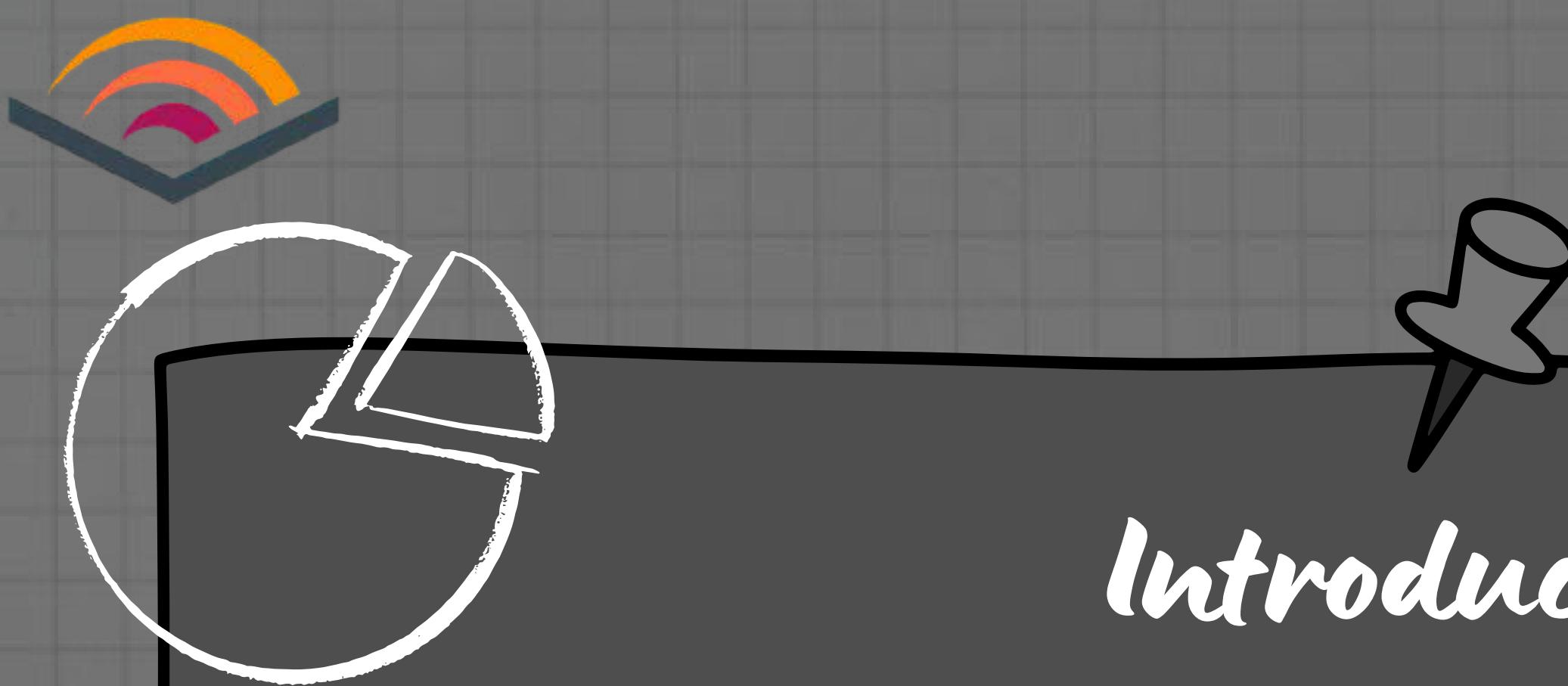




# Audible Data Cleaning Project

## PRESENTATION



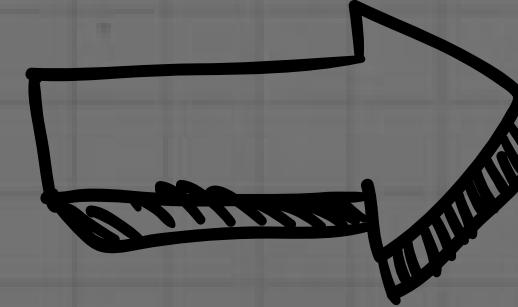
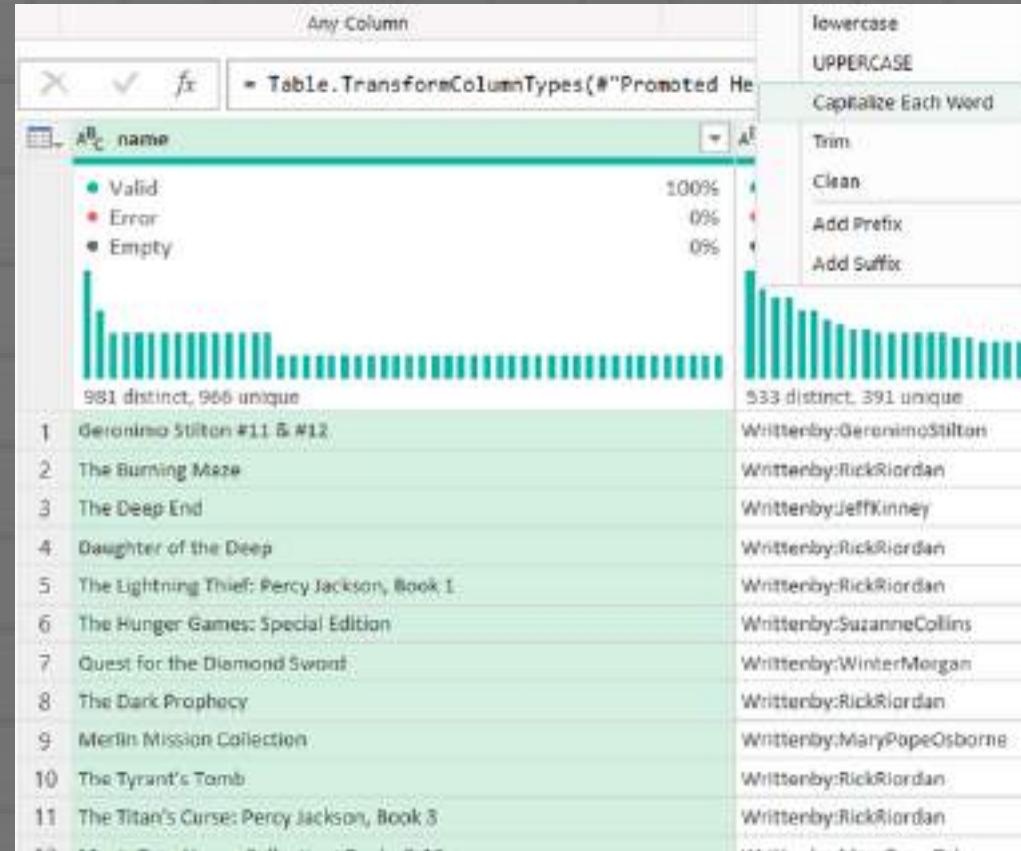
# *Introduction*

*The primary goal of this project was to clean and transform a raw dataset of Audible audiobooks using Power Query. The initial data suffered from significant quality issues, including inconsistent formatting, mixed data types, and unstructured text, making it unsuitable for reliable analysis.*





# Standardize the name column to ensure consistent title casing.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
Valid	981 distinct, 966 unique	100%	Error	0%	0%	Empty																					
1	Geronimo Stilton #11 & #12		Written by Geronimo Stilton		533 distinct, 391 unique																						
2	The Burning Maze		Written by Rick Riordan																								
3	The Deep End		Written by Jeff Kinney																								
4	Daughter of the Deep		Written by Rick Riordan																								
5	The Lightning Thief: Percy Jackson, Book 1		Written by Rick Riordan																								
6	The Hunger Games: Special Edition		Written by Suzanne Collins																								
7	Quest for the Diamond Sword		Written by Winter Morgan																								
8	The Dark Prophecy		Written by Rick Riordan																								
9	Merlin Mission Collection		Written by Mary Pope Osborne																								
10	The Tyrant's Tomb		Written by Rick Riordan																								
11	The Titan's Curse: Percy Jackson, Book 3		Written by Rick Riordan																								

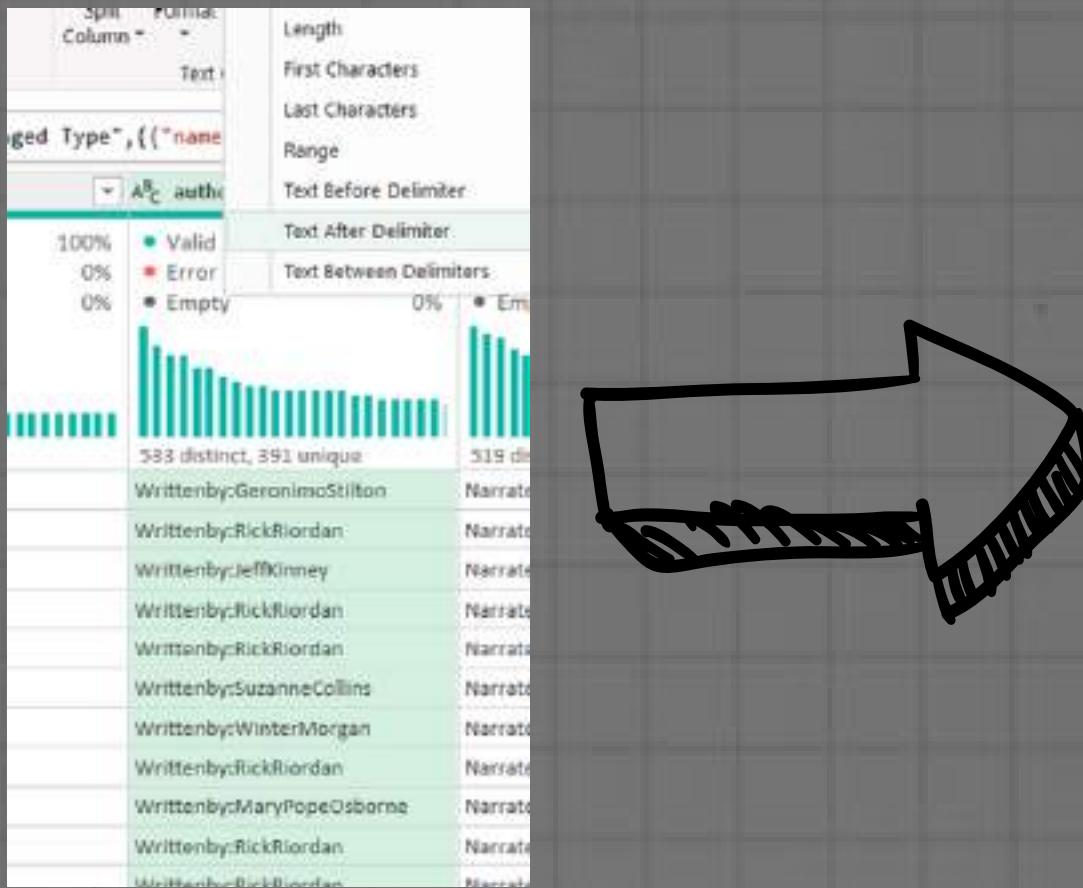
**Problem:** The book titles in the name column had inconsistent capitalization. This made the data look unprofessional and could lead to errors when sorting or grouping the titles.

