Power BI Users
- Power Query Various Tips -

Good we are all here again and moving fast toward our gaols.

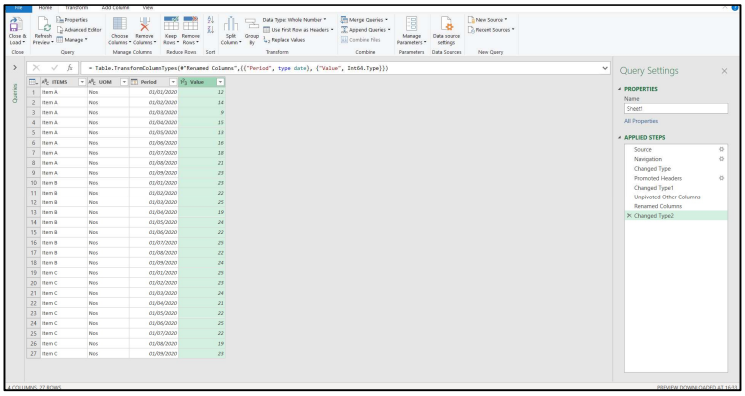
Today we will cover some Power Query tips to enhance our knowledge of the tools and improve our data import/cleaning workflow as well solve some of the potential compatibility Issues.

Ok let's start and see quickly what we will be exploring today:

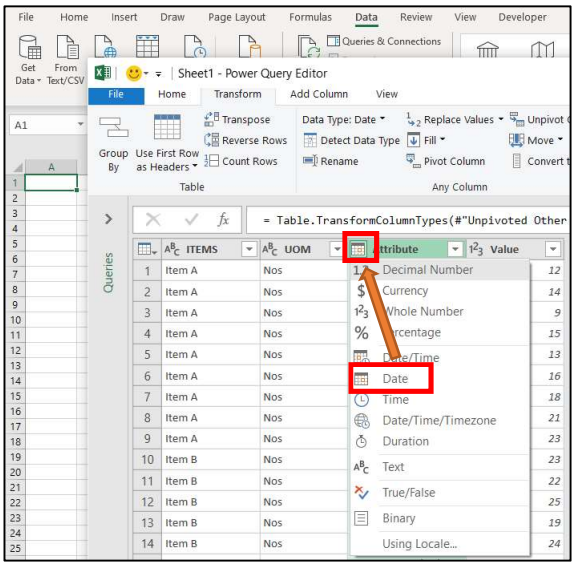
1) International Data Issues
2) Trim
3) Replace
4) Split Column
5) Letters Capitalization
6) Group By

1) International Data Issues

In the example we saw two session ago we change the data type of the unpivot month field into "Date"



Using the conventional data type from the drop-down menu



That's great, however when we are working in an international environment.

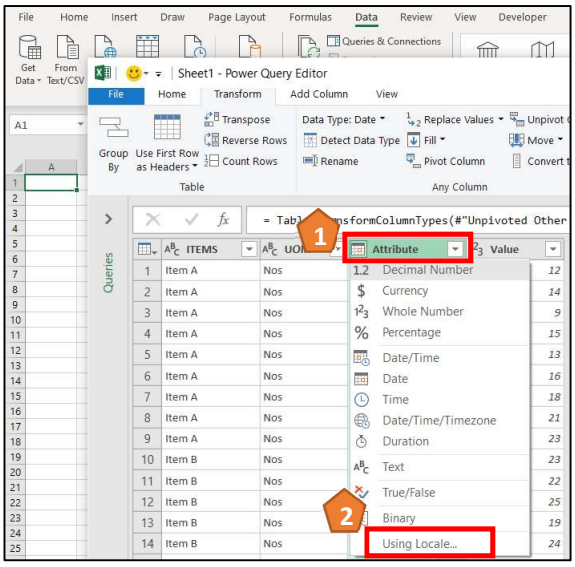
In US the date model is conventionally mm.dd.yyy while in Europe it is dd.mm.yyyy and here it is the problem when having a file coming from different source.

In this case Power BI will fail to recognise the date and the 12th July 2020 (07.12.2020 in US format) may be read as 07th December 2020 (12.07.2020) at the best or even found some errors with-out data.

