

Глава 19. Объекты Power Query

Это продолжение перевода книги Кен Пульс и Мигель Эскобар. Язык M для Power Query. Главы не являются независимыми, поэтому рекомендую читать последовательно.

[Предыдущая глава](#) [Содержание](#) [Следующая глава](#)

Прежде чем мы продолжим изучение языка M, сначала подробнее рассмотрим доступные объекты и функции. Обратите внимание, что все ключевые объекты в этой главе (таблицы, списки, записи, значения, двоичные файлы и ошибки) отображаются в столбцах редактора Power Query зеленым шрифтом. Кроме того, каждый объект можно просмотреть, щелкнув пробел рядом с ним.

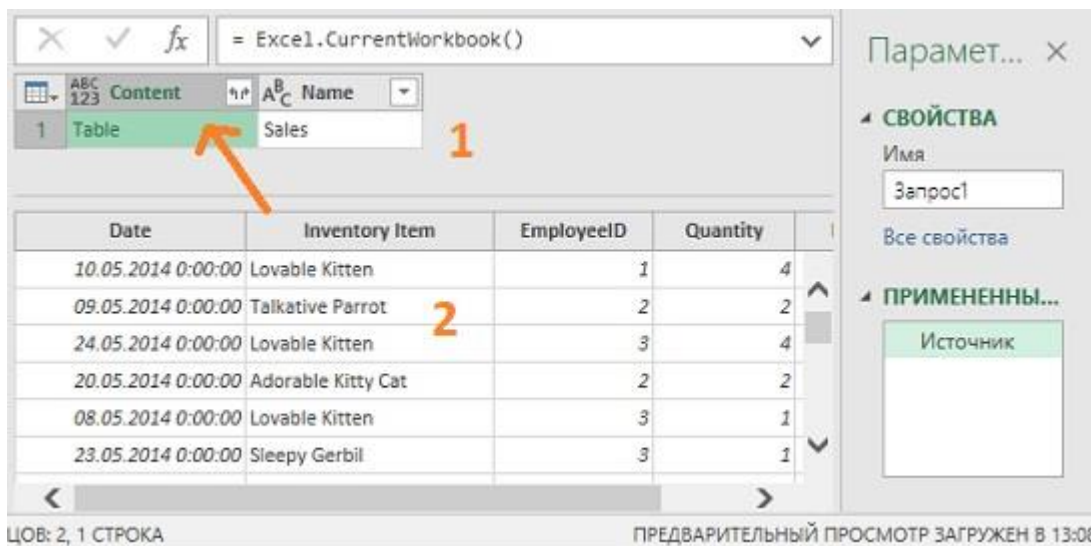


Рис. 19.1. Единственная таблица в просматриваемой книге

Таблицы

Объекты таблицы могут отображаться во многих местах в Power Query, и это весьма ценно, так как с ними очень легко работать. Выполните следующие действия, чтобы убедиться в этом. Откройте *Power Query Objects.xlsx*. Пройдите по меню *Данные* → *Получить данные* → *Из других источников* → *Пустой запрос*. Введите в строке формул: `=Excel.CurrentWorkbook()`. Нажмите Enter. В книге есть одна таблица (цифра 1 на рис. 19.1). Если кликнуть в пустое место рядом со словом *Table*, в нижней части окна появится предварительный просмотр таблицы (2).

Давайте завершим запрос, прежде чем двигаться дальше. Переименуйте запрос – *Table. Главная* – > *Закрыть и загрузить в...* → *Только создать подключение*.

Списки

В отличие от таблиц списки содержат только один столбец. Синтаксис: список можно идентифицировать по наличию фигурных скобок; элементы списка разделяются запятыми; текстовые элементы берутся в кавычки (по аналогии с использованием текстовых элементов в формулах Excel). Пример числового списка: `{1,2,3,4,5,6,7,8,9,10}`. Пример списка из текстовых элементов: `{"A","B","C","D","E","F","G","H","I","J"}`. Списки могут включать элементы разных типов данных, включая другие списки: `{1,465,"M","Data Monkey",{999,234}}`.

Создание списков

Создайте новую книгу Excel. Пройдите по меню *Данные* → *Получить данные* → *Из других источников* → *Пустой запрос*. В строке формул введите: `={1,2,3}`. Нажмите Enter.

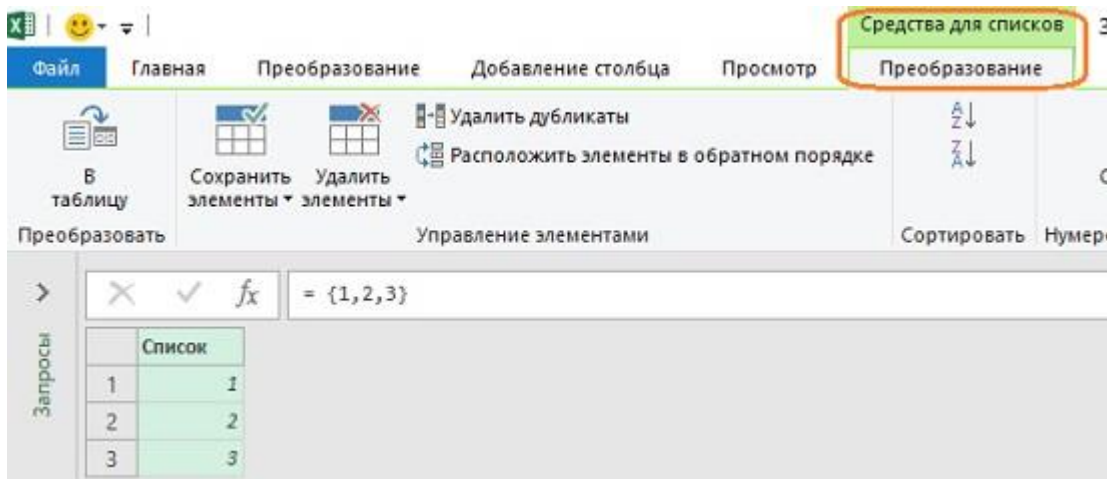


Рис. 19.2. Создание списка с нуля

Обратите внимание, появилась контекстная вкладка *Средства для списков* → *Преобразование*. В этот момент почти все команды на других вкладках будут неактивны.

