



OCHA

# Using Power Query for Excel

## In Humanitarian Context

---

### Introduction

Power Query is an excel build in function that can be used in different stages of the Information Management cycle. It allows you to:

- 1- Data collection: connect to data collected from different sources from different locations i.e. excel sheets, text files, sharepoint, kobo, google sheets and other online sources.
- 2- Data cleaning: collected data can be verified and corrected, categorized, merged and combined.

Steps of data operations are saved as “Applied Steps” that can be reviewed, amended, shared and automate future processes.

**Output dataset:** Power query output is processed and consolidated data to be used in later phases i.e. analysis and visualizations.



A key success of using power query is to develop the script in a way that allows it to be shared and minimal updates are needed to be done by the people intending to use it.

# Data Preparation

1. It's a good practice to standardize column names of the different datasets. This step helps in matching fields from different datasets so they can be "combined" regardless whether they were in same order or not. HXL allows you to do so. Add new row under the main header (in grey) and add the HXL tags (shown in yellow).

Lead Organization	Implementing Partner	Organization type	Provincia	Admin1CODE	Distrito	Admin2CODE	Posto	Admin3CODE	Village/Location
#org +funder +name	#org +impl +name	#org +impl +type +nar	#adm1 +name	#adm1 +code	#adm2 +name	#adm2 +code	#adm3 +name	#adm3 +code	#loc +name
Shelter lead org	Shelter imp1	Agências das N	Manica	MZ004	Gondola	MZ004603	Macate	MZ004603135	xx transit centre
Shelter lead org	Shelter imp2	Agências das N	Manica	MZ004	Sussundenga	MZ004609	Dombe	MZ004609157	Matarara
Shelter lead org	Shelter imp3	Agências das N	Sofala	MZ009	Cidade Da Beira	MZ009701	Cidade Da Beira	MZ009701318	orphanage

**HXL to created standard fields names**

2. It's highly recommended to have your dataset defined in tables in excel so you can only import only needed dataset.

The screenshot shows the Microsoft Excel interface. The 'Insert' tab is active, and the 'Table' button is highlighted with a blue arrow and the text "2 insert table". Below the ribbon, the spreadsheet shows a data range from A1 to I4. A blue arrow points to this range with the text "1 select the data you want to create as table".

#org +funder	#org +impl	#org +impl +type +name	#adm1	#adm1	#adm2 +name	#adm2	#adm3 +name	#adm3
Shelter lead org1	Shelter imp1	Agências das Nações Unidas	Manica	MZ004	Gondola	MZ004603	Macate	MZ004603135
Shelter lead org2	Shelter imp2	Agências das Nações Unidas	Manica	MZ004	Sussundenga	MZ004609	Dombe	MZ004609157
Shelter lead org3	Shelter imp3	Agências das Nações Unidas	Sofala	MZ009	Cidade Da Beira	MZ009701	Cidade Da Beira	MZ009701318

## Connection to data source and consolidating in master table

1. Create connection to your tables i.e. one connection per cluster, without importing the full table unless needed:

**The option "Only Create Connection" will connect to the table without having to import the data**

**Import Data**

Select how you want to view this data in your workbook.

Table

PivotTable Report

PivotChart

**Only Create Connection**

Where do you want to put the data?

Existing worksheet:

=SAS1

New worksheet

Add this data to the Data Model

Properties... OK Cancel

**Queries & Connect...**

Queries | Connections

2 queries **5**

Cluster\_Shelter  
Connection only.

Cluster\_WASH  
Connection only.

**The result shown here is list of queries that connects to as many data sources as needed.**

**Cluster\_Shelter**

#org +funder +name	#org +impl +nam
Shelter lead org1	Shelter imp1
Shelter lead org2	Shelter imp2
Shelter lead org3	Shelter imp3
Shelter lead org4	Shelter imp4
Shelter lead org5	Shelter imp5

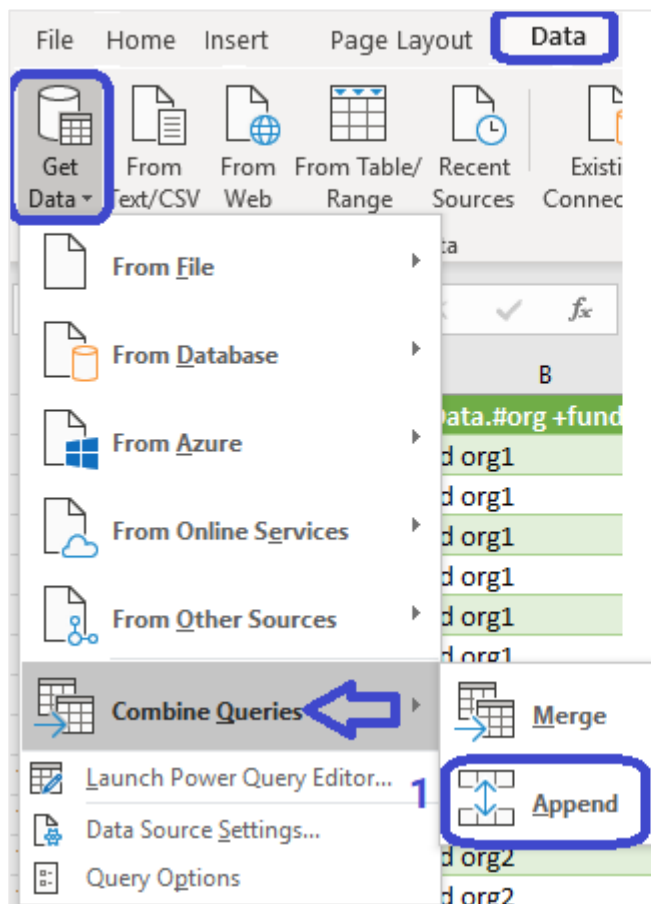
Load Edit Cancel

Load Load To...

Do this step for every dataset you want to use i.e. all clusters data

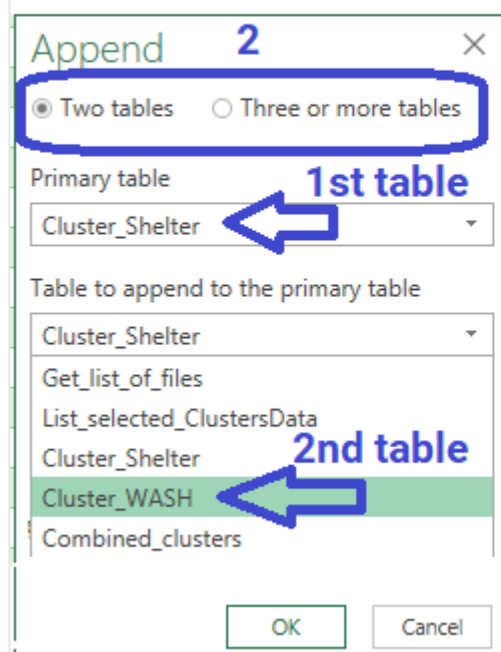
2. Combine datasets together i.e. clusters data consolidated all in master connection/sheet as explained below:

For successful application of the Combine function, it is important to have all columns of all tables has standard similar names i.e. HXL as explained in the first step page so that values of same columns go correctly under each other.



### 3 Resulting table contains data from both tables

80	Cluster_WASH	WASH_Lead_org80
81	Cluster_WASH	WASH_Lead_org81
82	Cluster_WASH	WASH_Lead_org82
83	Cluster_WASH	WASH_Lead_org83
84	Cluster_WASH	WASH_Lead_org84
85	Cluster_WASH	WASH_Lead_org85
86	Cluster_WASH	WASH_Lead_org86
87	Cluster_WASH	WASH_Lead_org87
88	Cluster_WASH	WASH_Lead_org88
89	Cluster_WASH	WASH_Lead_org89



90	Cluster_Shelter	Shelter lead org1
91	Cluster_Shelter	Shelter lead org2
92	Cluster_Shelter	Shelter lead org3
93	Cluster_Shelter	Shelter lead org4
94	Cluster_Shelter	Shelter lead org5
95	Cluster_Shelter	Shelter lead org6
96	Cluster_Shelter	Shelter lead org7
97	Cluster_Shelter	Shelter lead org8
98	Cluster_Shelter	Shelter lead org9
99	Cluster_Shelter	Shelter lead org10
100	Cluster_Shelter	Shelter lead org11
101	Cluster_Shelter	Shelter lead org12
102	Cluster_Shelter	Shelter lead org13
103	Cluster_Shelter	Shelter lead org14

## Applying data processing functions

1. Correct data Types of all columns:

## Right click the column you need to change its data type

A <sup>B</sup> <sub>C</sub> #adm1 +code	A <sup>B</sup> <sub>C</sub> #adm2 +name	A <sup>B</sup> <sub>C</sub> #adm2 +code	A <sup>B</sup> <sub>C</sub> #adm3 +name
MZ004			Macate
MZ004			Dombe
MZ009			Cidade Da Beira
		null	
		null	
		null	
		null	
		null	Mujana