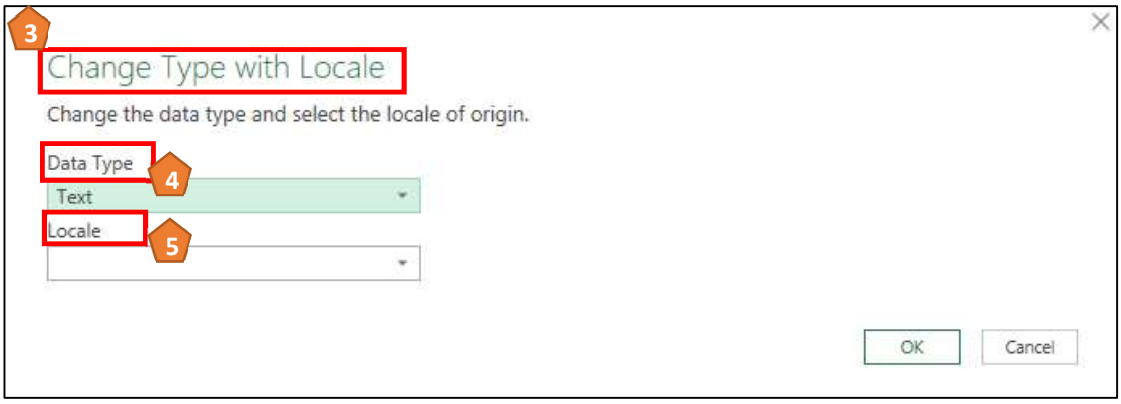
In our past life we used trim, length and some MsExcel function to isolate the day, month and year component of the date and rebuild the date after and the same we could be doing using the M code of Power Query (the language that we will explore in later sessions), but is there an easier solution?

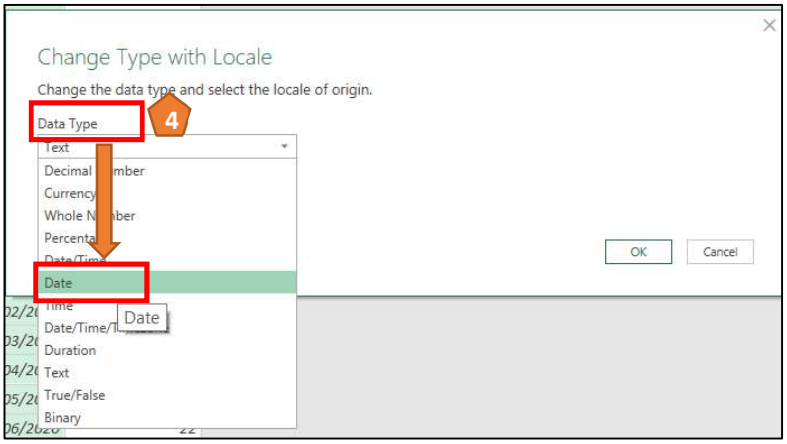
The good news is that Power Query has a really easy way of doing this for you, even if it's not immediately obvious where this functionality is. You don't need to change your Windows locale or split dates into their constituent parts and rearrange them in the correct order, or anything like that.

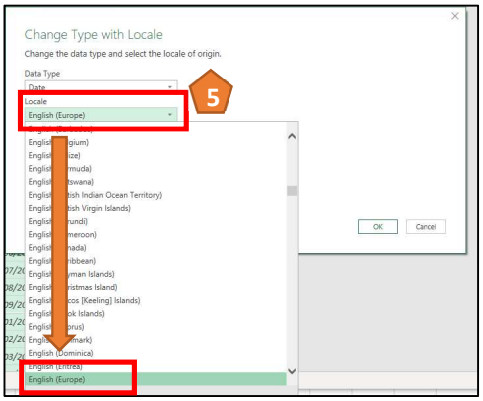
Ohh yes, we can use the "Using Locale" (2).



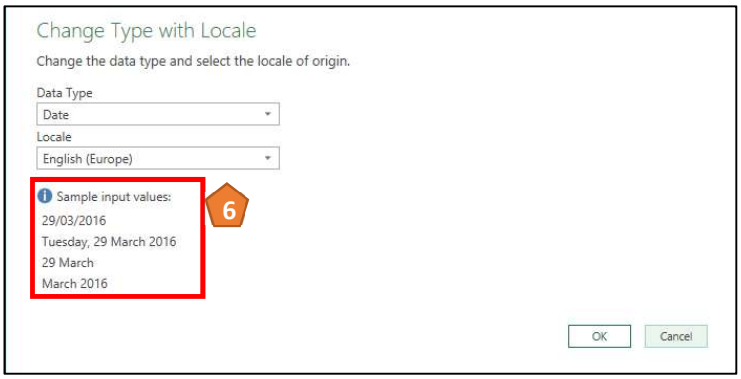
That will open a window “Change Type with Locale”, from which we do need to set the “Data Type” (3) and “Locale” (4)







With the “Data Type” and “Locale” selection we will be presented with a sample of the format (6)



A locale is simply a name for all of the rules for date and number formats and more associated with a particular language and region. So, for example, setting the Date column so that it is interpreted using the English (United States) locale, means that when the data is loaded into Power Query on my machine I see the date 3/2/2015.

What's happened is that the csv file contains the date "2/3/2015", I have told Power Query that the data source uses a US English format date, Power Query has then assumed this date is therefore in MM/DD/YYYY format, loaded the data for me and shown the date in my own locale (which is English (United Kingdom)) in DD/MM/YYYY format as 3/2/2015. The date itself hasn't changed, just the way it is displayed.

