

УДК 004.424
ББК 32.372
П88

Кен Пульс, Мигель Эскобар

П88 Приручи данные с помощью Power Query в Excel и Power BI / пер. с англ.
А. Ю. Гинько. – М.: ДМК Пресс, 2022. – 572 с.: ил.

ISBN 978-5-93700-105-4

Иногда нас называют мартышками, работающими с данными, но на самом деле мы чаще ходим на волшебников. Наши данные редко появляются на свет в готовом к работе виде, и у нас могут уходить долгие часы на их очистку, фильтрацию и преобразование. Power Query помогает сократить этот процесс при первичной обработке данных, а все последующие обновления позволяет свести к простому нажатию на кнопку. Когда дело касается импорта, очистки и преобразования исходных данных для дальнейшего анализа, освоить Power Query бывает гораздо легче, чем выучить формулы Excel или язык программирования VBA.

Нет сомнений, что Power Query навсегда изменит подход специалистов Excel к работе с данными. Если у вас есть Excel, значит, у вас уже есть Power Query – этот инструмент встроен в Excel 2016 и выше. Эта книга поможет вам извлечь максимум пользы из Power Query.

УДК 004.424
ББК 32.372

Master Your Data with Power Query in Excel and Power BI, published by Holy Macro! Books.
Copyright © 2021, Russian-language edition copyright © Оформление, издание ДМК Пресс, 2021.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN 978-1-61547-058-7 (англ.)
ISBN 978-5-93700-105-4 (рус.)

© Tickling Keys, Inc., 2021
© Перевод, оформление, издание,
ДМК Пресс, 2022

Оглавление

Предисловие от издательства	15
Предисловие	17
Как Power Query изменил НАШИ жизни	17
История Кена: «Кофе и Power Query»	17
История Мигеля: новый старт	18
Благодарности от авторов	19
Благодарности от Кена	20
Благодарности от Мигеля	21
Наши преданные читатели	22
И наконец	22
Глава 0. Революция данных	23
Общий сценарий для аналитиков данных	23
Преимущества и опасности черной магии	24
Будущее изменилось	26
Почему Power Query – это магия?	28
Извлечение	29
Преобразование	29
Загрузка	30
Возможности Power Query и интеграция с другими продуктами	31
Компоненты Power Query	32
Цикл обновлений Power Query	34
Power Query Online	34
Microsoft 365	35
Excel 2016/2019/2021	35
Excel 2010 & 2013	35
Power BI Desktop	35
Как использовать эту книгу	36
Где найти Power Query?	36
Excel 365	36
Power BI Desktop	37
Предыдущие версии Excel	37
Подключение к данным	37
Особые пометки	38
Сопроводительные файлы	38
Глава 1. Основы Power Query	39
Перед началом	39
Изменение настроек Power Query по умолчанию в Excel	40
Изменение настроек Power Query по умолчанию в Power BI	40
Извлечение	41

Настройки подключения (выбор данных).....	41
Аутентификация.....	42
Предварительный просмотр	42
Выбор назначения запроса.....	43
Преобразование.....	44
Редактор Power Query	45
Преобразования по умолчанию.....	46
Источник (Source).....	46
Повышенные заголовки (Promoted Headers).....	47
Измененный тип (Changed Type).....	47
Создание и изменение преобразований	48
Загрузка.....	50
Установка типов данных	50
Переименование запроса	52
Загрузка запроса в Excel.....	52
Загрузка запроса в Power BI.....	53
Обновление запросов	54
Редактирование запросов	55
Запуск редактора Power Query в Power BI.....	56
Запуск редактора Power Query в Excel.....	56
Просмотр шагов.....	57
Настройка шагов	57
Влияние Power Query	60

Глава 2. Управление запросами

Использование архитектуры со множеством запросов.....	61
Разделение запросов на E, T и L.....	61
Преимущества совмещения запросов	62
Преимущества разделения запросов	63
Влияние разделения запросов на производительность.....	63
Ссылки на запросы.....	65
Создание базового запроса.....	65
Ссылочные запросы.....	66
Визуализация дерева зависимостей запросов	69
Просмотр зависимостей при помощи Monkey Tools	70
Выбор места загрузки запроса	71
Выбор места загрузки запроса в Power BI	72
Выбор места загрузки запроса в Excel	72
Изменение места назначения.....	76
Организация запросов.....	78
Создание папок в Power Query	78
Перенос запросов в группы	79
Изменение порядка следования запросов и групп.....	80
Создание подпапок запросов	81
Разделение существующих запросов	81
Заключительные мысли об архитектуре запросов.....	82

Глава 3. Типы данных и ошибки	85
Типы и форматы данных.....	85
Форматы	85
Типы данных.....	86
Как устанавливать формат данных в Power Query?	90
Порядок шагов имеет значение	90
Важность определения типов данных	92
Распространенные ошибки в Power Query	93
Ошибки на уровне шага	94
Ошибки источников данных	95
Ошибки вида «столбец X не найден».....	97
Ошибки значений.....	99
Обнаружение ошибок.....	99
Ошибки из-за неправильного приведения типов	101
Ошибки по причине несовместимости типов данных.....	103
Проверка запросов на ошибки	105
Обнаружение источника ошибок	105
Исправление исходного запроса	106
Удаление запроса с ошибками	108
Заключительные мысли о типах данных и ошибках	108
Глава 4. Перенос запросов между Excel и Power BI	109
Перенос запросов между решениями	109
Перенос запросов Excel в новую рабочую книгу	110
Перенос запросов из Excel в Power BI.....	113
Перенос запросов из Power BI в Excel.....	114
Перенос запросов из Power BI в новый проект Power BI	115
Импорт запросов из Excel в Power BI.....	115
Только внешние источники данных.....	116
Импорт модели данных Excel в Power BI	118
Импорт данных на основе таблиц Excel – копирование	119
Таблицы Excel – сохранение подключения	126
Заключительные мысли о переносе запросов между решениями	128
Глава 5. Импортирование из плоских файлов	131
Понимание процесса импорта данных.....	131
Определение системных настроек.....	132
Как программа интерпретирует плоские данные.....	133
Импортирование файлов с разделителями.....	135
Источник данных	136
Извлечение данных	136
Задача.....	137
Использование локали для установки корректных типов данных	138
Импортирование файлов без разделителей	141
Подключение к файлу.....	142
Очистка файлов без разделителей	143

Разделение столбцов по позиции.....	145
Прелесть ошибок в Power Query.....	146
Удаление лишних столбцов	148
Объединение столбцов.....	149
Разделение столбцов по разделителю	150
Исключение дублирующихся пробелов.....	151
Минута славы Power Query	152
Глава 6. Импортрование из файлов Excel.....	155
Данные в активной рабочей книге	155
Подключение к таблицам Excel.....	156
Подключение к табличным диапазонам	158
Подключение к именованным диапазонам.....	161
Динамические именованные диапазоны.....	163
Подключение к рабочим листам Excel из той же книги.....	165
Данные из других рабочих книг.....	165
Подключение к файлу Excel.....	166
Подключение к таблицам	168
Подключение к именованным диапазонам.....	169
Подключение к рабочим листам.....	170
Заключительные мысли о подключении к данным Excel	174
Глава 7. Простые техники преобразования данных.....	177
Снимаем проклятие сводных данных.....	177
Подготовка данных	178
Отмена свертывания других столбцов	179
Повторное сведение данных при помощи сводной таблицы	181
Есть ли жизнь после обновления данных?	182
Разница между различными типами отмены свертывания	183
Сведение столбца	184
Разделение столбцов.....	186
Разделение столбца на несколько столбцов.....	187
Разделение столбца на строки	188
Разделение на столбцы с отменой свертывания против разделения на строки	190
Фильтрация и сортировка	191
Фильтрация значений	192
Применение контекстных фильтров.....	195
Сортировка данных.....	197
Группирование данных	198
Глава 8. Добавление данных.....	203
Базовые операции по добавлению данных.....	203
Добавление двух таблиц.....	205
Добавление дополнительных таблиц.....	208
Объединение запросов с разными заголовками	211

Добавление таблиц и диапазонов в текущем файле.....	213
Консолидация таблиц	214
Консолидация диапазонов и рабочих листов	218
Используйте =Excel.CurrentWorkbook() с осторожностью.....	221
Заключительные мысли о добавлении запросов.....	221
Глава 9. Объединение файлов	223
Практический пример	223
Описание процесса	225
Методология объединения файлов	225
Архитектура запросов при объединении файлов	225
Шаг 0: подключение к папке.....	227
Подключение к локальной/сетевой папке	228
Подключение к папке SharePoint.....	229
Подключение к OneDrive для бизнеса.....	231
Подключение к другим файловым системам	231
Шаг 1: фильтрация и страховка на будущее	232
Методология шага 1	232
Применение шага 1 к нашему примеру.....	233
Шаг 2: объединение файлов.....	235
Методология шага 2	236
Применение шага 2 к нашему примеру.....	236
Шаг 3: преобразование данных в запросе примера.....	239
Почему нужно использовать запрос «Преобразовать пример файла»?	239
Использование запроса «Преобразовать пример файла».....	240
Шаг 4: преобразование данных в мастер-запросе	243
Исправление ошибки на уровне шага в мастер-запросе	243
Сохранение свойств файлов	244
Добавление дополнительных шагов	246
Обновление	248
Использование данных	248
Добавление новых файлов	249
Повышение эффективности с помощью сохранения верхних строк.....	250
Глава 10. Объединение данных	253
Основы объединения данных.....	253
Создание подготовительных запросов	254
Выполнение объединения запросов	254
Типы соединений	257
Внешнее соединение слева.....	260
Внешнее соединение справа	262
Полное внешнее соединение.....	264
Внутреннее соединение	265
Антисоединение слева	265
Антисоединение справа.....	266

Полное антисоединение.....	267
Декартовы произведения.....	268
Методология	268
Пример.....	268
Случайные декартовы произведения	271
Объединения с приблизительными совпадениями.....	272
Методология	273
Пример.....	273
Поиск нечетких соответствий	277
Основы нечеткого поиска	278
Таблицы преобразования	279
Управление порогом подобия.....	281
Стратегии поддержки решений с нечетким поиском	283
Глава 11. Источники данных в интернете	285
Подключение к файлам данных в интернете	285
Подключение к веб-страницам	287
Подключение к данным на веб-странице	287
Естественные и предлагаемые таблицы.....	288
Добавление таблиц с использованием примеров	289
Подключение к страницам без таблиц.....	291
Предостережения при работе с данными из интернета	296
Сбор данных.....	296
Целостность данных	296
Надежность решения.....	297
Глава 12. Реляционные источники данных	298
Подключение к базам данных	298
Соединение с БД.....	298
Управление учетными данными	301
Не можете подключиться к нашей базе данных?.....	302
Использование навигатора.....	303
Исследование данных.....	304
Свертывание запросов.....	307
Что такое свертывание запросов?	308
Какие технологии поддерживает механизм свертывания запросов? ..	310
Распространенные мифы относительно свертывания запросов.....	311
Уровни конфиденциальности	313
Объявление уровней конфиденциальности данных	315
Управление уровнями конфиденциальности данных	315
Конфиденциальность и производительность.....	316
Отключение движка конфиденциальности	317
Оптимизация	320
Глава 13. Преобразование табличных данных.....	323
Сложные шаблоны сведения данных	323

Сведение сгруппированных данных.....	323
Сведение вертикально сгруппированных данных	330
Сведение горизонтально сгруппированных данных	332
Сложные шаблоны отмены свертывания данных.....	337
Отмена свертывания данных с подкатегориями	337
Эффективная отмена свертывания данных с подкатегориями.....	345
Изменение запроса отмены свертывания данных с подкатегориями	346
Сохранение значений null при отмене свертывания данных.....	349
Продвинутое техники группирования данных.....	352
Процент от целого.....	352
Ранжирование данных	355
Нумерация сгруппированных строк (номера строк по секциям).....	359
Глава 14. Условная логика в Power Query	364
Основа условной логики.....	364
Описание набора данных.....	364
Подключение к данным	365
Создание условной логики при помощи интерфейса пользователя.....	366
Условная проверка в ручном режиме.....	369
Воспроизведение функции Excel ЕСЛИОШИБКА (IFERROR)	372
Работа с несколькими условиями	376
Сравнение со следующей/предыдущей строкой	379
Столбцы из примеров.....	383
Глава 15. Значения в Power Query	389
Типы значений в Power Query	389
Таблицы	391
Списки.....	392
Синтаксис	392
Создание списков	392
Преобразование списка в таблицу	394
Создание списка из столбца таблицы	395
Создание списка списков.....	397
Записи	400
Синтаксис	400
Создание записи.....	401
Преобразование записи в таблицу	402
Создание нескольких записей.....	402
Преобразование нескольких записей в таблицу	403
Доступ к записям таблиц по позиции (индексирование строк)	405
Доступ к записям таблиц по критерию.....	407
Создание записей из каждой строки таблицы.....	410
Значения	412
Двоичные данные.....	413
Ошибки	413

