Guide to linking your CRM or Case Management database to PowerBI using a rest API

Contents

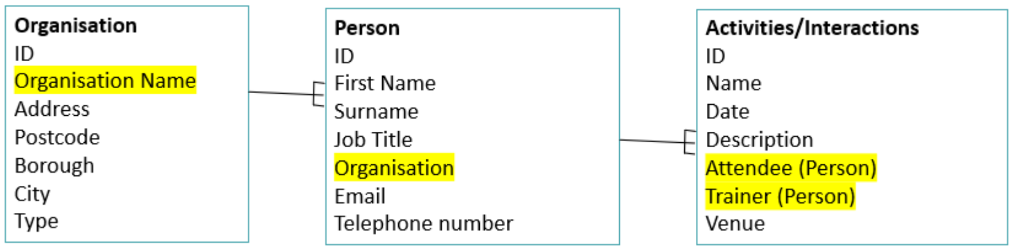
Before getting started

This guide was created for a specific database but as the principles are the same for databases we decided to share the guide more widely. Your API set up and the data tables being brought back might not look or act in the same way and so you might need your provider to help.

What is a CRM?

Although technically a CRM is defined as a customer relationship management system and is often used for sales in the commercial sector in the charity sector we use the term CRM as simply a database that collects and analyses data about the people you work with, how they engage with you and the impact of that. We often use the word Case Management database to make it's use clear but both are simply databases, and the principles are the same as any other database that data is stored in interconnected or relational tables.

A simple database design



What is an API?

Application Programming Interface or API is a mechanism that enables an application or service to access a resource within another application or service. The application or service doing the accessing is called the client, and the application or service containing the resource is called the server.

This documentation is for setting up an API that gets data from a CRM or Case Management database and streams it live into a Power BI report.

What is a REST API?

A REST API is an API that conforms to the design principles of the REST, or *representational state transfer* architectural style. The significance of this is that it guarantees a uniform design which makes it easy to link applications with a REST API to other applications.

Check with your CRM provider if they have an API and discuss with them what you are hoping to do. They might have a solution already in place which would mean you don't have to follow this procedure. If they don't the next question to ask them is if they have a API that is built on REST principles.

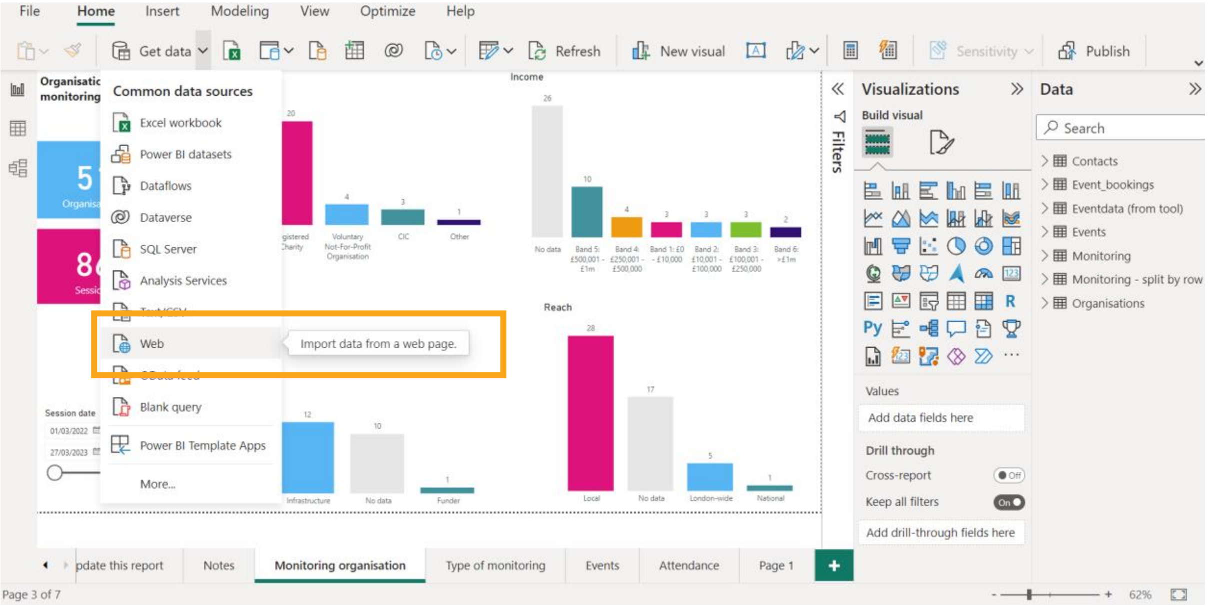
Set up your CRM API in Power BI

In the screenshots below you'll see that I've brought the data into an existing Power BI report, I have chosen to do this so that I don't need to rebuild the report but can simply change the data source for visualisations. If you don't already have a Power BI report then you can start from scratch with a new report.

You will need to repeat the steps below for each entity in your database that you want to bring into PowerBI e.g. Contacts, Organisations, Monitoring, Events, Event bookings etc.

Step 1 : In the Power BI Desktop application:

1. Go to the **Home** tab
2. Click on the arrow next to the **Get data** command button
3. Select the option **Web**



If this is a new Power BI report, don't forget to save the file. It's worth saving the file in its final destination as if you rename or move the report the links will all be broken and you'll have to go and edit them to reflect the move/name change.