Yeah at the end I used the USA and UK example, as most of the time we have to deal with them.

The same workflow can be used to import number with the dot or comma as decimal separator without messing around the International Settings.

2) Trim

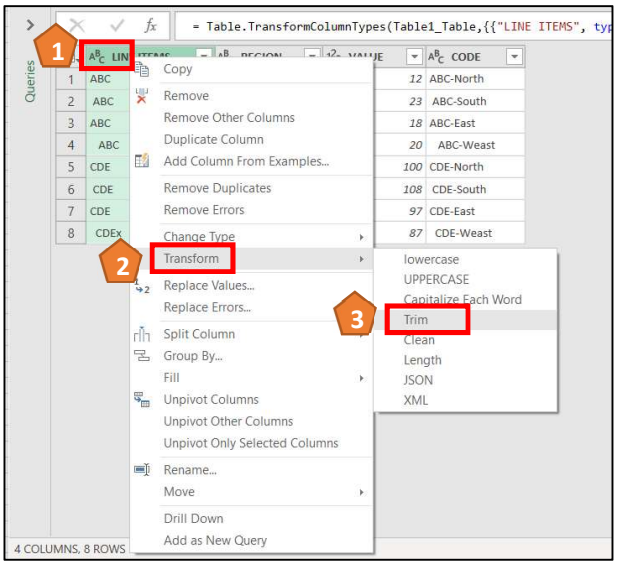
For the next few examples we will be using the data from the table below

LINE ITEMS	REGION	VALUE	CODE
ABC	North	12	ABC-North
ABC	South	23	ABC-South
ABC	East	18	ABC-East
ABC	Weast	20	ABC-Weast
CDE	North	100	CDE-North
CDE	South	108	CDE-South
CDE	East	97	CDE-East
CDEx	Weast	87	CDE-Weast

We notice that, for any reason the "Line Items" values come with various spaces in front of their values, but we know that regardless of this the line item "ABC" is the same regardless having one or more spaces in front of it.

During our import we do Need to "Trim" the values of the "Line Items" Field.

We right click (1) on the field we do want to trim to open the contextual window, we expand the "Transform" (2) option and click "Trim" (3)



	LINE ITEMS	REGION	VALUE	CODE
1	ABC	North	12	ABC-North
2	ABC	South	23	ABC-South
3	ABC		18	ABC-East
4	ABC		20	ABC-Weast
5	CDE		100	CDE-North
6	CDE	South	108	CDE-South
7	CDE	East	97	CDE-East
8	CDEx	Weast	87	CDE-Weast

All the entries for the Line Item Field been trimmed

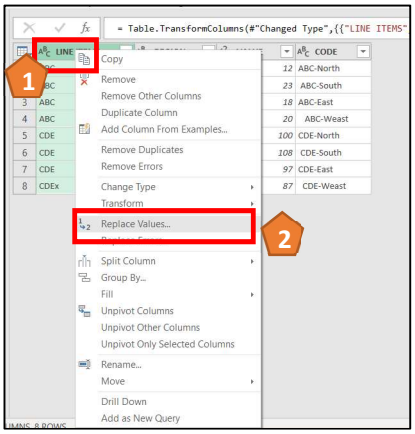
3) Replace

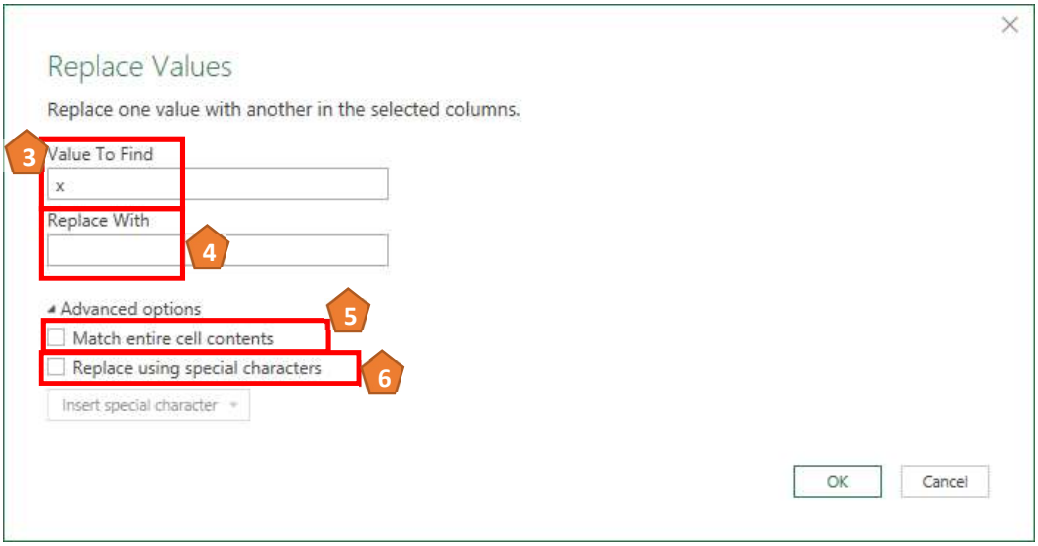
Let's use again the same example, but this time after we remove the spaces before the Line Items field we do want to replace the value "CDEx" with "CDE".