Copy	
Remove	
Remove Other Columns	
Duplicate Column	
Add Column From Examples...	
Remove Duplicates	
Remove Errors	
<b>Change Type</b>	<ul style="list-style-type: none"> <li>Decimal Number</li> <li>Currency</li> <li>Whole Number</li> <li>Percentage</li> <li>Date/Time</li> <li>Date</li> <li>Time</li> <li>Date/Time/Timezone</li> <li>Duration</li> <li><input checked="" type="checkbox"/> Text</li> <li>True/False</li> <li>Binary</li> <li>Using Locale...</li> </ul>
Transform	
Replace Values...	
Replace Errors...	
Split Column	
Group By...	
Fill	
Unpivot Columns	
Unpivot Other Columns	
Unpivot Only Selected Columns	
Rename...	
Move	
Drill Down	

2. Make necessary basic data manipulation, cleaning and processing needed for each data source, steps can be done differently than explained below:

**1 Right-click the query**

**2 Remove Blank Rows**

**3 Remove Other Columns**

**4 Replace values i.e. null with zeros**

**5 Change the column values i.e. remove spaces using Trim, Letters case...etc**

These blank rows will be deleted

these will be deleted and only selected columns will be kept

#org +funder +name	#org +impl +name	#adm1 +name	#adm2 +code
Shelter lead org1	Shelter imp1	Agências das Naç	Gondola MZ004603
Shelter lead org2	Shelter imp2	Agências das Naç	Sussundenga MZ004609
Shelter lead org3	Shelter imp3	Agências das Naç	Cidade Da Beira MZ009701
Shelter lead org16	Shelter imp16	ONGs nacionais	Sussundenga MZ004609

Replace Values

Value To Find: null

Replace With: 0

**Note:** The most important feature of Power Query is that it saves your work in steps so you can track, undo or customize steps (what has been done) without having to delete the steps. Below is a summary steps of what has been done in the above illustration:

**Query Settings**

**PROPERTIES**

Name: Cluster\_Shelter

**APPLIED STEPS**

- Source
- Navigation
- Changed Type
- Removed Blank Rows
- Changed Type1
- Trimmed Text
- Removed Other Columns

Steps run in this direction

3. Make Advanced data processing operations:

Split and merge columns

### Split Columns

**1** Select the column

**2** By Delimiter

**3** Select the separator

**4** Each occurrence of the delimiter

Split Column by Delimiter

Specify the delimiter used to split the text column.

Select or enter delimiter

- Custom--
- Colon
- Comma
- Equals Sign
- Semicolon
- Space
- Tab
- Custom--

Advanced options

Split at

- Left-most delimiter
- Right-most delimiter
- Each occurrence of the delimiter

OK Cancel

### Merge/concatenate columns

**1** Select columns to merge

**2** Merge Columns

Merge Columns

Concatenate the currently selected columns into one column.

A <sup>B</sup> C #org +funder +name	A <sup>B</sup> C #org +impl +name	A <sup>B</sup> C #date +start
Shelter lead org1	Shelter imp1	2019
Shelter lead org2	Shelter imp2	2019
Shelter lead org3	Shelter imp3	2019
Shelter lead org4	Shelter imp4	2019
Shelter lead org5	Shelter imp5	2019
Shelter lead org6	Shelter imp6	2019
Shelter lead org7	Shelter imp7	2019
Shelter lead org8	Shelter imp8	2019
Shelter lead org9	Shelter imp9	2019

Merge Columns

3 Select the separator between merged columns

Merge columns resulting in one

A <sup>B</sup> C Merged	A <sup>B</sup> C #date +start
Shelter lead org1:Shelter imp1	2019
Shelter lead org2:Shelter imp2	2019
Shelter lead org3:Shelter imp3	2019
Shelter lead org4:Shelter imp4	2019
Shelter lead org5:Shelter imp5	2019
Shelter lead org6:Shelter imp6	2019
Shelter lead org7:Shelter imp7	2019
Shelter lead org8:Shelter imp8	2019

## Add Conditional Columns

**Conditional Column**  
Create a new column that conditionally adds the values in the currently selected column.

**1 select the column to categorise**

Index	Agências das Nações Unidas
1	Agências das Nações Unidas
2	Agências das Nações Unidas
3	Agências das Nações Unidas
4	Agências das Nações Unidas
5	Agências das Nações Unidas

### 4 resulting new column with new values

Index	Data Type: Data.#org +impl +type +name	Index	Custom
1	Agências das Nações Unidas	1	UN
2	Agências das Nações Unidas	2	UN
3	Agências das Nações Unidas	3	UN
4	Agências das Nações Unidas	4	UN
5	Agências das Nações Unidas	5	UN
6	ONGs nacionais	6	NNGO
7	ONGs nacionais	7	NNGO
8	ONGs nacionais	8	NNGO

### Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name:

**Add IF conditions to make change values based on certain occurrences**

Condition	Column Name	Operator	Value	Output
If	Data Type: Data.#o...	contains	ABC 123 Unidas	ABC 123 UN
Else If	Data Type: Data.#o...	contains	ABC 123 ONGs	ABC 123 NNGO

Otherwise:

Buttons: Add rule, OK, Cancel



## Add Custom Columns

1

2 Add the formula needed using M language i.e. calculate # of households by deviding the # reached beneficiaries by 5

```

Custom Column
Create a new column in this table, based on a custom formula.
Custom Column
New column name: Custom_1
Custom column formula:
= if [{"DataType.Data.#reached +ind +num2"} <> "" then [{"DataType.Data.#reached +ind +num2"}]/5 else [{"DataType.Data.#reached +ind +num2"}]
Available columns:
DataType.Name
DataType.Data.#org +funder...
DataType.Data.#org +impl +n...
DataType.Data.#org +impl +t...
Custom
DataType.Data.#adm1 +name
DataType.Data.#adm1 +code
No syntax errors have been detected.
    
```

3 Resulting column

ABC 123	DataType.Data.#reached +ind +num2	ABC 123	Custom_1
1	375		75
2	360		72
3	100		20
4	500		100
5	500		100
6	500		100
7	500		100
8	3500		700
9	440		88
10	275		55
11	null		null
12	null		null
13	null		null

# Standardizing/categorizing values

This step is critical in case you want to create standard names or categorize values i.e. cluster names, type of organizations...etc. The steps are explained below:

1. Create lists of values; These should contain original names that should be replaced together with the standard name format that will be replaced with. We will call these lists as vocabularies, sometimes they are called data dictionary.

**1 Create table of the data that has been categorised**

**2**

**3**

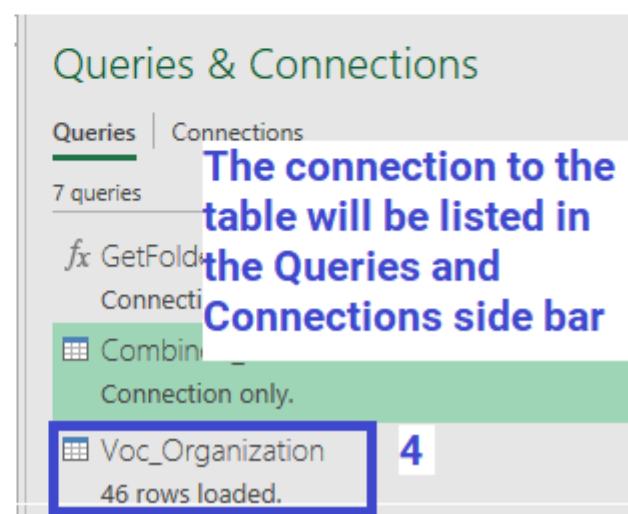
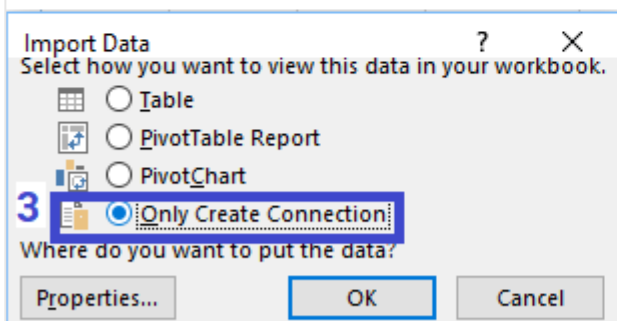
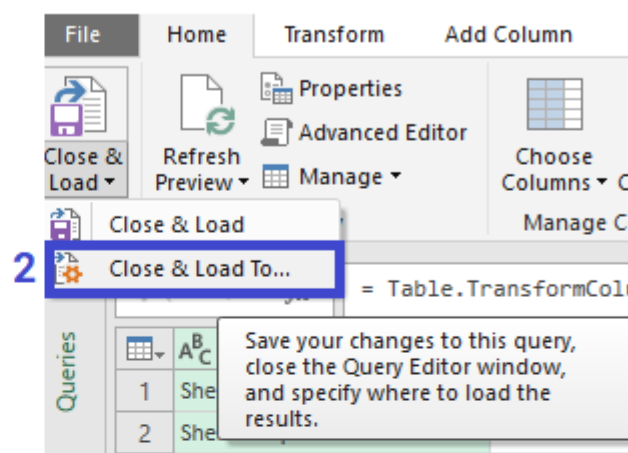
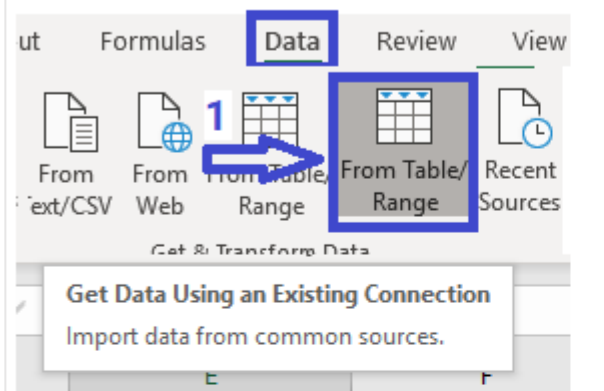
**5 i.e. Voc\_Organizations**