Step 2: Enter your API information

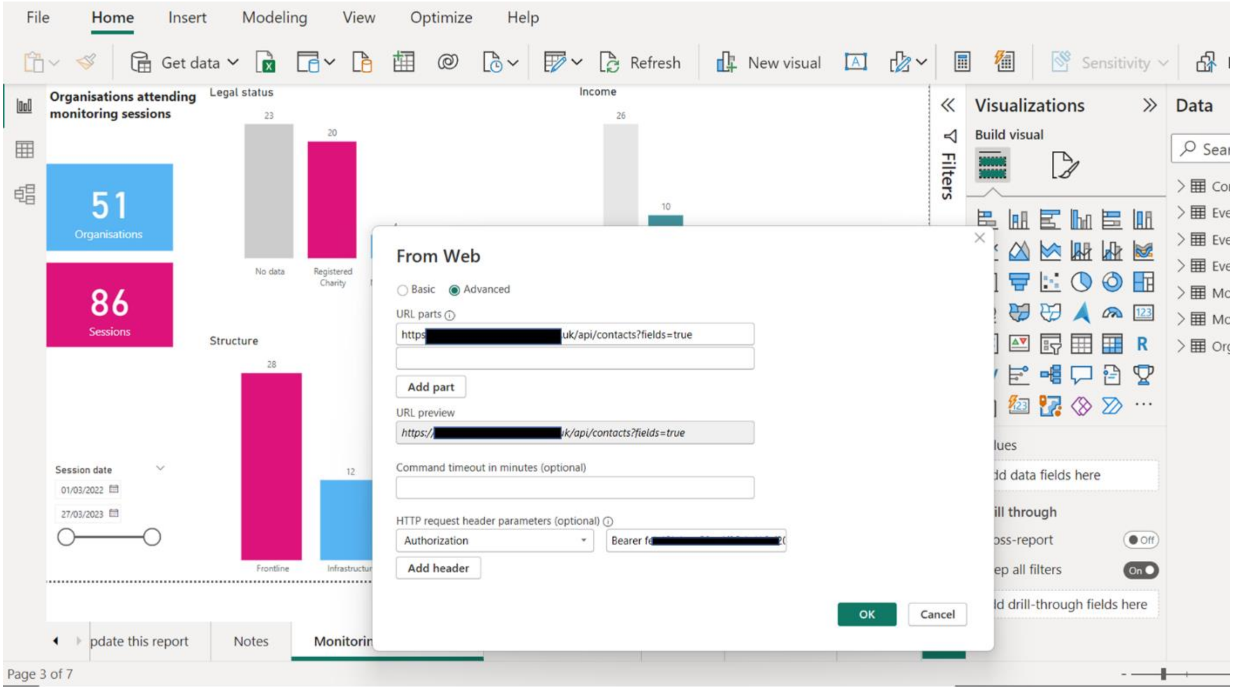
Contact your database provider to ask for **API URLs** and an **API Key** for all the different tables you want to draw data from. For example, organisations, contacts, events, sessions, outcomes etc.

The **API URL** is the unique URL for each database table.

The **API key** is the unique identifier that authenticates requests to your database. This is like a password into your database so should be stored in a secure location.

Once you have these 2 bits of information you'll need to enter them as follows:

1. Select the option **Advanced**
2. Enter the API URL in the field **URL parts**
3. In the **HTTP request header parameters (optional)** field type the word **Authorization** into the first box (don't worry that it isn't on the drop down list, this is just a list of the most frequently used requests but you can simply type in your own request) and in the second box type the word Bearer, then a space, then your API code.
4. Click on **OK**



From Web

○ Basic ● Advanced

URL parts

https[redacted]uk/api/contacts?fields=true

Add part

URL preview

https://[redacted]uk/api/contacts?fields=true

Command timeout in minutes (optional)

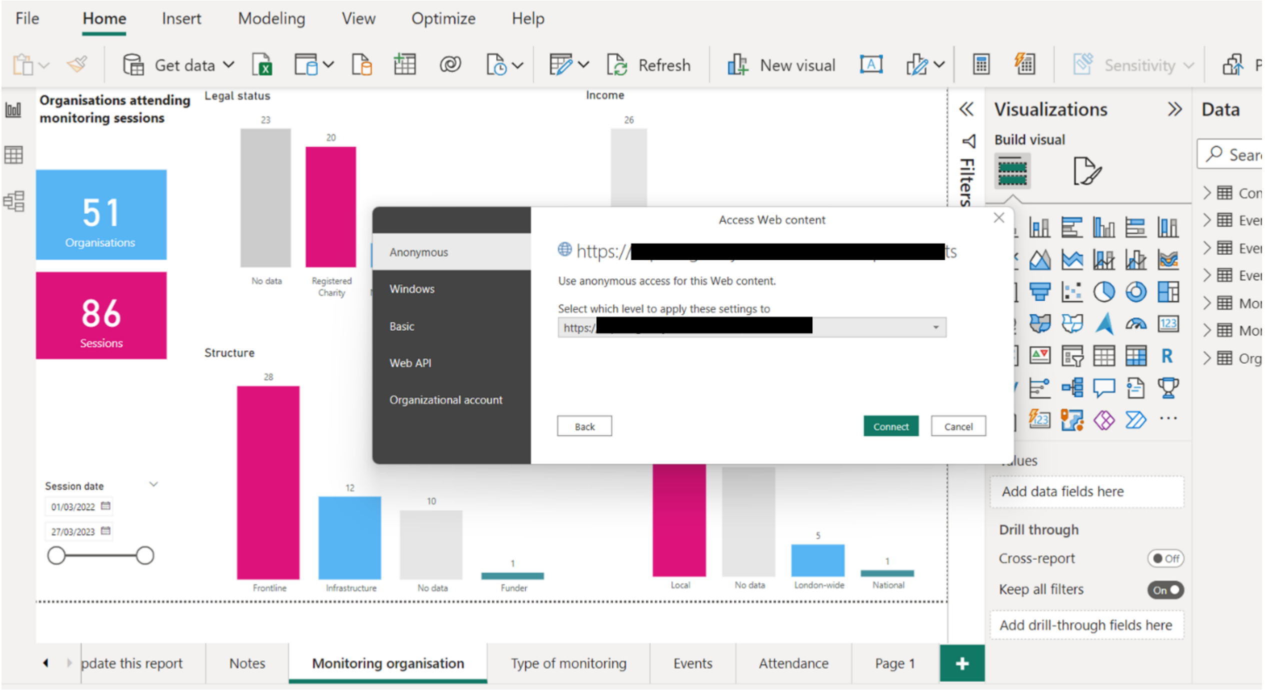
HTTP request header parameters (optional)