LINE ITEMS	REGION	VALUE	CODE
ABC	North	12	ABC-North
ABC	South	23	ABC-South
ABC	East	18	ABC-East
ABC	Weast	20	ABC-Weast
CDE	North	100	CDE-North
CDE	South	108	CDE-South
CDE	East	97	CDE-East
CDEx	Weast	87	CDE-Weast

We right click (1) on the field we do want to trim to open the contextual window, we click on “Replace Values” (2) and a window will pop up.

At this point in “Value to Find” (3) we could choose “x” and the replace in “Replace With” (4) with a no character “” or find “CDEx” and replace with “CDE”.

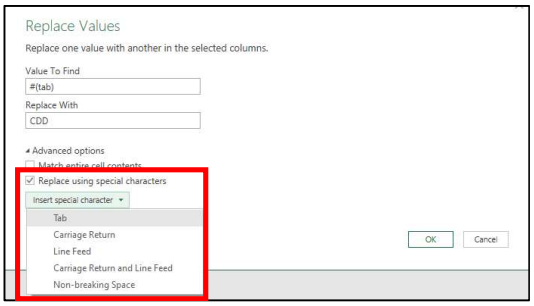




	LINE ITEMS	REGION	VALUE	CODE
1	ABC	North	12	ABC-North
2	ABC	South	23	ABC-South
3	ABC	East	18	ABC-East
4	ABC	Weast	20	ABC-Weast
5	CDE	North	100	CDE-North
6	CDE	South	108	CDE-South
7	CDE	East	97	CDE-East
8	CDE	Weast	87	CDE-Weast

Click “Match entire cell contents” (5) option to replace cells from the entire contents; otherwise, “Replace Values” replaces each Value To Find match.

Last click “Replace Using Special Characters” (6) option to replace using wild characters as below



4) Split Column

Let’s use the same file and we know we do have all the information in the “Line Items” and “Region” field, but assume we did not, and we do need to use the “Code” field to extract the “Line Item” and “Region”.

LINE ITEMS	REGION	VALUE	CODE
ABC	North	12	ABC-North
ABC	South	23	ABC-South
ABC	East	18	ABC-East
ABC	Weast	20	ABC-Weast
CDE	North	100	CDE-North
CDE	South	108	CDE-South
CDE	East	97	CDE-East
CDEx	Weast	87	CDE-Weast

We should first of all Trim the values within the “Code” field, using the workflow that we just saw and then split the field content using the “-” as separator.

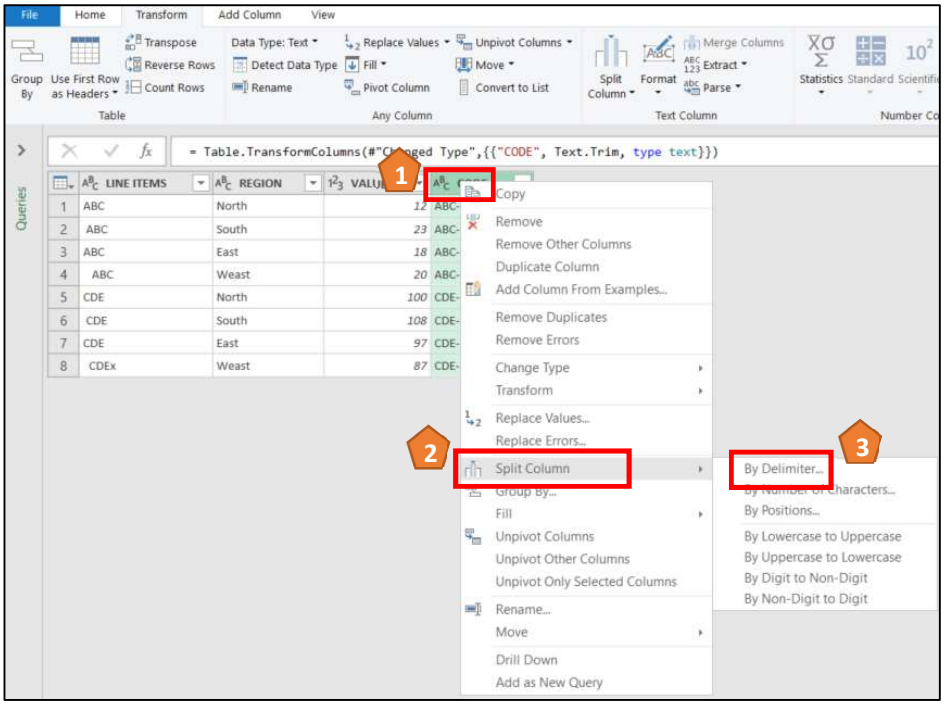
We upload the table into Power Query.