**Action:** I right-clicked the name column header, selected the Transform option, and then chose "Capitalize Each Word" from the menu.

**Result:** All book titles are now uniformly formatted in a professional title case. This ensures the data is clean and consistent for any future analysis or reporting.

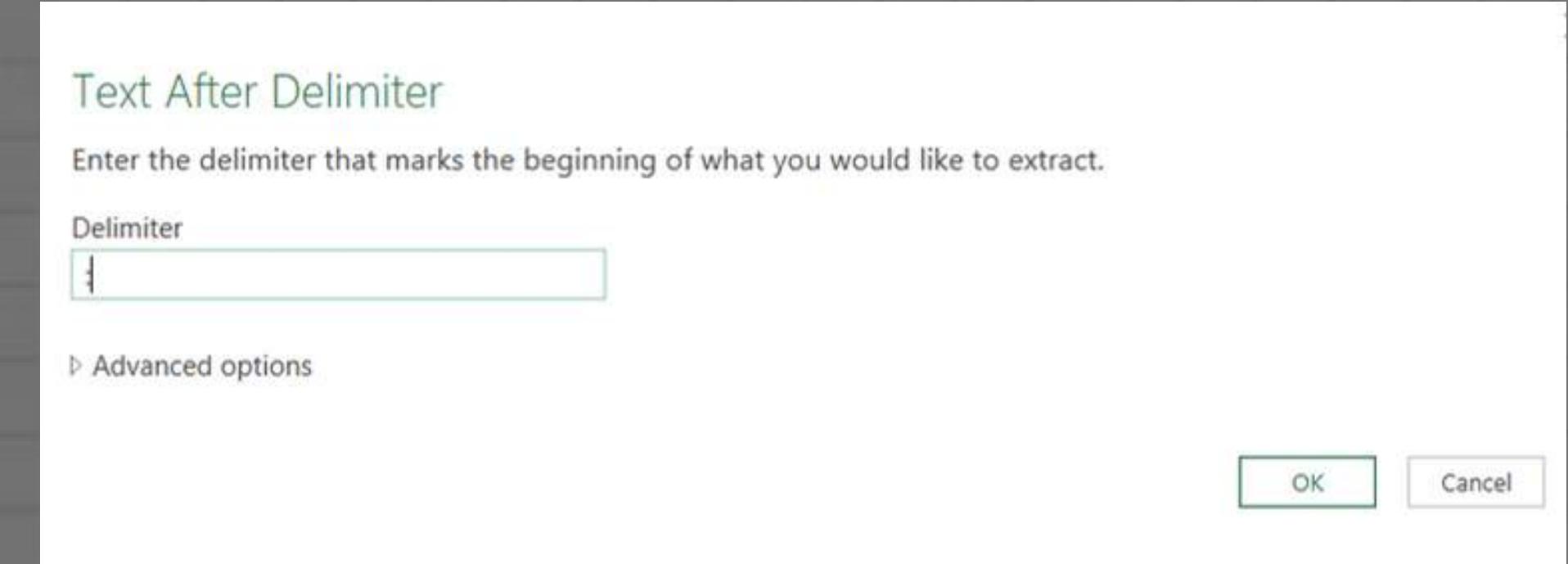


# Separate combined names in the author column if there are multiple authors.



**Problem:** The author column contained the unnecessary prefix "Writtenby:" before each name. This made the data look unprofessional and would cause errors in filtering or grouping by author.

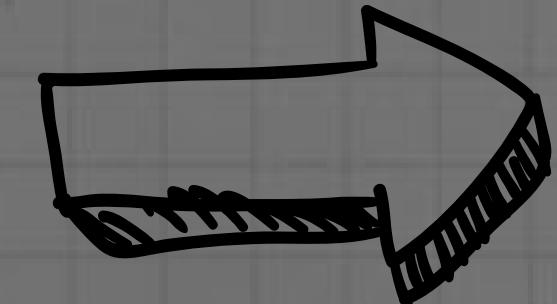
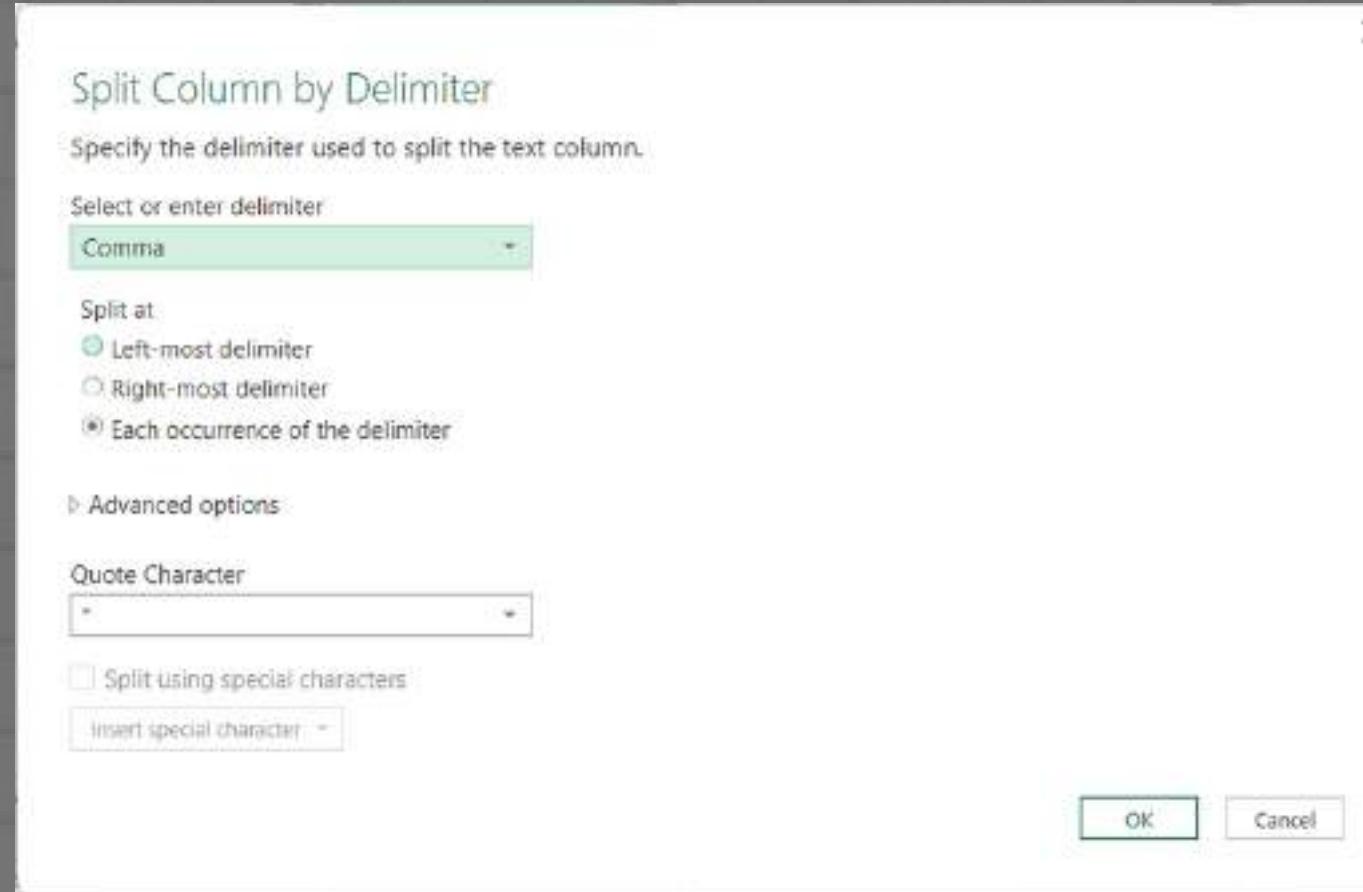
**Result:** The prefix "Writtenby:" was removed from all entries, leaving a clean column with only the authors' names. The data is now properly formatted for accurate sorting and analysis.



**Action:** Used the "Extract" > "Text After Delimiter" function on the author column. I entered a colon (:) as the delimiter to extract everything after "Writtenby:".