Для создания больших списков с нуля можно воспользоваться специальными символами. Например, если вам нужен список 365 дней, введите в строку формул: `= {1..365}`. Таким же образом можно создавать последовательные алфавитные списки при условии, что символы заключены в кавычки и используется только один символ. Например, `= {"a".."J"}` будет работать, но `= {"AA".."ZZ"}` не будет. Вы также можете использовать запятые внутри списков, при условии, что они находятся внутри кавычек. Введите в строке формул: `= {"Пульс, Кен", "Эскобар, Мигель"}`, и вы получаете список из двух элементов, показывающие имена авторов книги:

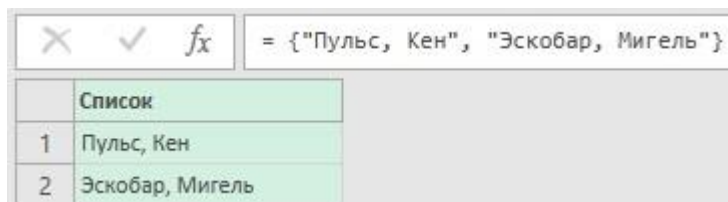


Рис. 19.3. Список из имен авторов книги

Преобразование списков в таблицы

Если вы хотите разделить список на несколько столбцов, сначала преобразуйте его в таблицу. Для этого нажмите кнопку *В таблицу* (на рис. 19.2 слева). Появится диалог:

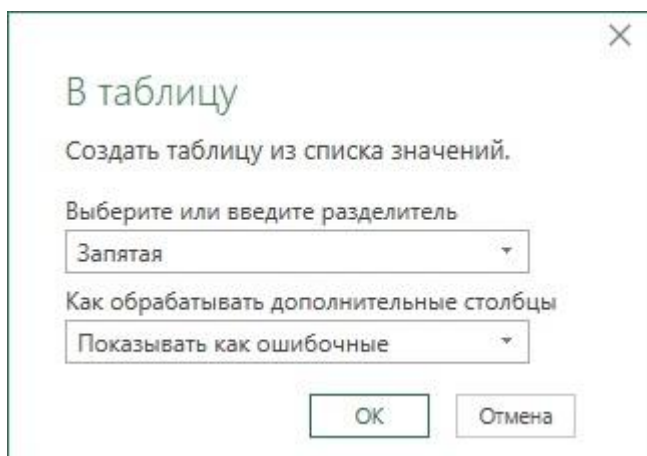


Рис. 19.4. Какой знак использовать в качестве разделителя?

Установить в качестве разделителя запятую, второй оставьте по умолчанию, нажмите *Ок*:

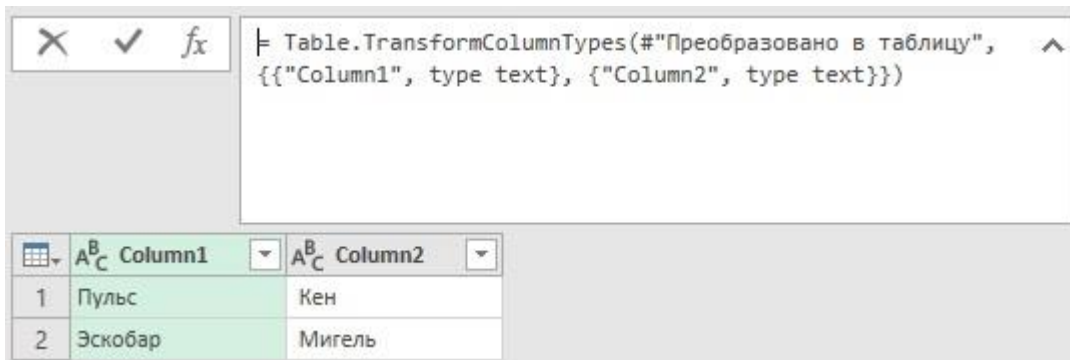


Рис. 19.5. Данные, загруженные из списка, разделенного запятыми

Завершить этот запрос. Измените имя запроса на List_Authors. Главная → Закрывать и загрузить в... → Только создать подключение.

Создание списков из столбцов таблицы

Иногда может возникнуть задача извлечь данные из одного столбца запроса в список. Давайте посмотрим, как это работает. Откройте файл *Power Query Objects.xlsx*. Перейдите на лист *Sales*. Кликните на любую ячейку *Таблицы*. Пройдите по меню *Данные* → *Из таблицы/диапазона*:

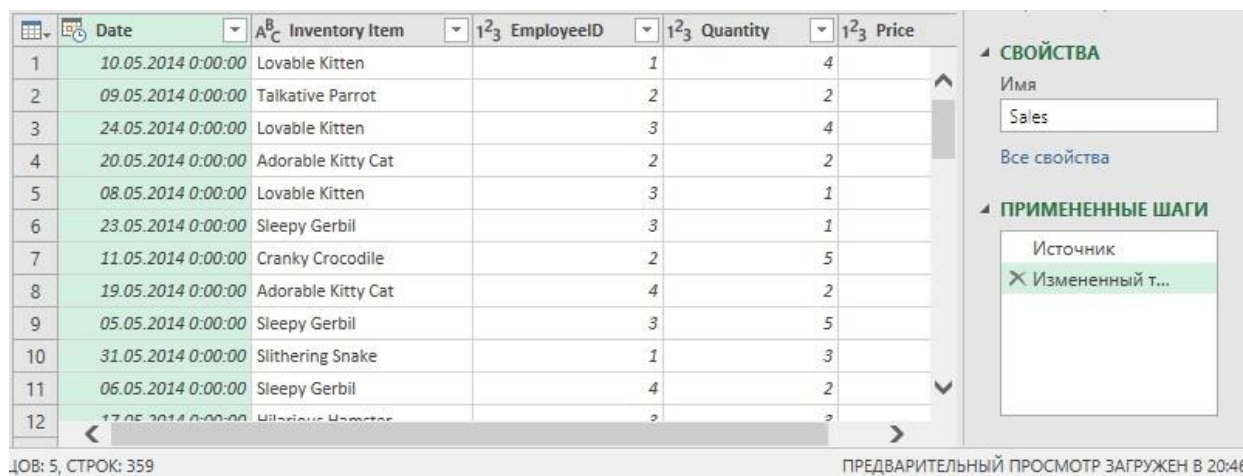


Рис. 19.6. Исходная таблица запроса

Если вам нужен уникальный список *Inventory Item*, вы можете удалить все остальные столбцы, а затем перейти на вкладку *Главная* → *Удалить строки* → *Удалить дубликаты*. Проблема в том, что объект все еще будет в форме таблицы, и вы не сможете передать его в функцию, если вам это нужно. Вы можете получить уникальные элементы в виде списка, что даст вам требуемую гибкость. Если вы экспериментировали с таблицей, удалите шаги, вернувшись к ситуации, как на рис. 19.6 – только два примененных шага. Щелкните правой кнопкой мыши столбец *Inventory Item* → *Детализация*. Вы получите список всех запасов:

= #"Измененный тип"[Inventory Item]	
	Список
1	Lovable Kitten
2	Talkative Parrot
3	Lovable Kitten
4	Adorable Kitty Cat
5	Lovable Kitten
6	Sleepy Gerbil
7	Cranky Crocodile
8	Adorable Kitty Cat
9	Sleepy Gerbil
10	Slithering Snake
11	Sleepy Gerbil

Рис. 19.7. Столбец, извлеченный из таблицы в список

Обратите внимание на строку кода: = #"Измененный тип"[Inventory Item]. В общем виде она может быть записана так = #Источник[Столбец]. Это шаблон языка М для извлечения всех значений столбца в список без использования команд пользовательского интерфейса. Запомните на будущее.