Authorization | Bearer [redacted]

Add header

OK Cancel

5. Click on **Connect**.



Access Web content

https://[redacted]ts

Use anonymous access for this Web content.

Select which level to apply these settings to

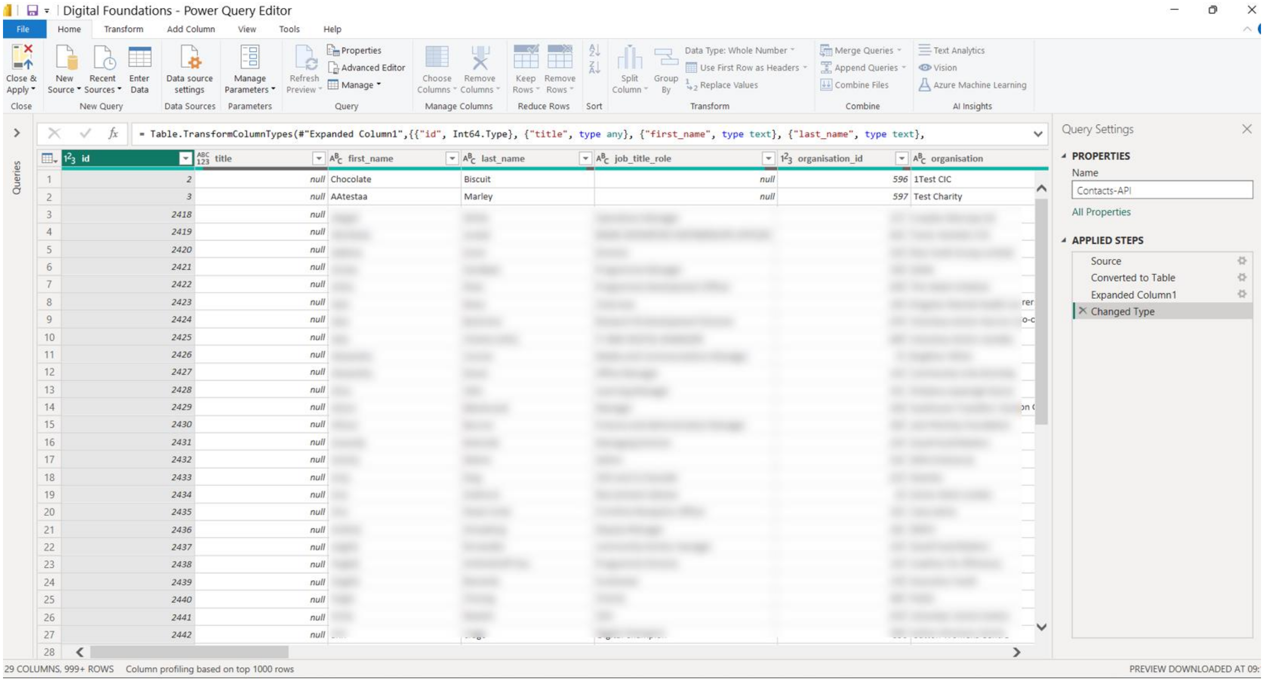
https:/[redacted]

Step 3: Loading the results in Power Query

Assuming all goes to plan the results of your API call should load as a query in the **Power Query Editor**.

Rename your query so that it's easy to identify in your data model.

Close and load the query.



```
= Table.TransformColumnTypes(#"Expanded Column1",{{"id", Int64.Type}, {"title", type any}, {"first_name", type text}, {"last_name", type text},
```

	id	title	first_name	last_name	job_title_role	organisation_id	organisation
1	2	null	Chocolate	Biscuit	null	596	1Test CIC
2	3	null	AAtestaa	Marley	null	597	Test Charity
3	2418	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
4	2419	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
5	2420	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
6	2421	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
7	2422	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
8	2423	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
9	2424	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
10	2425	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
11	2426	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
12	2427	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
13	2428	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
14	2429	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
15	2430	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
16	2431	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
17	2432	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
18	2433	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
19	2434	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
20	2435	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
21	2436	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
22	2437	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
23	2438	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
24	2439	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
25	2440	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
26	2441	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
27	2442	null	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]

