

## Объединение несовпадающих таблиц в Power Query

Это фрагмент книги [Гил Равив. Power Query в Excel и Power BI: сбор, объединение и преобразование данных](#).

[Предыдущий раздел](#)

[К содержанию](#)

[Следующий раздел](#)

Несовпадающие таблицы в контексте этой главы — это таблицы, которые описывают одни и те же семантические сущности и факты, но имеют различающиеся названия столбцов. Например:

Столбцы	Таблица1	Таблица2	Причина несоответствия
Столбец1	Employee	Name	Различные имена столбцов
Столбец2	age	Age	Различные имена столбцов из-за чувствительности к регистру
Столбец3	Gender	F/M	Различные имена столбцов

Рис. 1. Пример несоответствия имен столбцов

В большинстве случаев Power Query объединяет таблицы в новую таблицу, которая включает расширенный набор всех столбцов из исходных таблиц. Каждая строка из первой таблицы копируется в объединенную таблицу, причем пустые значения отображаются в столбцах, находящихся только во второй таблице. Каждая строка из второй таблицы будет скопирована таким же образом с пустыми значениями в столбцах, которые являются исключительными для первой таблицы. Это называется разделением данных. Вместо объединения значений из двух таблиц в один столбец Power Query сохраняет исходные столбцы с унаследованными несвязанными данными.

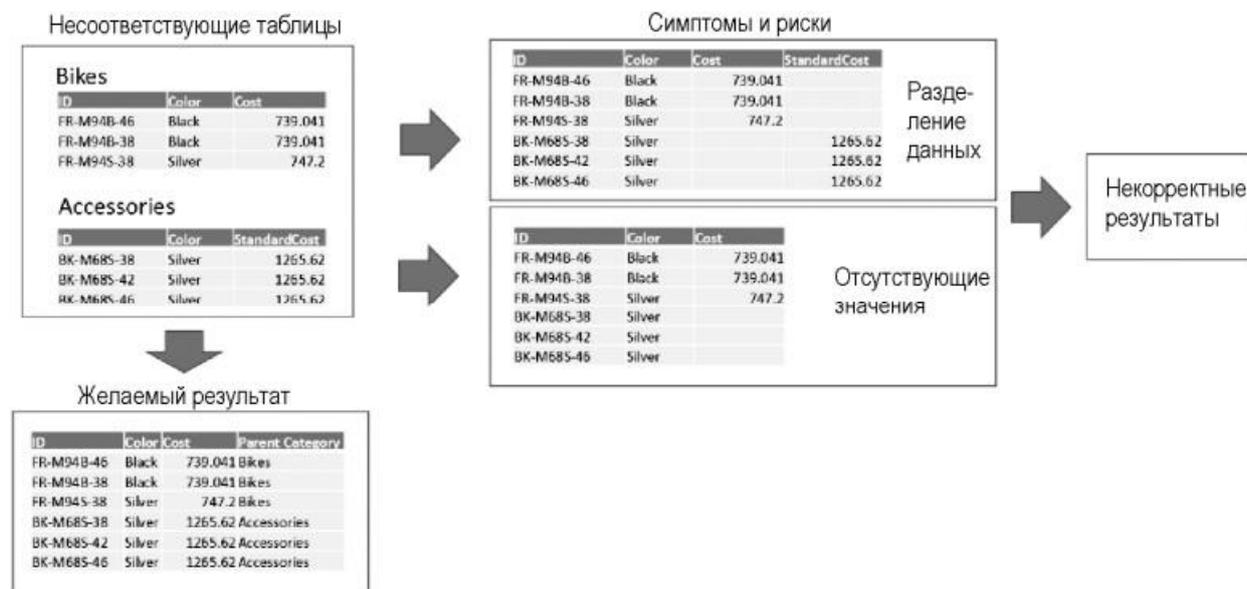


Рис. 2. Несоответствующие таблицы

### Устранение несоответствующих названий столбцов

Загрузить файлы: C04E01 - Accessories.xlsx и C04E01 - Bikes.xlsx. Стоимость товара указана в столбце Cost для Bikes и в столбце StandardCost в таблице Accessories. Откройте новую рабочую книгу в Excel и импортируйте две исходные книги в режиме *Только создать подключение*. В редакторе PQ выберите запрос *Accessories* и выполните команду *Добавить запросы* → *Добавить запросы в новый*. Обратите внимание, что в новом запросе *Добавить1* столбцы *Cost* и *StandardCost* включены в объединенные результаты:

Запросы [3] = Table.Combine({Accessories, Bikes})

	A <sup>B</sup> ID	A <sup>B</sup> Color	1.2 StandardCost	1.2 Cost
24	LT-T990		null	5,7709
25	LT-H902		null	14,4334
26	LT-H903		null	18,5584
27	HL-U509-B	Blue	13,0863	null
28	HL-U509-R	Red	13,0863	null
29	HL-U509	Black	13,0863	null
30	BK-T44U-60	Blue	null	755,1508
31	BK-T79Y-46	Yellow	null	1481,9379
32	BK-T79Y-50	Yellow	null	1481,9379
33	BK-T79Y-54	Yellow	null	1481,9379
34	BK-T79Y-60	Yellow	null	1481,9379
35	BK-T181J-54	Blue	null	461 444R

Рис. 3. Объединенный запрос

Выберите запрос *Accessories* и переименуйте столбец *StandardCost* на *Cost*. Теперь, при повторном выборе запроса *Append1* видно, что столбцы объединены корректно.