	LINE ITEMS	REGION	VALUE	CODE
1	ABC	North	12	ABC-North
2	ABC	South	23	ABC-South
3	ABC	East	18	ABC-East
4	ABC	Weast	20	ABC-Weast
5	CDE	North	100	CDE-North
6	CDE	South	108	CDE-South
7	CDE	East	97	CDE-East
8	CDEx	Weast	87	CDE-Weast

We trim the “CODE” field values as previously shown.

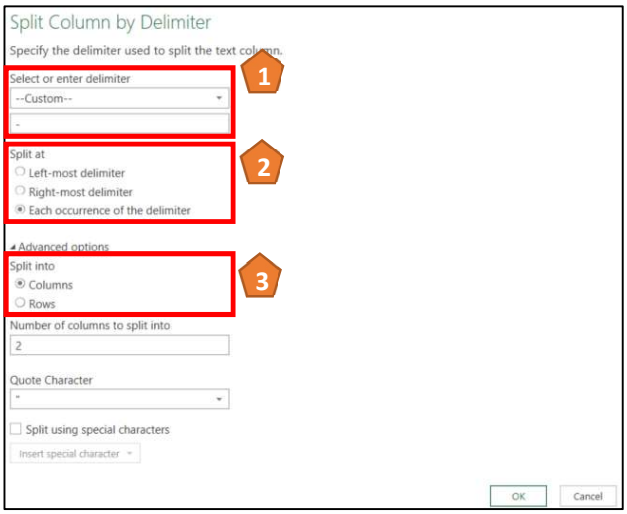
	LINE ITEMS	REGION	VALUE	CODE
1	ABC	North	12	ABC-North
2	ABC	South	23	ABC-South
3	ABC	East	18	ABC-East
4	ABC	Weast	20	ABC-Weast
5	CDE	North	100	CDE-North
6	CDE	South	108	CDE-South
7	CDE	East	97	CDE-East
8	CDEx	Weast	87	CDE-Weast

At this point we are ready to perform the Split, we right click (1) on the field we need to transform, in the context window we choose “Split Column” (2) and the “By Delimiter” (3).

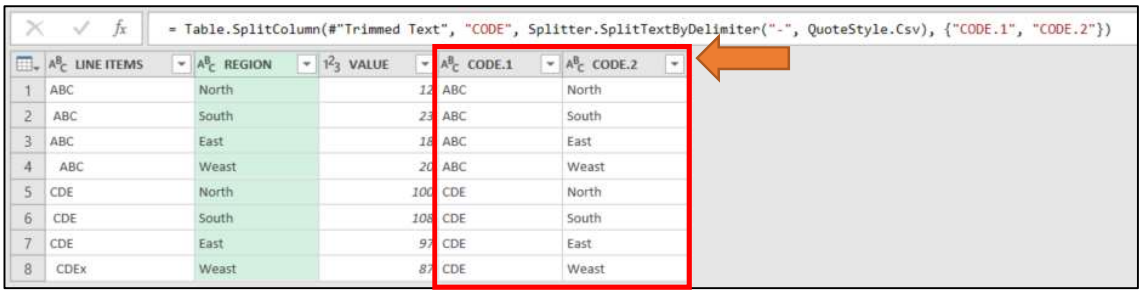


At this point a window open with the various parameters to perform our transformation / split:

1. Select the character/delimiter that trigger the split, in this case we choose a custom one “-”;
2. Select when and how to do the split, in this case for each occurrence of the delimiter; and
3. How the system will return the split, by column or row, in this case by columns.



We have now the "CODE" field split into two "CODE.1" and "CODE.2"



	LINE ITEMS	REGION	VALUE	CODE.1	CODE.2
1	ABC	North	12	ABC	North
2	ABC	South	23	ABC	South
3	ABC	East	18	ABC	East
4	ABC	Weast	20	ABC	Weast
5	CDE	North	100	CDE	North
6	CDE	South	108	CDE	South
7	CDE	East	97	CDE	East
8	CDEx	Weast	87	CDE	Weast