Query Settings

PROPERTIES

Name: Contacts-API

All Properties

APPLIED STEPS

- Source
- Converted to Table
- Expanded Column1
- Changed Type

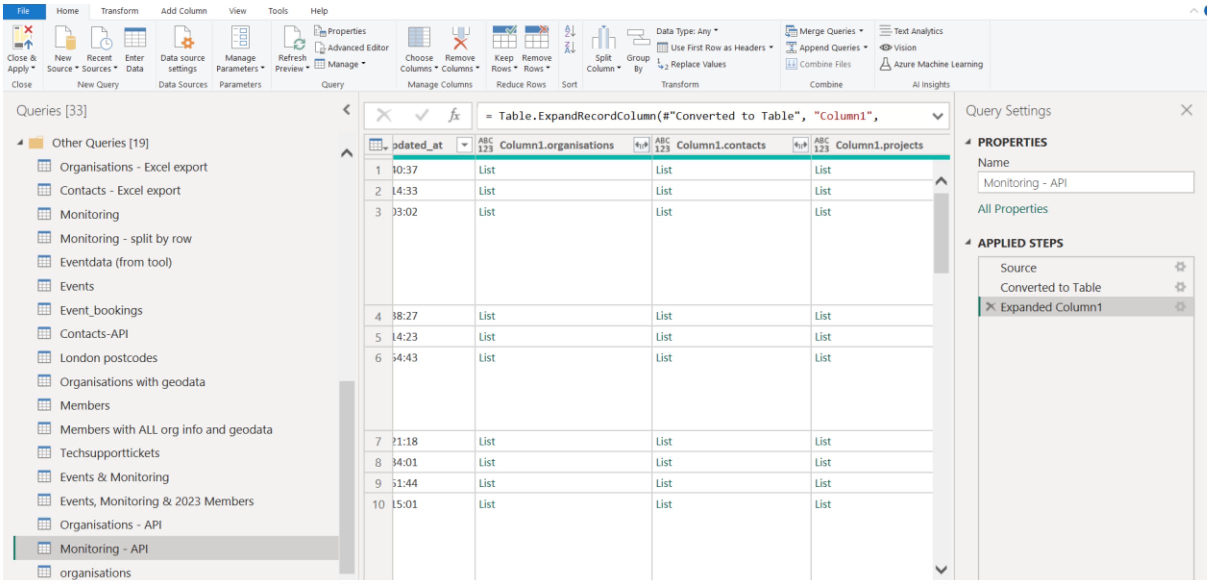
29 COLUMNS, 999+ ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 09:

Extract nested data in queries

For some tables there will be nested data where fields have a one-to-many relationship. For example, in Monitoring, the Contacts column is nested because more than one person can attend one monitoring session. This is the same for Organisations & Projects.

In columns with nested data, instead of a value the word List will appear (screenshot below)



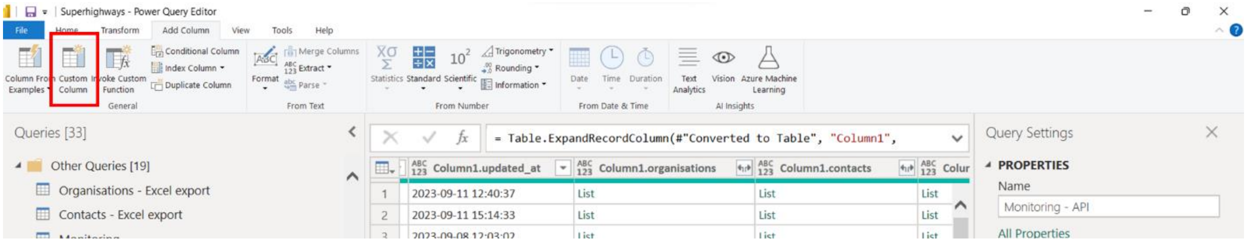
= Table.ExpandRecordColumn(#"Converted to Table", "Column1",

Name: Monitoring - API

	pdated_at	Column1.organisations	Column1.contacts	Column1.projects
1	10:37	List	List	List
2	14:33	List	List	List
3	03:02	List	List	List
4	38:27	List	List	List
5	14:23	List	List	List
6	54:43	List	List	List
7	21:18	List	List	List
8	34:01	List	List	List
9	51:44	List	List	List
10	15:01	List	List	List

To convert these nested fields into data you can use for reporting you need to create custom columns which extract the data. The steps below are for organisation but are almost identical for contacts & projects with just the column names & the error handling message needing alteration.

1. Open the query in the query editor
2. Duplicate the original API query by right clicking on it and selecting duplicate
3. Rename & open the new query e.g. Monitoring API - extracted
4. Find the organisations column and make a note of the column name i.e. *Column1.organisations* so that you can refer to it when writing your formula later in the process
5. In the **Add column** menu, click on **Custom column** button in the ribbon