Ошибки на уровне строки.....	413
Ошибки на уровне шага	414
Функции.....	415
Ключевые слова в Power Query	418
#binary.....	419
#date, #datetime и #datetimezone	420
#time	421
#duration	422
type	423
#table	426
Глава 16. Изучаем язык M.....	429
Структура запроса на языке M	429
Структура запроса	430
Область определения запроса и идентификаторы	432
Обобщенные идентификаторы.....	434
Комментарии к коду	435
Собираем все воедино	437
Понимание процесса выполнения запроса.....	438
Что такое ленивое вычисление?	439
План выполнения запроса	440
Итераторы (построчное выполнение).....	443
Ремарка по поводу рекурсивных функций в Power Query	443
Ключевые слова each и _	444
Другие техники	449
Получение первого значения из столбца таблицы	449
Замена на null при ошибке навигации	451
Создание динамического списка заголовков типизированных столбцов	452
Создание динамического списка заголовков нетипизированных столбцов	456
Глава 17. Параметры и пользовательские функции	461
Воссоздание метода объединения файлов	461
Создание примера файла (Sample File).....	462
Создание параметра Sample File Parameter	463
Создание преобразования файла (Transform Sample).....	465
Создание функции Transform Function.....	466
Вызов функции Transform Function.....	467
Обновление функции Transform Function.....	467
Ключевые выводы	468
Создание настраиваемых функций с помощью параметров	469
Создание параметра FilePath	470
Создание запроса Timesheet Transform	471
Создание функции Timesheet Function.....	473
Обновление запроса Timesheet.....	473

Создание настраиваемых функций вручную.....	477
Построение сценария разового применения	477
Преобразование запроса в функцию	478
Вызов функции.....	481
Отладка настраиваемых функций.....	482
Восстановление функциональности	484
Таблицы динамических параметров.....	485
Проблема с динамическими путями к файлам	485
Реализация таблицы динамических параметров	487
Создание таблицы параметров.....	487
Реализация функции fnGetParameter.....	489
Вызов функции.....	490
Применение таблиц параметров	492
Глава 18. Техники работы с датой и временем.....	494
Определение границ календаря.....	494
Динамическое создание границ календаря.....	495
Корректировка начальной и конечной дат для нестандартных финансовых периодов.....	497
Корректировка начальной и конечной дат для 364-дневного календаря	499
Календари с последовательными датами.....	501
Создание календаря.....	501
Обогащение календаря за счет дополнительных столбцов.....	503
Столбцы для финансовых периодов в 12-месячном календаре	503
Столбцы-идентификаторы периодов для 364-дневного календаря ...	504
Столбцы финансовых периодов для календарей 4-4-5 (и их разновидностей).....	506
Что находится в файле с примерами?.....	509
Заполнение особых диапазонов даты и времени.....	510
Заполнение определенного количества дат	510
Заполнение определенного количества часов по каждой дате	512
Заполнение определенного количества дат с заданными интервалами	513
Разнесение данных на основе таблиц с датами.....	515
Разнесение данных по дням.....	515
Разнесение данных по целым месяцам.....	518
Разнесение данных по заданному количеству месяцев от начальной даты	522
Заключительные штрихи к разнесению данных.....	525
Глава 19. Оптимизация запросов	527
Оптимизация настроек Power Query	527
Глобальные параметры загрузки данных.....	527
Глобальные параметры редактора Power Query	528
Глобальные параметры безопасности	528

Глобальные параметры конфиденциальности.....	529
Настройки текущей книги (файла) – фоновые данные.....	529
Настройки текущей книги (файла) – другие.....	531
Использование функций буферизации.....	531
Форсирование вычисления значения.....	532
Буферизация вычисления значения.....	534
Снижение временных лагов во время разработки.....	537
Стратегия уменьшения временных лагов.....	538
Пример борьбы с временными лагами при разработке.....	539
Адаптация решения для снижения временных лагов.....	541
Изменение данных в предпросмотре.....	544
Ошибка Formula Firewall.....	544
Ошибка Formula.Firewall №1: несовместимость уровней конфиденциальности.....	545
Ошибка Formula.Firewall № 2: доступ к источнику данных.....	545
Вызов ошибки перестроения сочетания данных.....	546
Перестроение сочетания данных против создания цепочек запросов.....	548
Перестроение сочетания данных против выравнивания запросов....	551
Перестроение сочетания данных при передаче значений в SQL.....	553
Заключительные мысли об ошибках Formula.Firewall.....	555
Глава 20. Автоматизация обновлений.....	557
Варианты автоматического обновления в Excel.....	557
Обновления в Excel без VBA.....	557
Фоновое обновление.....	558
Обновление каждые X минут.....	558
Обновление при открытии рабочей книги.....	559
Быстрая загрузка данных.....	559
Автоматизация обновлений запросов в Excel с помощью VBA.....	560
Обновление одного подключения.....	560
Обновление в определенном порядке.....	563
Обновление всех запросов.....	565
Проблемы с синхронным обновлением.....	565
Расписание обновлений в Power BI.....	566
Предметный указатель.....	568



Предисловие

Как Power Query изменил НАШИ жизни

История Кена: «Кофе и Power Query»

Именно такое название было указано в моем календаре Outlook для ноябрьской встречи в далеком уже 2013 году. Это было во время одного из слетов Microsoft MVP. Незадолго до этого инструмент сменил свое прежнее имя Data Explorer, и у меня была запланирована встреча за кофе с Мигелем Льописом (Miguel Llopis) и Фейсалом Мохамудом (Faisal Mohamood) из команды Power Query, где мы собирались обсудить преимущества и недостатки этого инструмента с позиции пользователя Excel.

В той беседе я высказал свое мнение о том, что Power Query прелестен, но в то же время по своей сути является не более чем неуклюжей заменой SQL Server Management Studio. Я по сей день отчетливо помню ту часть дискуссии. Тогда я много работал с SSMS и Power Query и был раздосадован тем, что новый продукт фактически выполняет ограниченный круг задач Management Studio. В то же время у меня не получалось заставить его работать так же.

То, что произошло дальше, перевернуло все мои жалобы и стенания с ног на голову. Я восстанавливаю события по памяти, но ответ был примерно следующий:

«Кен, Power Query – это вовсе не замена SSMS. Мы создали этот инструмент для пользователей Excel... Чтобы им никогда не пришлось изучать и использовать язык SQL».

Тем, кто меня знает, прекрасно известно, что я редко лезу за словом в карман, но в тот момент я ничего не смог сказать. Этот ответ буквально перевернул мой мир.

Вы должны понимать, что я не совсем обычный пользователь Excel. Я прекрасно знаю SQL – до опасного прекрасно. К тому же у меня есть большой опыт работы с языками VBA, VB.NET, C#, XML и другими. И хотя я люблю неизведанные технологии и новые вызовы, все эти языки я изучал по необходимости. Как правило, мои требования бывают довольно сложными, и я бросаюсь в новые для себя области по принципу «утону или выплыву».

Встреча, о которой я поведал, изменила мое отношение к Power Query навсегда. Я сделал шаг назад и взглянул на этот инструмент под другим углом. После этого я начал использовать его по его прямому назначению – посредством интерфейса и без написания запросов SQL везде, где это возможно. И вы знаете, Power Query расцвел для меня новыми красками – он позволил мне добраться до самой сути данных и решать задачи, ранее мне неподвластные.

Я обожаю Power Query. И не только за то, что он позволяет мне делать, но и за то, как просто он позволяет это делать – без необходимости писать код

вообще. Да, в этом инструменте есть свой язык программирования, к услугам которого вы можете прибегать по мере роста сложности ваших задач, но это совсем не обязательно. И именно это делает Power Query таким привлекательным – он обладает одним из лучших интерфейсов, какие я только видел, и, по сути, пишет код за вас, пока вы нажимаете на кнопки. Также мне нравится это средство за то, как быстро наши ученики способны освоить его и начать реализовывать свои первые решения с его помощью, приносящие реальную пользу. Этот продукт целиком и полностью нацелен на применение в бизнесе.

Что касается меня лично, то Power Query позволил мне уйти из офиса и организовать собственное дело. Мы проводим очное и заочное обучение, а также распространяем платное дополнение к Excel – Monkey Tools, способное значительно облегчить выполнение задач при работе с Power Query и Power Pivot в Excel. В конце концов, я не знаю большего счастья, чем видеть благодарного и просветленного ученика, понявшего, как можно улучшить свою работу и при этом еще и сэкономить время.

История Мигеля: новый старт

Перед тем как начать собственный бизнес в качестве фрилансера в 2013 году, я прочно застолбил за собой репутацию продвинутого пользователя, так что решил сохранить это прозвище после ухода из офиса и назвать так свой канал на YouTube и новый сайт – *The Power User*.

Я никогда не был айтишником, но всегда был тем парнем, который старается выжать максимум возможного из имеющихся под рукой технологических средств, и по большей части таким средством был обычный Excel, и далеко не всегда последней версии. В какой-то момент Excel и сводные таблицы стали моим вторым именем.

В 2013 году я познакомился с Power Query. Я даже не могу вспомнить, как именно это было, но помню, что в моей повседневной рутинной работе тогда приходилось довольно часто фильтровать данные, удалять столбцы, повышать заголовки, разворачивать столбцы и т. д. Я не знал VBA (и до сих пор не знаю), так что Power Query буквально открыл для меня новый мир с новыми возможностями, которых раньше у меня не было. Мне больше не было нужды думать о том, чтобы становиться экспертом в VBA или SQL – все, что мне нужно было, – это освоить Power Query, с которым все мои проблемы с манипуляцией данными остались бы в прошлом.

Интерфейс пользователя Power Query полностью меня захватил. Он казался мне настолько интуитивно понятным, словно можно было усесться в него, пристегнуться ремнями и рассекать между моими данными так, как мне нравится. В то время это был совершенно новый инструмент, и в интернете не было абсолютно никакой информации о том, как выжать из него максимум. Так что я решил заполнить это зияющую пустоту и начать создавать полезный обучающий контент о Power Query самостоятельно.

В процессе ведения блога и записи видео я познакомился с массой людей, среди которых были Роб Колли (Rob Collie) и Билл Джелен (Bill Jelen), и через них я узнал о Кене, в то время активно увлекающемся Power Query. Ни разу не

встречаясь лично, мы с ним решили объединить усилия, поскольку чувствовали, что наши знания в области Power Query дополняют друг друга, к тому же мы оба страстно хотели донести до людей полезную информацию о таком богатом инструменте. Мы запустили проект под названием PowerQuery. Training, и информация, накопленная за время его работы, легла в основу первого издания нашей книги. Во время ее написания и даже до этого мы прекрасно осознавали всю мощь Power Query и его способность максимально облегчить жизнь пользователям Excel. ***Для нас этот инструмент был тогда и остается сейчас самым большим прорывом в области средств обслуживания данных.***

С момента выхода первого издания книги прошло много времени, на протяжении которого читатели, друзья и коллеги то и дело напоминали нам о том, что некоторые картинки и приемы в ней постепенно устаревают, одновременно признавая пользу книги в деле освоения Power Query и раскрытия его потенциала. Признаться, это и было нашей изначальной целью – мы хотели изменить жизнь людей так же, как когда-то Power Query изменил нашу жизнь, сделав процесс преобразования данных простым и интуитивно понятным.

С 2015 по 2021 год мы с Кеном получили огромное множество подтверждений того, что Power Query действительно изменил не одну жизнь – косвенно или напрямую. И такие подтверждения всякий раз вызывают улыбку на наших лицах. Это в том числе повлияло на наше желание написать второе издание книги и сделать его таким, каким оно в итоге получилось. Мы хотели все сделать правильно, и ради этого нам пришлось ждать идеального момента для выхода книги.

Глава 0

Революция данных

Общий сценарий для аналитиков данных

Производим ли мы базовый ввод информации, строим ли простые отчеты или разрабатываем полноценные бизнес-решения с использованием VBA, SQL и/или других языков, мы так или иначе работаем с данными. Наши наборы навыков и умений могут сильно варьироваться, но в основном работа с информацией включает в себя:

- извлечение данных из источника;
- преобразование данных под свои нужды;
- добавление наборов данных;
- объединение разных данных вместе;
- обогащение данных для расширенного анализа.