**4 Resulting table**

Organization Name	Organization Type
Shelter imp3	UN
Shelter imp4	UN
Shelter imp5	UN
Shelter imp6	UN
Shelter imp13	NNGO
Shelter imp14	NNGO
Shelter imp15	ACD
Shelter imp16	ACD
Shelter imp19	INOG
Shelter imp20	INOG
WASH_imp1	INOG
H_imp2	INOG
H_imp3	INOG

2. Now create connection to this table (vocabulary):

### Create a connection to the table i.e. Voc\_Organizations



3. Now you can add this table to the master connection/table that has consolidated cluster data. The reason for that is to add the standard column(s)/categories to the master sheet.

**1** Right click the Master connection that has consolidated dataset

**2** Select Merge to join tables

**3** Join the Master connection to the vocabulary table i.e. Voc\_Organizations in Merge Window

**4** Vocubary table

**5** Expand and select the columns from the newly merged table

**6** Now the new master table has the organizations with one more additional piece of data that is the organization type

Organization Name	Organization Type
Shelter imp3	UN
Shelter imp4	UN
Shelter imp5	UN
Shelter imp6	UN
Shelter imp7	NNGO

Organization Name	Organization Type
Shelter imp3	UN
Shelter imp4	UN
Shelter imp5	UN
Shelter imp6	UN
Shelter imp7	NNGO
Shelter imp8	NNGO
Shelter imp10	NNGO
Shelter imp11	NNGO
Shelter imp12	NNGO
Shelter imp13	NNGO
Shelter imp14	NNGO
Shelter imp15	ACD

#### 4. Unpivot data i.e. summaries multiple columns values into rows

Unpivot Columns

2 select unpivot columns

"Beneficiaries-Boys"}, {"Girls"}, "Beneficiaries-Girls"}, {"Men", "Beneficiaries-Men"}, {"Women", "Beneficiaries-Women"}, {"USD", "Cash Disbursed-USD"}))

Activity Description	Activity Condition	Beneficiaries-Boys	Beneficiaries-Girls	Beneficiaries-Men	Beneficiaries-Women	Cash Disbursed-USD
sh fo		a 20	b 15	c 22	d 17	e 806
sh fo		73	59	46	40	1910
sh fo		20	15	22	17	806
sh for winterization	Un-conditional	73	59	46	40	2866
sh for NFIs	Un-conditional	52	46	31	36	2060
sh for winterization	Un-conditional	52	46	31	36	2060
sh for NFIs	Un-conditional	60	43	28	35	2149
sh for winterization	Un-conditional	60	43	28	35	1433
sh for NFIs	Un-conditional	45	44	20	21	1701
sh for winterization	Un-conditional	45	44	20	21	1701
sh for NFIs	Un-conditional	47	62	27	30	2060
sh for winterization	Un-conditional	47	62	27	30	2060
ulti-Purpose cash	Un-conditional	71	86	42	47	0

1 Select the columns that want them unpivoted

3 The result is two columns one is called "Attributes" and its values are the unpivoted column headers, and one called "Value" which will be the values that were in one row turned into columns as shown in the illustration where columns a to e turned into columns

Attribute	Value
Beneficiaries-Boys	a 20
Beneficiaries-Girls	b 15
Beneficiaries-Men	c 22
Beneficiaries-Women	d 17
Cash Disbursed-USD	e 806
Beneficiaries-Boys	73
Beneficiaries-Girls	59
Beneficiaries-Men	46
Beneficiaries-Women	40
Cash Disbursed-USD	1910
Beneficiaries-Boys	20
Beneficiaries-Girls	15
Beneficiaries-Men	22
Beneficiaries-Women	17
Cash Disbursed-USD	806

# Connecting to different data sources

## Connect to Sharepoint:

**5** open excel file to which you want data to be imported from share point, then selected Data --> From Web

**1** The first step is to reset/clear permissions of all data sources

**2** Global permissions

**3** Clear All Permissions

**4** Open the excel file that need to have it imported into Power Query, Click File --> Info -> select and copy the path to clipboard

**5** open excel file to which you want data to be imported from share point, then selected Data --> From Web

**6** Paste the link you copied here and delete the last part from the question mark till the end

**7** Organizational account

**8** Sign in using your UniteID or sharepoint credentials

**9** Connect

**10** The resulting connection will list all tables,sheets, Select the data you would like to import and press load to.

**11** Load

**Data Source settings**

Manage settings for the data sources used in queries.

Data sources in current workbook  Global permissions

Search data source settings

https://unitednations-my.sharepoint.com/

Edit Permissions... Clear Permissions Clear All Permissions

**Access Web content**

Anonymous Windows Basic Web API Organizational account

https://unitednations-my.sharepoint.com/personal/...

You aren't signed in.

Sign in

Select which level to apply these settings to

https://unitednations-my.sharepoint.com/

Back Connect

**Navigator**

Select multiple items

Display Options

https://unitednations-my.sharepoint.com/

DataProcessing\_Results

Voc\_Organization

Voc\_Organization\_2

1DataSource

2Vocabularies

**DataProcessing\_Results**

Data Type Name	Data Type Data.#org +funder
Cluster_Shelter	Shelter lead org1
Cluster_Shelter	Shelter lead org2
Cluster_Shelter	Shelter lead org3
Cluster_Shelter	Shelter lead org4
Cluster_Shelter	Shelter lead org5
Cluster_Shelter	Shelter lead org6
truncated due to size limits.	

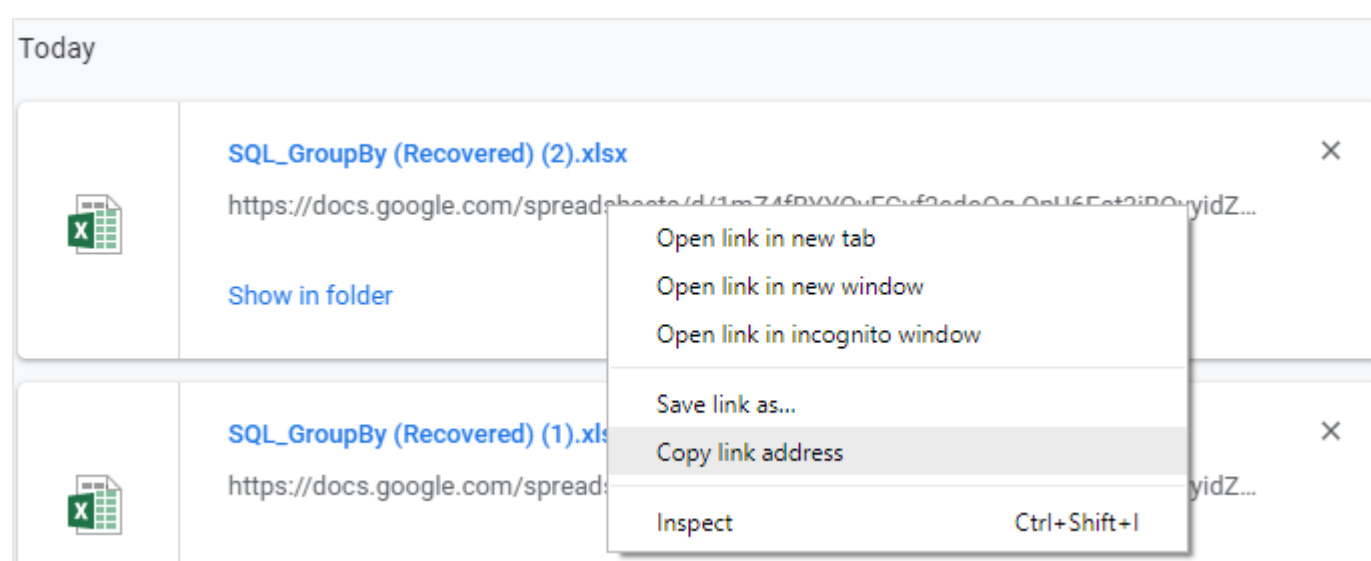
Load Transform Data Cancel

## Connect to KOBO:

1. Create connection to Kobo API and import collected data. A step by step guide can be found here: (<http://support.kobotoolbox.org/en/articles/1559346-pulling-your-data-into-excel-power-query>)
2. The importance of importing the data collected in kobo is that excel data will be updated with new collected data every time you refresh the connection in Power Query.