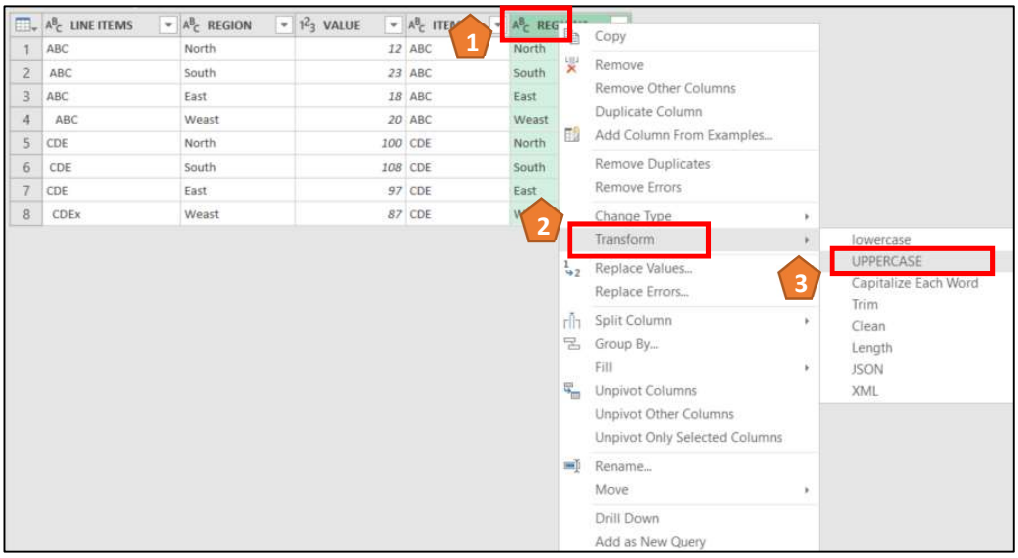
If wanted we can change their heading.

	LINE ITEMS	REGION	VALUE	ITEM	REGION2
1	ABC	North	12	ABC	North
2	ABC	South	23	ABC	South
3	ABC	East	18	ABC	East
4	ABC	Weast	20	ABC	Weast
5	CDE	North	100	CDE	North
6	CDE	South	108	CDE	South
7	CDE	East	97	CDE	East
8	CDEx	Weast	87	CDE	Weast

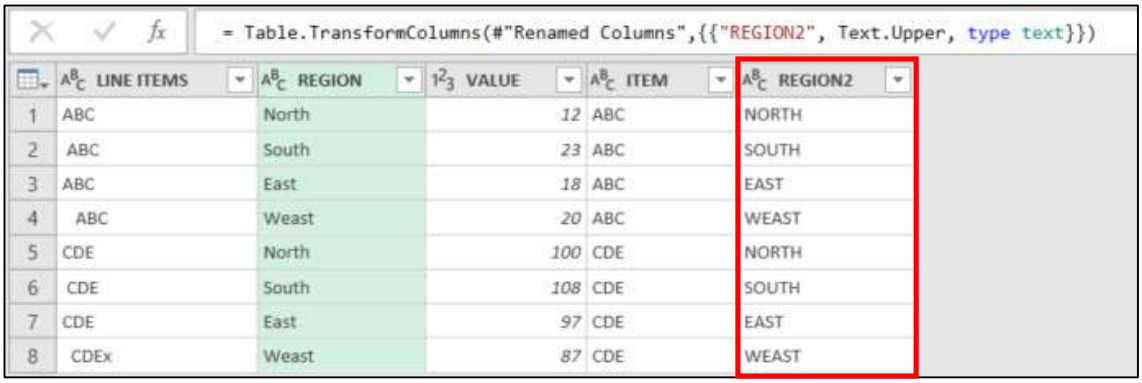
5) Letters Capitalization

Using the same source data file, in case the "REGION.2" field created in the previous tip is not consistently formatted, in terms of letter capitalization, we could perform a further transformation:

1. Right click on the field we do want to transform;
2. On the sensitive menu click "Transform"; and
3. Select "UPPERCASE".

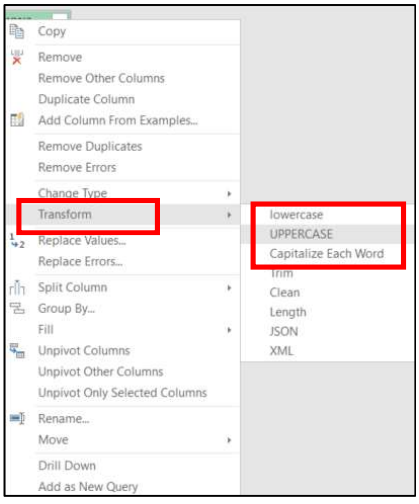


All the records associated with the "REGION2" field are not capitalised.



Similarly we could perform the following text transformations (that are self explanatory):

- lowercase;
- UPPERCASE; and
- Capitalise Each Word.