Мы работаем с данными, и как бы мы ни именовали наши должности официально, наша задача – получить данные, очистить их и превратить в информацию. Какого-то лоска и великой славы в нашей работе нет, но она невероятно важна, поскольку без нее аналитическая информация будет оставаться разрозненной и ненадежной.

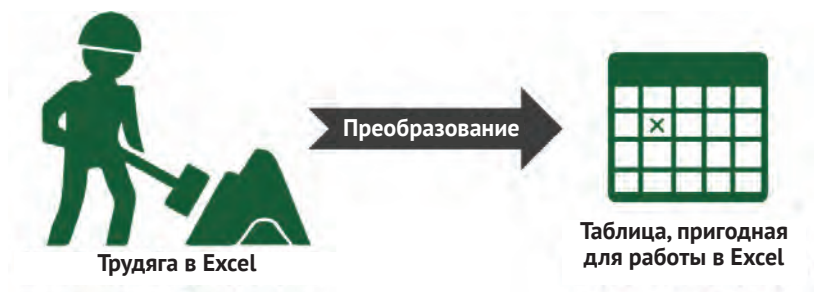


Рис. 0.1. За кулисами все мы работаги от Excel и информационные трудяги, пытающиеся достичь своих целей в работе с данными

Долгие годы нашим основным рабочим инструментом был Microsoft Excel. И хотя Excel обладает достаточной функциональностью для построения пол-

ноценных систем бизнес-аналитики на основе данных, в вопросе превращения сырых данных в пригодную для анализа информацию этот инструмент сам по себе не так силен. Иногда большую часть своего рабочего времени мы тратим как раз на то, чтобы получить данные из источника, подготовить их для анализа, преобразовав в привычные нам таблицы, и передать системе анализа или формирования отчетности.

Но те, кто знают нашу работу поближе, понимают, что мы не просто трудяги, а настоящие Волшебники данных! Наши исходные данные редко появляются на свет в виде, пригодном для анализа, – гораздо чаще нам приходится тратить долгие часы на их очистку, фильтрацию и преобразование в приемлемый вид.

После завершения процесса преобразования данных мы можем с легкостью использовать весь доступный арсенал инструментов для аналитики. Условное форматирование, фильтрация, сводные таблицы, диаграммы, срезы и многое другое – все эти средства можно применять для свершения настоящей магии и удивления аудитории.

Но мы вступаем в игру на гораздо более ранней стадии процесса. Нам поступают грязные и сырые данные, обычно хранящиеся в текстовых файлах или в Excel (а если ОЧЕНЬ повезет, в базе данных), и мы используем все имеющиеся у нас средства для их очистки и подготовки к использованию. В конечном счете наша цель проста: максимально быстро привести исходные данные к табличной форме, сохранив при этом их точность и масштабируемость. Для каждой задачи данные могут поступать разрозненно из самых разных источников, и наша задача – собрать их все воедино. Без магии здесь точно не обойтись...



Рис. 0.2. Черная магия: что происходит за сценой при консолидации данных

Преимущества и опасности черной магии

Настоящие волшебники Excel используют в своем деле различные колдовские техники – иногда отдельно, иногда в сочетании с другими инструментами. Вот лишь некоторые из них:

- *формулы Excel* (Excel formulas). Зачастую к этой технике разработчики прибегают в первую очередь, используя полное многообразие фор-

мул, включая *ВПР* (VLOOKUP), *ИНДЕКС* (INDEX), *ПОИСКПОЗ* (MATCH), *СМЕЩ* (OFFSET), *ЛЕВСИМВ* (LEFT), *ДЛСТР* (LEN), *СЖПРОБЕЛЫ* (TRIM), *ПЕЧСИМВ* (CLEAN) и др. Хотя формулами в Excel пользуются практически все, их сложность и реализуемые с их помощью задачи зависят в большей степени от опыта пользователя или разработчика;

- *VBA* (Visual Basic for Applications). Мощный язык программирования, способный обеспечить полноценное и динамическое преобразование данных. В то же время эта техника обычно доступна только продвинутым пользователям Excel, обладающим определенным опытом в программировании и ощущающим в себе силы разобраться во всех нюансах языка;
- *выражения SQL* (SQL Statements). Эта техника подразумевает использование еще одного языка – на этот раз языка запросов SQL – для манипулирования данными. Это очень мощный язык, с помощью которого можно выбирать, сортировать, фильтровать, группировать данные и производить с ними множество других действий. В реальности же этой техникой также пользуются только профессионалы, тогда как обычный пользователь Excel даже не знает, как подступиться к этой теме. Язык *SQL* – это главный инструмент разработчиков баз данных по всему миру, но и пользователям Excel, честно говоря, было бы неплохо познакомиться с его основами.

Объединяет все эти инструменты то, что в прежние времена при преобразовании данных в полезную информацию мы были ограничены только ими.

Однако, несмотря на все преимущества этих средств работы с данными, им присущи два серьезных недостатка: время, требуемое на реализацию решения, и время, необходимое для освоения инструмента.

И хотя самым могущественным волшебникам по силам строить полноценные решения по загрузке и преобразованию исходных данных при помощи только одних этих средств, на освоение этой древней магии у них уходит не один год, а затем они вынуждены тратить много времени на разработку, тестирование и поддержку своих решений. Кроме того, таким решениям зачастую катастрофически не хватает гибкости: при небольшом изменении формата исходных данных разработчику может потребоваться немало времени на корректировку процедуры импорта, как и при добавлении нового источника.

Это ведет к третьей опасности наличия в компании таких волшебников: все они строят потрясающие решения, которые перестают работать сразу после их ухода. И тогда в компании понимают, что ничего не понимают в созданном алгоритме и что никто разобраться в нем не в силах.

А иногда задачу сбора и подготовки данных поручают тем, кто просто не в состоянии освоить все эти магические премудрости. И хотя в этом случае компания не будет так сильно страдать во время отказа настроенной волшебником системы, хорошего здесь тоже мало, поскольку такой человек будет бесконечно растрачивать временные и финансовые ресурсы компании, на ежедневной основе совершая одни и те же никому не нужные операции импорта и трансформации данных.

Остановитесь на секунду и подумайте, сколько времени в вашей организации тратится каждый день на повторяющиеся операции по очистке входных данных в Excel. Умножьте это время на среднюю зарплату в вашей компании, а теперь на количество подобных компаний в вашей отрасли. Нам продолжать? Стоимость неэффективной производительности будет очень и очень высока.

Нет, нам нужно что-то другое. Какой-то продукт, который можно будет достаточно быстро освоить и который позволит автоматизировать процесс загрузки и очистки данных, чтобы мы могли сконцентрироваться на превращении данных в информацию и тем самым сэкономить деньги компании.

И такой продукт уже есть! Встречайте **Power Query**!

Будущее изменилось

Power Query – решение всех проблем в работе с данными, поскольку этот инструмент практически лишен недостатков, присущих всем перечисленным выше техникам. Освоить Power Query можно достаточно быстро, ведь он обладает одним из наиболее интуитивно понятных интерфейсов, с которыми нам доводилось работать. Кроме того, все производимые действия фиксируются в последовательности шагов, которые впоследствии можно просмотреть или изменить при необходимости. И все, что было сделано при помощи Power Query, можно воспроизвести посредством пары щелчков мыши.

Для нас – людей, которые долгие годы реализовывали решения в Excel с помощью перечисленных выше техник, – Power Query явился глотком свежего воздуха, и сразу по нескольким причинам.

Первая из них состоит в простоте этого инструмента, работе в котором очень легко обучиться. В процессе выполнения импорта и преобразования данных освоить Power Query можно даже быстрее, чем формулы Excel, а сложные решения с его помощью можно реализовывать легче, чем в VBA.

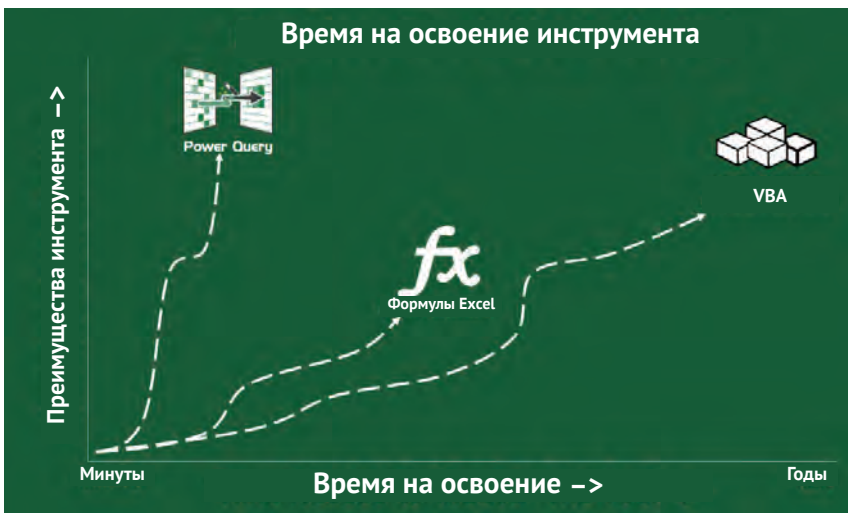


Рис. 0.3. Power Query задумывался как простое в использовании средство преобразования данных

Именно простотой использования Power Query мы объясняем повсеместное исчезновение с арены прежних волшебников, ловко манипулирующих данными при помощи старых средств. Даже если один из таких волшебников наворотит какое-нибудь сложное решение в Power Query, то после его ухода можно будет с гораздо меньшими усилиями найти того, кто с минимальными расходами на обучение сможет поддерживать и расширять прежний функционал. Затраты в этом случае будут исчисляться не неделями, а часами.

Каким бы странным это ни казалось профессиональным разработчикам Excel, но рядовым сотрудникам компании не доставляет удовольствия разбираться в сложных формулах этого программного пакета. Они хотят просто открыть инструмент, подключиться с его помощью к источнику данных, нажать пару кнопок, чтобы очистить набор данных, загрузить результат в программу и построить необходимый им отчет. По этой причине Power Query пользуется большой популярностью не только у разработчиков, хорошо владеющих формулами в Excel. С такими богатыми возможностями интерфейса пользователю в большинстве случаев не придется запоминать никаких формул и функций для получения необходимого результата.

ИСПОЛЬЗОВАНИЕ СРЕДСТВ ПРЕОБРАЗОВАНИЯ

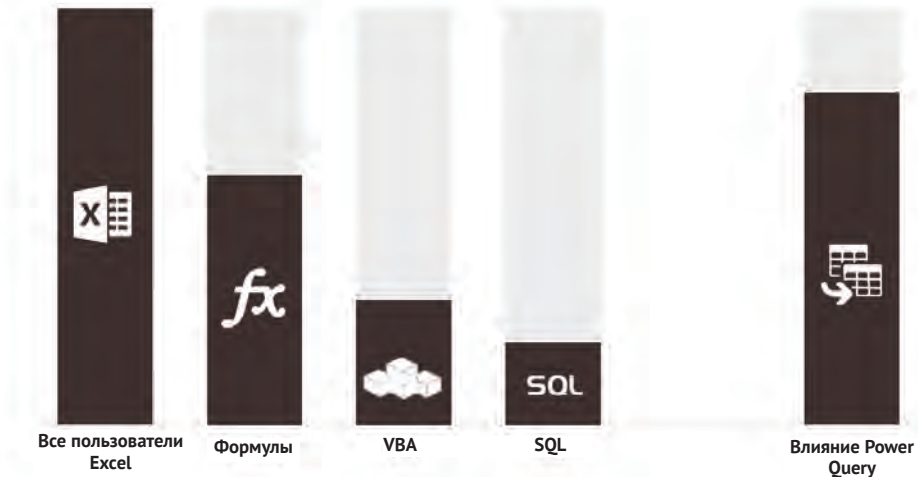


Рис. 0.4. Простота использования Power Query привлекает не только сторонников классических методов работы

У нас нет ни малейших сомнений в том, что Power Query полностью изменит подход в работе с данными всех пользователей Excel, включая продвинутых разработчиков. Мы хотим особо отметить, что ни в коем случае не принижаем значимости формул Excel, а также языков VBA и SQL. В конце концов, без этих инструментов не было бы и нас таких, какими мы являемся. Формулы до сих пор абсолютно незаменимы во многих аспектах работы с Excel, не касающихся преобразования данных, – Power Query вам здесь никак

не поможет. Знание VBA позволит пользователю реализовывать более сложные алгоритмы, включая подключение к другим приложениям и создание макросов для импорта и экспорта данных. Ну а запрос на языке SQL, написанный настоящим профессионалом, всегда будет давать фору по скорости выполнения запросу Power Query.