## Connect to Google sheet:

1. Open the google sheet (with sharing settings set to Edit – anyone with the link)/
2. Get the web link of your sheet to be used in Power Query:
  - Go to File>Download as>Microsoft Excel OR Comma-separated values OR Tab-Separated values
  - CTRL+J and you will see the history of downloaded files, right click the sheet and **copy link address**



- Paste the link as URL for data source from Data-> "From Web" in Power Query.

## Connect to FINANCIAL TRACKING SERVICE (FTS)

Create a function that collect the data for X country(s) using the API. The exercise can be generalized to be used for funding received within the Plan, requirements and total funding.

To connect and get updated to recent dataset do the following:

1. Connect to APIs (for more information about the HPC APIs please check here: <https://api.hpc.tools/docs/v1/>)

**Plan Funding:** [https://api.hpc.tools/v1/public/fts/flow?planid="&Number.ToText\(appealID\)"&limit=100000&format=json](https://api.hpc.tools/v1/public/fts/flow?planid=)

**Plan Requirement:** [https://api.hpc.tools/v1/public/fts/flow?planid="&Number.ToText\(appealID\)"&groupBy=cluster&format=json](https://api.hpc.tools/v1/public/fts/flow?planid=)

**Total Funding:** [https://api.hpc.tools/v1/public/fts/flow?locationid="&Number.ToText\(ctryID\)"&year=2019&limit=100000&format=json](https://api.hpc.tools/v1/public/fts/flow?locationid=)

Notes:

- Replace "*planid=&Number.ToText(appealID)*" with the country PlanID.
- APIs are better to be called from within functions.

2. Create Vocabularies Tables

- o Create a table listing previous year's donors (exact names as per FTS) in one column and their acronym in the other.
- o Create a table listing clusters names in one column (exact names as per FTS) within the country Plan and "correct version" (how you would like the clusters names to appear on your visual) in the other column.
- o Create a table listing in the first column the appealing organizations (exact names as per FTS) and the second column their acronyms
- o Create a table listing in the first column the country name (exact name as per FTS), in the second column the ISO code and in the third column the country's PlanID

3. Link Table to Functions

Call each Function by following the below steps:

- o On Data (in the Menu), click on Get Data → From Other Sources -> Blank Query
- o Create a list of ISO3 with the above-mentioned columns
- o Convert it to table and give It a title

4. STEP3: MERGING THE TABLES

- o Merge the Appealing organizations, Donors and Clusters tables with the Plan Funding Table
- o Merge the Clusters tables with the Requirement Table then Expand
- o The Plan funding table will show the list of organizations that have requested funds within the Plan and have received the funds within it, as well as the donor per project and the corresponding fund
- o The Requirement table will show the funding per cluster

The Total Funding will show the amount that was funded inside and outside the Plan. In order to extract the total amount received outside the Plan, subtract the sum of the Total Funding with the sum of the sum of the Plan Funding. It is important to note that it cannot be disaggregated by cluster/sector.



---

## Connect to DEEP:

Create connection to DEEP and import collected data. A step by step guide can be found here: (<https://docs.google.com/document/d/1TBeRRi5UgZX0APzAua1HX12V57RPAzEjaqhXZjCAOhg/edit>)

And also Illustrative video ([https://drive.google.com/file/d/18W0r9fkWKe\\_DoIH-9IZ0sD-E3yw7cKPR/view](https://drive.google.com/file/d/18W0r9fkWKe_DoIH-9IZ0sD-E3yw7cKPR/view))

### Note:

To successfully establish connection to kobo, make sure to append the string **'?limit=25&format=json'** at the end of the link to load limited data while using browser. i.e. if you are planning to get the list of projects i.e the api link to the projects would be something like this ('<https://api.thedeep.io/api/v1/projects/>') then you need to add use ('<https://api.thedeep.io/api/v1/projects/?limit=25&format=json>') for Power Query to work which limits the number of rows to 25. However Power BI can import unlimited number of rows.

Due to large number of data set, trying to pull all at once will give timeout error so it's best to use pagination to load large data using limit. You can use the limit parameter to provide how much entity you want per requests as explained earlier so appending ?limit=10 will provide 10 entity per request for the used API.

Power query/PBI doesn't provide pagination right out of box, but you can construct query to load all the required data.

<https://stackoverflow.com/questions/46904641/how-to-get-paginated-data-from-api-in-power-bi>

so that you would use <https://api.thedeep.io/api/v1/projects/?limit=10> for the url and use offset instead of page similar to this <https://api.thedeep.io/api/v1/projects/?limit=10&offset=10>.

## Connect to ACLED

**Note:** The Armed Conflict Location & Event Data Project (ACLED) is a disaggregated conflict analysis and crisis mapping project. ACLED is the highest quality, most widely used, realtime data and analysis source on political violence and protest in the developing world. Practitioners, researchers and governments depend on ACLED for the latest reliable information on current conflict and disorder patterns. Visit it here <https://www.acleddata.com>.

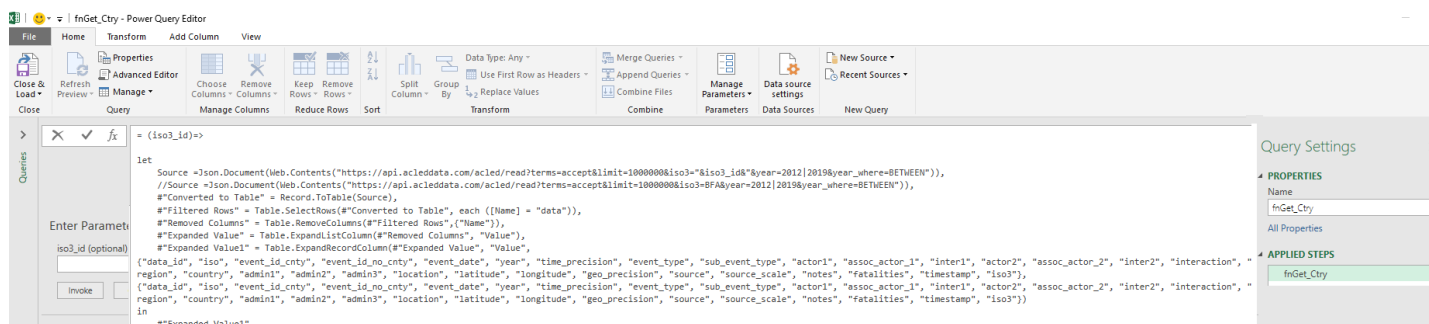
To connect and get updated to recent dataset do the following:

1. Create a function in Power Query that's main functionality is to connect to specific country(s) data using the API as follow:
  - o Data on the menu -> Get Data -> From Web
  - o The API to connect is: Web.Contents ("https://api.acleddata.com/acled/read?terms=accept"). A full guide for using ACLED API can be found here (<https://www.acleddata.com/wp-content/uploads/2013/12/API-User-Guide-August-2017.pdf>)

Name the Function (in the example: **fnGet\_Ctry**). Below is an example for such function:

```
(= (iso3_id)=>

let
    Source
    =Json.Document(Web.Contents("https://api.acleddata.com/acled/read?terms=accept&limit=1000000&iso3="&iso3_id&"&year=2012|2019&year_where=BETWEEN")),
    //Source
    =Json.Document(Web.Contents("https://api.acleddata.com/acled/read?terms=accept&limit=1000000&iso3=BFA&year=2012|2019&year_where=BETWEEN")),
    #"Converted to Table" = Record.ToTable(Source),
    #"Filtered Rows" = Table.SelectRows(#"Converted to Table", each ([Name] = "data")),
    #"Removed Columns" = Table.RemoveColumns(#"Filtered Rows",{"Name"}),
    #"Expanded Value" = Table.ExpandListColumn(#"Removed Columns", "Value"),
    #"Expanded Value1" = Table.ExpandRecordColumn(#"Expanded Value", "Value", {"data_id", "iso", "event_id_cnty", "event_id_no_cnty", "event_date", "year", "time_precision", "event_type", "sub_event_type", "actor1", "assoc_actor_1", "inter1", "actor2", "assoc_actor_2", "inter2", "interaction", "region", "country", "admin1", "admin2", "admin3", "location", "latitude", "longitude", "geo_precision", "source", "source_scale", "notes", "fatalities", "timestamp", "iso3"}, {"data_id", "iso", "event_id_cnty", "event_id_no_cnty", "event_date", "year", "time_precision", "event_type", "sub_event_type", "actor1", "assoc_actor_1", "inter1", "actor2", "assoc_actor_2", "inter2", "interaction", "region", "country", "admin1", "admin2", "admin3", "location", "latitude", "longitude", "geo_precision", "source", "source_scale", "notes", "fatalities", "timestamp", "iso3"})
in
    #"Expanded Value1")
```