6) Group By

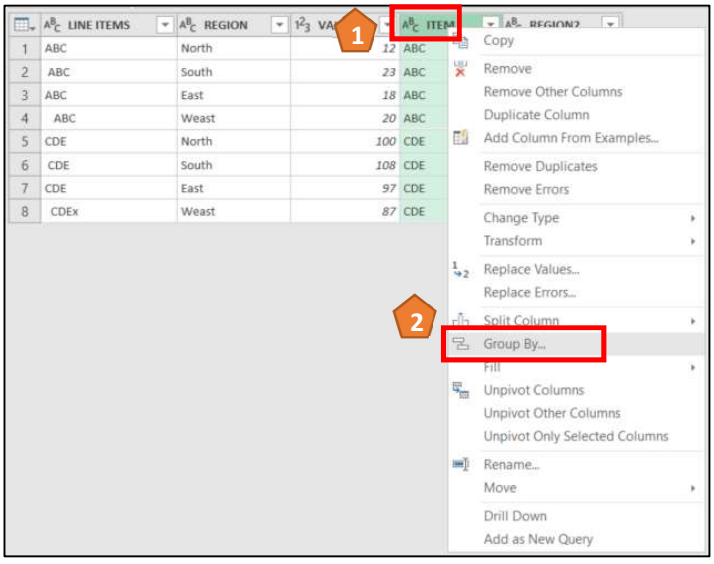
Using the same source data file, we will import the data already Grouped.

You will be asking the reason of this, as we could perform this in Power BI or Power Pivot, and you are right, however think of working with millions of records.

Is always good practice to understand the minimum level of drill down that is required on our data and upload the essential data (memory utilisation optimisation that lead to system good performance and reaction time).

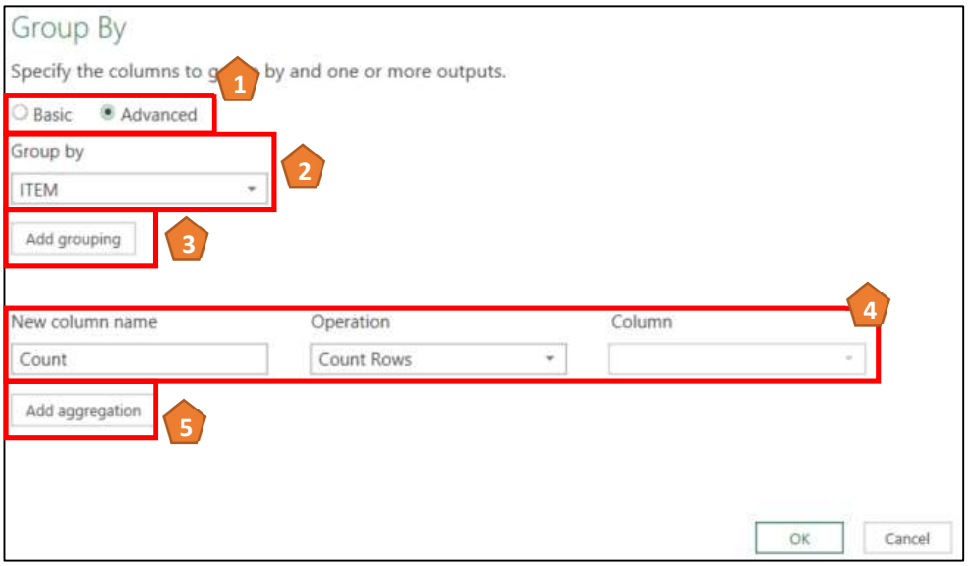
LINE ITEMS	REGION	VALUE	CODE
ABC	North	12	ABC-North
ABC	South	23	ABC-South
ABC	East	18	ABC-East
ABC	Weast	20	ABC-Weast
CDE	North	100	CDE-North
CDE	South	108	CDE-South
CDE	East	97	CDE-East
CDEx	Weast	87	CDE-Weast

1. Right click on the field we do want to group by; and
2. On the sensitive menu click “Group By…”.



The context window will open and we will:

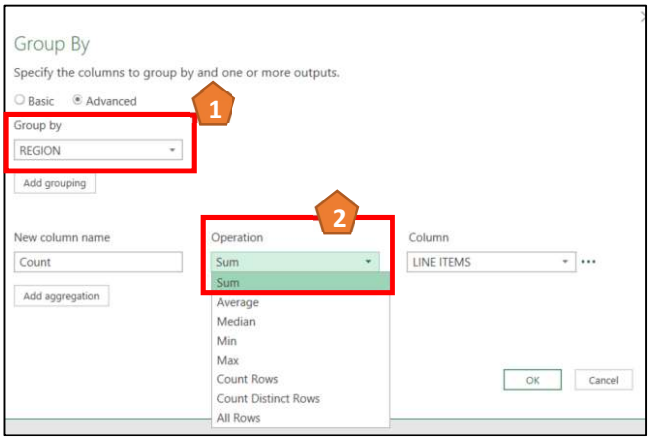
1. Click for the “Advanced” grouping option;
2. Select the filed we do want group by;
3. Add additional field to group by;
4. Insert the grouping aggregation calculation; and
5. Add further grouping aggregation calculations as necessary.