Однако что касается задачи подключения к источнику, импорта исходных данных, их очистки и преобразования, а также загрузки в Excel – тут у Power Query конкурентов нет, ведь только этот инструмент позволит полностью автоматизировать весь процесс с минимальными затратами времени. А с учетом регулярных обновлений, выпускаемых командой Power Query, разрывы в эффективности между запросами SQL и Power Query постоянно и стремительно уменьшаются.

Также стоит помнить, что Power Query относится не только к Excel. В прошлом, если бы вы построили систему преобразования и загрузки данных в Excel, она навсегда осталась бы в этой среде, а для ее переноса на другую платформу потребовалось бы полностью ее переписывать. В то же время в основе Power Query изначально заложены возможности переноса и масштабируемости ваших решений. Таким образом, одно и то же технологическое решение, реализованное посредством Power Query, будет работать как в Excel, так и в Power BI Desktop, Power Automate и Power BI Dataflows. Это означает, что процесс преобразования данных, выполненный при помощи Power Query и проверенный в Excel, может быть с легкостью перенесен в Power BI Desktop или Power BI Dataflows без потери функционала.

Помимо создания масштабируемых и универсальных решений, Power Query, по сути, дает вам возможность освоить расширяемую технологию, и вы сможете применить полученные знания на любых платформах. А самое главное, что команда Power Query в ближайшее время вряд ли остановится на пути расширяемости своего продукта.

Также стоит отметить, что подобные принципы интеграции позволяют разработчикам эффективно использовать несколько платформ одновременно. К примеру, вы можете при необходимости направлять запросы SQL непосредственно в Power Query, настраивать обновления посредством VBA в Excel или планировать их в Power BI, напрямую загружать запросы Power Query в модель данных или сущности и многое другое.

Почему Power Query – это магия?

Первоочередная задача, с которой сталкивается любой специалист по работе с данными при реализации надежных и устойчивых решений, – это доступ, очистка и преобразование исходных данных. Всем нам нужен был инструмент с загадочной аббревиатурой, о которой далеко не все слышали, – *ETL*.

И в виде такого инструмента ETL предстал Power Query. В его задачи как раз входит извлечение (Extract) данных почти из любых источников, преобразование (Transform) их в любой желаемый вид и загрузка (Load). Но что означают эти самые действия?



Рис. 0.5. Извлечение, преобразование и загрузка данных

Извлечение

Извлечение (extraction) данных, как мы сказали выше, может производиться практически из любых источников, включая текстовые файлы, файлы CSV, базы данных и веб-страницы. В дополнение команда Power Query разработала массу коннекторов, позволяющих получать данные из источников, из которых иначе извлечь информацию было бы очень проблематично. Это, например, такие источники данных, как Microsoft Exchange, Salesforce и разное программное обеспечение как услуга (Software As A Service – SAAS). Также существуют коннекторы *ODBC* и *OLEDB*, с помощью которых можно подключаться к базам данных, еще не охваченным разработчиками. Сегодня практически не важно, где находятся ваши исходные данные, – почти к любым из них вы сможете подключиться при помощи инструмента Power Query.

Преобразование

Говоря о *преобразовании* (transformation) данных, мы подразумеваем следующие операции:

- *очистка данных* (data cleansing) – может состоять в исключении информации о ненужных отделах, удалении пустых значений или строк. Также на этапе очистки может быть выполнено преобразование строк из верхнего регистра в нижний, разделение данных на колонки и конвертация дат в формат вашего региона. В общем, на стадии очистки происходит первоначальная подготовка данных для их использования;
- *интеграция данных* (data integration) – если вы используете в Excel функции *ВПР* (VLOOKUP), *ИНДЕКС* (INDEX), *ПОИСКПОЗ* (MATCH) или новую *ПРОСМОТРХ* (XLOOKUP), значит, вы знакомы с интеграцией нескольких наборов данных. Power Query умеет объединять данные как вертикально, так и горизонтально, позволяя вам *добавлять* (append) одну таблицу к другой, создавая длинную таблицу, или *объединять* (merge) таблицы по горизонтали без необходимости пользоваться функцией *ВПР* (VLOOKUP). Также на этом этапе вы можете выполнять и другие операции, включая группировку данных;
- *обогащение данных* (data enrichment) – на этой стадии преобразования данных вы можете добавлять новые столбцы и выполнять вычисления

с вашим набором данных. К примеру, вам может потребоваться вычислить общие суммы продаж, добавив столбец *Gross Sales*, в котором значения из колонки *Sales Quantity* будут умножены на значения из колонки *Sales Price*. Или вы захотите ввести новые форматы даты в соответствии с исходной колонкой в таблице транзакций. Все это можно сделать на этапе обогащения данных в Power Query. Фактически с помощью Power Query вы можете создавать новые динамические таблицы на базе значений на листах Excel, наборов данных SQL и даже содержимого веб-страниц. Нужна динамическая таблица с календарем за предыдущие пять лет? Power Query справится с этим легко и просто.

Что действительно впечатляет в Power Query, так это то, как много преобразований в нем можно выполнить, пользуясь одними лишь кнопками и меню в интерфейсе, – без необходимости писать код. Этот инструмент был разработан для конечных пользователей, и с помощью него действительно можно несколькими щелчками мыши выполнить преобразования, которые сложно было бы произвести даже посредством языков SQL или VBA. Это просто здорово!

Если же вы принадлежите к тем пользователям, которые любят докапываться до самой сути, крутить формулы и писать код, с Power Query вы также не соскучитесь. Хотя для того, чтобы комфортно работать с этим инструментом, нет необходимости изучать особый язык, этот язык все же есть, и с помощью него Power Query фиксирует все выполненные вами действия. Язык этот называется M, и мы любим шутить, что, видимо, буквы от A до L просто были заняты. Язык M позволит вам писать еще более эффективные запросы в Power Query и делать просто удивительные вещи.

Использовать язык M или нет – решать только вам. Вне зависимости от того, какой путь вы выберете, вы будете поражены тем, сколько всего в этом инструменте можно сделать, не используя навыки программирования.

Загрузка

Поскольку все программные пакеты, имеющие поддержку Power Query, обладают своим назначением, по расположению *загружаемых* данных они также будут различаться.

1. *Excel*: загрузка в таблицы Excel, модель данных Power Pivot или просто в виде подключения.
2. *Power BI*: загрузка в модель данных или в виде подключения.
3. *Power Automate* (ранее Microsoft Flow): загрузка в рабочие книги Excel (надемся, в будущем появятся и другие варианты загрузки).
4. *Потоки данных* (dataflows): загрузка в хранилище Azure Data Lake Storage, Dataverse или в виде подключения.

Фраза *в виде подключения* может кого-то сбивать с толку, но это означает лишь то, что мы можем создавать запросы, которые можно использовать в других запросах. Это позволяет реализовывать достаточно интересные подходы, о которых мы поговорим подробно далее в этой книге.

Хотя кому-то может быть интересно узнать, куда конкретно загружаются данные в том или ином случае, в целом это не самый важный момент в процессе работы инструмента ETL. Важнее то, **как** осуществляется загрузка, точнее даже то, как выполнить эту загрузку **снова**.

По своей сути Power Query можно назвать средством записи макросов, поскольку он тщательно фиксирует все действия, которые вы выполняете в процессе работы с данными. Таким образом, вы лишь раз определяете свой запрос и указываете, куда хотите его загрузить. После этого все, что вам нужно, – это просто обновлять свой запрос.



Рис. 0.6. Процесс объявления преобразований происходит один раз, дальше вы пользуетесь тем, что сделали

Вы только задумайтесь. Вы импортируете свой текстовый файл – тот самый, на загрузку и очистку данных в котором у вас уходит минут по 20 каждый месяц. Power Query позволит вам вдвое сократить это время в первый раз, а в дальнейшем новые файлы будут загружаться уже без вашего участия.

До сих пор вы тратили эти 20 минут, поражая Excel своими навыками работы в нем, и ежемесячно выполняли одни и те же заученные действия с файлами... Подождите, а вы уверены, что вам это нужно?

Мы же предлагаем вам просто перезаписывать старый текстовый файл и нажимать на кнопку обновления данных в Excel или Power BI. И все. Да нет, серьезно! А если вы публикуете свои данные в Power BI или настраиваете поток данных, вы можете создать автоматическое расписание, чтобы не делать даже этого!

В этом и заключается истинная мощь Power Query. Мало того, что его легко использовать, его очень легко использовать **повторно**! С его помощью вы можете превратить свой тяжелый труд в инвестицию в будущее и высвободить время для чего-то стоящего.

Возможности Power Query и интеграция с другими продуктами

Power Query – это технология, произведшая настоящую революцию в мире бизнес-аналитики. Официально она появилась в Excel в 2013 году в виде надстройки, а сегодня она полноценно входит в состав восьми различных

продуктов, включая Excel, Power BI Desktop, SQL Server Integration Services и Azure Data Factory. А когда вы будете читать эту книгу, возможно, интеграцией с этим инструментом смогут похвастаться и другие программные продукты для работы с данными от Microsoft.

Влияние Power Query просто феноменально – всего за несколько лет этот инструмент сумел буквально перевернуть жизни многих специалистов по работе с данными из самых разных областей. Недостатком такой высочайшей интеграции в разные продукты является то, что она никогда не проходит бесследно. И разработчикам Power Query приходится очень нелегко, балансируя между многочисленными функциями и возможностями своего инструмента с учетом связи со всеми этими продуктами, имеющими свои требования. В результате они вынуждены идти на многочисленные компромиссы в отношении функционала в зависимости от продукта интеграции.

Компоненты Power Query

Power Query можно сравнить с луковицей – у этого инструмента, как и у лука, есть чешуйки, представляющие собой ключевые компоненты Power Query.

Глядя на Power Query, мы почти буквально видим эти чешуйки. В процессе чтения данной книги вы будете понимать, как много всего происходит «под капотом» Power Query во время работы с этим инструментом. В частности, все это время за вас трудится язык M, видимый для пользователя, и внутренний движок Power Query, скрытый от наших глаз. На рис. 0.7 представлен Power Query в разрезе.



Рис. 0.7. Слои Power Query

Всего Power Query насчитывает три слоя, но в некоторых интеграциях могут участвовать только первые два. Эти слои:

- **движок M** (M Engine) – внутренний движок выполнения запросов, обрабатывающий инструкции, написанные на внутреннем языке Power Query – M;
- **запросы M** (M Query) – набор команд, доступных языку M;

○ **интерфейс пользователя Power Query** (Power Query User Interface) – также известен как *редактор Power Query* (Power Query Editor). Графический интерфейс, помогающий пользователю выполнять различные действия, в том числе:

- создавать и модифицировать запросы на языке M путем простого взаимодействия с интерфейсом;
- визуализировать запросы и их результаты;
- управлять запросами путем создания групп запросов, добавления метаданных и т. д.

Как минимум любая интеграция Power Query с тем или иным продуктом включает в себя два из перечисленных слоев: *движок M* и *запросы M*. В табл. 0.1 приведены все присутствующие компоненты в различных интеграциях.

Таблица 0.1. Не все интеграции Power Query включают в себя все слои Power Query

Компоненты	Движок M	Запросы M	Интерфейс пользователя Power Query
Excel	Да	Да	Да
Power BI Desktop	Да	Да	Да
Потоки данных Power BI	Да	Да	Да
SQL Server Integration Services	Да	Да	Нет

Если сравнивать возможности Power Query в Excel и возможности Power Query при работе с потоками данных Power BI в 2021 году, можно заметить некоторые различия. Потоки данных Power BI используют пользовательский интерфейс *Power Query Online*, тогда как в Excel и Power BI возможности базируются на интерфейсе *Power Query Desktop*. И хотя эти два интерфейса отличаются, процессы, лежащие в их основе, по большей мере схожи.

Если же вы проведете это сравнение в 2024 году, мы уверены, разница окажется не столь существенной. Причина этого в том, что команда разработчиков Power Query стремится к полной универсализации и унификации интерфейса инструмента во всех продуктах интеграции.