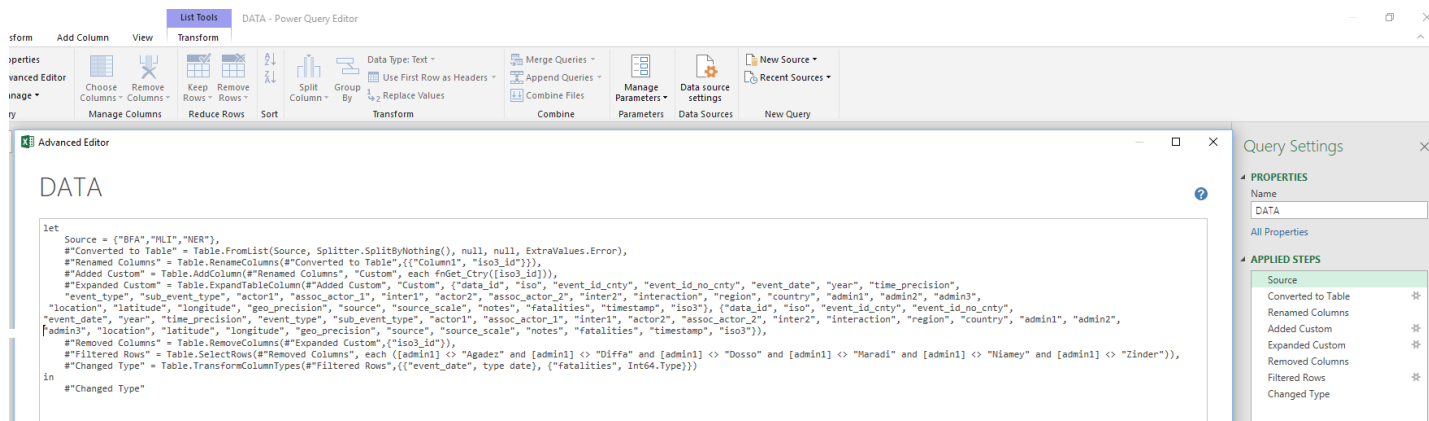
2. Using the above function to collect data of certain country(s):
  - o Open Blank Query again → Create a list of countries → Convert it to table and name it.
  - o Add a custom column then call the function you just created **fnGet\_Ctry** and apply the parameter to the function (ISO3) then expand to see all columns retrieved
  - o Expand

Below is the script that can be used:

```
let
    Source = {"BFA","MLI","NER"},
```

```

#"Converted to Table" = Table.FromList(Source, Splitter.SplitByNothing(), null, null, ExtraValues.Error),
#"Renamed Columns" = Table.RenameColumns("#Converted to Table",{"Column1", "iso3_id"}),
#"Added Custom" = Table.AddColumn("#Renamed Columns", "Custom", each fnGet_Ctry([iso3_id]),
#"Expanded Custom" = Table.ExpandTableColumn("#Added Custom", "Custom", {"data_id", "iso", "event_id_cnty", "event_id_no_cnty", "event_date", "year",
"time_precision", "event_type", "sub_event_type", "actor1", "assoc_actor_1", "inter1", "actor2", "assoc_actor_2", "inter2", "interaction", "region", "country", "admin1",
"admin2", "admin3", "location", "latitude", "longitude", "geo_precision", "source", "source_scale", "notes", "fatalities", "timestamp", "iso3"}, {"data_id", "iso",
"event_id_cnty", "event_id_no_cnty", "event_date", "year", "time_precision", "event_type", "sub_event_type", "actor1", "assoc_actor_1", "inter1", "actor2",
"assoc_actor_2", "inter2", "interaction", "region", "country", "admin1", "admin2", "admin3", "location", "latitude", "longitude", "geo_precision", "source", "source_scale",
"notes", "fatalities", "timestamp", "iso3"}),
#"Removed Columns" = Table.RemoveColumns("#Expanded Custom",{"iso3_id"}),
#"Filtered Rows" = Table.SelectRows("#Removed Columns", each ([admin1] <> "Agadez" and [admin1] <> "Diffa" and [admin1] <> "Dosso" and [admin1] <>
"Maradi" and [admin1] <> "Niamey" and [admin1] <> "Zinder")),
#"Changed Type" = Table.TransformColumnTypes("#Filtered Rows",{"event_date", type date}, {"fatalities", Int64.Type})
in
#"Changed Type"
    
```



3. See below sample results which you could update regularly and automatically:

The screenshot shows an Excel spreadsheet with a table of event data. The columns are: data\_id, iso, event\_id\_cnty, event\_id\_no\_cnty, event\_date, year, time\_precision, event\_type, sub\_event\_type, and assoc\_actor\_1. The data includes various event types such as 'Strategic developments', 'Violence against civilians', 'Battles', and 'Armed clash', with sub-event types like 'Change to group/activity', 'Attack', 'Looting/property destruction', and 'Other'. Some rows have an 'assoc\_actor\_1' value, such as 'Katiba Macina' and 'Prisoners (Burkina Faso)'.

	A	B	C	D	E	F	G	H	I	K
	data_id	iso	event_id_cnty	event_id_no_cnty	event_date	year	time_precision	event_type	sub_event_type	assoc_actor_1
1	5975273	854	BFO2012	2012	9/13/2019	2019	1	Strategic developments	Change to group/activity	
3	5975514	854	BFO2009	2009	9/13/2019	2019	1	Violence against civilians	Attack	
4	5975728	854	BFO2011	2011	9/13/2019	2019	1	Violence against civilians	Attack	
5	5975729	854	BFO2014	2014	9/13/2019	2019	1	Strategic developments	Looting/property destruction	
6	5975730	854	BFO2010	2010	9/13/2019	2019	1	Violence against civilians	Attack	Katiba Macina
7	5975270	854	BFO2005	2005	9/12/2019	2019	1	Battles	Armed clash	
8	5975271	854	BFO2006	2006	9/12/2019	2019	1	Battles	Armed clash	
9	5975727	854	BFO2013	2013	9/11/2019	2019	1	Strategic developments	Looting/property destruction	
10	5975274	854	BFO2015	2015	9/10/2019	2019	2	Strategic developments	Other	Prisoners (Burkina Faso)
11	5975513	854	BFO2008	2008	9/10/2019	2019	1	Violence against civilians	Attack	
12	5975268	854	BFO2003	2003	9/9/2019	2019	1	Battles	Armed clash	
13	5975269	854	BFO2004	2004	9/9/2019	2019	1	Battles	Armed clash	
14	5975266	854	BFO2001	2001	9/8/2019	2019	1	Battles	Armed clash	
15	5975267	854	BFO2002	2002	9/8/2019	2019	1	Battles	Armed clash	
16	5975272	854	BFO2007	2007	9/8/2019	2019	1	Violence against civilians	Attack	
17	5975275	854	BFO2017	2017	9/8/2019	2019	1	Evolutionary/Born violence	Born violence/landmine/IED	

## Preparing the environment

- To easily use the below steps, it will be good practice to store the latest clusters data in the same location, together with the master excel file that will be used to process these data using Power Query as shown in the below screenshot.