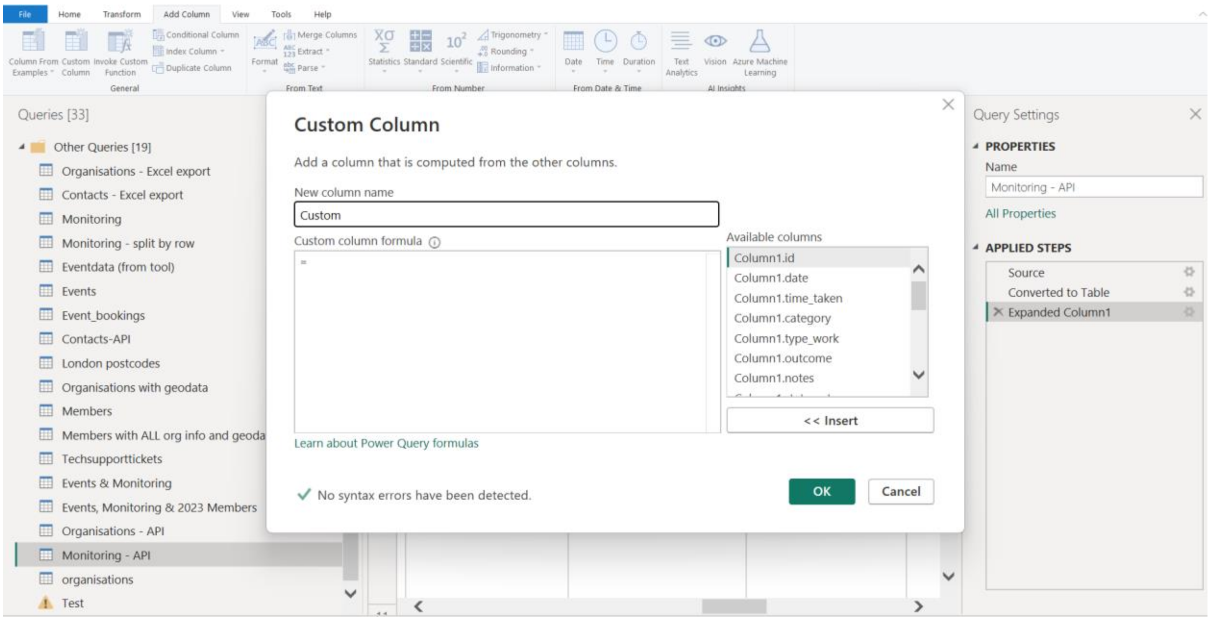


= Table.ExpandRecordColumn(#"Converted to Table", "Column1",

Name: Monitoring - API

	Column1.updated_at	Column1.organisations	Column1.contacts	Colur
1	2023-09-11 12:40:37	List	List	List
2	2023-09-11 15:14:33	List	List	List
3	2023-09-08 12:03:02	List	List	List

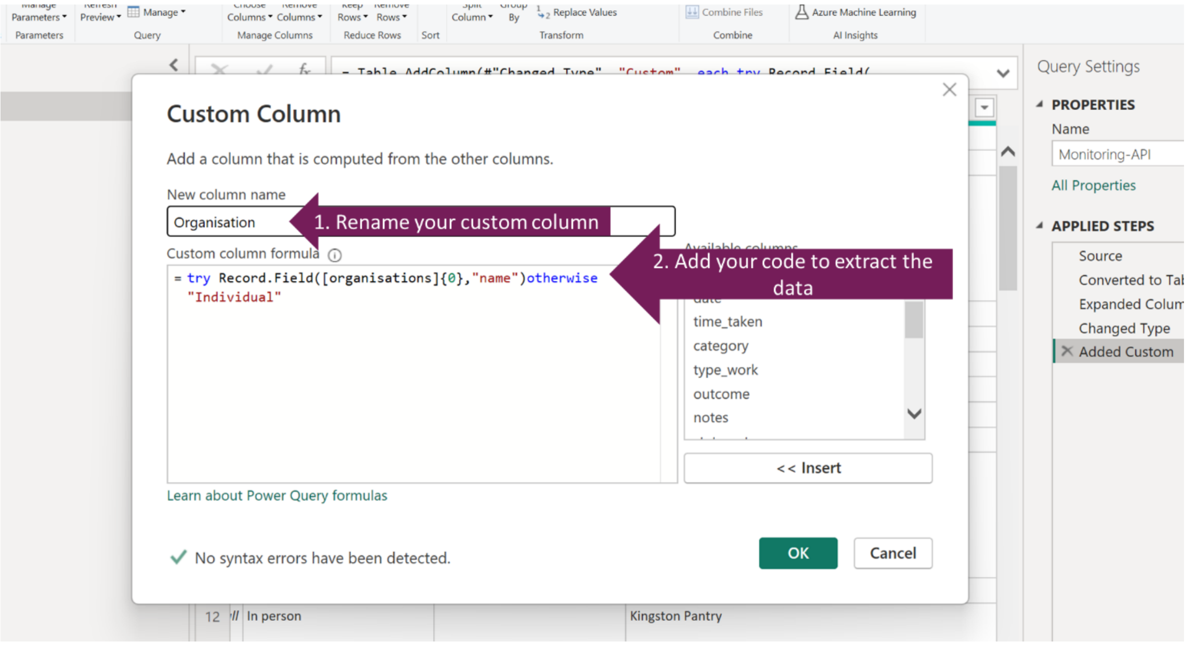
6. This will open a new window where you can define what you want to appear in the custom column.



7. **Rename** the custom column to Organisation name

   Add the following code to the custom formula box at the = sign:
   try Record.Field([organisations]{0},"name")otherwise "Individual"

   The Otherwise Individual part of the code is an error handling statement. As if there is no organisation associated



New column name: Organisation

Custom column formula:

```
= try Record.Field([organisations]{0},"name")otherwise
  "Individual"
```