# Separate combined names in the author column if there are multiple authors.



A <sub>B</sub> author.1	A <sub>B</sub> author.2	A <sub>B</sub> author.3
Valid 0% Empty 100%	Valid 21% Error 0% Empty 79%	Valid 2% Error 0% Empty 98%
478 distinct, 340 unique	110 distinct, 78 unique	16 distinct, 12 unique
GeronimoStilton	null	null
RickRiordan	null	null
JeffKinney	null	null
RickRiordan	null	null
RickRiordan	null	null
SuzanneCollins	null	null
WinterMorgan	null	null
RickRiordan	null	null
MaryPopeOsborne	null	null
RickRiordan	null	null
RickRiordan	null	null
MaryPopeOsborne	null	null
MaryPopeOsborne	null	null

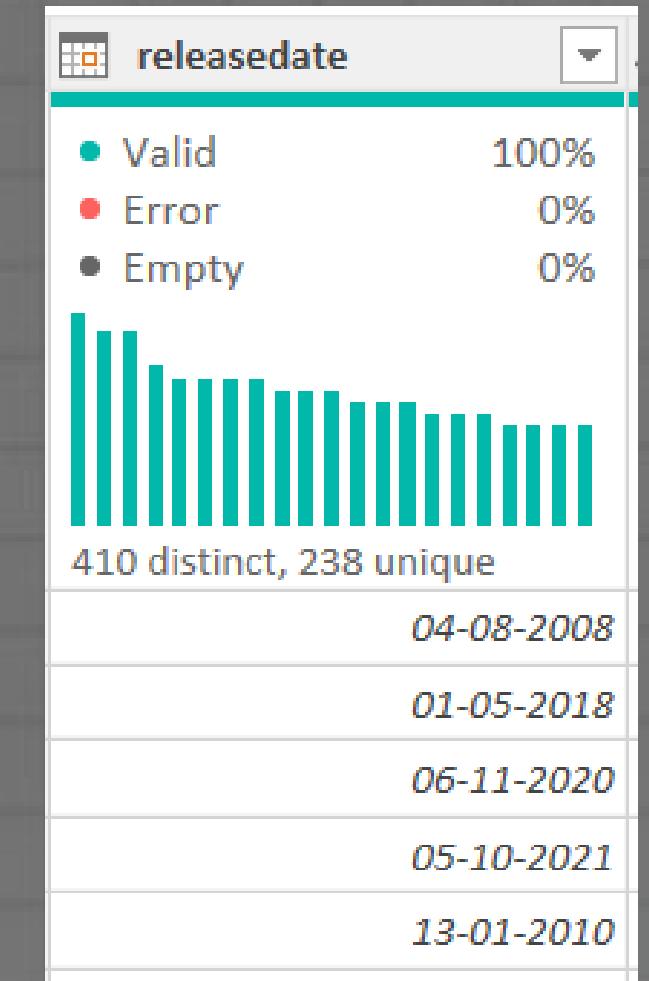
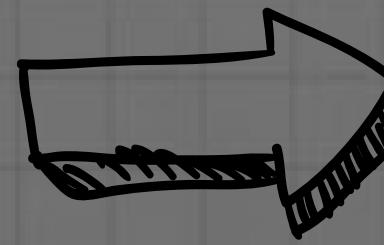
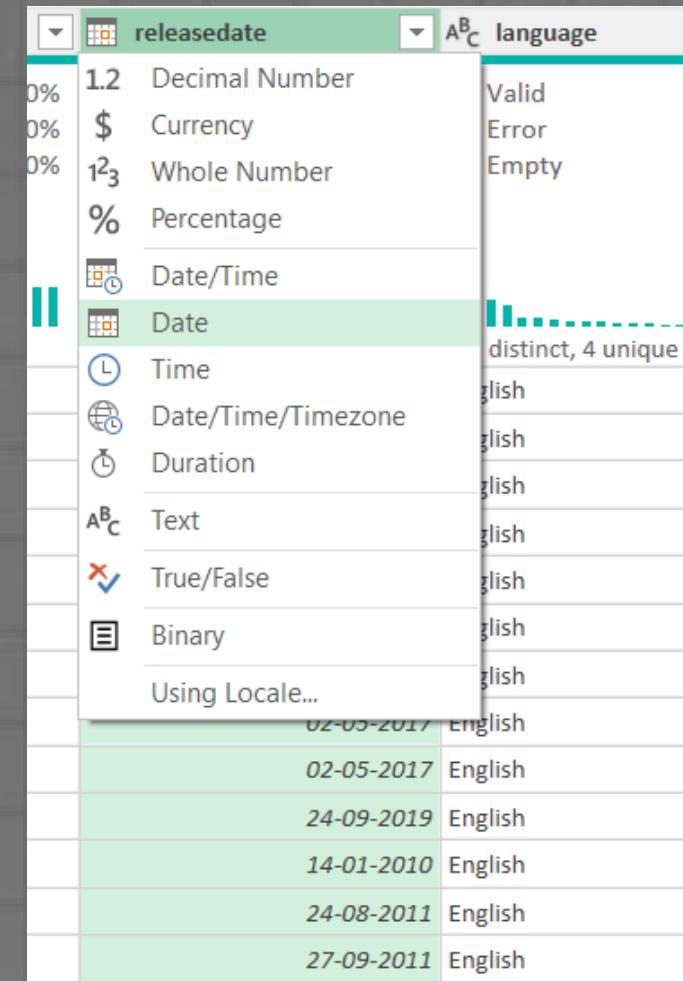
**Problem:** Some books listed multiple authors in a single cell, separated by commas. This "many-to-one" format made it impossible to analyze the data by individual author.

**Action:** I selected the author column and used the "Split Column" > "By Delimiter" tool. As shown in the screenshots, I specified a comma as the delimiter and chose the option to split at each occurrence, which created new columns for each author

**Result:** The data is now more structured, with each co-author separated into their own distinct column (author.1, author.2, author.3). This allows for clear and accurate analysis of every author's contribution.



# Ensure all entries in the releasedate column follow a consistent date format (DD-MM-YYYY).



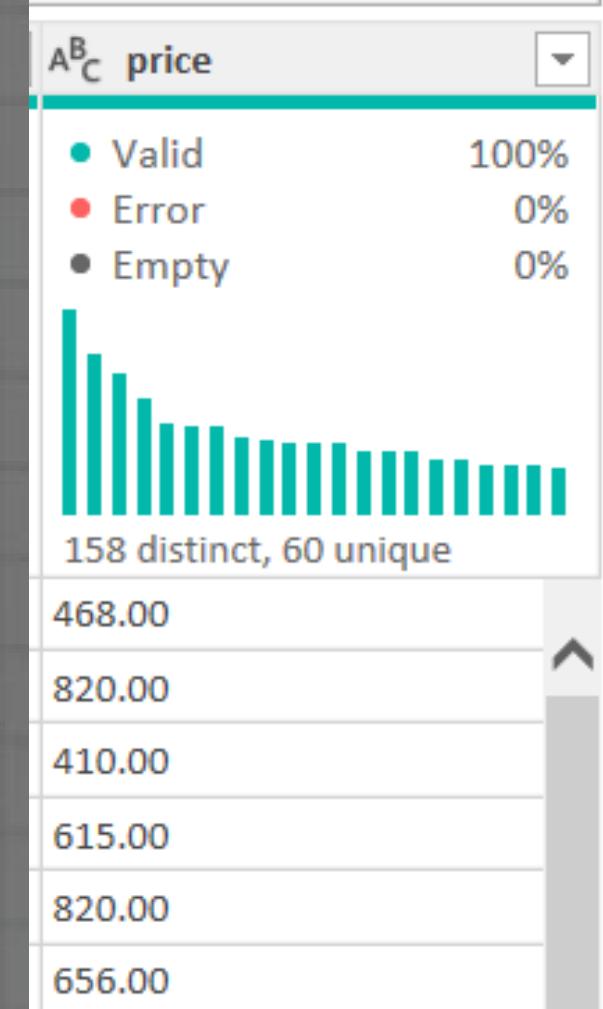
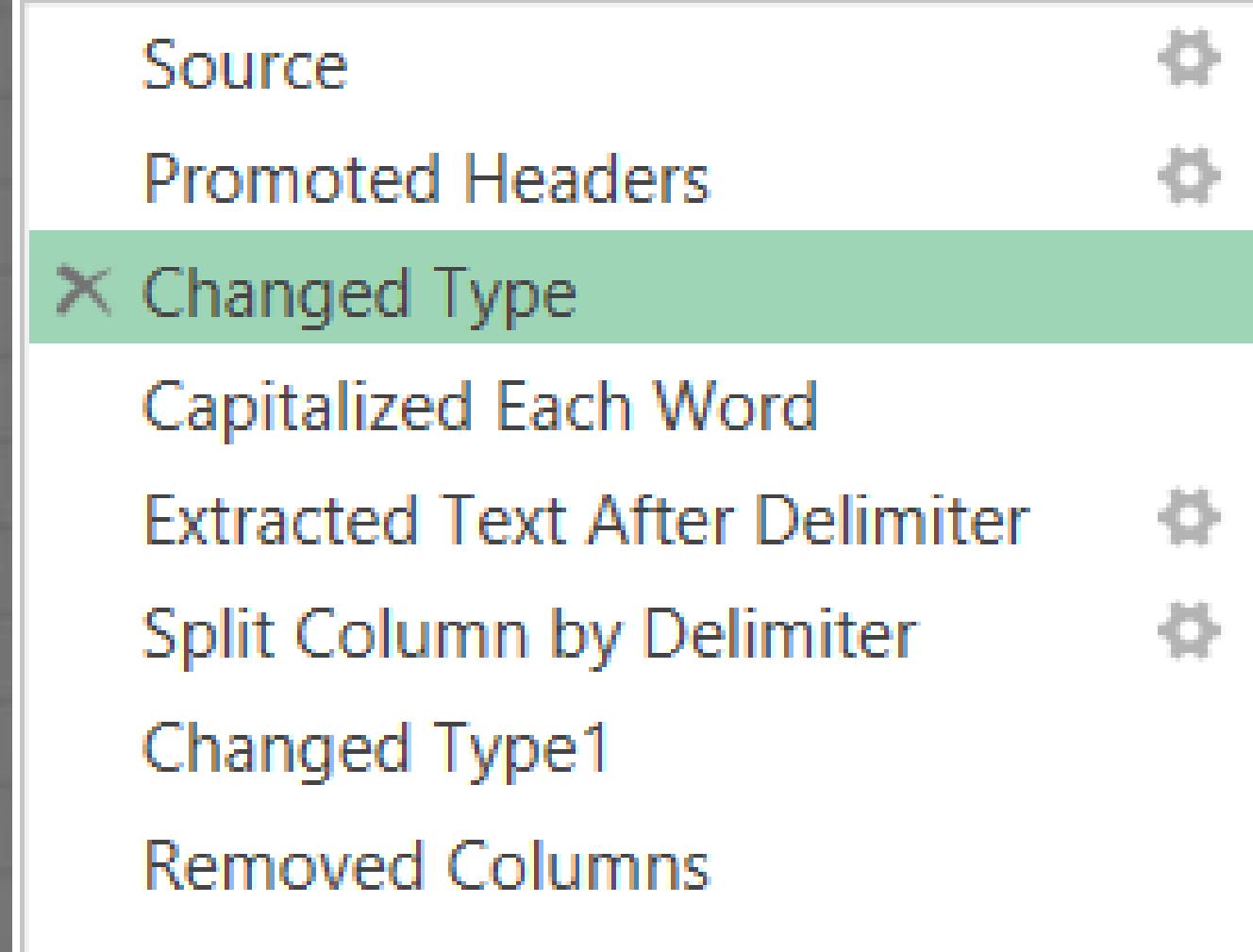
**Problem:** The releasedate column was initially formatted as text. This prevents correct sorting (e.g., sorting alphabetically instead of by date) and makes it impossible to perform date-based calculations or use time-intelligence features.