### Объединение несоответствующих таблиц из одной папки

Допустим четверо менеджеров ведут запасы товаров. При попытке объединить их данные обнаруживается, что они используют разные имена столбцов:

C04E02 - Accessories.xlsx	Product	ID	Color	StandardCost	ListPrice
C04E02 - Bikes.xlsx	Name	Product_Number	Color	Standard_Cost	List_Price
C04E02 - Clothing.xlsx	Product_name	Prod_num	Pro_color	Prod_cost	Prod_price
C04E02 - Components.xlsx	Name	Product Number	Color	Standard Cost	List Price

Рис. 4. Имена столбцов в четырех категориях продуктов

Загрузите файлы *C04E02 - Accessories.xlsx*, *C04E02 - Bikes.xlsx*, *C04E02 - Components.xlsx*, *C04E02 - Clothing.xlsx* в отдельную папку. В Excel откройте новую рабочую книгу, перейдите *Данные* → *Получить данные* → *Из файла* → *Из папки*. В окне *Обзор* выберите папку, в которой сохранили эти 4 файла (я назвал ее *Товары*) и нажмите *Открыть*, хотя ни один файл не выбран. В следующем окне файлы из папки будут представлены четырьмя строками. Пройдите *Объединить* → *Объединить и преобразовать данные*. В окне *Объединить файлы* раскрывающееся меню *Пример файла* позволяет выбрать один из исходных файлов и применить его формат и заголовки ко всем файлам в папке:

### Объединить файлы

Выберите объект, который следует извлечь из каждого файла. *Дополнительные сведения*

Пример файла: Первый файл

Отобразить пар

C04E02 - Accessories.xlsx C:\Users\Сепрей\Dropbox\Сайт\6\_Эффективность\R...  
C04E02 - Bikes.xlsx C:\Users\Сепрей\Dropbox\Сайт\6\_Эффективность\Ра...  
C04E02 - Clothing.xlsx C:\Users\Сепрей\Dropbox\Сайт\6\_Эффективность\Ра...  
C04E02 - Components.xlsx C:\Users\Сепрей\Dropbox\Сайт\6\_Эффективность\...

	StandardCost	List
Mountain Bottle Cage	BC-M005	null
Road Bottle Cage	BC-R205	null
Patch Kit/8 Patches	PK-7098	0,8565
Hitch Rack - 4-Bike	RA-H123	44,88

Sheet1

Рис. 5. Выбор файла для заголовков

Оставьте выбор по умолчанию, выделите *Sheet1* и кликните *Ok*.

В окне редактора PQ прокрутите вниз, пока строки не изменятся от *Accessories* до *Bikes*. Ряд столбцов содержат нулевые значения. В редакторе PQ перейдите на панель *Запросы* и выберите

запрос *Преобразовать пример файла*. Он играет роль примера запроса. Этот запрос сгенерирован в результате объединения файлов из папки. После выбора примера запроса можно увидеть на главной панели редактора PQ, что данные из таблицы Accessories отображаются правильно.

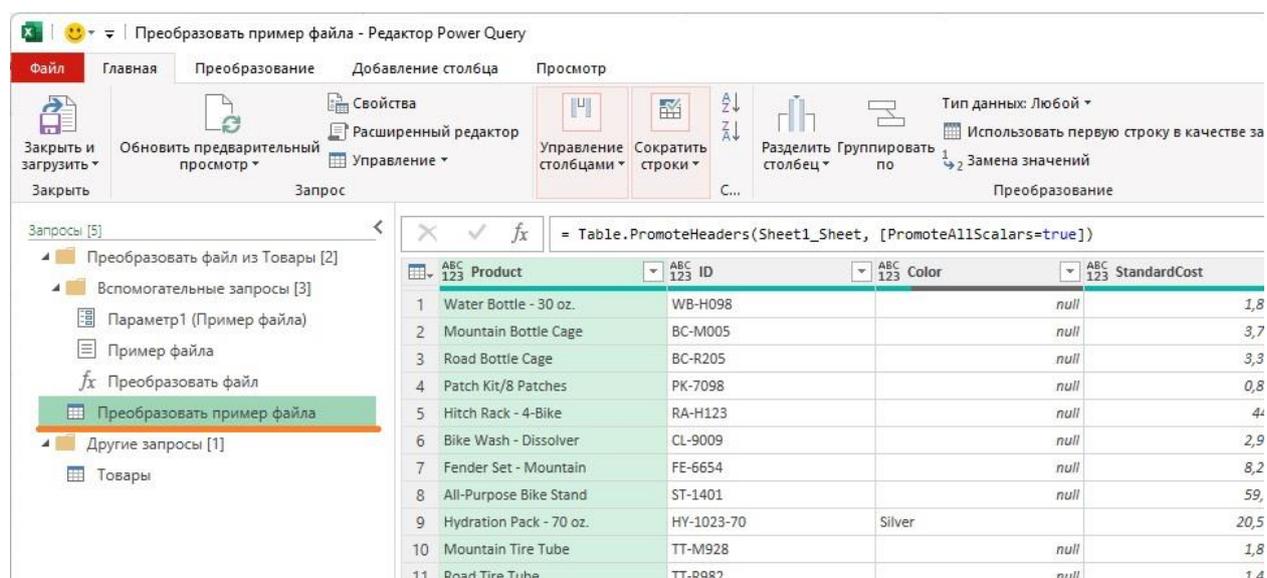


Рис. 6. Запрос *Преобразовать пример файла*

При объединении файлов из папки Power Query создает:

- функцию *Преобразовать файл*,
- пример файла,
- Параметр 1 (Пример файла) и
- запрос *Преобразовать пример файла*
- запрос *Товары*; этот запрос будет назван по имени папки, из которой вы ведете импорт.

Эти элементы помогают объединять несколько файлов из папки на основе выбранных данных. Основное преобразование для каждого файла реализовано в функции. Для того чтобы настроить преобразование для каждого файла, можно вносить изменения в пример запроса. Эти изменения распространяются на функцию и применяются ко всем файлам. Всякий раз при объединении файлов из папки найдите запрос, который начинается с *Преобразовать пример файла*. Этот запрос рассматривает как изменение каждого файла, так и его добавление.

#### *Предположение об одинаковом порядке столбцов*

В редакторе PQ выберите запрос *Преобразовать пример файла* и переименуйте его в *Products Sample*. На панели *Примененные шаги* удалите последний шаг *Повышенные заголовки*. После этого объединенная таблица будет иметь общие столбцы с именами Column1, Column2, Column3 и т.д. Если все таблицы имеют одинаковую последовательность столбцов, то данные в добавленной таблице будут корректными.

Выберите запрос *Товары* и просмотрите объединенную таблицу. Обратите внимание, что при переходе от Accessories к Bikes данные отражаются корректно. Правда появилась ошибка *Столбец "Product" таблицы не найден*. На панели *Примененные шаги* удалите последний шаг *Измененный тип*. Теперь в объединенной таблице отсутствуют пропущенные значения. Можно убедиться в этом, выполнив прокрутку снова и удостоверившись, что включены все значения в таблице Accessories.

Теперь можно повысить первую строку до уровня заголовка, выбрав команду *Использовать первую строку в качестве заголовков*. Осталось избавиться от промежуточных заголовков. На панели *Запросы* выберите запрос *Products Sample*. Пройдите *Добавление столбца* → *Столбец индекса*. Имена столбцов каждой таблицы имеют индекс ноль. Вернитесь к запросу *Товары* и отфильтруйте строки с нулевым индексом. Для этого кликните мышью на элементе управления фильтром в последнем столбце, который помечен символом 0. Снимите галочку напротив значения 0, нажмите *Ок*.

Удалите первый столбец (столбец "C04E02 - Accessories.xlsx") и последний столбец (столбец 0). Можно также повысить надежность скрипта, изменив удаление столбца с именем "C04E02 - Accessories.xlsx" на удаление первого столбца. Измените код:

```
= Table.RemoveColumns("#Строки с примененным фильтром", {"C04E02 - Accessories.xlsx", "0"})
```

...на:

```
= Table.RemoveColumns("#Строки с примененным фильтром", {Table.ColumnNames("#Строки с примененным фильтром"){0}, "0"})
```

Теперь можно загрузить объединенную таблицу на лист Excel и приступить к анализу.

### Простая нормализация

Если нельзя полагаться на согласованный порядок следования столбцов, можно попробовать применить нормализацию имен столбцов. Нормализация – это изменение текста заголовков, устраняющее незначительные различия. Например, можно заменить все символы подчеркивания пробелами и использовать только заглавные буквы.

Загрузите файлы C04E03 - Accessories.xlsx, C04E03 - Bikes.xlsx, C04E03 - Components.xlsx, C04E03 - Clothing.xlsx. Откройте новую книгу в Excel и пройдите *Данные* → *Получить данные* → *Из файла* → *Из папки*. В окне *Обзор* выберите папку, в которой сохранили эти 4 файла и нажмите *Открыть*, хотя ни один файл не выбран. В следующем окне файлы из папки будут представлены четырьмя строками. Пройдите *Объединить* → *Объединить и преобразовать данные*. В окне *Объединить файлы* выделите *Sheet1* и кликните *Ok*.