Конечно, в будущем у разных продуктов также могут сохраняться свои особенности при работе с Power Query. К примеру, импорт данных в таблицу непосредственно из активной рабочей книги может остаться прерогативой использования этого инструмента исключительно в Excel, но в целом интерфейсы заметно сближаются. Некоторые из существующих различий охватывают все три описанных выше слоя Power Query, тогда как другие касаются только интерфейса пользователя (как, например, различия в иконках в разных продуктах).

В последние годы компания Microsoft вкладывает огромные средства в развитие инструментария интерфейса Power Query Online. По окончании тестирования внедренного решения оно переносится в предварительную

версию выпуска Power Query Desktop, а затем следует официальное обновление этого интерфейса. Это означает, что если вы хотите воспользоваться всеми новейшими средствами и возможностями Power Query, вам следует обратиться к продукту, взаимодействующему с Power Query с помощью интерфейса Online, – например, к потокам данных Power BI.

Ни для кого не секрет, как быстро и стремительно развивается Power Query – как в плане возможностей, так и в отношении пользовательского интерфейса. В связи с этим мы отдаем себе отчет в том, что писать книгу по Power Query и надеяться, что скриншоты в ней не устареют, было бы очень странно. К примеру, выход этой книги был отложен на целых два года в ожидании того, когда одна важная возможность появится в интерфейсе Power Query, доступном в Excel.

И хотя в книге мы будем постоянно приводить пошаговые описания тех или иных действий, мы обязаны предостеречь вас на случай того, если в вашем случае внешне та или иная процедура будет выглядеть несколько иначе по причине вышедших обновлений. Неизменно будет одно – рецепты и подходы, лежащие в основе примеров. Это основы, которым мы и собираемся вас научить, – как обращаться с данными вне зависимости от имеющегося интерфейса пользователя. И в этом отношении мы надеемся, что наша книга будет полезна пользователям и разработчикам еще не один год.

Цикл обновлений Power Query

Перед тем как начать говорить о том, как добраться до Power Query, давайте пару слов скажем об *обновлениях* этого инструмента. Вам может показаться, что мы бежим впереди паровоза, но на то есть свои причины.

Команда Power Query выпускает обновления ежемесячно. Мы говорим не об исправлениях ошибок (хотя они тоже, конечно, присутствуют), а о новых возможностях и расширениях. Некоторые обновления незначительные, другие довольно весомые. В начале 2015 года, помнится, разработчики выпустили обновление, сократившее время загрузки запросов на 30 %. В июле того же года они пофиксили серьезные проблемы с обновлением данных в Power Pivot. В последующие годы были добавлены типы объединения таблиц, условные столбцы и многое другое. Что касается последних трех лет, то здесь можно вспомнить обновления, в которых появились *добавление столбца из примеров* (Columns from Example), *нечеткие соответствия* (Fuzzy Matching) и другие возможности.

Так как же получать эти обновления? Здесь все зависит от того, в каком именно продукте вы работаете с Power Query.

Power Query Online

Интерфейс Power Query Online используется при обращении к Power Query онлайн в таких продуктах, как Power Automate, и при работе с потоками данных Power BI. Это все веб-службы, и обновлять в них вам ничего не требуется. Все новые возможности в этом случае устанавливаются автоматически, а вам просто нужно следить за новинками.

Microsoft 365

Мы предпочитаем пользоваться Excel и другими продуктами семейства Office по подписке *Microsoft 365*. Если у вас есть подписка, все программное обеспечение будет обновляться автоматически в соответствии с выбранным каналом обновления (Channel) для вашей версии продукта.



Подробнее о каналах обновления можно почитать по адресу <https://docs.microsoft.com/ru-ru/deployoffice/overview-update-channels>.

Excel 2016/2019/2021

Как мы уже отметили, Power Query не стоит на месте, а постоянно развивается. Если вспомнить Excel 2016, официально вышедший в сентябре 2015 года, то в нем Power Query впервые был интегрирован на уровне базовой установки. Однако изначально в нем присутствовали ошибки при работе с именованными диапазонами, а также использовался устаревший алгоритм объединения файлов. Кроме того, в этой версии не были представлены типы объединения, условные столбцы и столбцы из примеров.

Хорошие новости состоят в том, что хоть Excel 2016 и 2019 не входят в подписку Microsoft, без обновлений Power Query не остается. И мы настоятельно рекомендуем вам установить последнюю доступную версию продукта перед продолжением чтения этой книги.

Секрет получения этих обновлений кроется в том, чтобы Windows при загрузке очередных новинок обновляла все, что нам нужно. Добиться этого в Windows 10 можно следующим образом:

- нажмите на кнопку Windows и введите *Windows*;
- выберите пункт **Параметры центра обновления Windows** (Windows Update Settings);
- перейдите в раздел **Дополнительные параметры** (Advanced Options);
- убедитесь, что установлен флажок **При обновлении Windows получать обновления для других продуктов Майкрософт** (Give me updates for other Microsoft products when I update Windows).

Excel 2010 & 2013

Для версий Excel ниже 2016 Power Query необходимо скачать и установить вручную с адреса <https://go.microsoft.com/fwlink/?LinkId=317450>. Последнее обновление для Excel 2010 и 2013 было выпущено в марте 2019 года.

Power BI Desktop

В *Power BI Desktop* предусмотрено два механизма поставки. Если вы установили этот пакет посредством *Microsoft Store*, он будет обновляться автоматически. Если же Power BI Desktop был установлен по ссылке **Дополнительные параметры скачивания** (Advanced download options) по адресу <https://>

powerbi.microsoft.com/ru-ru/downloads, вам необходимо будет вручную скачивать и устанавливать обновления.



Примечание. Пользователям Power BI повезло больше остальных – они первыми получают новинки Power Query. Информация об обновлениях скрывается в окне **Параметры** (Options) на вкладке **Предварительные версии возможностей** (Preview Features). Если вы хотите узнать, какие обновления пойдут в Excel, вам сюда. Новые возможности обычно сначала появляются в Power BI Desktop, а затем, по приобретении статуса общего доступа через два-три месяца, отправляются в Excel.

Как использовать эту книгу

В первую очередь мы бы хотели, чтобы эта книга могла стать вашим главным практическим помощником в отношении Power Query в целом и языка M в частности – вне зависимости от того, являетесь вы новичком или профессионалом в области ETL. Наша цель – помочь читателю справиться с распространенными задачами, возникающими на практике ежедневно, и показать, как для их решения можно применять Power Query. Также мы рассмотрим множество продвинутых сценариев с использованием языка M, чтобы вы научились не только строить решения при помощи Power Query, но и знали, как сделать их надежными и устойчивыми.

Большая часть примеров и иллюстраций, используемых в этой книге, будут показаны применительно к версии Excel Microsoft 365. И если не указано обратное, все сценарии должны одинаково корректно работать и в Excel, и в Power BI.

Где найти Power Query?

Чтобы использовать в работе Power Query, нужно знать, где он находится.

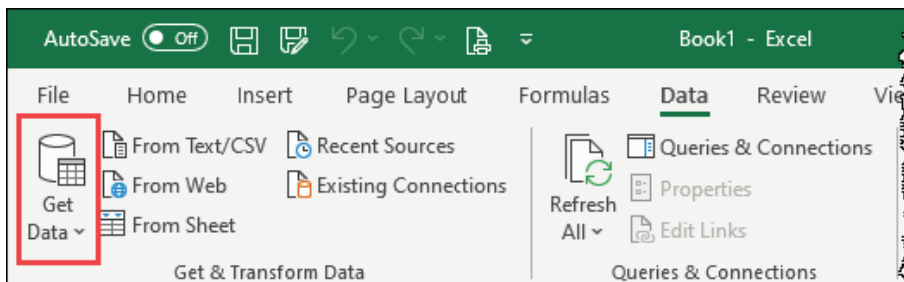


Рис. 0.8. Расположение Power Query в Excel

Excel 365

В версии Excel, выпущенной в рамках продукта Microsoft 365 (далее мы будем называть его просто Excel 365), доступ к командам Power Query можно по-

лучить в группе меню **Получить и преобразовать данные** (Get & Transform) на вкладке **Данные** (Data). Тогда как для распространенных источников данных предусмотрены отдельные кнопки, к полному списку источников можно получить доступ, нажав на кнопку **Получить данные** (Get Data).

Power BI Desktop

Работая с Power BI Desktop, для доступа к Power Query вам даже не придется покидать вкладку **Главная** (Home) – большая кнопка **Получить данные** (Get Data) будет прямо у вас перед глазами, не промахнетесь.

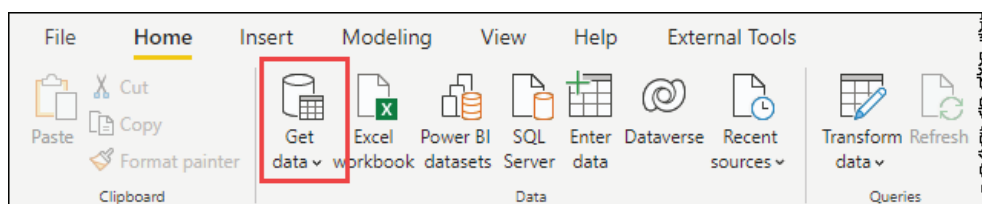


Рис. 0.9. Расположение Power Query в Power BI Desktop

Предыдущие версии Excel

Несмотря на то что в этой книге мы будем ориентироваться на Excel 365, большинство описываемых нами сценариев будут прекрасно работать и в более ранних версиях. Разве что доступ к командам Power Query везде будет осуществляться по-разному:

- **Excel 2019:** по большей части интерфейс Excel 2019 совпадает с Excel 365. Единственным заметным отличием на момент написания книги является то, что кнопка **Из листа** (From Sheet), которая будет появляться на наших скриншотах регулярно, в Excel 2019 называется **Из таблицы/диапазона** (From Table/Range);
- **Excel 2016:** как и в продуктах Office 2019/365, входной точкой в Power Query является вкладка **Данные** (Data), но сама кнопка называется **Создать запрос** (New Query) и расположена посередине вкладки **Данные**;
- **Excel 2010/2013:** в этих версиях Excel инструмент Power Query после загрузки и установки будет располагаться на отдельной вкладке. Когда вы увидите в этой книге, что вам необходимо нажать на кнопку **Получить данные** (Get Data) на вкладке **Данные** (Data), то переходите на вкладку **Power Query** и ищите соответствующую команду там.

Подключение к данным

С помощью Power Query вы можете подключаться к самым разным источникам данных, варианты которых можно найти, нажав на кнопку **Получить данные** (Get Data) на вкладке **Данные** (Data) в Excel или на вкладке **Главная**

(Home) в Power BI Desktop. Тогда как в Excel источники данных разбиты на подкатегории прямо в пунктах меню, в Power BI Desktop для просмотра иерархии необходимо нажать на кнопку **Другие** (More).

С целью соблюдения единообразия мы будем использовать следующую последовательность команд для открытия файла CSV:

- создайте новый запрос ⇒ **Из файла** ⇒ **Из файла Text/CSV**.

В Excel это действие будет осуществляться следующим образом:

- открыть вкладку **Данные** (Data) ⇒ нажать на кнопку **Получить данные** (Get Data) ⇒ **Из файла** (From File) ⇒ **Из текстового/CSV-файла** (From Text/CSV).

В Power BI Desktop действия будут такими:

- открыть вкладку **Главная** (Home) ⇒ нажать на кнопку **Получить данные** (Get Data) ⇒ **Другие** (More) ⇒ **Файл** (File) ⇒ **Текстовый или CSV-файл** (Text/CSV).

Если вы по-прежнему используете Excel 2016 или более ранней версии, последовательность будет следующей:

- Excel 2016: открыть вкладку **Данные** (Data) ⇒ нажать на кнопку **Создать запрос** (New Query) ⇒ **Из файла** (From File) ⇒ **Из текстового/CSV-файла** (From Text/CSV);
- Excel 2010/2013: открыть вкладку **Power Query** ⇒ **Из файла** (From File) ⇒ **Из текстового/CSV-файла** (From Text/CSV).

Особые пометки



Примечание. Примечания в книге будут выделяться абзацами с соответствующей иконкой. В них я буду рассказывать о каких-то полезных возможностях и трюках, которые помогут вам более эффективно использовать Power Query.



Предупреждение. Предупреждения также будут выделяться абзацами, и в них я буду подмечать моменты, которые в будущем могут приводить к проблемам. Пропускать предупреждения вы можете только на свой страх и риск.

Сопроводительные файлы

Перед тем как двигаться дальше, я настоятельно рекомендую вам скачать все рабочие книги по указанному ниже адресу, чтобы следить за происходящим и параллельно пробовать все самим: <https://www.skillwave.training/book-master-your-data-examples>.