**Action:** I clicked the data type icon next to the releasedate column header and selected Date from the dropdown menu.

**Result:** The column is now correctly formatted as a date type. This enforces a consistent format (DD-MM-YYYY) across all entries and unlocks the ability to sort chronologically, filter by date ranges, and perform accurate time-based analysis.



# Ensure the price column is in a numeric format, and identify any non-numeric values.



**Problem:** As shown in the images, Power Query's automatic "Changed Type" step failed. This is because the price column contains mixed data types: it has numbers and also the text value "Free", which prevents a successful conversion to a numeric format and creates errors.

**Action:** First, I removed the automatic "Changed Type" step from the Applied Steps pane to fix the initial error.

**Result:** This fixed the conversion errors, making the price column a clean, numeric field ready for any calculations.



# Ensure the price column is in a numeric format, and identify any non-numeric values.



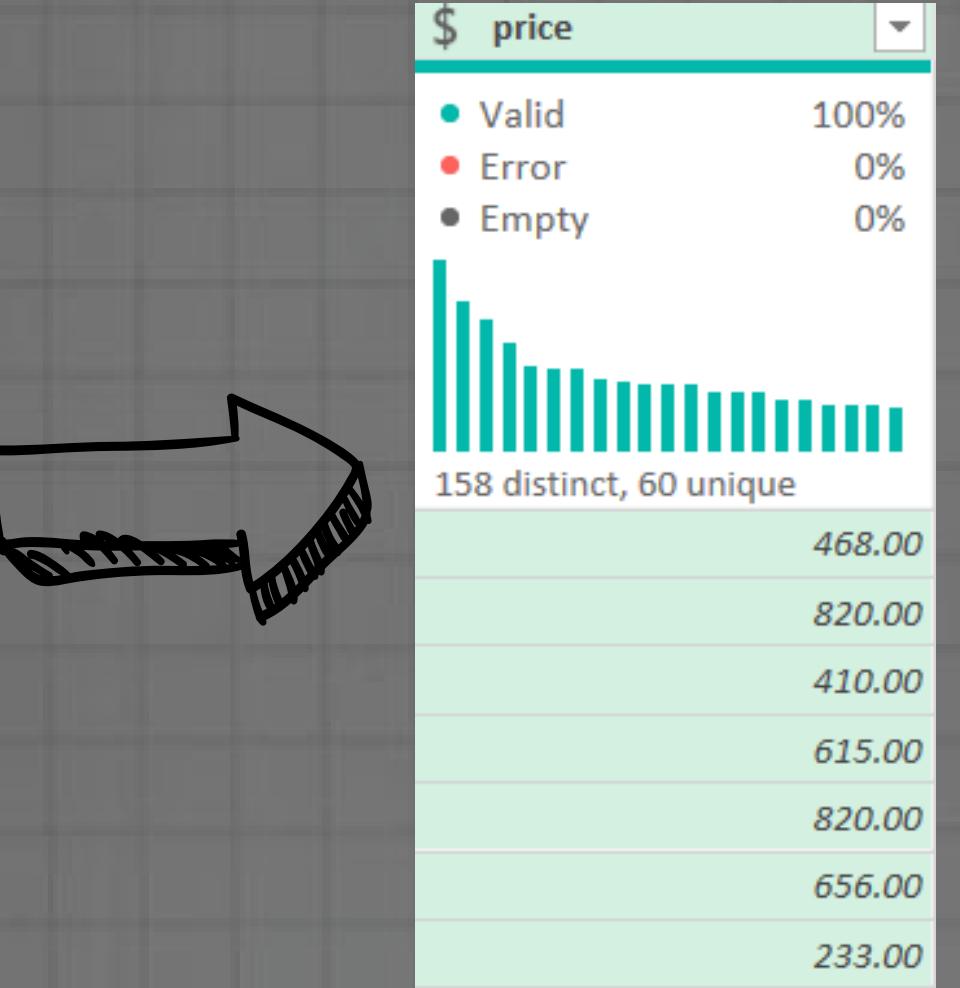
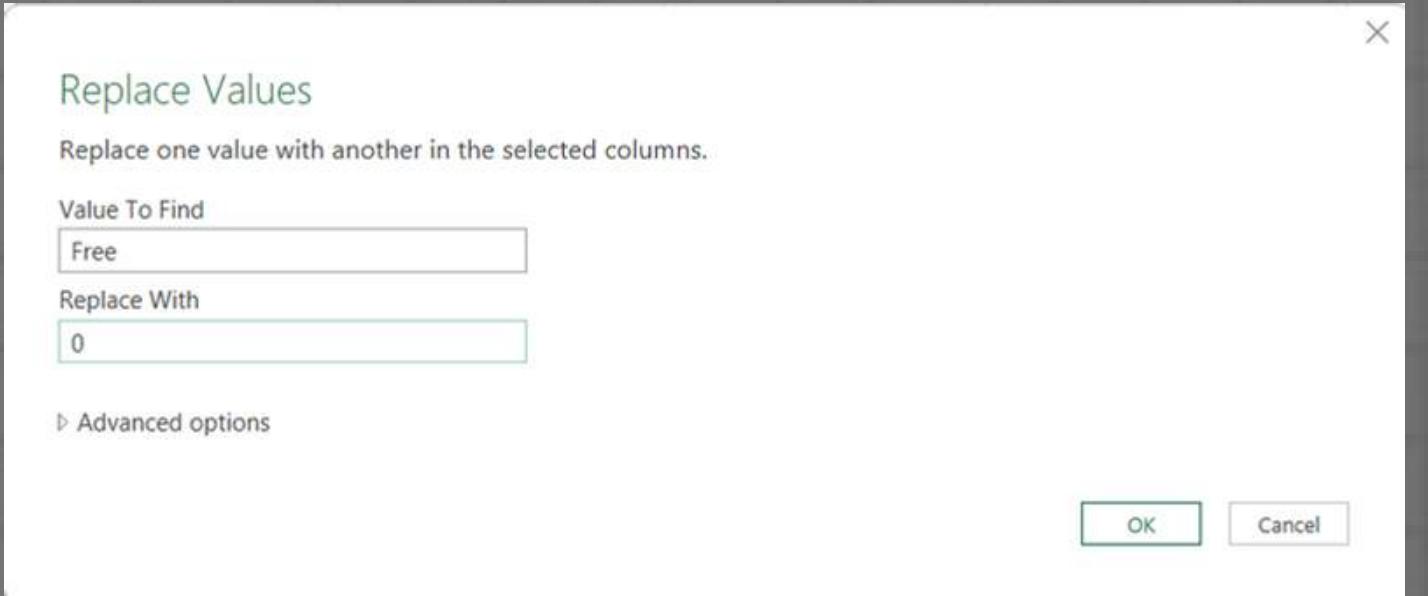
Sort Ascending  
Sort Descending  
Clear Sort  
Clear Filter  
Remove Empty  
Text Filters

Search

- Free
- 821.00
- 836.00
- 854.00
- 866.00
- 873.00
- 888.00
- 903.00
- 904.00
- 93.00
- 930.00
- 933.00
- 949.00
- 957.00
- 959.00
- 968.00
- 99.00
- Free

List may be incomplete. Load more

OK Cancel



**Action:** Next, as shown in your first image, I used "Replace Values" to find every instance of "Free" and replace it with "0". Finally, as shown in your second image, I changed the column's data type to Currency to ensure it was formatted correctly for financial analysis.