1. Rename your custom column

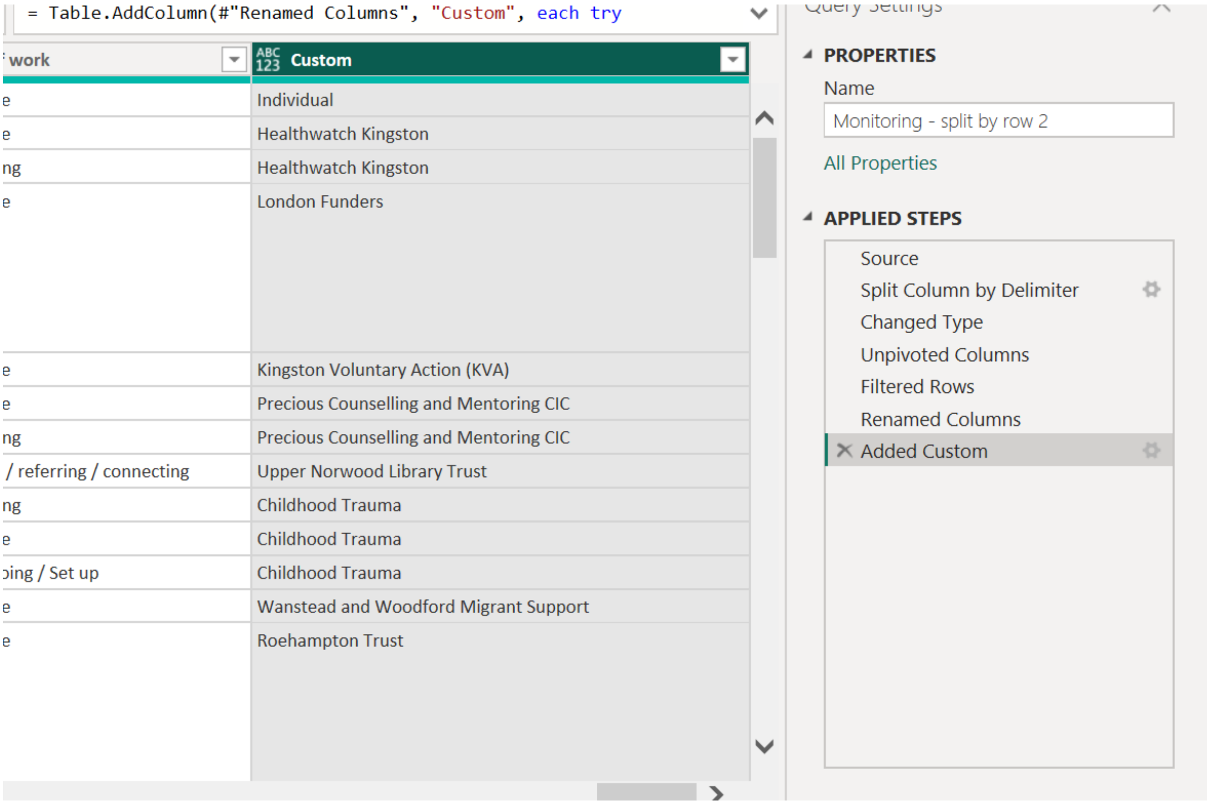
2. Add your code to extract the data

8. **Repeat** the steps above to add a custom column for the Organisation ID. Amend the code to replace “name” with “id” . So the code will be as follows:

   try Record.Field([organisations]{0},“id")otherwise "Individual"

9. **Rename** the original Organisations column with the word list in it to make it clear which one to use in relationship mapping & visuals e.g. Unextracted Organisations
10. **Close & Apply** the Query
11. Cick on OK

A new column will appear called Custom



```
= Table.AddColumn(#"Renamed Columns", "Custom", each try
```

work	Custom
e	Individual
e	Healthwatch Kingston
ng	Healthwatch Kingston
e	London Funders
e	Kingston Voluntary Action (KVA)
e	Precious Counselling and Mentoring CIC
ng	Precious Counselling and Mentoring CIC
/ referring / connecting	Upper Norwood Library Trust
ng	Childhood Trauma
e	Childhood Trauma
ɔing / Set up	Childhood Trauma
e	Wanstead and Woodford Migrant Support
e	Roehampton Trust

Query Settings

PROPERTIES

Name

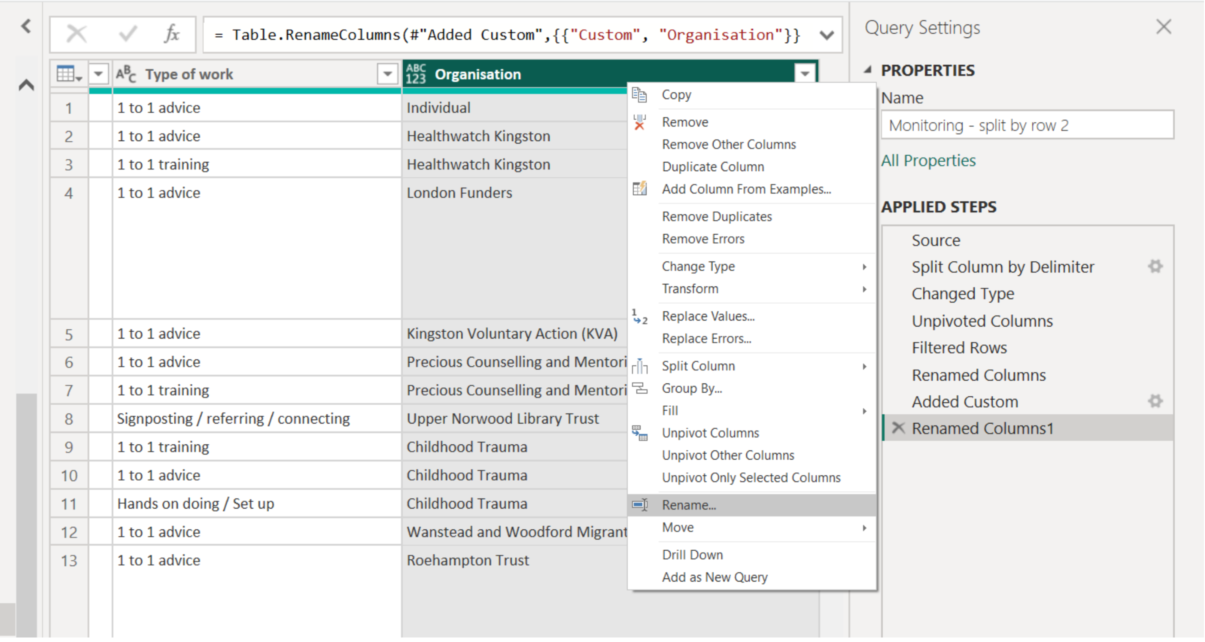
Monitoring - split by row 2

All Properties

APPLIED STEPS

- Source
- Split Column by Delimiter
- Changed Type
- Unpivoted Columns
- Filtered Rows
- Renamed Columns
- Added Custom