Выберите запрос *Преобразовать пример файла* и переименуйте его в *Products Sample*. При выбранном запросе *Products Sample* щелкните на значке *f<sub>x</sub>* в строке формул. На панели *Примененные шаги* создается новый шаг – *Пользовательская1*, который имеет формулу:

```
= #"Повышенные заголовки"
```

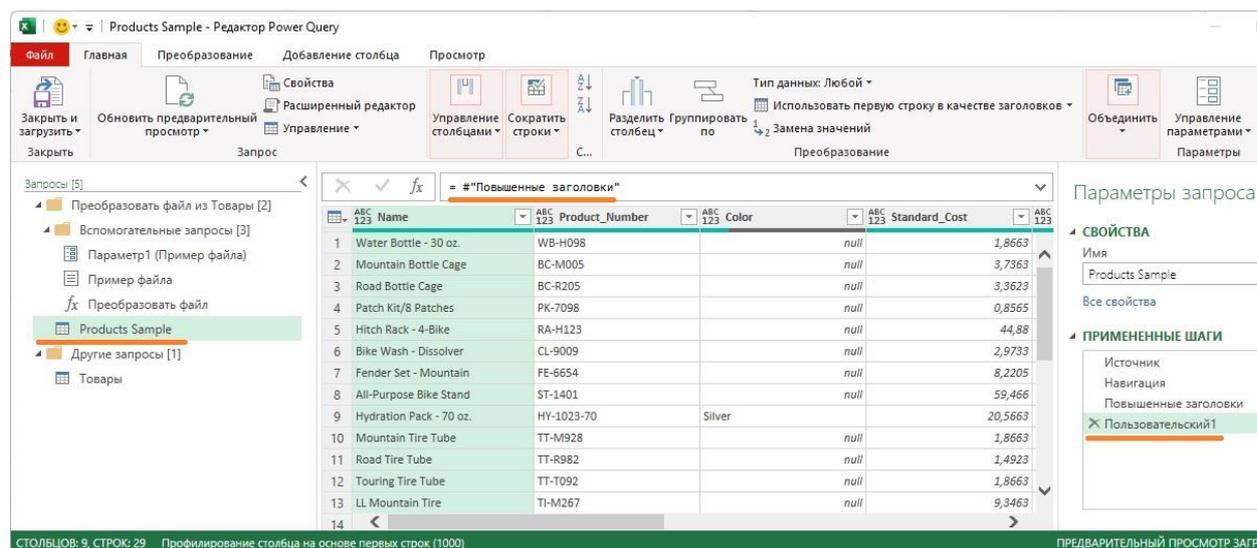


Рис. 7. Код M, соответствующий щелчку на значке *f<sub>x</sub>*

Шаг *Повышенные заголовки* был последним шагом на панели *Примененные шаги* перед тем, как выполнен щелчок на кнопке *f<sub>x</sub>*.

*#"Повышенные заголовки"* – переменная, используемая для вывода этого шага. Поскольку данная переменная возвращает таблицу с несовпадающими именами столбцов, к ней можно применить функцию *Table.TransformColumnNames* с помощью метода *Text.Lower*, что позволит задать строчные буквы в именах столбцов. Для этого измените формулу в строке на:

```
= Table.TransformColumnNames("#Повышенные заголовки", Text.Lower)
```

Нажмите Enter и обратите внимание, что все имена столбцов переведены в нижний регистр. Если предпочтительнее заглавные буквы в именах столбцов, то можно в предыдущей формуле заменить функцию Text.Lower на Text.Proper.

Для замены символа подчеркивания в именах столбцов пробелами снова щелкните на значке  $f_x$  в строке формул. На панели *Примененные шаги* появился новый шаг, *Пользовательская2*, а в строке формул:

= Пользовательский1

Примените функцию Table.TransformColumnNames для замены подчеркивания пробелами в именах столбцов, изменив формулу следующим образом:

= Table.TransformColumnNames(Пользовательский1,  
each Replacer.ReplaceText(\_, "\_", " "))

Нажмите Enter и обратите внимание на то, что все имена столбцов теперь содержат пробелы вместо подчеркивания.

Выберите запрос *Товары* и удалите последний шаг *Измененный тип*. Заметьте, что все файлы корректно сведены вместе, без каких-либо дополнительных признаков пропущенных значений.

### Таблица преобразования

К сожалению, простой нормализации может не хватить для исправления разношерстных заголовков. Для обозначения одного по сути столбца разные менеджеры используют 4 заголовка: ID, Product\_Number, Product\_num и Product Number. Нормализовать имена этих столбцов с помощью простых операций над текстом нельзя. Вместо этого можно сформировать таблицу преобразования исходных имен в нормализованные.



Рис. 8. Таблица преобразования исходных имен в нормализованные

Все несоответствия, которые вы будете выявлять при работе с исходными файлами, удобно отражать в таблице во внешней книге Excel. Постепенно будут появляться новые несоответствия, и формироваться новые строки таблицы преобразования. При подключении таблицы преобразования к запросам не нужно редактировать запрос для нормализации новых несоответствующих имен столбцов. Вместо этого просто добавьте в таблицу преобразования новые строки и обновите отчет.

Рассмотрим несколько методов, позволяющих работать с таблицами преобразования.

### Методика транспонирования

Для нормализации несовпадающих столбцов с помощью таблицы преобразования необходимо временно преобразовать имена столбцов в таблицу с одним столбцом, объединить таблицу преобразования, заменить несоответствующие имена столбцов требуемыми целевыми и преобразовать имена столбцов обратно в заголовки.