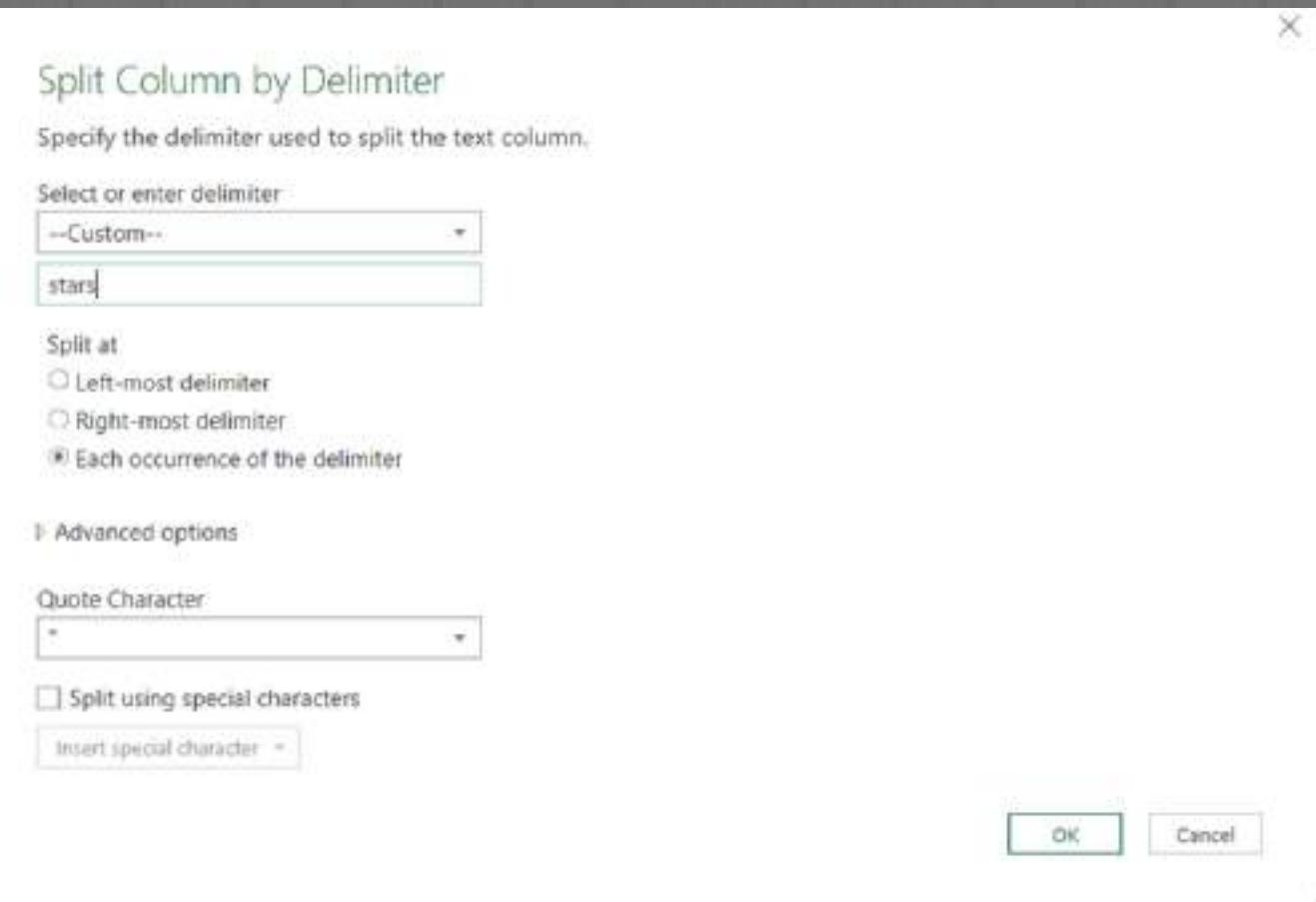
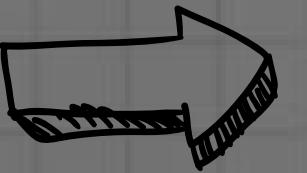
**Result:** The price column is now a clean, error-free numeric column. This allows for accurate mathematical calculations and ensures data consistency for reporting.



# Convert text ratings in the stars column to numeric values.



A	B	C	stars
● Valid			100%
● Error			0%
● Empty			0%
			64 distinct, 37 unique
5 out of 5 stars	34 ratings		
4.5 out of 5 stars	41 ratings		
4.5 out of 5 stars	38 ratings		
4.5 out of 5 stars	12 ratings		
4.5 out of 5 stars	181 ratings		



A	B	C	stars.1	stars.2
● Valid			100%	23%
● Error			0%	0%
● Empty			0%	77%
			6 distinct, 1 unique	38 distinct, 16 unique
5 out of 5			5 out of 5	34 ratings
4.5 out of 5			4.5 out of 5	41 ratings
4.5 out of 5			4.5 out of 5	38 ratings
4.5 out of 5			4.5 out of 5	12 ratings
4.5 out of 5			4.5 out of 5	181 ratings
5 out of 5			5 out of 5	72 ratings
5 out of 5			5 out of 5	11 ratings
5 out of 5			5 out of 5	50 ratings
5 out of 5			5 out of 5	5 ratings
				...

**Problem:** The stars column was a single text field containing two different pieces of information: the average star rating (e.g., "4.5 out of 5") and the total number of ratings (e.g., "130 ratings"). This format is unusable for any numerical analysis or calculations.

**Action:** First, I used "Split Column by Delimiter" with a custom delimiter of "stars" to separate the average rating from the rating count into two new columns (stars.1 and stars.2).

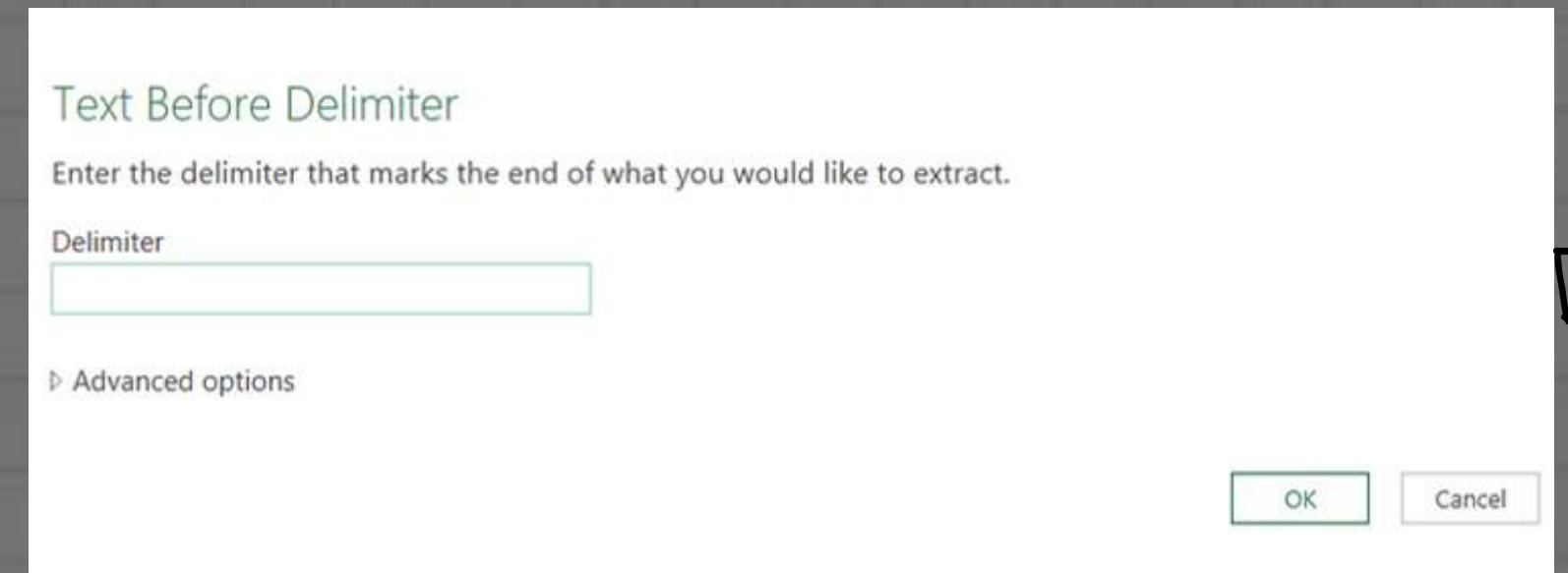
**Result:** I successfully created two clean numeric columns—one for the star rating and another for the rating count—which are now ready for accurate analysis.



# Convert text ratings in the stars column to numeric values.



A <sup>B</sup> <sub>C</sub> stars.1	
Valid	100%
Error	0%
Empty	0%
	
6 distinct, 1 unique	
5 out of 5	
4.5 out of 5	
4.5 out of 5	
4.5 out of 5	
4.5 out of 5	
4.5 out of 5	
5 out of 5	



1.2 stars out of 5	A <sup>B</sup> <sub>C</sub> Ratings
Valid	100%
Error	0%
Empty	0%
	
6 distinct, 1 unique	
5	23%
4.5	0%
4.5	77%
	
38 distinct, 16 unique	
5	34 ratings
4.5	41 ratings
4.5	38 ratings
4.5	12 ratings
4.5	181 ratings
5	72 ratings