Ну что ж, пришло время познакомиться с этим потрясающим инструментом поближе! Приступим!

Глава 1

Основы Power Query

Главная задача Power Query – собирать и преобразовывать исходные данные в желаемый формат, после чего загружать их в таблицы для нужд бизнес-аналитики. В простейшем виде процесс, который Power Query будет стараться выполнить без вашего вмешательства, можно представить так, как показано на рис. 1.1.



Рис. 1.1. Общие принципы работы Power Query

Разумеется, мы можем вмешиваться в этот процесс на любой его стадии. И именно это мы будем делать на протяжении всей книги. Но для начала полезно будет пройти по всем пунктам и понять, что на той или иной стадии делает Power Query.

Перед началом

Прежде чем отправляться в путешествие по Power Query, мы советуем вам кое-что изменить в интерфейсе. Для чего? Дело в том, что компания Microsoft по умолчанию отключила некоторые полезные опции, чтобы не перегружать ваше сознание, но, к сожалению, некоторые из них критически важны для полноценной работы с этим инструментом. А раз вы держите в руках данную книгу, значит, вы хотите научиться правильно обращаться с Power Query.

Изменение настроек Power Query по умолчанию в Excel

Добраться до *настроек инструмента Power Query* в Excel можно следующим образом:

- перейдите на вкладку **Данные** (Data), раскройте выпадающую кнопку **Получить данные** (Get Data) и выберите пункт **Параметры запроса** (Query Options);
- в разделе **Глобальные** (Global) откройте секцию **Загрузка данных** (Data Load) и убедитесь, что флажок **Быстрая загрузка данных** (Fast Data Load) установлен. Эта настройка приведет к блокированию пользовательского интерфейса Excel во время обновления данных, но позволит обеспечить актуальность информации при работе;
- в разделе **Глобальные** (Global) откройте секцию **Редактор Power Query** (Power Query Editor) и убедитесь, что все флажки установлены. Самым важным здесь является флажок **Отобразить строку формул** (Formula Bar is showing), но остальные опции также имеют большое значение, и в данной книге мы будем предполагать, что они у вас установлены;
- нажмите на кнопку **ОК**.

В этом диалоговом окне есть и другие интересные настройки, но для начала хватит и этих.

Изменение настроек Power Query по умолчанию в Power BI

Чтобы изменить настройки по умолчанию в Power BI Desktop, сделайте следующее:

- на вкладке **Файл** (File) откройте раздел **Параметры и настройки** (Options & settings) и выберите пункт **Параметры** (Options);
- в разделе **Глобальные** (Global) откройте секцию **Редактор Power Query** (Power Query Editor) и убедитесь, что все флажки установлены. В особенности нас интересует пункт **Отобразить строку формул** (Formula Bar is showing), но остальные опции также имеют значение;
- нажмите на кнопку **ОК**.



Примечание. Находясь в окне настроек в Power BI Desktop, вы можете также открыть секцию **Предварительные версии возможностей** (Preview Features) в разделе **Глобальные** (Global) и посмотреть список новых опций. Поскольку все новинки сначала появляются именно в Power BI Desktop, здесь можно увидеть, какими возможностями в ближайшем будущем может пополниться Power Query в Excel.

Извлечение

В данной главе мы будем импортировать простые файлы CSV посредством Power Query в Excel и Power BI для демонстрации возможностей Power Query, его интерфейса и различий между двумя платформами.

Процесс *ETL* всегда начинается с этапа извлечения данных, который, в свою очередь, делится на четыре стадии, показанные на рис. 1.2.



Рис. 1.2. Четыре составляющие процесса извлечения данных

Настройки подключения (выбор данных)

На первом шаге необходимо выбрать и настроить *подключение к данным*, с которыми мы собираемся работать. Давайте создадим запрос, использующий простейший коннектор к файлу CSV в Excel:

- нажмите на выпадающую кнопку **Получить данные** (Get Data) и в меню **Из файла** (From File) выберите пункт **Из текстового/CSV-файла** (From Text/CSV).

В Power BI Desktop аналогичные действия выполняются следующим образом:

- нажмите на кнопку **Получить данные** (Get Data), выберите пункт **Другие** (More) и в разделе **Файл** (File) укажите вариант **Текстовый или CSV-файл** (From Text/CSV).

Стоит заметить, что импортировать текстовые или CSV-файлы можно и гораздо проще. Поскольку к таким файлам аналитики обращаются достаточно часто, разработчики предусмотрели более быстрый способ для их загрузки. В Excel вы увидите соответствующую кнопку непосредственно справа от кнопки **Получить данные** (Get Data) на вкладке **Данные** (Data). А в Power BI нужный вам коннектор располагается на первом уровне вложенности в выпадающей кнопке **Получить данные** (Get Data), так что вам нет необходимости добираться до него через пункт **Другие** (More). Но мы упоминаем полный путь к подключению, поскольку далее в этой книге будем работать с самыми разными источниками данных.

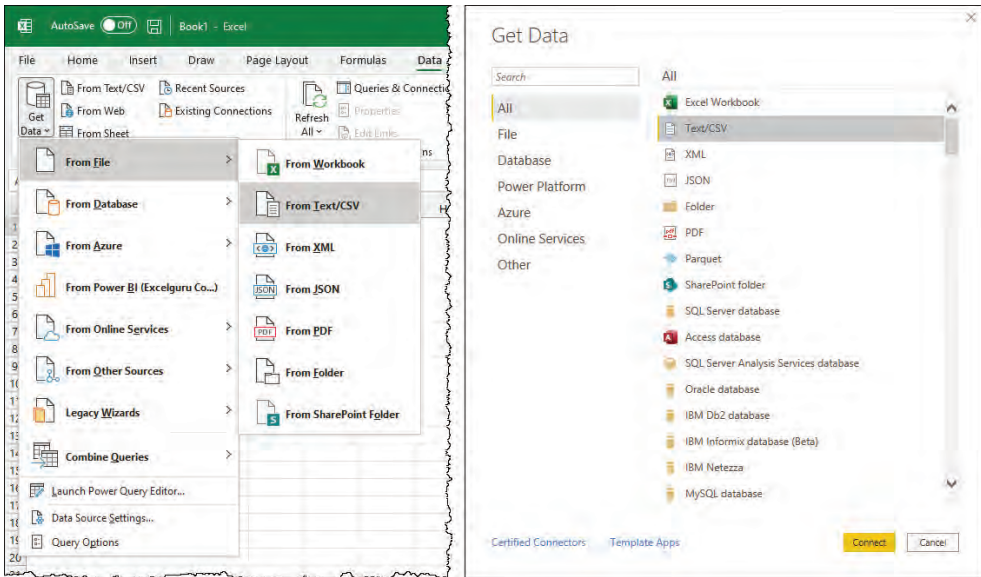


Рис. 1.3. Подключение к текстовому/CSV-файлу в Excel (слева) и Power BI Desktop (справа)



Примечание. В Power BI Desktop предусмотрена возможность подключения к большему количеству источников данных по сравнению с Excel. По установившейся традиции разработчики стремятся добавлять тестовые коннекторы сначала в Power BI, а после прохождения полной проверки переносят их в Excel.

Теперь, когда мы нашли нужный тип подключения, пришло время выбрать источник данных в виде файла. В нашем случае откроем следующий файл: *Ch01 Examples\Basic Import.csv*.

Аутентификация

Многие источники данных требуют дополнительной *аутентификации* (authentication) перед подключением. Если ваш источник данных относится к подобному случаю, на этом этапе вы получите запрос на ввод личных данных. К счастью, мы имеем дело с файлом, расположенным локально на вашем компьютере, к которому у вас есть полный доступ.

Предварительный просмотр

После выбора файла откроется *окно*, похожее на то, что показано на рис. 1.4.

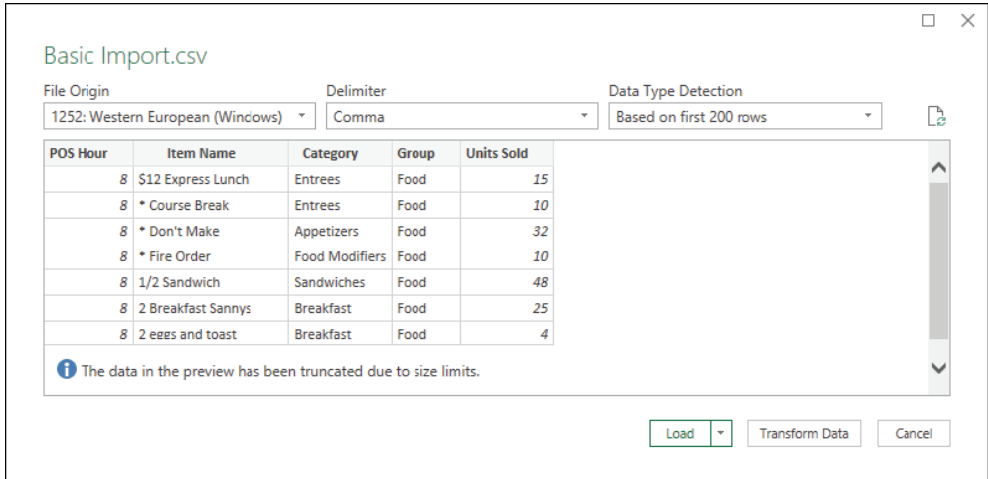


Рис. 1.4. Окно предварительного просмотра в Power Query

Целью является показать вам, как Power Query видит исходные данные, чтобы вы могли внести необходимые коррективы перед началом процесса преобразования. Обычно на этом этапе мало что требуется менять, поскольку Power Query в абсолютном большинстве случаев делает правильные предположения о характере данных. Несмотря на это, в верхней части окна расположены следующие выпадающие списки для настройки данных:

- **источник файла** (File Origin). Здесь вы можете выбрать кодировку исходного файла. Вам редко нужно будет пользоваться этой опцией;
- **разделитель** (Delimiter). В этом списке вам также очень редко понадобится менять настройки, поскольку в большинстве случаев Power Query прекрасно справляется с определением разделителя. При необходимости вы можете выбрать разделитель из списка, установить пользовательский разделитель и даже задать фиксированную длину столбцов;
- **обнаружение типов данных** (Data Type Detection). Здесь вы можете задать способ определения типов исходных данных на основе первых 200 строк или всего набора данных. Также есть вариант вовсе не определять типы данных.

Очень важно обратить внимание на информационное сообщение о том, что данные в предварительном просмотре были усечены из-за ограничения размера.

Выбор назначения запроса

Чаще всего вам не придется вносить какие-то изменения в окне предварительного просмотра. Главная цель этого окна – дать вам понять, в каком виде представлены данные, и ответить на три следующих вопроса.

1. Это неправильный набор данных? В этом случае вы всегда можете нажать кнопку **Отмена** (Cancel).
2. С данными все в порядке? Тогда нажимайте кнопку **Загрузить** (Load).
3. Данные нуждаются в трансформации или обогащении? Тогда ваш выбор – кнопка **Преобразовать данные** (Transform Data).



Примечание. По нашим предположениям, около 80–90 % данных нуждаются в преобразовании перед их использованием. Сложность преобразований может варьироваться от самых элементарных (вроде изменения названий столбцов) до действительно серьезных. Вне зависимости от того, что вам требуется, вашим выбором по умолчанию всегда должна быть кнопка **Преобразовать данные**, а не **Загрузить**.

Теперь, когда вы увидели предварительный снимок данных и решили, что это именно та информация, которая вам требуется, нажмите на кнопку **Преобразовать данные**, чтобы перейти в Power Query. Редактор откроется в новом окне, как показано на рис. 1.5.

