship=0; classification=-2; rank=6; status=1; score=115; singleMatch=5,\"matchType\":\"EXACT\",\"kingdom\":\"Ani urticae\",\"kingdomKey\":1,\"phylumKey\":54,\"classKey\":2	Animalia, Arthropoda, Insecta, Lepidoptera, Nymphalidae
248.	{\"usageKey\":8020665,\"acceptedUsageKey\":1922937,\"canonicalName\":\"Agriades optilete (Knoch, 1781)\",\"canonicalName\":\"Agriades optilete\",\"rank\":\"SPECIES\",\"status\":\"SYNONYM\"	Animalia, Arthropoda, Insecta, Lepidoptera, Lycaenidae

6. Under **higherClassification** follow the route **Edit column > Split into several columns...**, leave the settings as shown:

Split column higherClassification into several columns

How to Split Column

by separator

Separator regular expression

Split into columns at most (leave blank for no limit)

by field lengths

List of integers separated by commas, e.g., 5, 7, 15

After Splitting

Guess cell type

Remove this column

7. Now you know how to obtain the taxonomic categories of a given taxon if this is available in the GBIF API. Column names can be edited in **Edit column > Rename this column**.
8. For the purpose of the workshop, the columns created in this exercise (Higher taxonomy) must be deleted. Under **All**, which is the first column, go to **Edit columns > Re-order / remove columns...**
9. Drop the columns as shown and click **OK**:

Re-order / Remove columns

Drag columns to re-order

- eventID
- identificationVerificationStatus
- taxonID
- scientificName
- Api_name
- kingdom
- phylum
- class
- order
- family
- genus
- specificEpithet
- taxonRank
- scientificNameAuthorship
- vernacularName
- collectionCode

Drop columns here to remove

- higherClassification 1
- higherClassification 2
- higherClassification 3
- higherClassification 4
- higherClassification 5

4. USEFUL LINKS AND REFERENCES

- Name validation Tutorial:
https://docs.google.com/document/d/1tkDRXIYhmassYAk5T4v5oac5prF0jAiSMr_IEGTvhRo/edit
- Higher Taxonomy Tutorial:
https://docs.google.com/document/d/1XZ_pM9gldQzHzl8wfUCVea-52yub5T_3tc-snBgPRa0/edit
- Documentation
<https://github.com/OpenRefine/OpenRefine/wiki/Documentation-For-Users>
- Resources list for OpenRefine:
<https://github.com/OpenRefine/OpenRefine/wiki/External-Resource>

Exercise concept and content developed by Néstor Beltrán.

Updated: 03 July 2019. Sophie Pamerlon

Updated: 11 July 2019. Laura Russell, Sophie Pamerlon.

Updated: 20 April 2026. Laura Russell