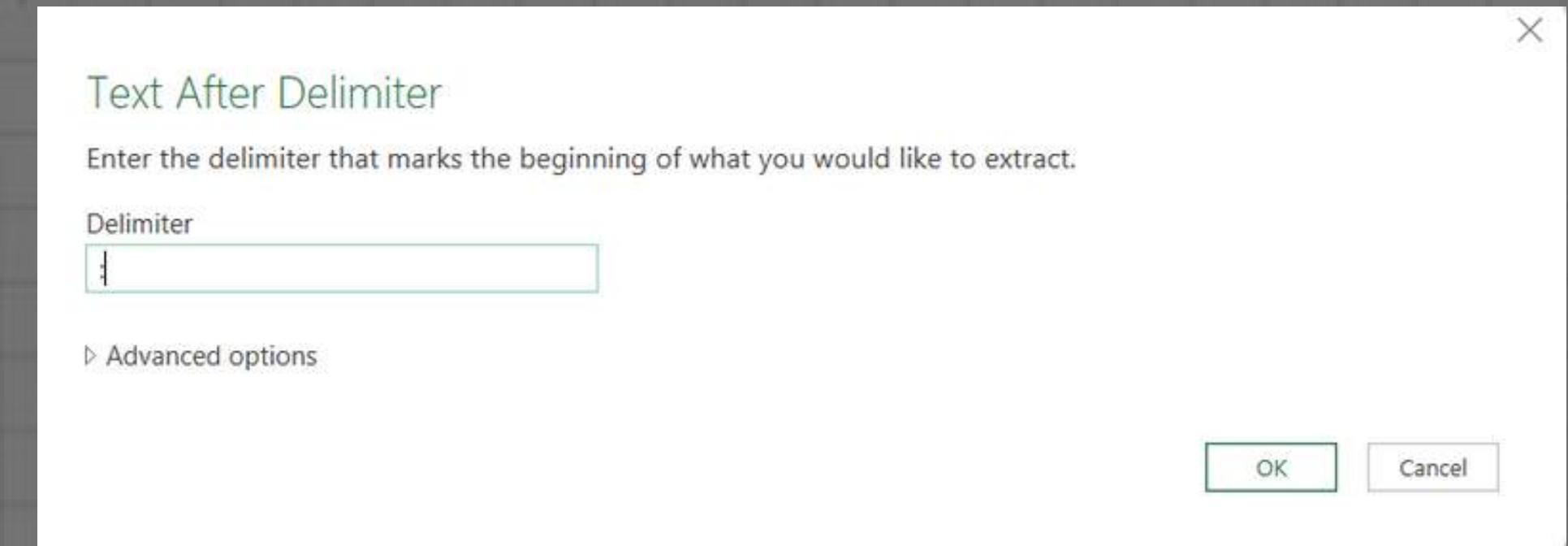
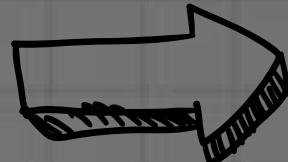
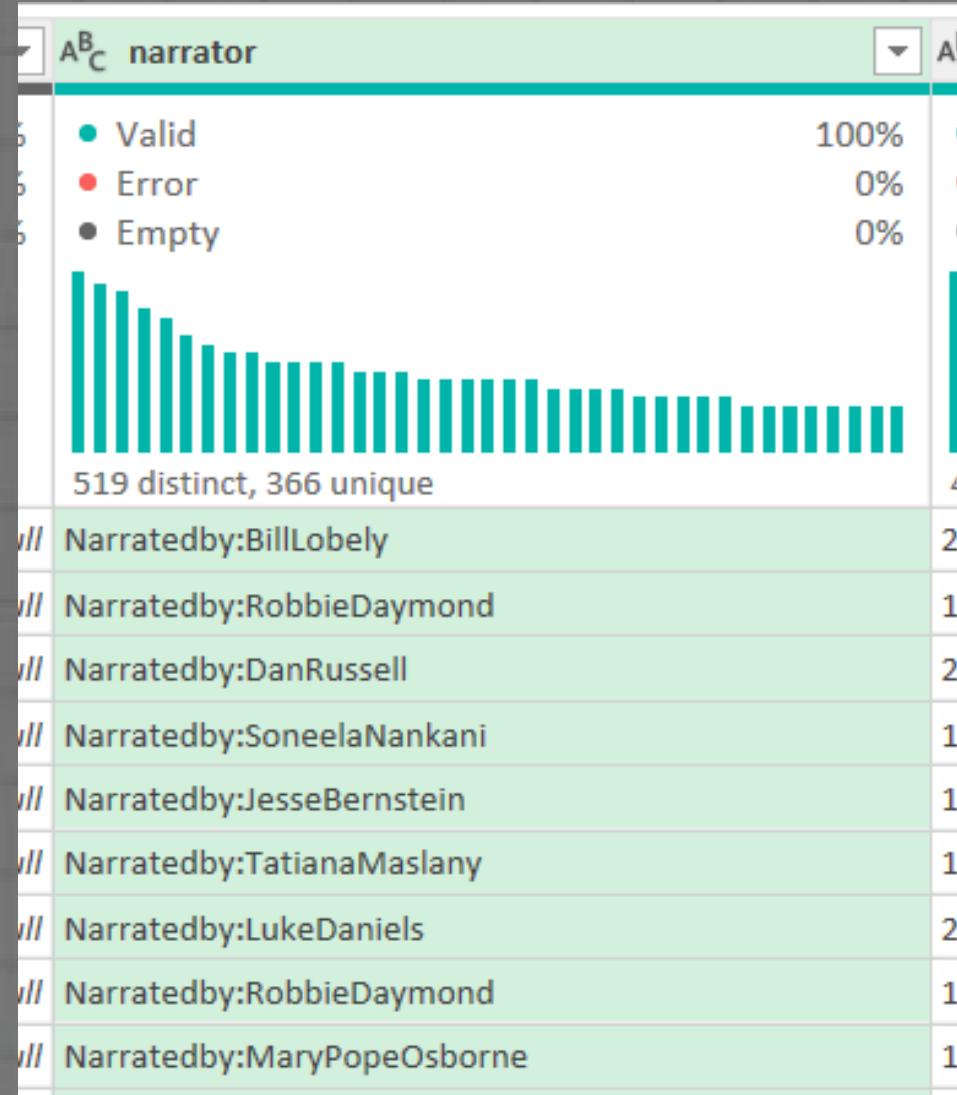
**Problem:** After splitting the stars column, the new stars.1 column still contained extra text (e.g., "4.5 out of 5"). This prevented it from being used as a number for calculations.

**Action:** I selected the stars.1 column and used the "Extract" > "Text Before Delimiter" function. By entering a space (" ") as the delimiter, I isolated the numeric rating at the beginning of the text. I then finished the process by changing the column's data type to Decimal Number.

**Result:** The column now contains only the clean, numeric star rating (e.g., 4.5, 5). This makes the data accurate and ready for mathematical analysis and visualization.



# Split the narratedby column into multiple columns if multiple narrators are listed.

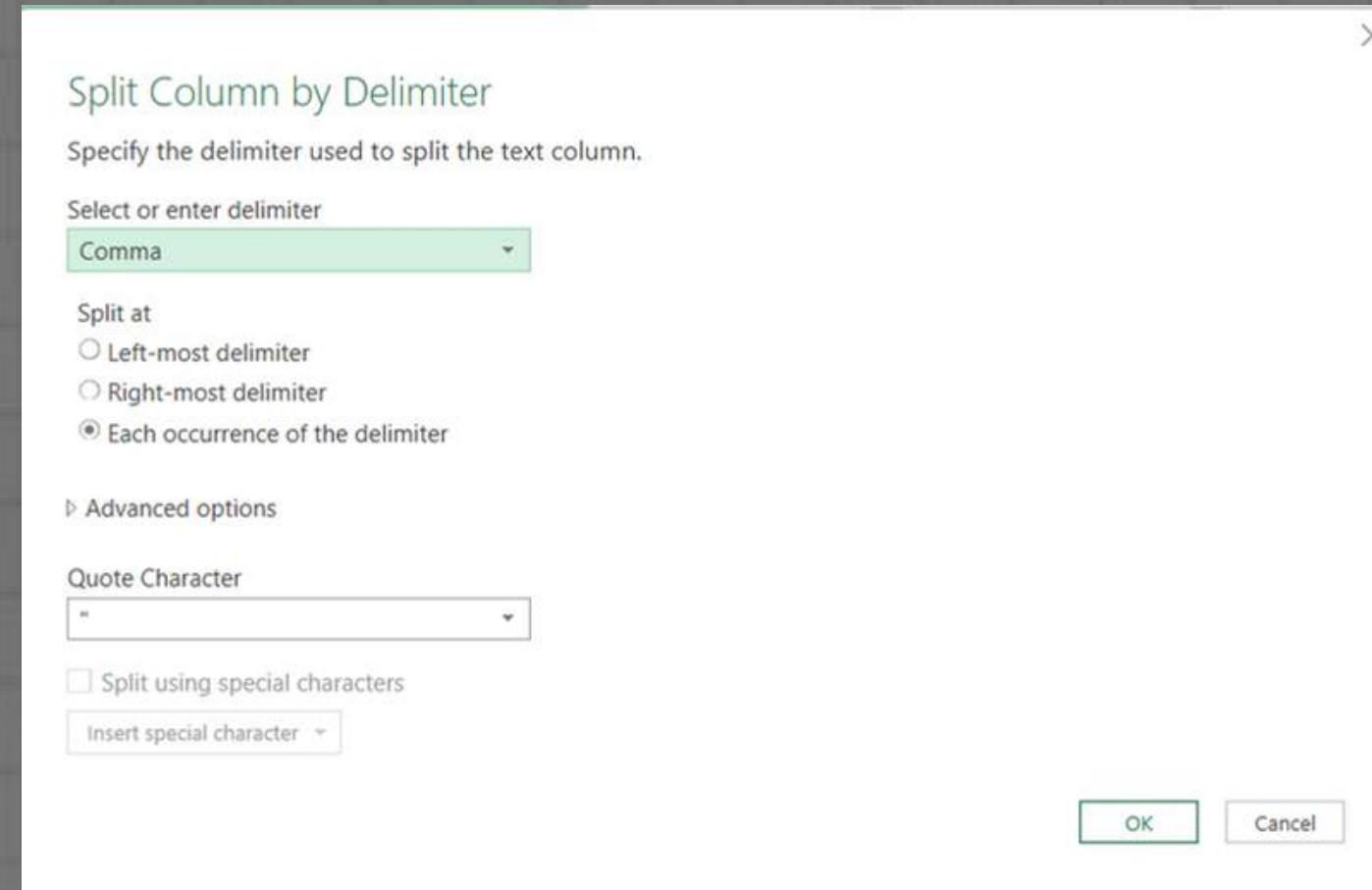


**Problem:** The narrator column data was messy and unusable for analysis. As seen in the first image, each entry had an unnecessary "Narratedby:" prefix, and some cells contained multiple narrators separated by commas.

**Action:** I performed a multi-step process to clean this column. First, I used "Extract" > "Text After Delimiter" with a colon (:) to remove the "Narratedby:" prefix. Next, as shown in your second image



# Split the narratedby column into multiple columns if multiple narrators are listed.



	narrator.1	narrator.2	narrator.3
Valid	100%	7%	3%
Error	0%	0%	0%
Empty	0%	93%	97%
	496 distinct, 336 unique	59 distinct, 49 unique	23 distinct, 18 unique
BillLobley	null	null	null
RobbieDaymond	null	null	null
DanRussell	null	null	null
SoneelaNankani	null	null	null
JesseBernstein	null	null	null
TatianaMaslany	null	null	null
LukeDaniels	null	null	null
RobbieDaymond	null	null	null
MaryPopeOsborne	null	null	null
RobbieDaymond	null	null	null
JesseBernstein	null	null	null
MaryPopeOsborne	null	null	null
MaryPopeOsborne	null	null	null
MichaelCrouch	null	null	null
PhilipPullman	fullcast	RuthWilson	
BillLobley	null	null	null
MaryPopeOsborne	null	null	null
CaitlinKelly	null	null	null

**Problem:** After removing the prefix, the narrator column still contained multiple names in a single cell, separated by commas. This "many-to-one" format made it impossible to properly count or analyze data for each individual narrator.