Теперь на вкладке *Средства для списков* кликните *Удалить дубликаты*. У вас останется список уникальных элементов, которые вы можете использовать в другой функции. Переименуйте запрос – *List_FromColumn*. Главная → *Закрыть и загрузить в...* → *Только создать подключение*.

Список списков

Это может показаться странной концепцией, поэтому давайте рассмотрим пример. Предположим, что у вас есть ID менеджеров по продажам (от 1 до 4), а именно: Фреда, Джона, Джейн и Мэри. Было бы неплохо иметь возможность конвертировать эти значения в их имена, не создавая отдельную таблицу. Пройдите по меню *Данные* → *Получить данные* → *Из других источников* → *Пустой запрос*. В строке формул создайте список с нуля:

```
= {{1, "Фред"}, {2, "Джон"}, {3, "Джейн"}, {4, "Мэри"}}
```

Обратите внимание, что здесь у вас есть четыре списка, которые, в свою очередь, объединены в основной список. Нажмите Enter:

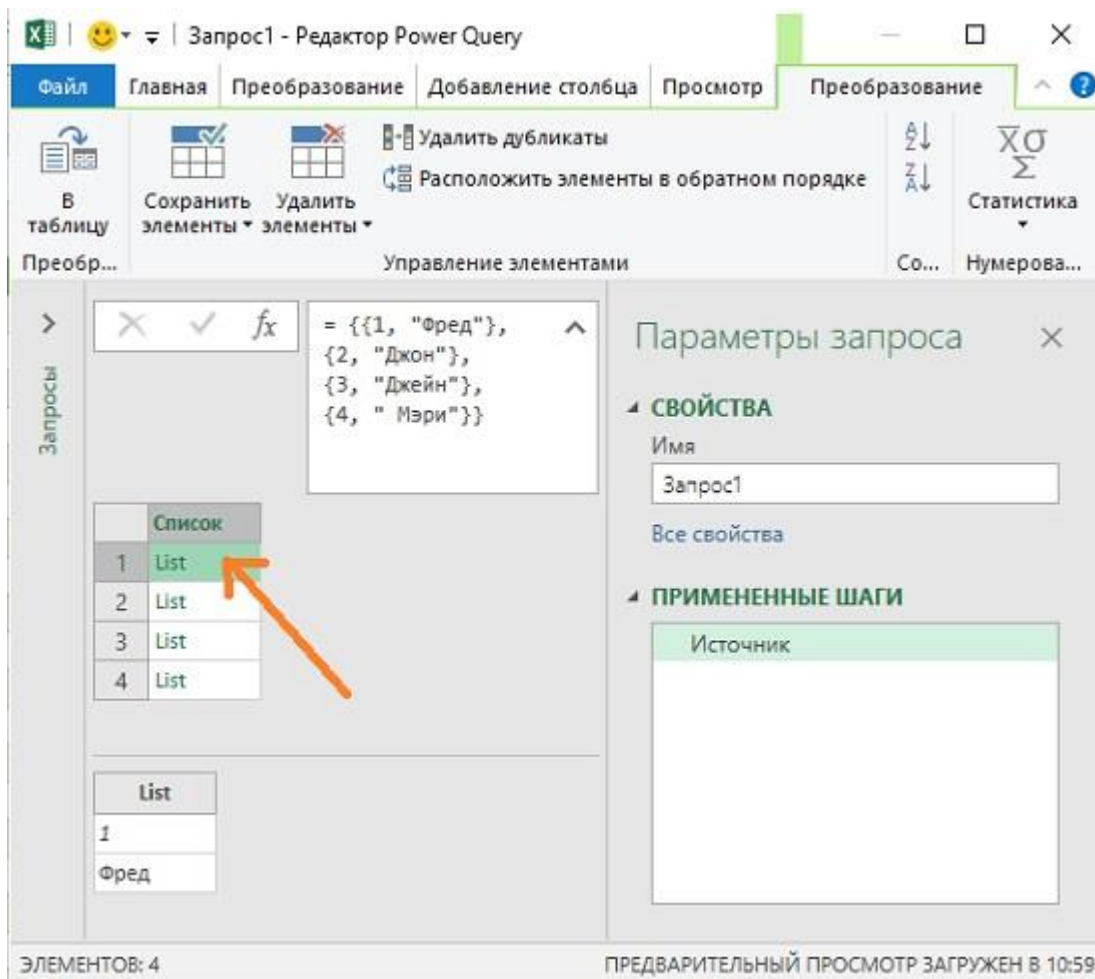


Рис. 19.8. Список, состоящий из списков

Предварительный просмотр первого списка показывает, что он содержит значения 1 и Fred. Нажмите кнопку *В таблицу*. Таблицу по-прежнему содержит списки, но появилась стрелка *Развернуть*:



Рис. 19.9. Список списков, преобразованный в таблицу

Нажмите на стрелку *Развернуть*. Вам будут предоставлены две опции:

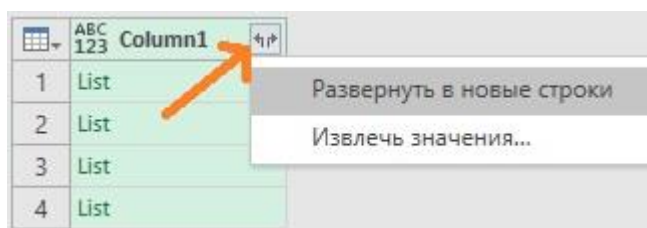


Рис. 19.10. Можно развернуть в новые строки или извлечь значения

При извлечении значений Power Query запросит вид разделителя. Я ввел запятую:

ABC 123	Column1
1	1
2	Фред
3	2
4	Джон
5	3
6	Джейн
7	4
8	Мэри

ABC Column1	
1	1,Фред
2	2,Джон
3	3,Джейн
4	4,Мэри

Рис. 19.11. Результаты: (1) разворачивания в новые строки, (2) извлечения значений

Переименуйте запрос *List_of_Lists*. Главная → Закрывать и загрузить в... → Только создать подключение.

При работе со списками необходимо учитывать два ключевых момента: списки могут содержать другие списки; развертывание списков списков не меняет их ориентацию.