12. Rename the column to Organisation by right clicking on the column heading and selecting Rename from the menu



```
= Table.RenameColumns(#"Added Custom",{{"Custom", "Organisation"}})
```

	Type of work	Organisation
1	1 to 1 advice	Individual
2	1 to 1 advice	Healthwatch Kingston
3	1 to 1 training	Healthwatch Kingston
4	1 to 1 advice	London Funders
5	1 to 1 advice	Kingston Voluntary Action (KVA)
6	1 to 1 advice	Precious Counselling and Mentori
7	1 to 1 training	Precious Counselling and Mentori
8	Signposting / referring / connecting	Upper Norwood Library Trust
9	1 to 1 training	Childhood Trauma
10	1 to 1 advice	Childhood Trauma
11	Hands on doing / Set up	Childhood Trauma
12	1 to 1 advice	Wanstead and Woodford Migrant
13	1 to 1 advice	Roehampton Trust

Query Settings

PROPERTIES

Name

Monitoring - split by row 2

All Properties

APPLIED STEPS

- Source
- Split Column by Delimiter
- Changed Type
- Unpivoted Columns
- Filtered Rows
- Renamed Columns
- Added Custom
- Renamed Columns1

You will now need to do the same process for Individuals & Projects

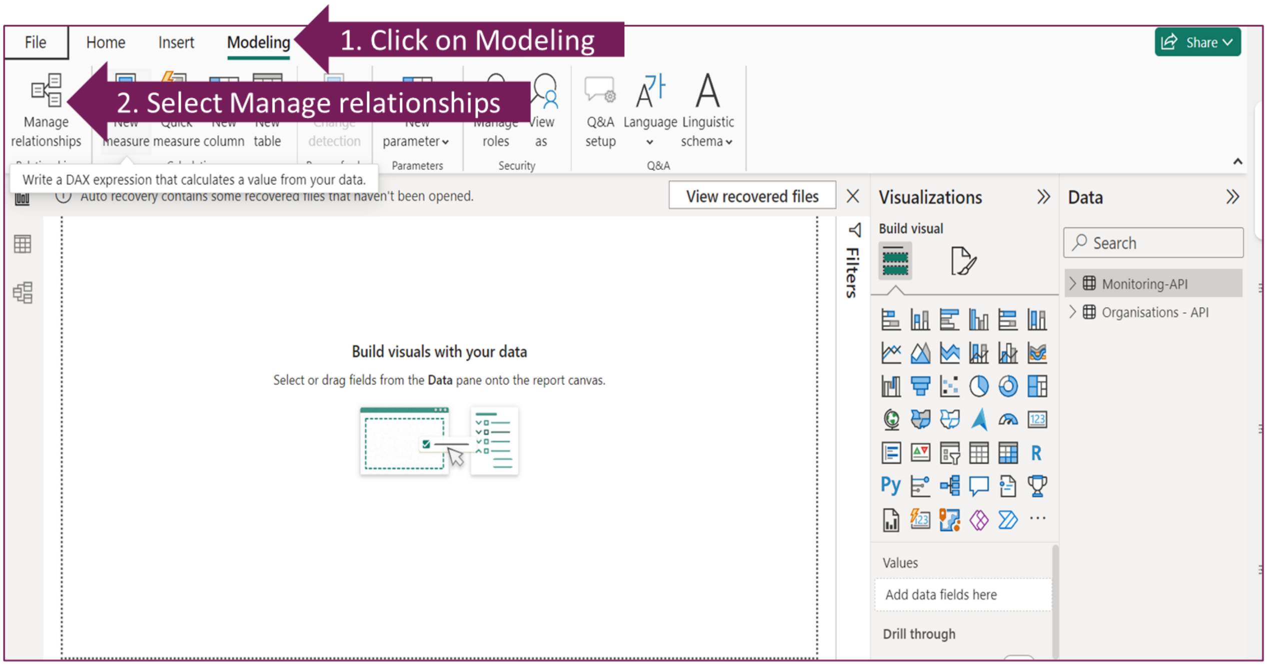
Manage relationships

In PowerBI you will need to recreate the relationships between the entities in your database e.g. link Contact to Organisations. PowerBI will attempt to spot relationships but you'll need to ensure they are correct.

There are 2 ways of managing relationships, one list format and one visual. It's worth trying both to see which one you prefer. Both have the same functionality:

Managing relationships in a list

1. Click on the **Modeling** tab
2. Click on the first button in the ribbon, **Manage relationships**



1. Click on Modeling

2. Select Manage relationships

Write a DAX expression that calculates a value from your data.