**Result:** The data is now more structured, with each narrator separated into distinct columns (narrator.1, narrator.2, narrator.3). This allows for clear and accurate analysis of every narrator's contribution..

**Action:** I selected the narrator column and used the "Split Column" > "By Delimiter" tool. I specified a comma as the delimiter and chose the option to split at each occurrence, which created the new columns shown in the screenshot.



Merge the releasedate and language columns into a single new column named releaseinfo with the format "DD-MM-YYYY, Language."



releasedate	A <sup>B</sup> C	language	A <sup>B</sup> C	stars out of 5	A <sup>B</sup> C	Ratings
• Valid ● Error ● Empty	100%	• Valid ● Error ● Empty	100%	• Valid ● Error ● Empty	100%	23% 0% 77%

### Merge Columns

Choose how to merge the selected columns.

Separator

Comma

New column name (optional)

Merged

OK

Cancel



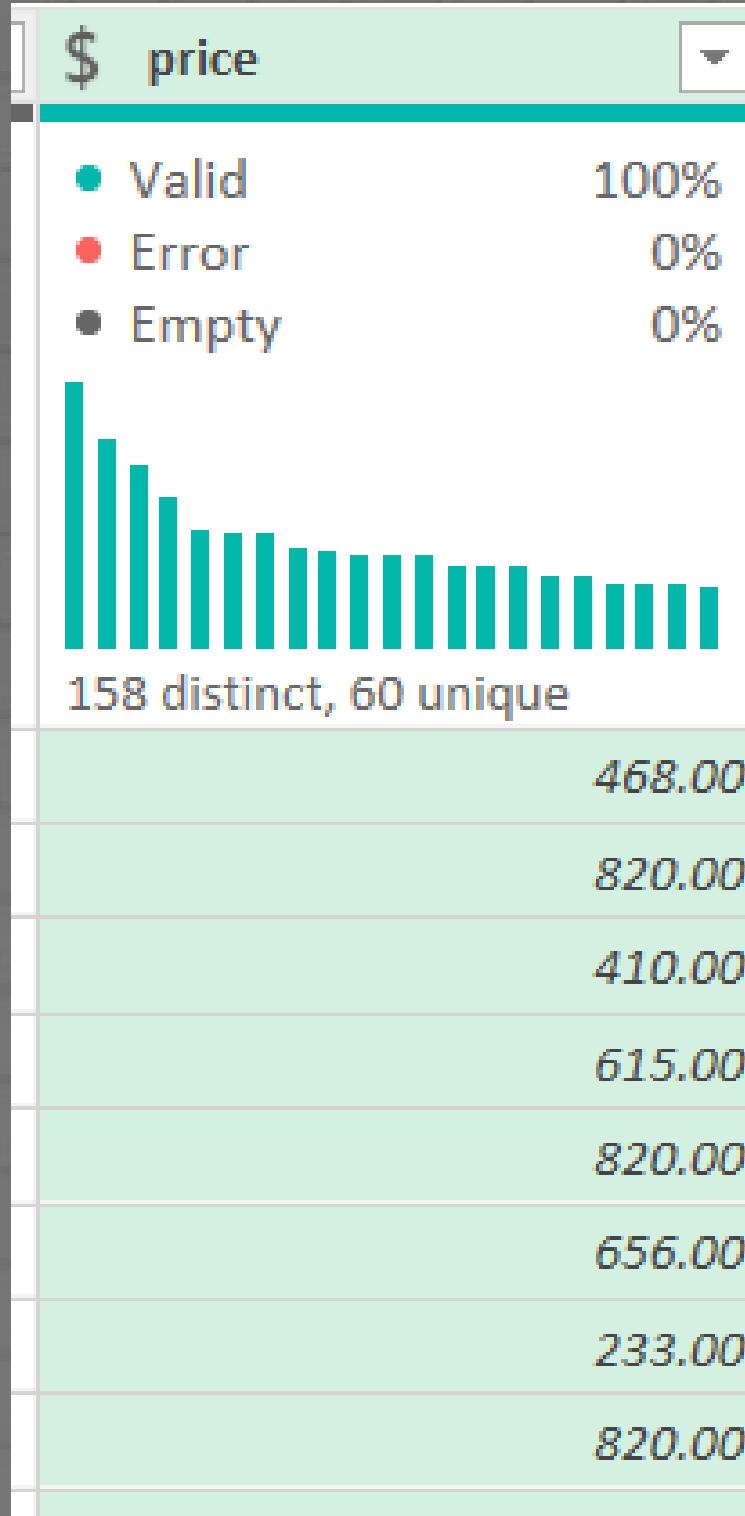
A <sup>B</sup> C	Releaseinfo	A <sup>B</sup> C
• Valid ● Error ● Empty	100% 0% 0%	• Valid ● Error ● Empty
	551 distinct, 355 unique	6
	04-08-2008,English	5
	01-05-2018,English	4.5
	06-11-2020,English	4.5
	05-10-2021,English	4.5
	13-01-2010,English	4.5
	30-10-2018,English	5
	25-11-2014,English	5
	02-05-2017,English	5

**Action:** I selected the releasedate and language columns in order. Then, I used the "Merge Columns" feature from the Transform tab. As shown in your screenshots, I chose a comma as the separator and named the new column "Releaseinfo"

**Result:** A new Releaseinfo column was successfully created, combining the two fields into the desired format (e.g., "04-08-2008, English"). This consolidates the data, making it easier to read and display in reports.



# Ensure all currency values in the price column are formatted consistently with two decimal places.

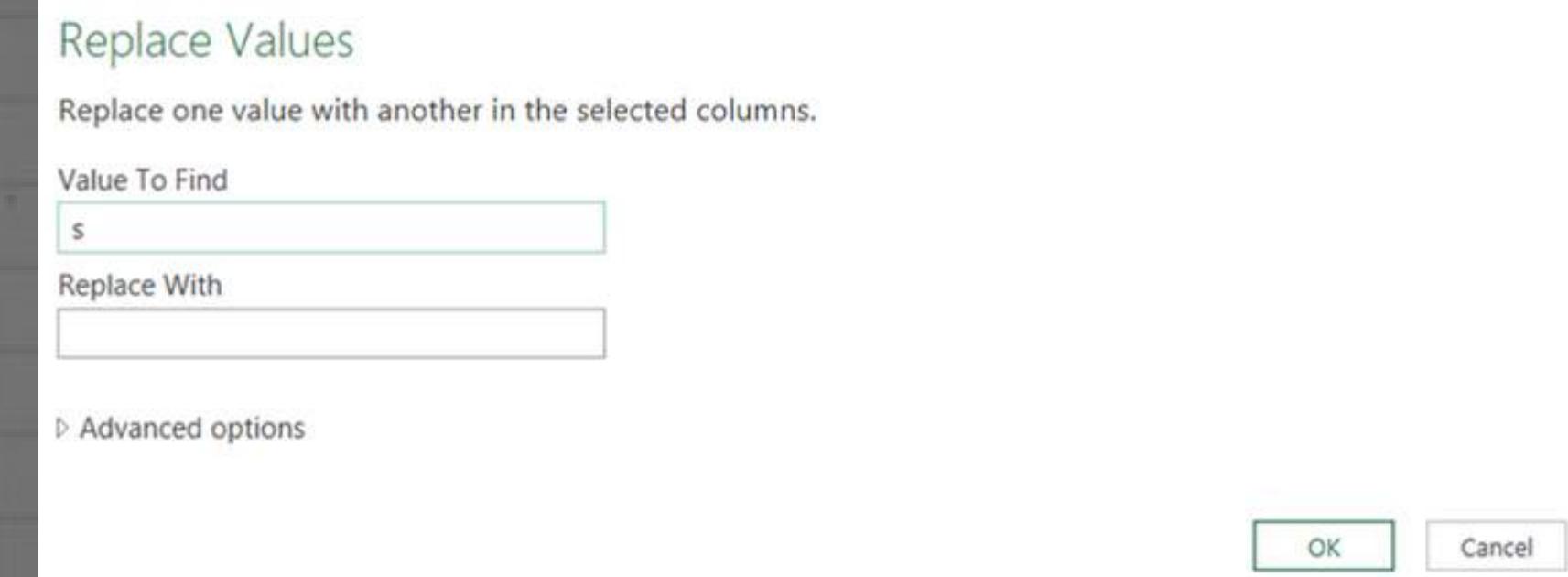
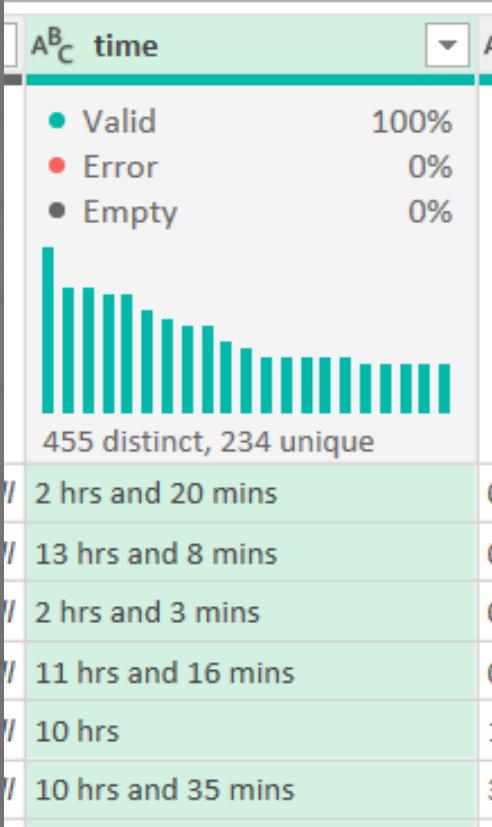


**Action:** I inspected the price column and verified that its data type was correctly set to a numeric format (Decimal Number or Currency).

**Result:** The price column is confirmed to be a clean, numeric field. This validation guarantees that all subsequent mathematical calculations and financial analyses will be accurate and reliable.