Запись

Хотя список можно описать как один вертикальный столбец данных, на самом деле, запись является горизонтальным, многоколоночным аналогом списка. Запись может быть визуализирована как одна строка базы данных, содержащая всю соответствующую информацию, относящуюся к конкретному клиенту или транзакции. В Power Query записи могут отображаться в столбцах таблицы или в списках при получении данных. Они также могут быть созданы на лету, если это необходимо.

Синтаксис

Записи немного сложнее списков, так как они должны иметь какие-то значения, и вы должны определить имена столбцов, например:

```
=[Name="Ken Puls", Country="Canada", Languages Spoken=2]
```

Обратите внимание:

- Каждая запись заключена в квадратные скобки;
- Каждое поле записи (столбец) должно иметь определенное имя, за которым следует символ =.
- Затем предоставляются данные для поля в кавычках, если это текстовые данные.
- Пара *имя поля + данные* разделяются запятыми.

Имена полей (столбцов) не нуждаются в какой-либо пунктуации вокруг них, независимо от того, включают ли они пробелы или нет.

Что происходит, когда вам нужно создать несколько записей сразу? Вы создаете список из записей:

```
={[Name="Ken Puls", Country="Canada", Languages Spoken=2], [Name="Miguel Escobar", Country="Panama", Languages Spoken=2]}
```

Создание записи с нуля

Вернемся к примеру в разделе *Список списков* (рис. 19.11). Данные → Получить данные → Из других источников → Пустой запрос. В строке формул создайте запись:

```
=[EmployeeID=1,EmployeeName="Fred"]
```

Нажмите Enter.

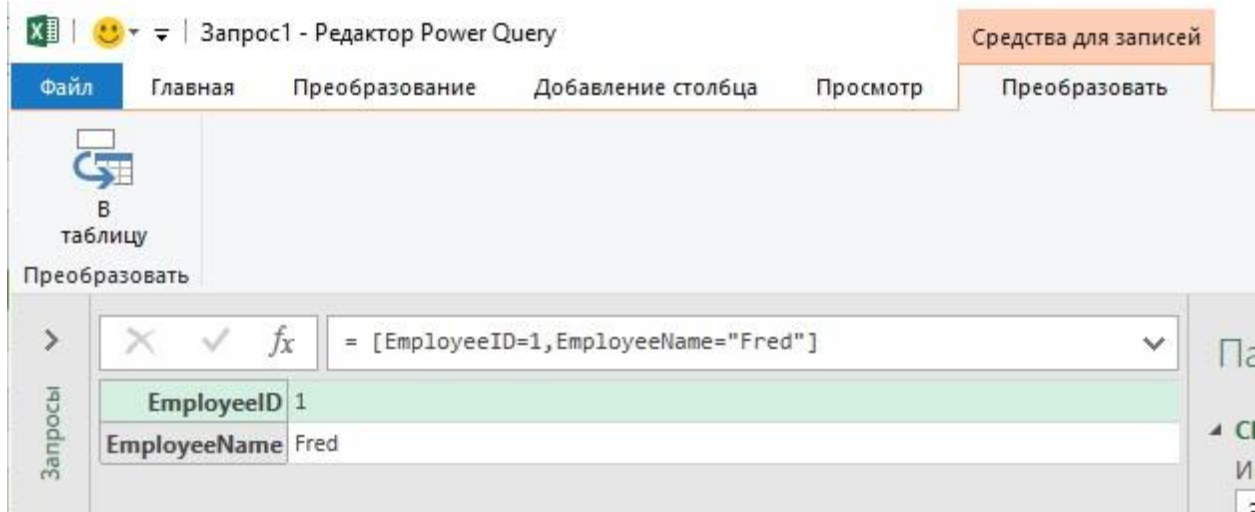


Рис. 19.12. Ваша первая запись

Как вы можете видеть, имена полей записи перечислены слева, а соответствующие данные – справа. Интересно, что данные расположены вертикально, а не горизонтально, как можно было бы ожидать. К этому нужно просто привыкнуть. Появилась контекстная вкладка *Средства для записей* → *Преобразовать*. Правда, на ней лишь одна кнопка, а большинство кнопок на других вкладках ленты не активны.

Преобразование записи в таблицу

Поскольку очевидно, что с записью мало что можно сделать, преобразуем ее в таблицу:

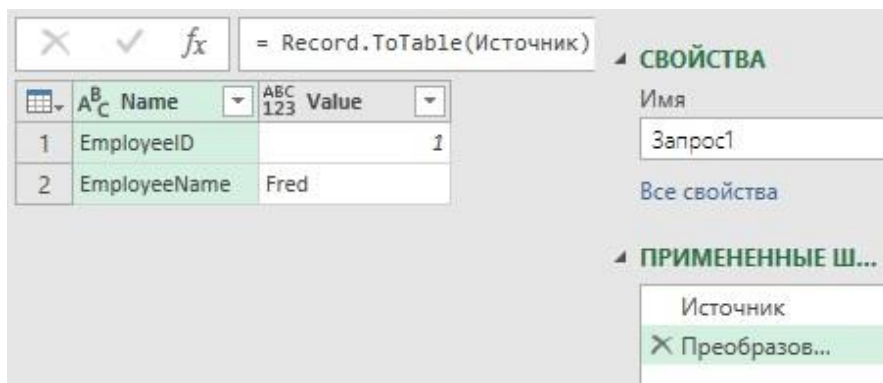


Рис. 19.13. Одна запись, преобразованная в таблицу

Возможно, вы ожидали, что таблица будет содержать имена полей в качестве заголовков столбцов. Хотя это и не так, исправить ситуацию не сложно. Перейдите на вкладку *Преобразование* → *Транспонировать*. *Главная* → *Использовать первую строку в качестве заголовков*. Именно на это вы надеялись с самого начала?



Рис. 19.14. Запись выглядит как правильная таблица

Сейчас это нормально, но что произойдет, если у вас есть множество записей, которые вы хотите преобразовать в таблицу? Завершите запрос, чтобы вы могли пойти дальше и узнать ответ на этот вопрос. Переименуйте запрос *Record_Single*. *Главная* → *Закреть и загрузить в...* → *Только создать подключение*.