Name	Date modified	Type	Size
3W_Food.xlsx	14/10/2019 5:14 PM	Microsoft Excel W...	42 KB
3W_Health.xlsx	14/10/2019 5:16 PM	Microsoft Excel W...	42 KB
3W_Nutrition.xlsx	14/10/2019 5:15 PM	Microsoft Excel W...	41 KB
3W_WASH.xlsx	14/10/2019 5:11 PM	Microsoft Excel W...	42 KB
Consolidated 3W Template.xlsx	15/10/2019 10:02 ...	Microsoft Excel W...	51 KB

- Path to data sources (i.e. 3W clusters dataset)

**Formula to get the path of current open blank excel sheet**

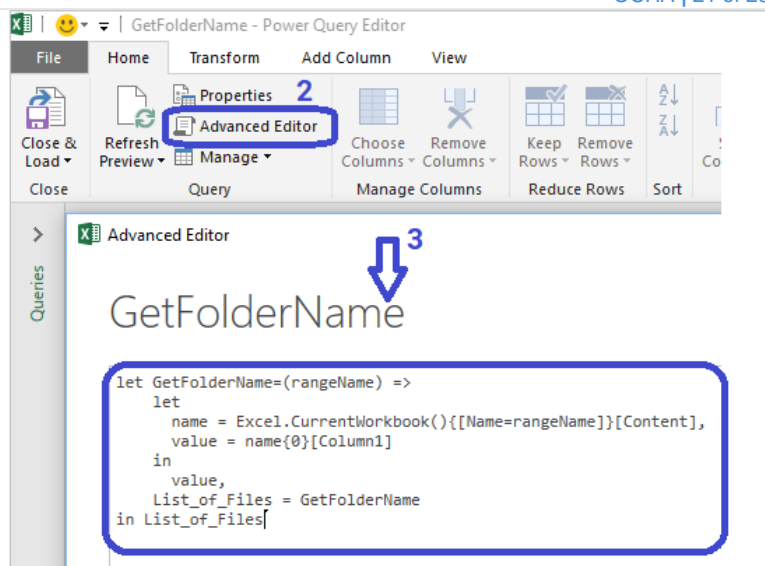
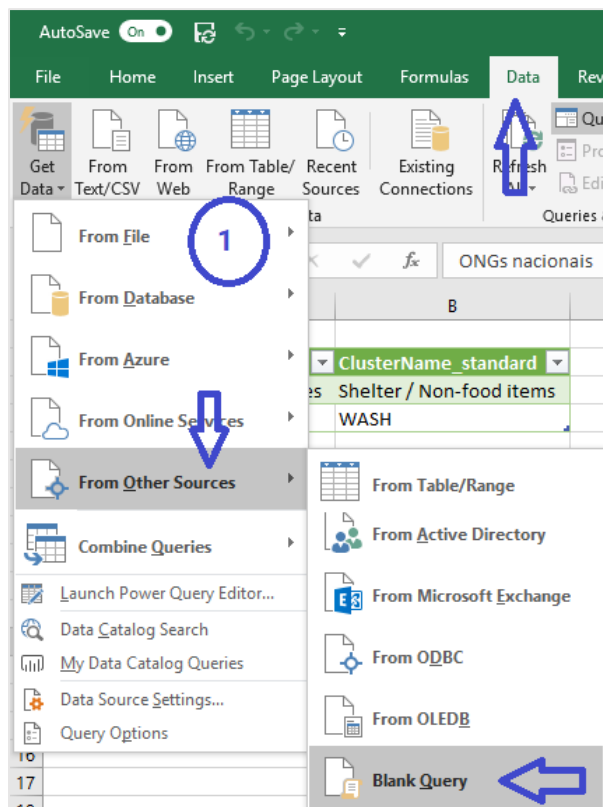
```
=LEFT(CELL("filename",S1),FIND("[",CELL("filename",S1))-2)
```

Current Path	Folder name i.e. where the data exists within current path (enter manually)	Data Path (automatic or manual)
C:\Users\Elter\OneDrive - United Nations\Useful tools\Excel\PowerQuery\Sample_3W_IMToolBox	ClusterData\Working	C:\Users\Elter\OneDrive - United Nations\Useful tools\Excel\PowerQuery\Sample_3W_IMToolBox\ClusterData\Working

**1 DataSource** | **2** Vocabularies | **3** Results | Instructions

**5 name the sheet datasource**      **2 path to datasets inside current path**      **3 final destination of the data location (could be entered manually)**      **4 name this cell as Vdatasource**

- Create custom function that will accept the directory path as input and generate list of files as its output.



4. Create another custom query that will run the previous function by passing the directory path and listing the files automatically. You should see list of files as displayed at the bottom of this screenshot.

The screenshot illustrates the process of creating a custom query in Power Query Editor. It is divided into two main sections:

**Top Section: Advanced Editor**

The 'Advanced Editor' window is open, showing the M query for 'Get\_list\_of\_files'. The code is as follows:

```
// get file list
let
  FolderName = GetFolderName("VDataSource"),
  Source = Folder.Files(FolderName)
in
  Source
```

A status message at the bottom of the editor indicates: "No syntax errors have been detected." Buttons for 'Done' and 'Cancel' are visible.

**Bottom Section: Power Query Editor**

The main Power Query Editor window shows the result of the query. The formula bar contains the expression: `= Folder.Files(FolderName)`. Below the formula bar, a table of results is displayed with the following columns: Content, Name, Extension, and Folder Path.

Content	Name	Extension	Folder Path
Binary	Cluster_Shelter.xlsx	.xlsx	C:\Users\Elter\OneDrive - United Nations\Useful tr
Binary	Cluster_WASH.xlsx	.xlsx	C:\Users\Elter\OneDrive - United Nations\Useful tr

5. Select the dataset you would like to process,

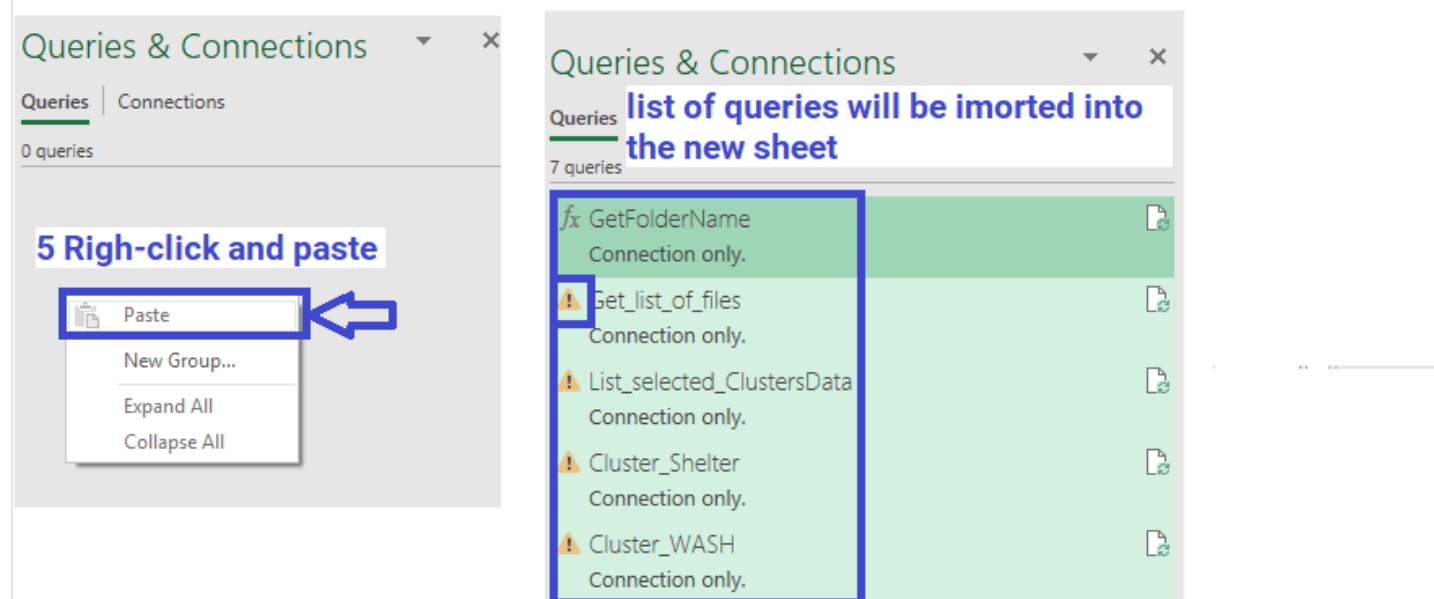
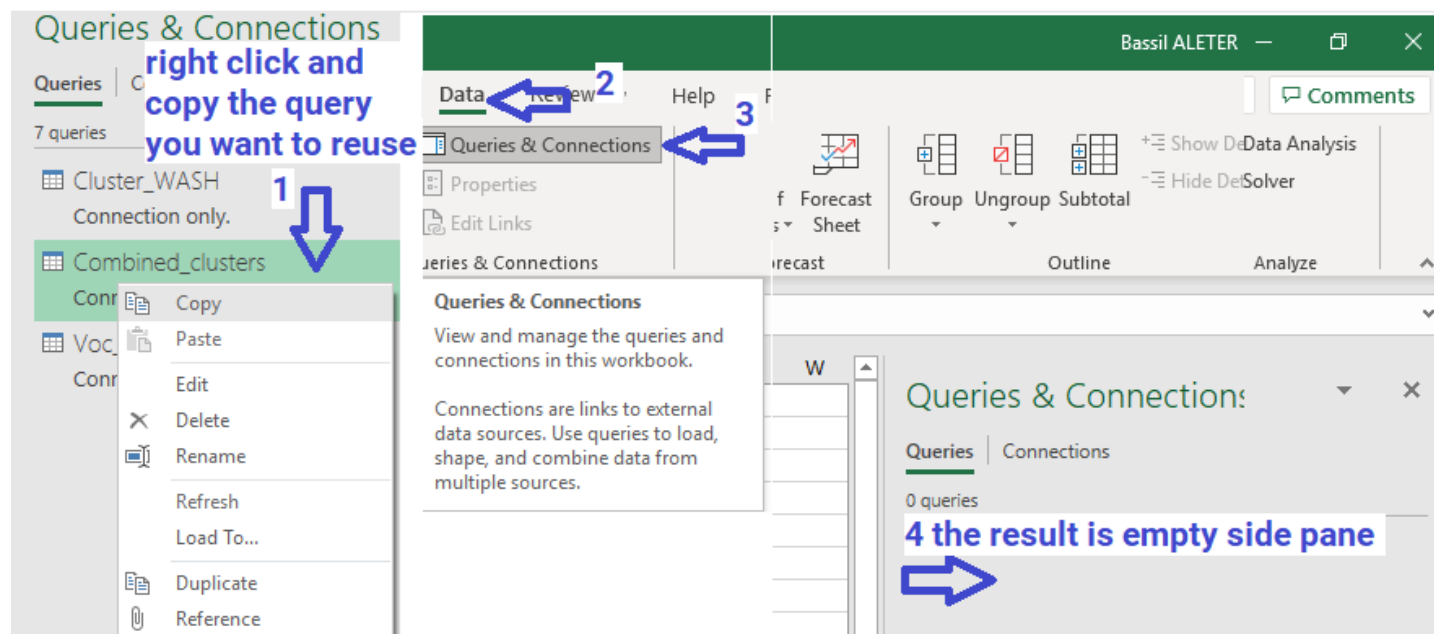
**Note:** It's highly recommended to have your dataset defined in tables in excel.