Загрузите файл C04E04 - Conversion Table.xlsx с таблицей преобразования. Загрузите файлы C04E04 - Accessories.xlsx, C04E04 - Bikes.xlsx, C04E04 - Components.xlsx, C04E04 - Clothing.xlsx в отдельную папку, например *Products*. Откройте новую книгу в Excel и пройдите *Данные* → *Получить данные* → *Из файла* → *Из книги*. Выберите файл C04E04 - Conversion Table.xlsx. Нажмите *Импорт*. В окне *Навигатор* выберите *Header\_Conversion*. Кликните *Загрузить* → *Загрузить в...*, выберите опцию

Только создать подключение. Теперь, при наличии таблицы преобразований в качестве нового запроса, можно загрузить содержимое папки *Products*.

Пройдите *Данные* → *Получить данные* → *Из файла* → *Из папки*. Выберите папку *Products* и нажмите *Открыть*, хотя ни один файл не выбран. В следующем окне файлы из папки будут представлены четырьмя строками. Пройдите *Объединить* → *Объединить и преобразовать данные*. В окне *Объединить файлы* выделите *Sheet1* и кликните *Ok*.

На панели *Запросы* выберите *Преобразовать пример файла* и переименуйте его в *Products Sample*. Напомним, что этот запрос является примером обработки одного файла из папки. Введенные здесь изменения повлияют на таблицу, добавленную в запросе *Products*. Переименуйте запрос *Products* в *Appended Products* и сохраните рабочую книгу Excel. Затем создайте три ее копии; они пригодятся при выполнении следующих заданий.

Первый способ нормализации имен столбцов основан на команде *Преобразование* → *Транспонировать*. С помощью транспонирования каждая ячейка в столбце X и строке Y будет помещена в столбец Y и строку X. При манипулировании значениями имен столбцов выполнить это в редакторе Power Query гораздо проще, если имена столбцов представлены вертикально в столбце.

Однако при попытке транспонирования таблицы обнаруживается, что исходные имена столбцов не сохраняются после преобразования. Для сохранения имен столбцов необходимо понизить имена столбцов до первой строки, а затем транспонировать таблицу. Для этого откройте сохраненную книгу и запустите редактор Power Query. На панели *Запросы* выберите запрос *Products Sample*, удалите шаг *Повышенные заголовки*. Теперь заголовки примера таблицы находятся в первой строке. Пройдите *Преобразование* → *Транспонировать*. Теперь столбец *Column1* включает имена столбцов, которые необходимо нормализовать.

ABC 123 Column1	ABC 123 Column2	ABC 123 Column3	ABC 123 Column4
1 Product	Water Bottle - 30 oz.	Mountain Bottle Cage	Road Bottle Cage
2 ID	WB-H098	BC-M005	BC-R205
3 Color		null	null
4 StandardCost	1,8663	3,7363	3,36
5 ListPrice	4,99	9,99	8,1
6 Size	null	null	null
7 Weight	null	null	null
8 Parent Category	Accessories	Accessories	Accessories
9 Category	Bottles and Cages	Bottles and Cages	Bottles and Cages

Рис. 9. Запрос *Products Sample* после транспонирования столбцов в строки

При выбранном запросе *Products Sample* перейдите *Главная* → *Объединить запросы*. Будьте внимательны, ранее мы использовали команду *Добавить запросы*. В окне *Слияние* выберите столбец *Column1* в таблице *Products Sample* (1), в раскрывающемся меню выберите запрос *Header\_Conversion* (2), выберите столбец *Source* (3):