Создание нескольких записей с нуля

Если вам нужна таблица, которая охватывает всех сотрудников, создайте список записей. *Данные* → *Получить данные* → *Из других источников* → *Пустой запрос*. В строке формул введите:

```
= {[EmployeeID=1,EmployeeName="Fred"], [EmployeeID=2,EmployeeName="John"], [EmployeeID=3,EmployeeName="Jane"], [EmployeeID=4,EmployeeName="Mary"]}
```

Обратите внимание, что вы используете синтаксис для одной записи, а записи разделили запятыми. Все записи взяли в фигурные скобки, показывая, что они элементы списка. Нажмите Enter. Формула возвращает список записей:

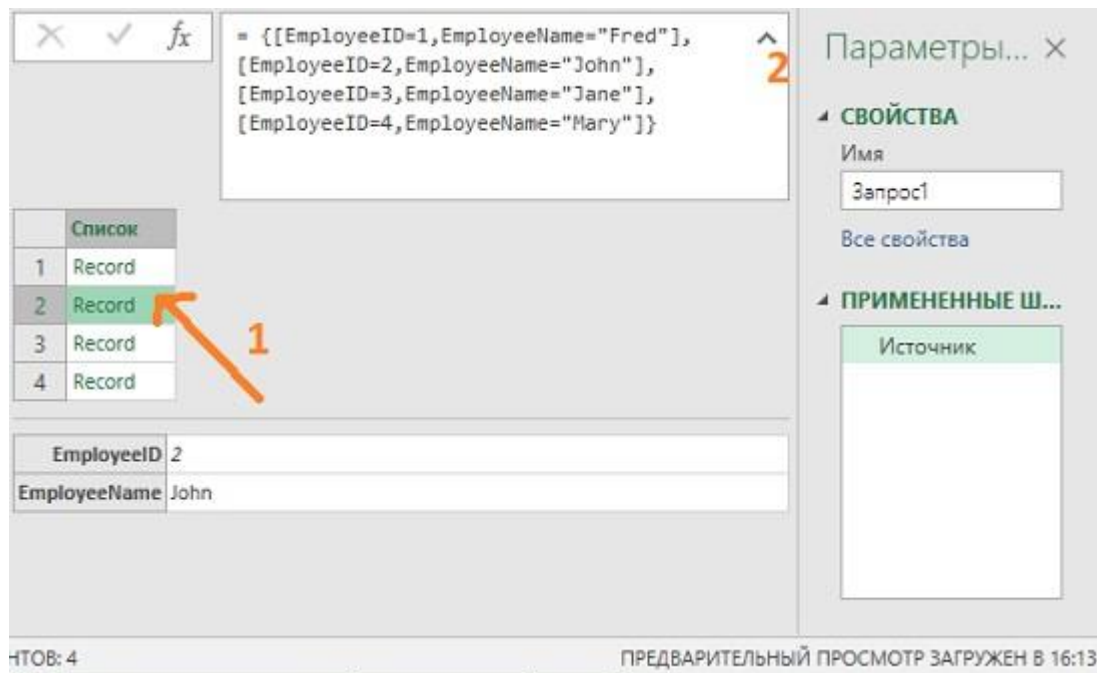


Рис. 19.15. Список записей с предварительным просмотром

Стрелка в правом верхнем углу (2) позволяет развернуть строку формул, чтобы показать сразу несколько строк.

Преобразование нескольких записей в таблицу

Теперь вы можете преобразовать этот список записей в таблицу. Пройдите по меню *Средства для списков* → *Преобразование* → *В таблицу*. Оставьте установки в окне *В таблицу* без изменений:

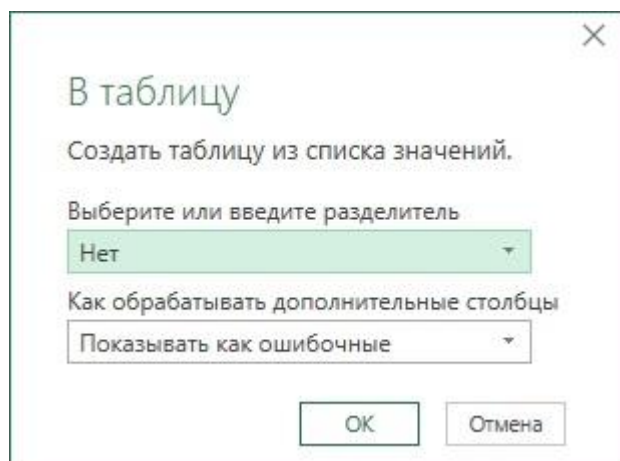


Рис. 19.16. Настройка параметров преобразования списка в таблицу

Если в качестве разделителя выбрать любой символ, преобразование в таблицу не удастся, и Power Query вернет ошибку:

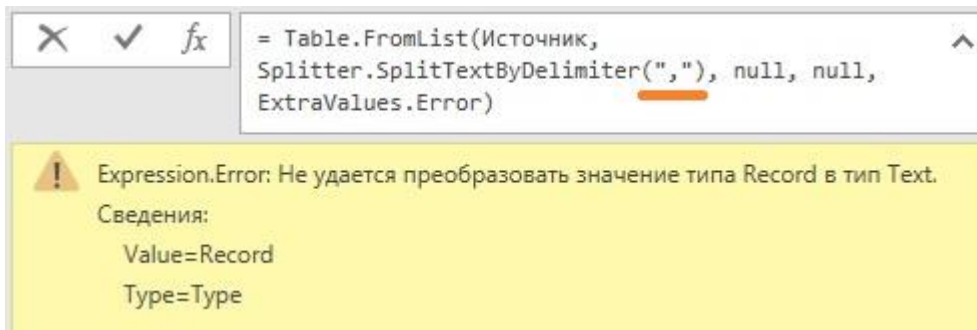


Рис. 19.17. Ошибка преобразования в таблицу при попытке указать разделить

Итак, после преобразования в таблицу на основе установок, как на рис. 19.16, вы опять получаете не то, что могли бы ожидать:

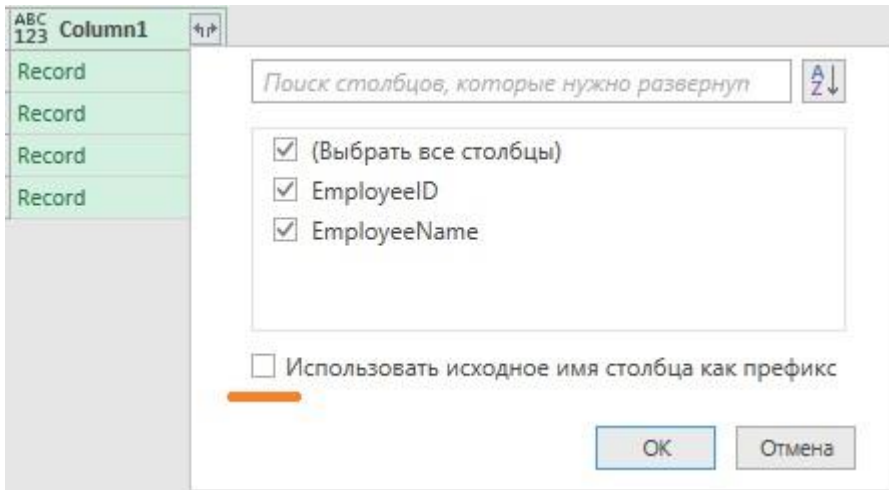


Рис. 19.18. Таблица включает список записей

Кликните на значке *Развернуть*. Выберите все столбцы, снимите галочку *Использовать исходное имя столбца как префикс*:

	ABC 123 EmployeeID	ABC 123 EmployeeName
1	1	Fred
2	2	John
3	3	Jane
4	4	Mary

Рис. 19.19. Наконец то получен правильный результат

Вы только что построили таблицу с нуля!