The screenshot displays a data processing interface. On the left, a dialog box for filtering data is open. It features a search bar and a list of columns with checkboxes. The columns 'Cluster\_Shelter' and 'Cluster\_WASH' are checked and circled in blue. Other columns include 'Admin1\_Linked\_Start', 'Admin2', 'Admin2\_Linked\_Pcode', 'Admin2\_Linked\_Start', 'Admin2\_Pcode', 'Admin2\_Start', 'Admin3', 'Admin3\_Pcode', 'Admin3\_Start', 'Admin\_List', 'Data\_Entry', 'DISTRIBUTION', 'Instruction', 'Lists', 'LST\_Modalities', and 'LST\_One-Time'. Below the list are 'OK' and 'Cancel' buttons.

On the right, the 'Query Settings' panel is visible. It has a 'Name' field containing 'List\_selected\_ClustersData'. Under the 'APPLIED STEPS' section, a list of operations is shown: 'Source', 'Added Custom', 'Filtered Rows', 'Added Index', 'Renamed Columns', 'Expanded DataType', and 'Filtered Rows1'. The 'Filtered Rows1' step is highlighted in green, and a blue arrow points to it from the filter dialog.

# Sharing Power Query Queries Between Excel Workbooks

1. Copy the query you would want to use from the original file and paste it in the blank sheet or in the workbook you want to use it in.




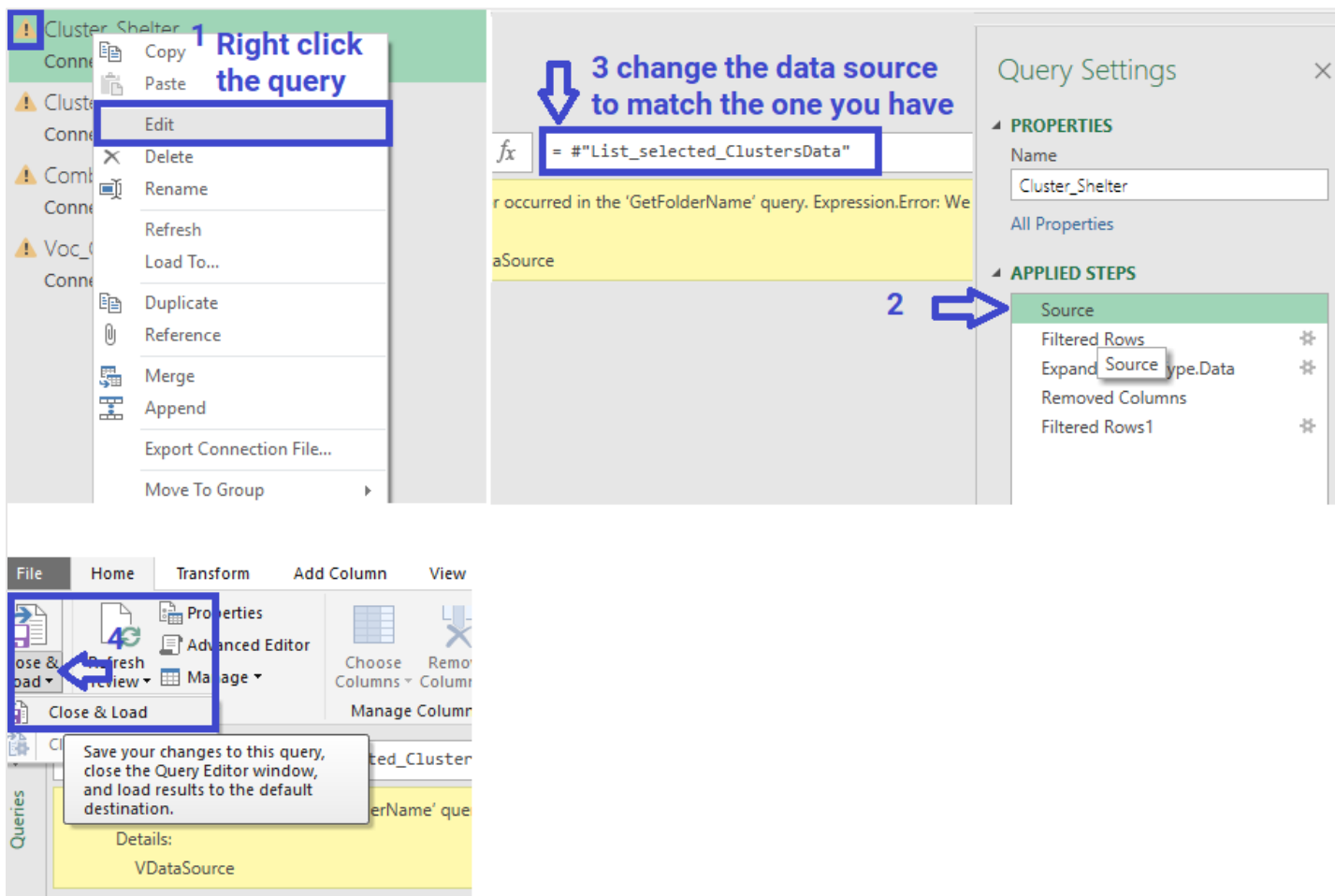
**Note:** If you selected and copied one query, it will copy not only the copied query but also all queries that are called by that specific query.



⚠ If you had copied and pasted the query to be used in different sheet and you got exclamation mark, it means the query will not run properly. Here are the things which should be updated in the pasted query (from Advanced Editor screen):

- Data sources is same or updated to the new location/data
- Tables and Column names are the same or should be updated to the new ones.

 A brilliant golden secret about power query is to think about shareability while developing it, so fewer changes are needed.



The screenshot illustrates the process of updating a query's data source in the Power Query Advanced Editor. It is divided into four numbered steps:

- 1 Right click the query:** A context menu is shown over the query name 'Cluster\_Shelter' in the left-hand pane, with the 'Edit' option highlighted.
- 2:** An arrow points to the 'Query Settings' pane on the right, specifically to the 'APPLIED STEPS' section where the 'Source' step is selected.
- 3 change the data source to match the one you have:** An arrow points to the formula bar at the top, where the data source is being updated from a default path to a specific data source: `= #"List_selected_ClustersData"`.
- 4:** An arrow points to the 'Close & Load' button in the ribbon, indicating the final step to refresh the data.

A tooltip at the bottom of the screenshot reads: "Save your changes to this query, close the Query Editor window, and load results to the default destination."