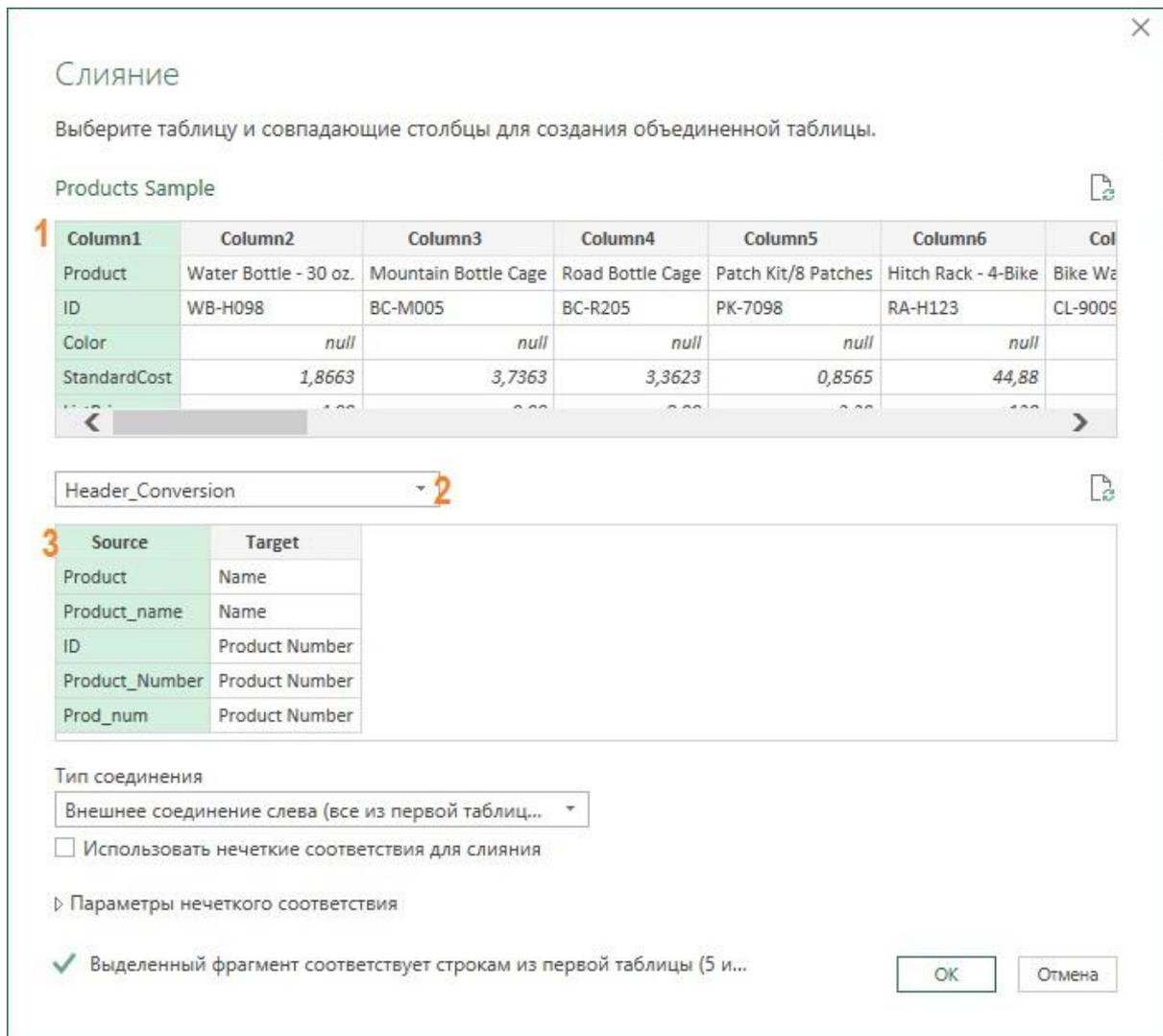


Рис. 10. Окно *Слияние*

Проверьте, что *Тип соединения* выбран *Внешнее соединение слева*, кликните *Ok*. В запрос *Products Sample* добавится новый столбец *Header\_Conversion* с объектами таблицы (Table). Разверните столбец *Header\_Conversion* щелкнув мышью на элементе управления справа от его заголовка, и установив опции:

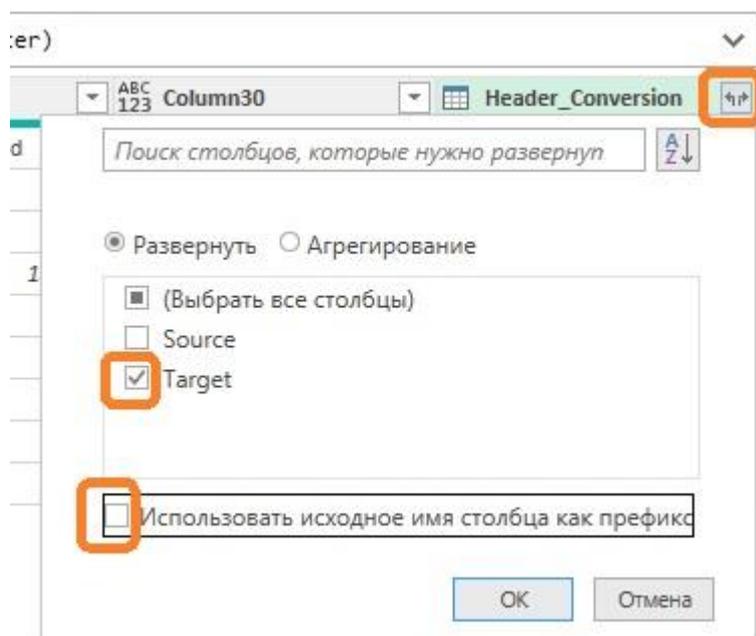
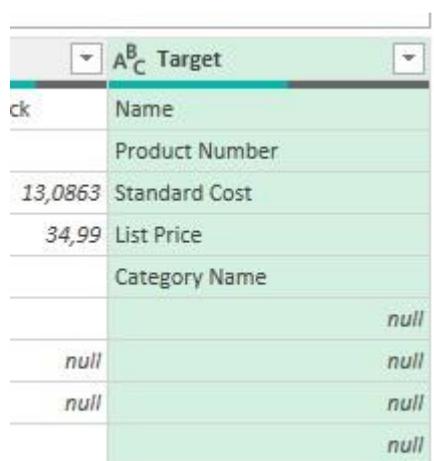


Рис. 11. Окно *Развернуть*

Столбец *Header\_Conversion* преобразуется в столбец *Target*:

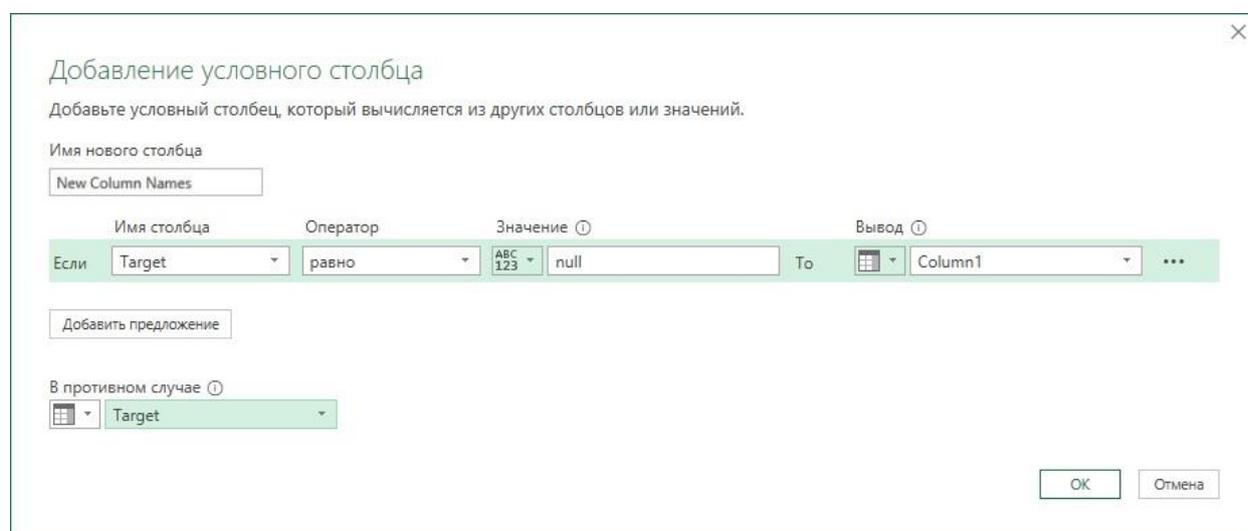


	Target
ck	Name
	Product Number
13,0863	Standard Cost
34,99	List Price
	Category Name
	null
null	null
null	null
	null

Рис. 12. Столбец *Target*

Значение *null* означает, что запрос *Header\_Conversion* не содержал значений для замены, т.е. исходные заголовки в столбце *Column1* отвечают требованиям. Добавим условный столбец для копирования целевых значений из строк, которые должны быть нормализованы, и исходных значений из прочих строк.

Пройдите *Добавление столбца* → *Условный столбец*. Настройте параметры:



Добавление условного столбца

Добавьте условный столбец, который вычисляется из других столбцов или значений.

Имя нового столбца  
New Column Names

Если	Имя столбца	Оператор	Значение	То	Вывод
	Target	равно	ABC 123 null		Column1

Добавить предложение

В противном случае  
Target

OK Отмена

Рис. 13. Окно *Добавление условного столбца*

Удалите столбцы *Column1* и *Target*. Переместите столбец *New Column Names* на место первого столбца таблицы. Просто перетащите и опустите его с правой на левую сторону или воспользуйтесь советом. Если в таблице много столбцов, а нужно переместить столбец в начало, вместо перетаскивания столбца можно выделить столбец и на вкладке *Преобразование* выбрать команду *Переместить* → *В начало*. Также можно щелкнуть правой кнопкой мыши на заголовке столбца и выбрать в контекстном меню команду *Переместить* → *В начало*.

Пришло время транспонировать таблицу обратно. Выполните *Преобразование* → *Транспонировать*, и *Преобразование* → *Использовать первую строку в качестве заголовков*.

Перейдите к запросу *Appended Products* и удалите последний шаг *Измененный тип*. Изучите запрос *Appended Products* в редакторе PQ или на листе Excel. Объединение должно быть выполнено корректно, несмотря на несоответствие в заголовках в четырех исходных файлах.

#### *Отмена свертывания, слияние и повторное сведение*

Метод транспонирования работает только при небольших наборах данных. Power Query поддерживает таблицы с 16 384 столбцами. Так что у вас не получится транспонировать таблицу с более чем 16 384 строками. Даже если испробовать этот метод на 15 000 строках, потребление памяти и низкая частота обновления наводят на мысль о необходимости альтернативного метода.

Откройте копию файла, сохраненного ранее. Напомню, мы остановились на шаге, когда переименовали запрос *Products* в *Appended Products*. Запустите редактор Power Query. Выберите запрос *Products Sample*. Пройдите *Добавление столбца* → *Столбец индекса*. Кликните правой кнопки мыши на столбце индекса и выберите *Отменить свертывание других столбцов*. Power Query создаст новую таблицу с тремя столбцами: *Индекс*, *Атрибут* и *Значение*. В столбце *Индекс* сохраняется исходный идентификатор строки, что поможет далее вернуть таблицу в исходное состояние. Столбец *Атрибут* включает имя столбца, а столбец *Значение* содержит исходные значения, которые имелись в таблице. Теперь можно выполнить подстановку в столбце *Атрибут*.