Теперь вы можете сохранить эту таблицу, и даже объединить ее с другими запросами. Переименуйте запрос *Table_From_Records*. Главная → Закрывать и загрузить в... → Только создать подключение.

Создание записи из строки таблицы

Когда вы работали со списками, вы видели, как можно преобразовать столбец в список. Вы также можете преобразовать строку в запись. Для этого можно начать с нового запроса. В файле Excel перейдите на лист *Sales* и выберите любую ячейку в *Таблице*. Пройдите по меню *Данные* → *Из таблицы/диапазона*. Чтобы извлечь первую запись, необходимо создать пустой шаг запроса. В области ПРИМЕНЕННЫЕ ШАГИ нажмите на первый шаг *Источник* (цифра 1 на рис. 19.20). Нажмите кнопку *fx* слева от строки формул (2):

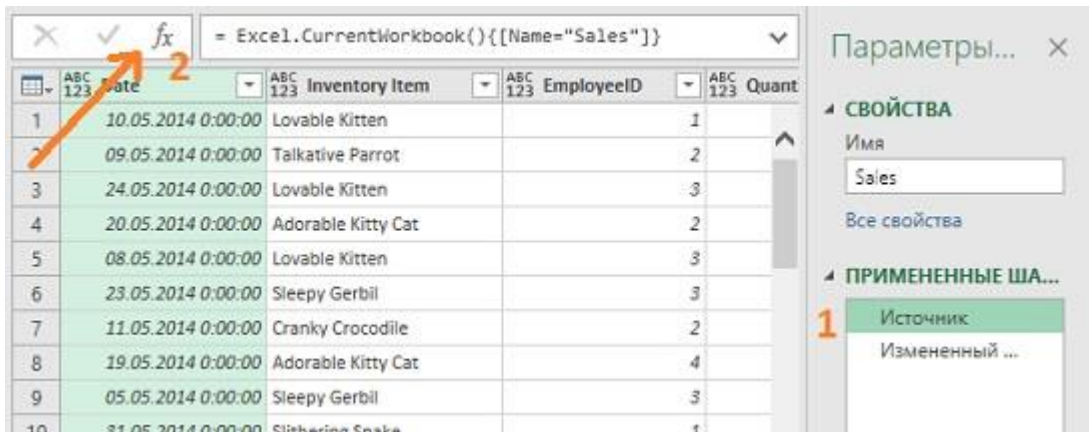


Рис. 19.20. Создание пустого шага запроса

Появится окно предупреждения о вставке шага. Подтвердите. В строке формул появится заготовка:

=Источник

Измените эту формулу на =Источник{0}

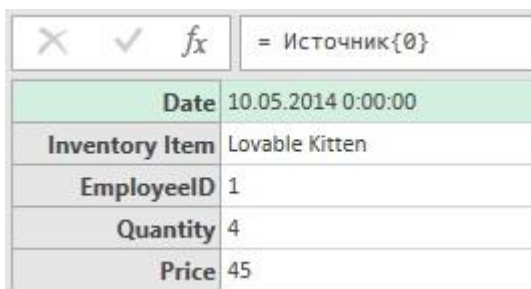


Рис.19.21. Из таблицы выделена первая запись

При первоначальном импорте *Таблицы* Excel исходный шаг возвращает список записей. Поскольку для Power Query базовым значением является ноль, формула =Источник{0} возвращает первое значение в списке.

Вы можете извлечь только одно поле первой записи, например, цену. Измените этот шаг запроса на: =Источник{0}[Price]



Рис. 19.22. Извлечение цены из первой записи

Это очень важно, и пригодится во многих ситуациях, например, при управлении фильтрами (подробнее см. главу 20).

Завершите запрос. Переименуйте его *Record_From_Table*. Главная → *Закрыть и загрузить в...* → *Только создать подключение*.

Создание записей из каждой строки таблицы

Чтобы преобразовать все строки таблицы запроса в записи, вам нужно использовать небольшой трюк. В файле Excel перейдите на лист *Sales* и выберите любую ячейку в *Таблице*. Пройдите по меню *Данные* → *Из таблицы/диапазона*. Таблица целиком загрузится в Power Query. Для создания записей вам нужен индекс каждой строки. Перейдите на вкладку *Добавление столбца* → *Столбец индекса*. Переименуйте этот шаг (а не столбец) в области ПРИМЕНЕННЫЕ ШАГИ – *AddedIndex* (без пробела):

fx = Table.AddIndexColumn("#Измененный тип", "Индекс", 0, 1)

	Date	Inventory Item	EmployeeID	Quantity	Price	1.2 Индекс
1	10.05.2014 0:00:00	Lovable Kitten	1	4	45	0
2	09.05.2014 0:00:00	Talkative Parrot	2	2	32	1
3	24.05.2014 0:00:00	Lovable Kitten	3	4	45	2
4	20.05.2014 0:00:00	Adorable Kitty Cat	2	2	35	3
5	08.05.2014 0:00:00	Lovable Kitten	3	1	45	4
6	23.05.2014 0:00:00	Sleepy Gerbil	3	1	39	5
7	11.05.2014 0:00:00	Cranky Crocodile	2	5	35	6
8	19.05.2014 0:00:00	Adorable Kitty Cat	4	2	35	7
9	05.05.2014 0:00:00	Sleepy Gerbil	3	5	39	8
10	31.05.2014 0:00:00	Slithering Snake	1	3	30	9
11	06.05.2014 0:00:00	Sleepy Gerbil	4	2	30	10

Панель параметров: СВОЙСТВА, Имя: Sales, Все свойства, ПРИМЕНЕННЫЕ...: Источник: Измененн..., X AddedIndex

Рис. 19.23. Добавлен столбец индекса и переименован шаг

Теперь вы можете создать пользовательский столбец для преобразования строк в записи. Хитрость заключается в создании формулы, использующей индекс столбца индекса, так как теперь у вас есть значение, необходимое для извлечения записей. Зачем тебе этот трюк? Вы не собираетесь работать на текущей строке, а скорее на выходе шага *AddedIndex*. Таким образом, вместо получения определенного значения (например, первой строки), вы можете динамически подавать его в запрос, чтобы получить каждую строку. Выполнить следующие действия:

Перейдите на вкладку *Добавление столбца* → *Настраиваемый столбец*. Назовите его *Records*. Используйте формулу: `=AddedIndex{[Индекс]}`.