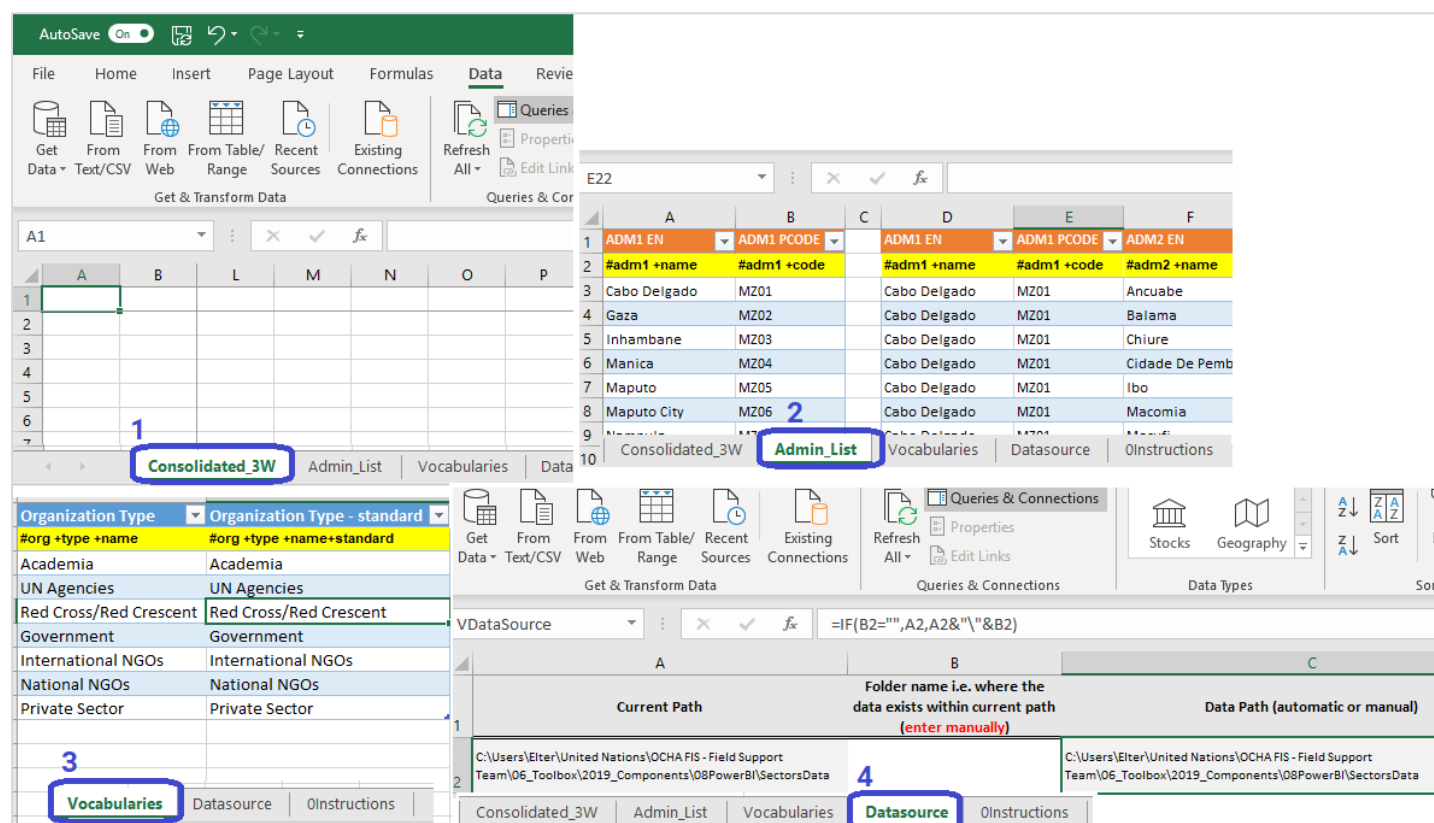
# Power Query Examples

## 3W including cash activities – download PQ\_Sample1

Steps – can vary from one dataset to another but these are the main operations:

1. Prepare the individual clusters dataset:
  - Add HXLated row for the 3W header in all clusters/sectors datasets (directly under the column headers).
  - Select the 3W raw dataset and create table of your dataset and name it Tbl\_3W\_WASH, Tbl\_3W\_Nutrition, ...etc.
  - Add all recent 3W dataset in one folder
2. Create new excel file (**Consolidated 3W Template**)– which will be used to consolidate clusters data:
  - **Consolidated\_3W sheet**: blank sheet which will have the final consolidated dataset added.
  - **Admin\_List sheet**: which can be used to Merge with the consolidated Consolidated\_3W to get P-codes...etc.
  - **Vocabularies sheet**: needs to be updated constantly with new organizations spelled differently, using vocabulary helps to create unique names across dataset and avoid duplications.
  - **Datasource sheet**: this sheet has automatic values added in the cells that will be used to get the files location.

The four sheets look like the below screenshots:



3. For each cluster:
  - Import/get data (**as connection** not to import the whole dataset):
    - Option 1: using the path obtained automatically in the **Datasource sheet**. See *Preparing the environment section*.
    - Option 2: for each cluster data go to Data tab → Get data → From File → From Workbook or
  - Process the data of each cluster separately:

- Promote the **HXL** row to become the table header
  - Change data type of all column to number, text or date as appropriate. i.e columns for Admin\_names1,2,3,4 as Text, and reached\_beneficiaries to number and same should be done for all other columns
  - Merge the cluster data with predefined set of vocabularies/standard **names** i.e. organizations, implementing partners.
    - Example1: voc\_beneficiaries\_type with first column has original value and the second with the categorized values of beneficiaries. i.e. records with values idp\_conflict, idp\_natural\_disaster would be categorized as IDPs.
    - Example2: Voc\_modality with first column has original values and the second with the standard correct values. i.e. mobile\_cash, e\_cash would be categorized as Mobile money.
  - Merge with **admin lists** including P-codes at the most granular level
  - Create **categories** as needed i.e. Modalities, Activity types:
    - Merge/link voc\_modality to the Master table using a key of original values so that they match to get the standard modality names added to the Master table.
    - Merge/link voc\_beneficiaries\_type to the Master table using a key of original values so that they match to get the beneficiaries categories added to the Master table.
    - You will need to expand the newly merged/linked table to be able to see the columns in the Master table.
  - Create **calculated fields** i.e. apply calculation as defined by cluster methodology for calculating the number of beneficiaries; calculate total individual beneficiaries from total household reached (create a custom column and its values would be the  $[total\_households\_reached] * [avg\_family\_size]$ ).
  - **Keep required columns**, and delete the other unnecessary columns. i.e. remove unnecessary columns (columns won't be removed from the original table but only from the query or the connection). In many cases, there are data collected that won't be used for specific type of analysis. i.e. Activity ID, descriptions...etc. and only keep columns that will be used in the analysis. These columns will not be deleted from the original file, but it will only apply to the query and you could revert changes anytime even after closing the file by deleting unwanted steps.
  - Split columns by Delimiter: i.e. split the date column by separator “\” to get it in separate columns for the day, month and year.
4. Combined clusters data in one table. If HXL is properly named, the data will automatically be merged in order. Resulting should look like displayed in this screenshot:

name	#adm1 +code	#adm2 +code	#adm2 +code	#adm3 +code
SX03	Sinjani	SX0308	Rawal	
SX03	Sinjani	SX0308	Lukerabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Ferguson	
SX03	Sinjani	SX0308	Rawal	
SX03	Sinjani	SX0308	Rawal	
SX01	Fergabad	SX0108	Ferguson	
SX01	Fergabad	SX0108	Ferguson	
SX03	Sinjani	SX0308	Rawal	
SX03	Sinjani	SX0308	Rawal	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Ferguson	
SX01	Fergabad	SX0108	Ferguson	
SX01	Fergabad	SX0108	Ferguson	
SX01	Fergabad	SX0108	Ferguson	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX01	Fergabad	SX0108	Zurabad	
SX03	Sinjani	SX0308	Lukerabad	
SX03	Sinjani	SX0308	Lukerabad	
SX01	Fergabad	SX0108	Zurabad	

Queries & Connecti...

Queries | Connections

8 queries

GetFolderName  
Connection only.

Get\_list\_of\_files  
Connection only.

List\_selected\_ClustersData  
Connection only.

1 Create connection to clusters data locations i.e. from same folder

WASH\_3W  
Connection only.

Food\_3W  
Connection only.

Health\_3W  
Connection only.

Nutrition\_3W  
Connection only.

2 Create connection for each cluster and process data individually

All\_3W  
31 rows loaded.

3 Consolidate processed cluster data

5. Add steps to apply to the resulting master table that has all clusters clean dataset consolidated.

- Rename columns. i.e replace the HXL names with meaning full names i.e. #adm1 +name to be Admin1\_Name...etc.
- Unpivot columns, i.e. unpivoting columns of reach beneficiaries disaggregated by age and gender (beneficiaries\_boys, beneficiaries\_girls, beneficiaries\_mens, beneficiaries\_women) to add two new columns:
  - One column is called **attributes** and its data would be for example (beneficiaries\_boys, beneficiaries\_girls, beneficiaries\_mens, beneficiaries\_women).
  - Another column is called **values** and its data would be the actual values of each attribute.