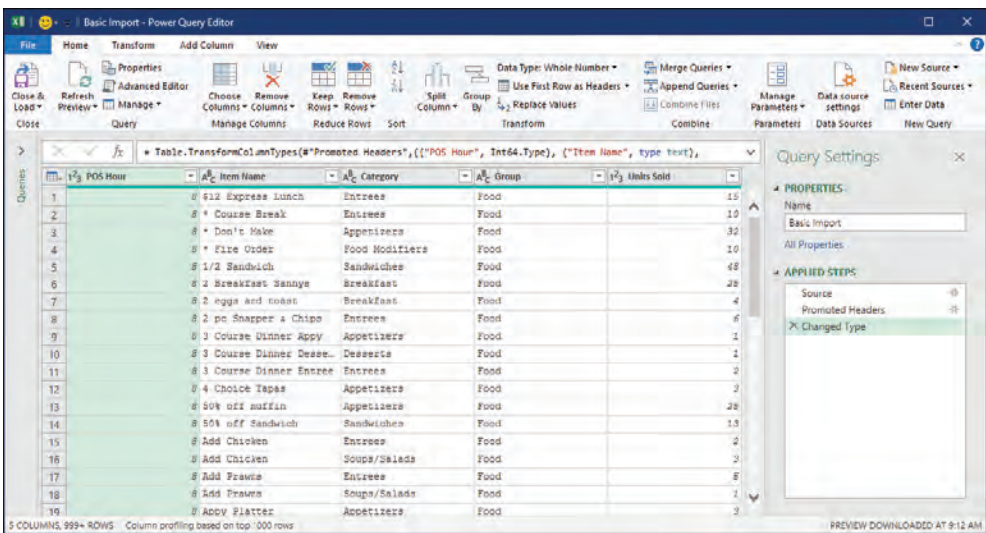


Рис. 1.5. Редактор Power Query в Excel

Преобразование

Следующим этапом процесса ETL является преобразование данных. В отличие от классических методов импорта данных в Excel, Power Query позволяет вам видеть и модифицировать преобразования, выполненные системой по умолчанию. Эти изменения можно делать в окне редактора Power Query, которое открывается после нажатия на кнопку **Преобразовать данные**.

Редактор Power Query

В редакторе Power Query присутствуют семь главных областей, к которым мы будем обращаться на протяжении всей книги. На рис. 1.6 они помечены цифрами.

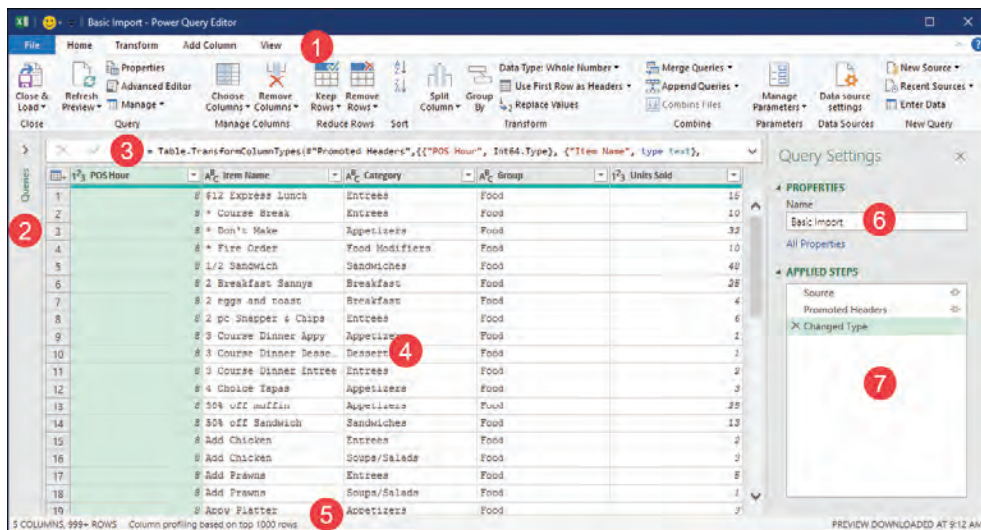


Рис. 1.6. Семь областей редактора Power Query

Предназначение областей в редакторе Power Query.

1. **Лента (Ribbon)**. Располагается в верхней части окна. Лента Power Query делится на четыре пункта меню: **Главная (Home)**, **Преобразование (Transform)**, **Добавление столбца (Add Column)** и **Просмотр (View)**.
2. **Панель навигатора запросов (Query Navigator Pane)**. В версиях Excel до 365 эта панель по умолчанию была свернута. Вы всегда можете нажать на кнопку со стрелкой выше слова **Запросы (Queries)**, чтобы раскрыть или скрыть эту панель. Обратите внимание, что в Excel 365 и Power BI панель навигатора по умолчанию раскрыта, и вы можете скрыть ее, нажав на ту же кнопку со стрелкой.
3. **Строка формул (Formula Bar)**. Если вы не видите эту область, значит, не выполнили наши рекомендации по настройке Power Query. Строка формул очень важна при работе с данными, так что мы настоятельно рекомендуем вам установить соответствующий флажок на вкладке **Просмотр (View)**.
4. **Окно текущего представления (Current View Window)**. Это ваше главное рабочее окно для преобразования данных и просмотра результатов. Также здесь могут отображаться схема данных и диаграмма.
5. **Строка состояния (Status Bar)**. Располагается в нижней части окна и содержит краткую сводку о количестве строк и столбцов в запросе,

а также о числе строк, на основании которых выполняется профилирование столбцов. В правой части строки состояния имеется информация о времени последнего обновления предварительного просмотра.

6. **Окно свойств** (Properties Window). В этой области содержится информация об имени запроса, которое наследуется из источника данных.
7. **Панель примененных шагов** (Applied Steps Window). В вашем путешествии по Power Query эта область станет одной из важнейших, поскольку в ней отображаются все шаги преобразований, примененные к предварительному просмотру, которые после импорта будут распространены на весь набор данных в целом.

Преобразования по умолчанию

После первоначального извлечения данных из файла бывает полезно узнать, какие действия *Power Query* выполнил по умолчанию. Для этого обратите внимание на панель **Примененные шаги** в правой части окна. Вы увидите, что три действия уже выполнены.

1. *Источник* (Source).
2. *Повышенные заголовки* (Promoted Headers).
3. *Измененный тип* (Changed Type).

Важно знать, что каждый шаг на этой панели можно выделить и посмотреть подробно. Давайте сделаем это.

Источник (Source)

По умолчанию первый шаг каждого загруженного запроса будет называться *Источник* (Source) – вне зависимости от источника данных. Выбор этого шага на панели **Примененные шаги** приведет к преобразованию окна предварительного просмотра, чтобы вы видели исходные данные в том виде, как видит их Power Query. Этот вид показан на рис. 1.7.

	A ^B _C Column1	A ^B _C Column2	A ^B _C Column3	A ^B _C Column4	A ^B _C Column5
1	POS Hour	Item Name	Category	Group	Units Sold
2	8	\$12 Express Lunch	Entrees	Food	15
3	8	* Course Break	Entrees	Food	10
4	8	* Don't Make	Appetizers	Food	32
5	8	* Fire Order	Food Modifiers	Food	10
6	8	1/2 Sandwich	Sandwiches	Food	48
7	8	3 Breakfast Sannys	Breakfast	Food	25

Рис. 1.7. Визуальное представление шага *Источник* (Source)

На этом этапе процесса ETL Power Query идентифицировал разделители-запятые в исходных данных, но больше никаких преобразований не выполнял.

При этом внутренние алгоритмы Power Query уже определили некоторые несоответствия в данных – к примеру, данные в первой строке сильно отличаются от дальнейших значений. Это очень напоминает... заголовки!

Повышенные заголовки (Promoted Headers)

Если выделить шаг *Повышенные заголовки* (Promoted Headers) на панели примененных шагов, вы увидите результат предположения, о котором мы сказали выше. Power Query взял значения из первой строки и заменил ими *Column1*, *Column2* и т. д. в первоначальных заголовках, что видно по рис. 1.8.

	A ^B _C POS Hour	A ^B _C Item Name	A ^B _C Category	A ^B _C Group	A ^B _C Units Sold
1	8	412 Express Lunch	Entrees	Food	15
2	8	* Course Break	Entrees	Food	10
3	8	* Don't Make	Appetizers	Food	32
4	8	* Fire Order	Food Modifiers	Food	10
5	8	1/2 Sandwich	Sandwiches	Food	48
6	8	2 Breakfast Sannys	Breakfast	Food	25
7	8	2 eggs and toast	Breakfast	Food	4

Рис. 1.8. Результат выполнения шага повышения заголовков

Измененный тип (Changed Type)

Заключительный шаг, который выполнил Power Query по умолчанию, называется *Измененный тип* (Changed Type), и результат его выполнения показан на рис. 1.9.

	1 ² ₃ POS Hour	A ^B _C Item Name	A ^B _C Category	A ^B _C Group	1 ² ₃ Units Sold
1	8	412 Express Lunch	Entrees	Food	15
2	8	* Course Break	Entrees	Food	10
3	8	* Don't Make	Appetizers	Food	32
4	8	* Fire Order	Food Modifiers	Food	10
5	8	1/2 Sandwich	Sandwiches	Food	48
6	8	2 Breakfast Sannys	Breakfast	Food	25
7	8	2 eggs and toast	Breakfast	Food	4

Рис. 1.9. В заголовках столбцов виден результат преобразования типов данных

Как действовал Power Query? Он просканировал первые 200 строк в каждом столбце и сделал определенные предположения о том, данные каких типов в них содержатся. После этого он закрепил определенные типы данных за столбцами, чтобы в таком виде информацию загрузить в место назначения. Наиболее популярные *типы данных*, которые вы будете встречать в Power Query:

- **Дата и время** (DateTime): отображается при помощи иконки с календарем и часами;
- **Целое число** (Whole number): этому типу соответствует иконка «123»;
- **Десятичное число** (Decimal number): обозначается иконкой «1.2»;
- **Текст** (Text): для этого типа используется иконка «ABC».



Примечание. В Power Query есть и другие типы данных, но мы будем говорить о них подробнее в следующих главах.

Создание и изменение преобразований

До сих пор мы видели, что Power Query помог нам выполнить первоначальное преобразование данных и ни разу не ошибся. А что, если мы захотим внести изменения в извлеченные данные?

Давайте начнем с *удаления столбца*, который нам не нужен, – *POS Hour*. Мы не собираемся анализировать представленные данные на этом уровне детализации. Для этого вы можете пойти одним из следующих путей.

1. Выделите столбец *POS Hour*, щелкните правой кнопкой мыши по его заголовку и выберите пункт **Удалить** (Remove).
2. Выделите столбец *POS Hour* и нажмите на клавишу **Delete** на клавиатуре.

Какой бы способ вы ни выбрали, столбец исчезнет из таблицы, а на панели **Примененные шаги** появится новый шаг с именем *Удаленные столбцы* (Removed Columns), как показано на рис. 1.10.

	A ^B C Item Name	A ^B C Category	A ^B C Group	1 ² 3 Units Sold	
1	\$12 Express Lunch	Entrees	Food	15	
2	* Course Break	Entrees	Food	10	
3	* Don't Make	Appetizers	Food	32	
4	* Fire Order	Food Modifiers	Food	10	
5	1/2 Sandwich	Sandwiches	Food	48	
6	2 Breakfast Sannys	Breakfast	Food	25	
7	2 eggs and toast	Breakfast	Food	4	
8	2 pc Snapper & Chips	Entrees	Food	6	
9	3 Course Dinner Appy	Appetizers	Food	1	
10	3 Course Dinner Desse	Desserts	Food	1	

On the right, the 'APPLIED STEPS' pane shows:

- PROPERTIES: Name (Basic Import), All Properties
- APPLIED STEPS: Source, Promoted Headers, Changed Type, **Removed Columns**

Рис. 1.10. Шаг **Удаленные столбцы** исключил из запроса колонку *POS Hour*

Но поостойте! А если мы передумаем и захотим вернуть удаленный столбец? Не проблема, достаточно просто удалить примененный шаг. Сейчас мы не будем этого делать, но если хотите попробовать, выделите последний шаг на панели **Примененные шаги** и нажмите на крестик слева от названия шага. Шаг тут же исчезнет, а колонка восстановит свое место в запросе. Вы можете воспринимать эту возможность как улучшенную *отмену действия* (Undo) в других приложениях. В отличие от отмены действия, которую невозможно выполнить после закрытия приложения, удалить шаг из панели примененных шагов вы можете всегда.

Но вернемся к модификации наших данных. Как насчет того, чтобы упростить названия колонок, начиная со столбца *Item Name*?

- щелкните правой кнопкой мыши на заголовке столбца *Item Name*, выберите пункт **Переименовать** (Rename) и введите новое имя *Item*.

На панели примененных шагов появится новый шаг с именем *Переименованные столбцы* (Renamed Columns). Вы уже понемногу начинаете привыкать к основному принципу работы Power Query: любое выполненное действие приводит к добавлению нового шага.