	Date	Inventory Item	1.2 Индекс	Records
1	10.05.2014 0:00:00	Lovable Kitten	0	Record
2	09.05.2014 0:00:00	Talkative Parrot	1	Record
3	24.05.2014 0:00:00	Lovable Kitten	2	Record
4	20.05.2014 0:00:00	Adorable Kitty Cat	3	Record
5	08.05.2014 0:00:00	Lovable Kitten	4	Record
6	23.05.2014 0:00:00	Sleepy Gerbil	5	Record
7	11.05.2014 0:00:00	Cranky Crocodile	6	Record
8	19.05.2014 0:00:00	Adorable Kitty Cat	7	Record
9	05.05.2014 0:00:00	Sleepy Gerbil	8	Record
10	31.05.2014 0:00:00	Slithering Snake	9	Record
11	06.05.2014 0:00:00	Sleepy Gerbil	10	Record

Date	10.05.2014 0:00:00
Inventory Item	Lovable Kitten
EmployeeID	1
Quantity	4
Price	45
Индекс	0

Рис. 19.24. Столбец записей

Строго говоря, переименовывать шаг, на котором был добавлен столбец индекса, было необязательным. Просто отсутствие пробела в имени шага упрощает синтаксис формулы. Удалите все остальные столбцы, оставив только столбец записей. Переименовать запрос *Records_From_Table*. Главная → Закрыть и загрузить в... → Только создать подключение.

Значение (Value)

При работе с базами данных иногда отображаются столбцы, содержащие *Значение*:

SalesLastYear	CostYTD	CostLastYear	Person.CountryRegion	Person.Sta
3298694.494	0	0	Value	Table
3607148.937	0	0	Value	Table
3205014.077	0	0	Value	Table
5366575.71	0	0	Value	Table
3925071.432	0	0	Value	Table
5693988.86	0	0	Value	Table
2396539.76	0	0	Value	Table

CountryRegionCode	US
ModifiedDate	6/1/2002 12:00:00 AM
Person.StateProvince	Table
Sales.CountryRegionCurrency	Table
Sales.SalesTerritory	Table

Рис. 19.25. Неуловимый объект Значение

Чтобы обнаружить этот объект, необходимо работать с базой данных, в которой между таблицами установлены связи. *Значение* – это способ, которым база данных возвращает связанную запись. Т.е., запись хранящаяся в другой таблице, а не в той, с которой вы сейчас работаете. Таблица на рис. 19.25 расположена в базе данных Adventureworks2012, к которой вы подключились в [главе 8](#). Направление связи определяет, что возвращается в связанный столбец при работе с базой данных. Если вы находитесь в таблице фактов, и ссылка идет в таблицу измерений, вы получите значение (запись). Если вы находитесь в таблице измерений и ссылка идет в таблицу фактов, вы получите таблицу (подробнее см. [Power Pivot. Мышление в нескольких таблицах](#)).

Двоичные данные

Двоичные данные являются файлами. Некоторые из них можно объединять и читать с помощью функции Csv.Document(). В тоже время двоичные книги Excel нельзя объединять, а для чтения данных используется функция Excel.Workbook(). Процесс извлечения данных подробно рассмотрен ранее в [главе 3](#) и [главе 4](#). Мы упоминаем их здесь, поскольку они являются объектом Power Query.

Ошибки

Существует два типа сообщений об ошибках: ошибки уровня шага и ошибки уровня строки.

Ошибки на уровне строк обычно возникают при попытке преобразовать данные в неверный тип или при попытке оперировать данными до их преобразования в правильный тип. Вы уже видели несколько примеров таких ошибок ранее:

TerritoryID	CountryRegionCode	Group	SalesYTD	Sales
1	Error	North America	7887186.788	
2	Error	North America	2402176.848	
3	Error	North America	3072175.118	

! DataFormat.Error: We couldn't parse the input provided as a Date value.
 Details:
 US

Рис. 19.26. Ошибка уровня строки, вызванная попыткой преобразовать страны в тип данных даты

Эти ошибки, как правило, не являются блокирующими, и они даже могут быть полезны при очистке данных, так как их можно обрабатывать двумя способами:

- Использовать в качестве фильтров для хранения / удаления строк
- Заменить другими данными, используя, например, команду *Преобразование* → *Замена значений* → *Заменить ошибки*.

Несмотря на то, что в Power Query нет механизма отладки, эти ошибки, как правило, идентифицируются и часто (хотя и не всегда) связаны с неверными типами данных.

Ошибки уровня шага

Эти ошибки серьезнее. Такие сообщения блокируют запрос, и в окне вывода ничего не отображается, кроме ошибки:



Рис. 19.27. Синтаксическая ошибка, вызванная отсутствием закрывающей скобки в конце формулы

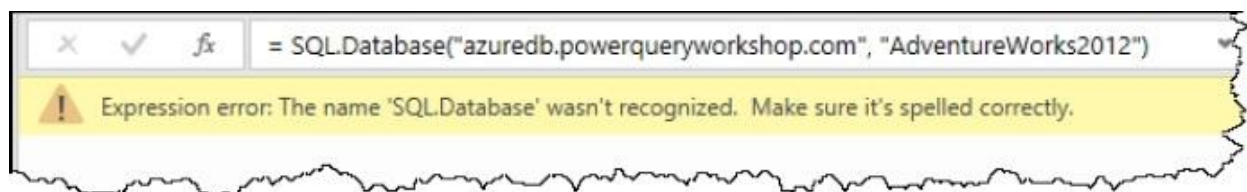


Рис. 19.28. Вызывается функция SQL.Database; правильно Sql.Database

К сожалению, средства отладки Power Query особенно слабы в следующей ситуации:

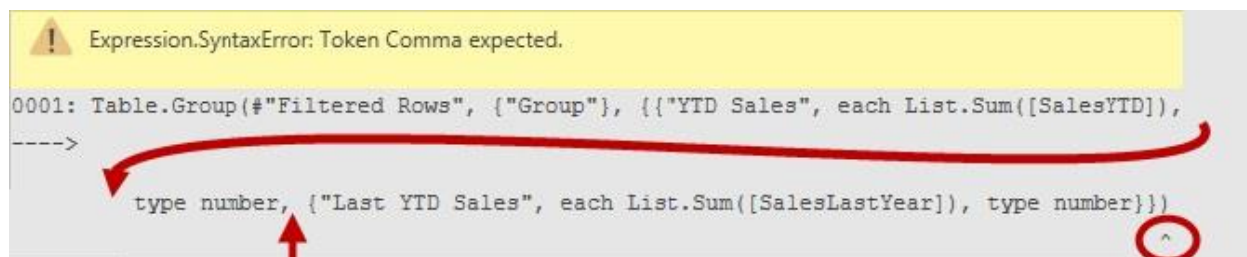


Рис. 19.29. Синтаксическая ошибка, вызванная отсутствием }, но сообщающая об ожидании запятой