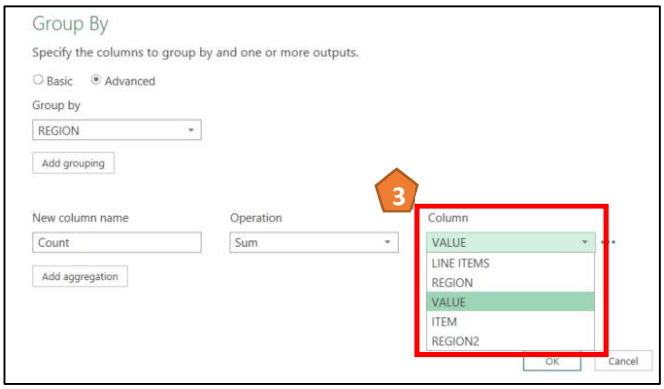


In this case we decide to group by "ITEM" field and for each group count the number of entries, number that will be saved into the new created "Count" field.

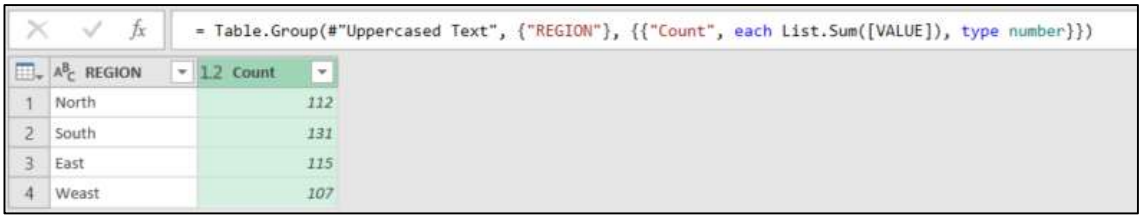
Now we do want to calculate the sum of "Value" grouped by "REGION", so we will:

1. Group by "REGION";
2. Select the "Sum" aggregation function; and
3. Sum to be performed on the "Value" field.

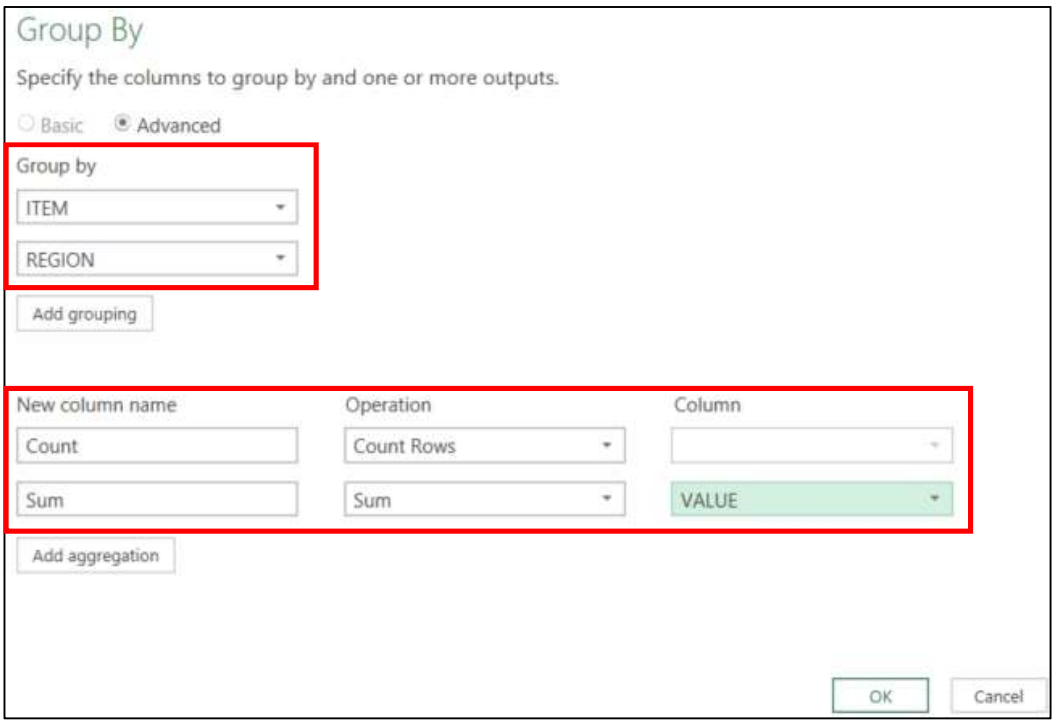




And here it is our grouped result.



Similarly if we wold group by “ITEM” and then “REGION” field, and return the count of entries for the combination of ITEM – REGION grouping along side the relevant sum of the “VALUE” field.



Good progress done with this session.

In these sessions I do no spend a lot of effort in explaining every single functionality, I rather prefer give you a solid basis from which you can start fly by yourself.

You will need to practice the topics presented in these sessions, and if want further enhance your knowledge with Mr Google. Yes Mr Google always knows everything as once a colleague of mine told me.

Soon we will complete the sessions on Power Query and move to the real magic of Power BI and Power Pivot the DAX language.