Индекс	Атрибут	Значение
1	Product	Water Bottle - 30 oz.
2	ID	WB-H098
3	StandardCost	1,8663
4	ListPrice	4,99
5	Parent Category	Accessories
6	Category	Bottles and Cages
7	Product	Mountain Bottle Cage
8	ID	BC-M005
9	StandardCost	3,7363
10	ListPrice	9,99
11	Parent Category	Accessories
12	Category	Bottles and Cages
13	Product	Road Bottle Cage

Рис. 14. Отменить свертывание других столбцов

Примените последовательность слияния, описанную выше. Пройдите *Главная* → *Объединить запросы*. Настройте окно *Слияние*, как на рис. 10. Разверните столбец *Header\_Conversion*, как показано на рис. 11. Добавьте условный столбец с параметрами, как на рис. 13. Удалите столбцы *Атрибут* и *Target*. Переместите столбец *New Column Names* на место второго столбца таблицы.

Для отмены развертывания выберите столбец *New Column Names*, пройдите *Преобразование* → *Столбец сведения*. В окне *Столбец сведения* выберите *Значение* (вместо *Индекс*), откройте раздел *Расширенные параметры*. Установите *Не агрегировать*, кликните *Ок*. Удалите столбец *Индекс*. Перейдите к запросу *Appended Products*, удалите шаг *Измененный тип*. Наслаждайтесь качественно скомбинированной таблицей!

### Применение языка M для нормализации имен столбцов

Это наиболее эффективный метод нормализации имен столбцов. Будет создана пользовательская функция, которая нормализует имена столбцов в соответствии с правилами из таблицы преобразования. Выполним одно изменение, которое упрощает код и ускоряет поиск в таблице преобразования. Ранее таблица преобразования была сформирована для пар значений *Source / Target*. Транспонируйте эту таблицу и используйте первую строку в качестве заголовков. Таким образом, новая таблица преобразования включит значения *Source* в качестве заголовков и значения *Target* в качестве первой строки.

Откройте копию файла, сохраненного ранее. Запустите редактор Power Query. На панели *Запросы* выберите *Header\_Conversion*. Пройдите *Преобразование* → *Транспонировать*, а затем *Использовать первую строку в качестве заголовков*. Удалите шаг *Измененный тип*, созданный автоматически. В нем нет необходимости, но этот шаг может привести к ошибкам обновления в будущем.

Выберите запрос *Products Sample* и щелкните мышью на значке *f<sub>x</sub>*. Измените формулу:

```
= "#Повышенные заголовки"
```

на:

```
= Table.TransformColumnNames("#Повышенные заголовки",
each try Table.Column(Header_Conversion, _) {0} otherwise _)
```

Нажмите *Enter*, изучите заголовки. Они нормализованы. Выберите запрос *Appended Products* и убедитесь, что все несоответствующие таблицы объединены правильно.

Обратите внимание на второй аргумент функции

```
each try Table.Column(Header_Conversion, _) {0} otherwise _
```

Комбинация `each` и `_` используется в качестве ярлыка для определения функции. В этом случае `Table.TransformColumnNames` требуется функция в качестве второго аргумента, и с помощью `each` и символа подчеркивания выполняется вызов этой функции для каждого имени столбца.

Для получения доступа к ячейке первой строки в одном из столбцов таблицы преобразования указан параметр `{0}`. В языке M все основано на индексах, начинающихся с нуля, поэтому для доступа к первой ячейке задан нуль. Фигурные скобки служат для доступа к элементу в списке по его нулевому индексу.

Допустим, столбец `Column1` находится в таблице `Header_Conversion`. Для получения доступа к списку значений в `Column1` применяется следующая формула:

```
Header_Conversion[Column1]
```

В этом случае имя столбца становится входным параметром для определенной функции с помощью символа подчеркивания. Для получения доступа к каждому из столбцов (в виде списков) по их именам в таблице `Header_Conversion` можно написать следующую функцию:

```
each Table.Column(Header_Conversion, _)
```

Заметим, что здесь применяется функция `Table.Column`. Следующая формула не будет работать:

```
each Header_Conversion[_]
```

С помощью этой формулы механизм языка M выполнит поиск столбца, действительное название которого представляет собой символ подчеркивания `_`. В результате получим следующую ошибку:

```
Expression.Error: The column '_' of the table wasn't found (Столбец '_' таблицы не найден).
```

Чтобы избежать появления подобной ошибки, применим функцию `Table.Column`, которая получает таблицу и имя столбца и возвращает нужный столбец в виде списка.

Напомним, что таблица преобразования была предназначена для хранения не всех возможных имен столбцов, а только несоответствующих имен столбцов. Поэтому выражение включает элементы `try/otherwise`. Если значение `Source` не обнаружено в таблице `Header_Conversion`, часть `try` потерпит неудачу, запуская другую часть, которая вернет значение `Source`.

Функцию...

```
= Table.TransformColumnNames("#Повышенные заголовки",  
each try Table.Column(Header_Conversion, _) {0} otherwise _)
```

...можно прочитать ее как инструкцию на естественном языке: «Возьмите таблицу, приведенную на шаге `Повышенные заголовки`, и преобразуйте имена столбцов по следующему правилу: для каждого столбца найдите его в качестве имени столбца в таблице преобразования и верните соответствующее значение в ячейку таблицы преобразования. Если этого сделать невозможно (значит, такого имени столбца в таблице преобразования нет), то верните имя столбца `Повышенные заголовки` в качестве выходных данных».