# Convert the time column from text format to a duration format that Excel recognizes.



**Problem:** The time column was formatted as text in a human-readable format (e.g., "2 hrs and 20 mins"), which is unusable for calculations. The text was also inconsistent, using both "hr" and "hrs" as well as "min" and "mins".

**Action:** I performed a series of transformations to convert this text into a proper duration. First, as shown in the screenshot, I standardized the text by replacing "s" with nothing to handle plurals. Next, I replaced " hr and " with a colon ":" and any solitary " hr" with ":00" to create a standard H:MM format. I also replaced any values like "Less than 1 minute" with "0:0:0". Finally, I converted the cleaned column's data type to Duration.

**Result:** The text-based time column was successfully converted into a standardized Duration format. This makes the data accurate and allows for mathematical calculations, such as finding the average or total listening time across all audiobooks.



# Convert the time column from text format to a duration format that Excel recognizes.



Custom Column

Add a column that is computed from the other columns.

New column name: Custom

Custom column formula:

```
= if not Text.Contains([time], ":") then "0:" & [time] else [time]
```

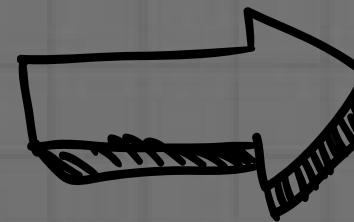
Available columns

- name
- author.1
- author.2
- author.3
- narrator.1
- narrator.2
- narrator.3

Learn about Power Query formulas

✓ No syntax errors have been detected.

OK Cancel



	ABC	Custom
	123	
Valid	100%	100%
Error	0%	0%
Empty	0%	0%
2:20min		
13:8min		
2:3min		
11:16min		
10:00		
10:35min		

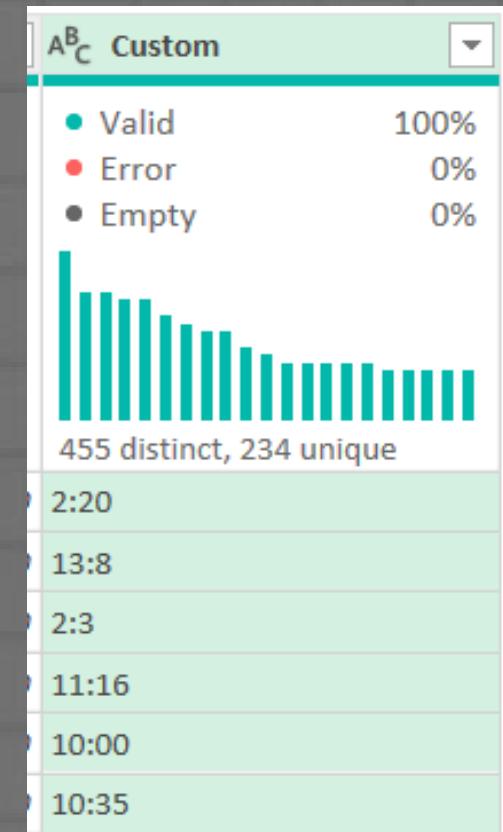
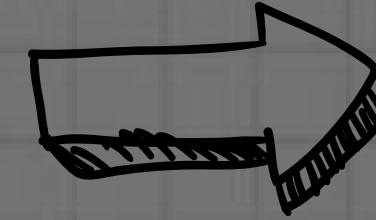
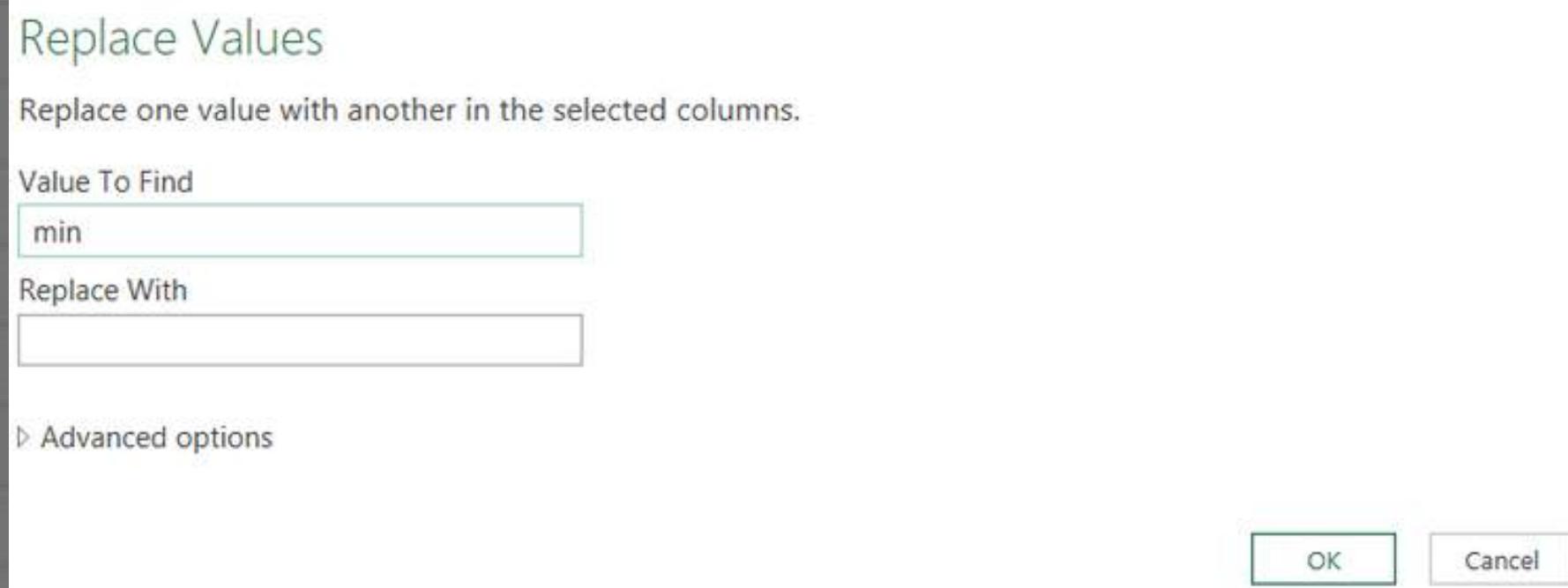
**Problem:** After initial cleaning, some entries in the time column that represented only minutes (e.g., "20min") lacked the "0:" prefix for hours, which is necessary for a consistent duration format (like "0:20min"). This inconsistency would lead to errors or incorrect interpretation when converting to a proper duration data type.

**Result:** The new custom column (which will replace the original time column) now has a consistent "H:MM" format. All minute-only entries are correctly prefixed with "0:", ensuring uniformity and preparing the data for a smooth conversion to a proper Duration data type.

**Action:** I created a Custom Column using an if-then-else statement. As shown in the image, the formula checks if the time value does NOT contain a colon (:). If it doesn't, it means the value is likely in minutes only, so it prepends "0:" to the existing time value. Otherwise, it keeps the time value as is..



# Convert the time column from text format to a duration format that Excel recognizes.



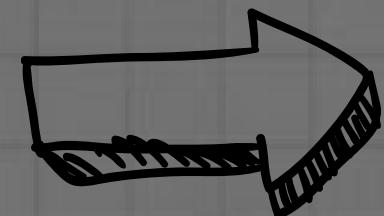
**Problem:** After creating the custom column to standardize the hours and minutes, the text still contained the "min" suffix (e.g., "2:20min"). This non-numeric text prevented the final, crucial step of converting the column to a proper duration data type.

**Result:** The "min" suffix was successfully removed from all entries in the column, leaving a clean, text-free format of hours and minutes separated by a colon (e.g., "2:20"). The column is now perfectly prepared for its final conversion to a Duration data type.

**Action:** I selected the Custom column and used the "Replace Values" feature. As shown in your screenshot, I entered "min" in the "Value To Find" field and left the "Replace With" field empty to effectively delete the text.



# Convert the time column from text format to a duration format that Excel recognizes.



A	B	C	Custom
1.2	Decimal Number		
\$	Currency		
123	Whole Number		
%	Percentage		
⌚⌚	Date/Time		
⌚📅	Date		
⌚🕒	Time		
⌚🌐	Date/Time/Timezone		
⌚🕒	Duration		
A	B	C	Text
✗✓	True/False		
☒	Binary		
			Using Locale...

A	B	C	time - Copy	D	E	Custom
●	Valid	100%	455 distinct, 234 unique	0.02:20:00	0.02:20:00	0.02:20:00
●	Error	0%	455 distinct, 234 unique	0.13:08:00	0.13:08:00	0.13:08:00
●	Empty	0%	455 distinct, 234 unique	0.02:03:00	0.02:03:00	0.02:03:00
			2 hrs and 20 mins	0.11:16:00	0.11:16:00	0.11:16:00
			13 hrs and 8 mins	0.10:00:00	0.10:00:00	0.10:00:00
			2 hrs and 3 mins	0.10:35:00	0.10:35:00	0.10:35:00
			11 hrs and 16 mins	0.02:23:00	0.02:23:00	0.02:23:00
			10 hrs	0.12:32:00	0.12:32:00	0.12:32:00
			10 hrs and 35 mins	0.10:56:00	0.10:56:00	0.10:56:00
			2 hrs and 23 mins	0.13:22:00	0.13:22:00	0.13:22:00
			12 hrs and 32 mins			
			10 hrs and 56 mins			
			13 hrs and 22 mins			

**Problem:** After all the text-based cleaning, the Custom column contained a clean time format (e.g., "2:20"), but it was still stored as a Text data type. In this format, it's impossible to perform any mathematical calculations like finding the average or total duration.

**Result:** As the "before and after" screenshot clearly shows, the text was successfully converted into a true Duration format (e.g., "0.02:20:00"). The column is now correctly formatted and can be used for accurate time-based calculations and analysis.

**Action:** I performed the final conversion step by clicking the data type icon in the Custom column header and selecting Duration from the dropdown menu, as shown in the image.



# Thank you!

*Chandra Prakash Choudhary*

 [LinkedIn.com](https://www.linkedin.com/in/chandra-prakash-choudhary/)