В самом конце строки находится полезный символ ^, который указывает, где Power Query считает, что нужна запятая. Однако, проблема в том, что не хватает фигурной скобки для закрытия списка "YTD Sales" в месте, на которое указывает красная стрелка. Жаль, что в редакторе нет встроенной подсветки ошибок. Но Power Query часто обновляется, и мы надеемся увидеть прогресс в этой области в будущем. Сейчас же, отладка заключается в просмотре строки, открывающих и закрывающих символов, запятых и т.п.

Функции

Функции используются в двух местах:

- Они могут находиться внутри базы данных, где они указывают на хранимую процедуру.
- Они могут быть возвращены в списке Power Query.

Об использовании и вызове функций мы расскажем позже. Сейчас же покажем один трюк, который покажет, как функции проявляются, а также откроет список функций Power Query. *Данные* → *Получить данные* → *Из других источников* → *Пустой запрос*. В строке формул введите: =#shared. Появится список записей. Перейдите на вкладку *Средства для записей* →

Преобразовать → В таблицу. Power Query генерирует таблицу всех элементов Power Query в текущей книге, а также список всех функций, включенных в продукт.

	ABC 123	Name	Value
10		Value.ResourceExpression	Function
11		Resource.Access	Function
12		Kusto.Contents	Function
13		Kusto.Databases	Function
14		DataLake.Contents	Function
15		DataLake.Files	Function
16		List.NonNullCount	Function
17		List.MatchesAll	Function
18		List.MatchesAny	Function
19		List.Range	Function

Рис. 19.30. Таблица функций

Вы можете получить доступ к документации для любой функции из этого списка, если кликните на слово *Function*, например:

Table.Last

Возвращает последнюю строку из `table` или необязательное значение по умолчанию `default`, если таблица пуста.

Введите параметры

table

default (необязательно)

function (table as table, optional default as any) as any

Пример: Нахождение последней строки таблицы.

Использование:
`Table.Last (Table.FromRecords ({[CustomerID = 1, Name = "Bob", Phone = "123-4567"], [CustomerID = 2, Name = "Jim", Phone = "987-6543"], [CustomerID = 3, Name = "Paul", Phone = "543-7890"]})))`

Выходные данные:
`[CustomerID = 3, Name = "Paul", Phone = "543-7890"]`

Пример: Найти последнюю строку таблицы ({}) или вернуть [a = 0, b = 0], если она пуста.

Использование:
`Table.Last (Table.FromRecords ({}), [a = 0, b = 0])`

Выходные данные:
`[a = 0, b = 0]`

Рис. 19.31. Синтаксис функции Table.Last

Более того, на переднем плане появится диалоговое окно вызова функции:

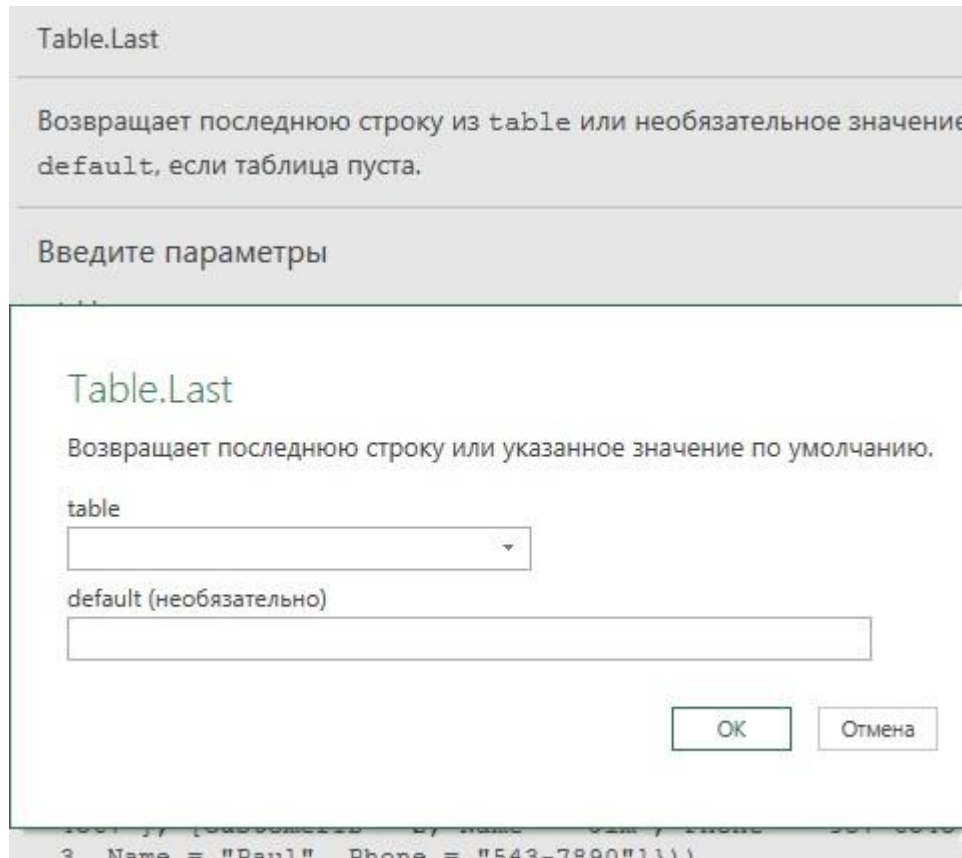


Рис. 19.32. Диалоговое окно функции Table.Last

Здесь можно протестировать функцию, а при нажатии кнопки *Отмена* окно исчезает.

Вам не обязательно создавать пустой запрос для вызова описания функции Power Query. Вы также можете в любой момент, когда вам это потребуется, кликнуть на *fx* слева от строки формул, чтобы добавить новый шаг. Замените код в новом шаге на `=#shared`. Нажмите *Enter*. Преобразуйте записи в таблицу. Можете использовать фильтр, чтобы найти функцию по имени. Кликните на слово *Function* напротив той функции, которую хотите изучить. Затем вы можете вернуться к другим шагам и продолжить формирование запроса. После того, как информация об этой или иной функции более не требуется, можете удалить шаги `=#shared`.