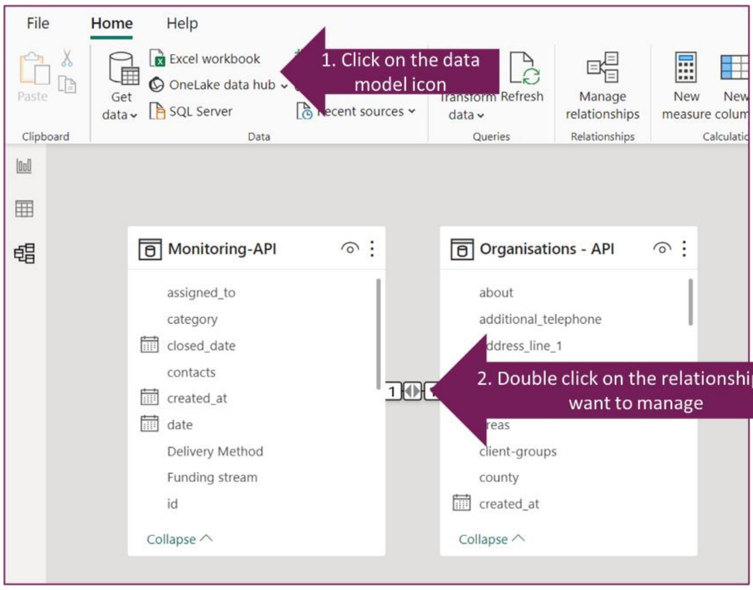
Auto recovery contains some recovered files that haven't been opened.

Build visuals with your data

Select or drag fields from the **Data** pane onto the report canvas.

Managing relationships in a data model diagram

1. Click on the **Data model** icon
2. Double click on the relationship you want to manage



1. Click on the data model icon

2. Double click on the relationship you want to manage

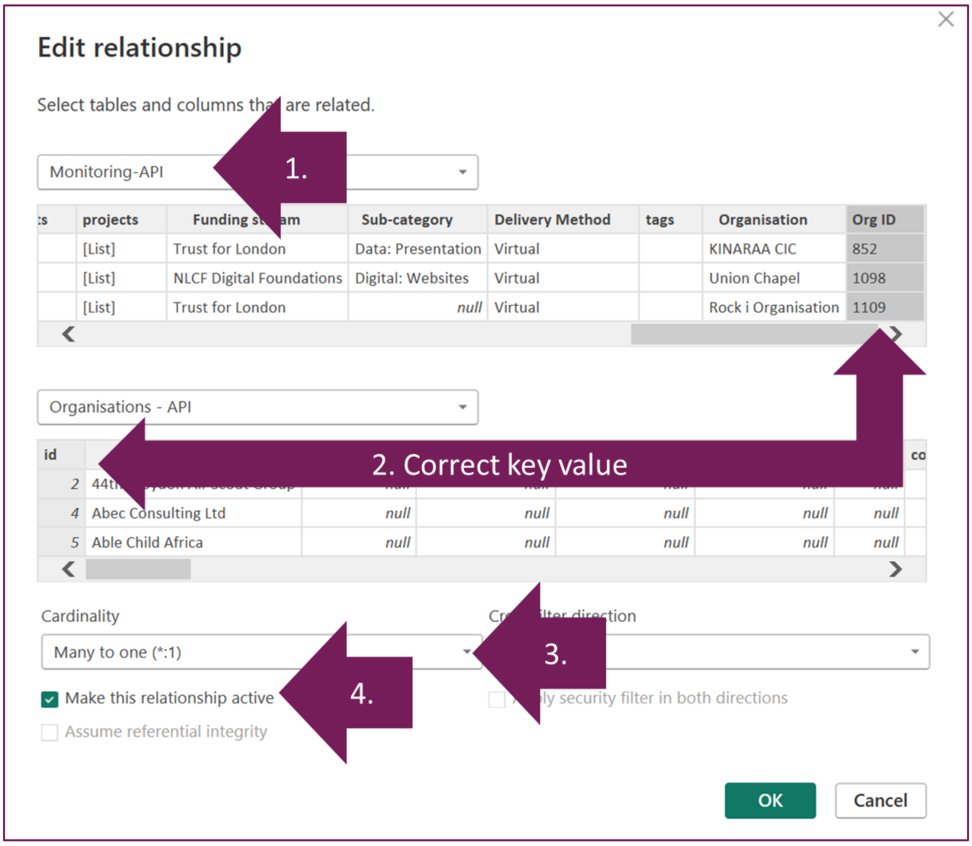
Editing a relationship

The PowerBI resource site has a comprehensive guide on relationships to follow https://learn.microsoft.com/en-us/power-bi/transform-model/desktop-create-and-manage-relationships

There are 4 key things to check when editing a relationship:

1. Are the correct tables selected?
2. Are the correct key value fields selected?
3. Is the relationship type correct e.g. one to one or one to many?
4. Is the relationship active?

The screen shot below shows how the relationship between Monitoring & Organisations should look



Edit relationship

Select tables and columns that are related.

Monitoring-API

ts	projects	Funding stream	Sub-category	Delivery Method	tags	Organisation	Org ID
	[List]	Trust for London	Data: Presentation	Virtual		KINARAA CIC	852
	[List]	NLCF Digital Foundations	Digital: Websites	Virtual		Union Chapel	1098
	[List]	Trust for London	null	Virtual		Rock i Organisation	1109

Organisations - API

id							co
2	44t[illegible] Air Scout Group	null	null	null	null	null	
4	Abec Consulting Ltd	null	null	null	null	null	
5	Able Child Africa	null	null	null	null	null	

Cardinality: Many to one (*:1)

Cross filter direction:

☑ Make this relationship active

□ Assume referential integrity

□ Apply security filter in both directions

Visualising the data

Now you have set up your relationships you can use data from both tables in one visual.

Join our PowerBI training, download our PowerBI resources from our Datawise London site or join the Superhighways' Power BI user group to access learning resources and a peer support community