Примечание. Возможность отменить предыдущее действие очень важна, но у показанной здесь особенности есть и другое, не менее важное предназначение, состоящее в том, что вы можете щелкать на любые шаги, чтобы посмотреть, какие преобразования они выполняют. Power Query всегда работает с копией ваших данных, так что исходные данные вы никак не нарушите. Это дает вам возможность нажимать на все кнопки, чтобы посмотреть, что будет. Если не понравится результат, просто удалите шаг. Мы рекомендуем вам самостоятельно проделать это со всеми командами, которые вам незнакомы. При этом вы не только освоите новый функционал, но и научитесь сопоставлять описания шагов на панели **Примененные шаги** с действиями, которые они выполняют.

Такое поведение Power Query очень важно понимать. В отличие от традиционного Excel, в котором выполненные действия нигде не сохраняются, в Power Query ведется полный учет примененных трансформаций. По умолчанию каждое действие, которое вы выполните при помощи интерфейса пользователя, будет отражено на панели примененных шагов. И даже если вы не знаете, какой именно шаг привел к тому или иному изменению данных, вы по крайней мере можете видеть, какой тип действия выполняется на определенном шаге. А если выбрать предыдущий шаг, можно посмотреть, в каком состоянии находились данные перед тем, как действие было выполнено. После этого можно выделить интересующий вас шаг, чтобы посмотреть результат выполненного преобразования.

А что произойдет, если мы захотим переименовать еще один столбец? Снова добавится шаг? Давайте проверим. Как и в Excel, в Power Query можно практически любое действие выполнить несколькими способами. На этот раз давайте переименуем столбец следующим образом:

- дважды щелкните мышью по заголовку столбца *Units Sold*;
- измените текст заголовка на *Units*.

Обратите внимание, что столбец переименовался, но шаг на панели примененных шагов не добавился! У нас по-прежнему лишь один шаг с именем *Переименованные столбцы* (Renamed Columns), что видно по рис. 1.11.

	A ^B C Item	A ^B C Category	A ^B C Group	1 ² 3 Units
1	12 Express Lunch	Entrees	Food	15
2	* Course Break	Entrees	Food	10
3	* Don't Make	Appetizers	Food	32
4	* Fire Order	Food Modifiers	Food	10
5	1/2 Sandwich	Sandwiches	Food	48
6	2 Breakfast Sannys	Breakfast	Food	25
7	2 eggs and toast	Breakfast	Food	4
8	2 pc Snapper & Chips	Entrees	Food	6
9	3 Course Dinner Appy	Appetizers	Food	1
10	3 Course Dinner Desse...	Desserts	Food	1
11	3 Course Dinner Entree	Entrees	Food	2

PROPERTIES

Name

Basic Import

All Properties

APPLIED STEPS

- Source *
- Promoted Headers *
- Changed Type
- Removed Columns
- Renamed Columns**

Рис. 1.11. Два действия по переименованию столбцов объединились в один шаг

Заметьте, что любой способ переименования столбцов приведет к одному и тому же результату. При этом, если выполнить два схожих действия подряд, Power Query попытается объединить их в один шаг. Причина этого проста: это позволит не раздувать без необходимости список примененных шагов и сделает его более легким для чтения. Многие файлы требуют выполнения похожих действий, так что это весьма полезная особенность поведения инструмента.



Примечание. Конечно, при желании у этой особенности можно найти и негативное последствие. Скажем, что делать, если вы переименовали шесть столбцов подряд, а затем поняли, что один из них переименовывать не нужно было? Если удалить шаг, будут потеряны все произведенные изменения. В качестве альтернативы вы можете добавить шаг с восстановлением имени столбца, переименованного по ошибке. А можете и вручную подправить код на языке M. Как это сделать – мы будем изучать в данной книге позже.

Загрузка

На данном этапе у нас есть запрос, в котором выполнены следующие действия:

- осуществлено подключение к файлу CSV;
- повышены заголовки из первой строки и установлены типы данных;
- удалены ненужные столбцы;
- переименованы два столбца, чтобы их имена были более понятными.

Для этого набора данных произведенных действий вполне достаточно. Наши данные приведены в понятный табличный формат, очищены и готовы к обработке аналитиком. Пришло время завершить процесс с помощью загрузки данных.

Установка типов данных

Перед загрузкой данных всегда бывает полезно *переопределить типы данных* для всех столбцов в наборе. О причинах этого мы расскажем чуть позже, но хотим, чтобы у вас с самого начала вырабатывались правильные привычки при работе с Power Query. Фактически так делает сама Microsoft, и именно поэтому по умолчанию последним шагом в запросах Power Query присутствует действие по изменению типов данных.

И хотя вы можете щелкать отдельно по иконке с типом данных в заголовке каждого столбца в вашем наборе данных и менять его, скорее всего, это займет достаточно много времени. Лучше позволить Power Query самому попытаться определить все типы данных в запросе, а затем скорректировать его действия. Для этого выполните следующее:

- выделите любой столбец;
- нажмите сочетание клавиш **CTRL+A** для выделения всех колонок;
- перейдите на вкладку **Преобразование** (Transform) и нажмите на кнопку **Определить тип данных** (Detect Data Type).

В результате на панели примененных шагов появится новый шаг с именем *Измененный тип1* (Changed Type1), как показано на рис. 1.12.

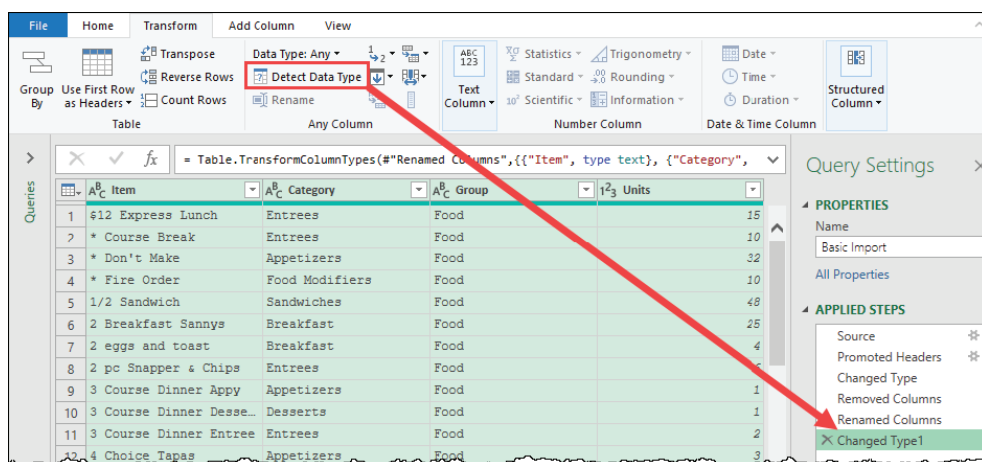


Рис. 1.12. Результат переопределения типов данных столбцов

Почему же *Измененный тип1*? Дело в том, что еще один шаг с изменением типов данных уже присутствует в списке примененных шагов. Этот шаг Power Query автоматически применил после повышения заголовков. Кстати, это помогает понять еще несколько важных принципов, по которым работает Power Query.

1. Каждый шаг в списке примененных шагов должен иметь уникальное имя.
2. При добавлении шага движок Power Query будет добавлять единицу или увеличивать на единицу имя последнего шага аналогичного типа.
3. Схожие шаги, выполненные подряд, будут объединяться в один шаг, а если между ними есть другие шаги, то не будут.

Обязаны ли вы оставлять то имя шага, которое было выбрано Power Query по умолчанию? Совсем нет. Хотя в основном мы советуем оставлять имена шагов по умолчанию и наблюдать за тем, какие команды интерфейса какие шаги создают, вы вправе *менять названия шагов* по своему усмотрению. Для этого сделайте следующее:

- щелкните правой кнопкой мыши по шагу, имя которого хотите изменить, и в контекстном меню выберите пункт **Переименовать** (Rename);
- введите новое имя шага – *Lock in Data Types*.



Примечание. Единственный шаг, который вам не удастся переименовать таким способом, – это *Источник* (Source). Для этого необходимо вручную редактировать код запроса на языке M.

Переименование запроса

По умолчанию Power Query устанавливает имя источника данных в качестве имени запроса. Но поскольку *Basic Import* – отнюдь не лучшее имя для запроса, вы всегда можете *переименовать его*. Для этого выполните следующие действия:

- перейдите на панель **Примененные шаги** (Query Settings Pane) и в разделе **Свойства** (Properties) выделите содержимое поля **Имя** (Name);
- введите новое имя запроса – *Transactions*.

Итоговый запрос должен выглядеть так, как показано на рис. 1.13.

	A ^B Item	A ^B Category	A ^B Group	T ³ Units
1	\$12 Express Lunch	Entrees	Food	15
2	* Course Break	Entrees	Food	10
3	* Don't Make	Appetizers	Food	32
4	* Fire Order	Food Modifiers	Food	10
5	1/2 Sandwich	Sandwiches	Food	48
6	2 Breakfast Sannys	Breakfast	Food	25
7	2 eggs and toast	Breakfast	Food	4
8	2 pc Snapper & Chips	Entrees	Food	6
9	3 Course Dinner Appy	Appetizers	Food	1
10	3 Course Dinner Desse...	Desserts	Food	1
11	3 Course Dinner Entree	Entrees	Food	2
12	4 Choice Tapas	Appetizers	Food	3

PROPERTIES

Name
Transactions

All Properties

APPLIED STEPS

Source ✖

Promoted Headers ✖

Changed Type

Removed Columns

Renamed Columns

Lock in Data Types

Рис. 1.13. Результат переименования запроса

Загрузка запроса в Excel

Теперь нам осталось *загрузить подготовленный запрос в Excel*. Для этого выполните следующие действия:

- перейдите на вкладку **Главная** (Home) в Power Query;
- нажмите на кнопку **Заккрыть и загрузить** (Close & Load).

В этот момент Power Query применит наши шаги не только к данным в окне предварительного просмотра, но и ко всему набору данных в источнике в целом. В зависимости от объема исходных данных и сложности преобразований это может занять некоторое время. По окончании загрузки вы увидите обработанные данные в таблице на новом рабочем листе, как показано на рис. 1.14.

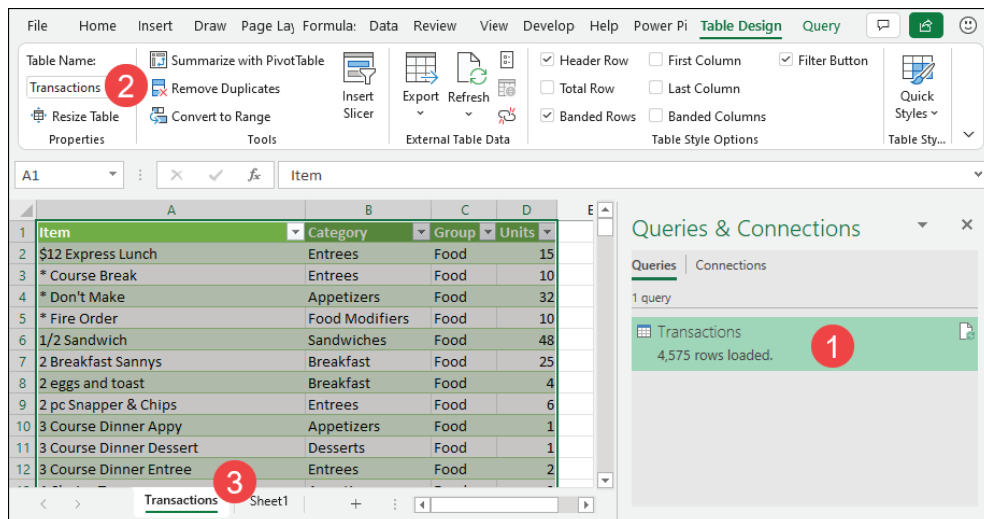



Рис. 1.14. Запрос Transactions, загруженный в Excel

На рис. 1.14 мы также поместили цифрами три важные области в интерфейсе Excel.

1. Панель **Запросы и подключения** (Queries & Connections). Запросы, перечисленные здесь, будут в точности соответствовать по именам запросам, подготовленным в Power Query.
2. Секция **Имя таблицы** (Table Name). Здесь также обычно будет содержаться имя запроса, но все недопустимые символы в нем будут заменены на символ подчеркивания (). Кроме того, в случае конфликтов с именами других листов в рабочей книге к имени запроса будет добавлено уникальное числовое окончание.
3. Имя рабочего листа. Здесь также обычно будет находиться имя запроса с символами подчеркивания, имя может быть усечено при необходимости, а в случае конфликтов с именами других листов к имени запроса будет добавлено уникальное числовое окончание в круглых скобках.

 **Примечание.** Каждый из перечисленных элементов может быть переименован, и называться они могут в итоге совершенно по-разному, это не повлияет на работоспособность книги.

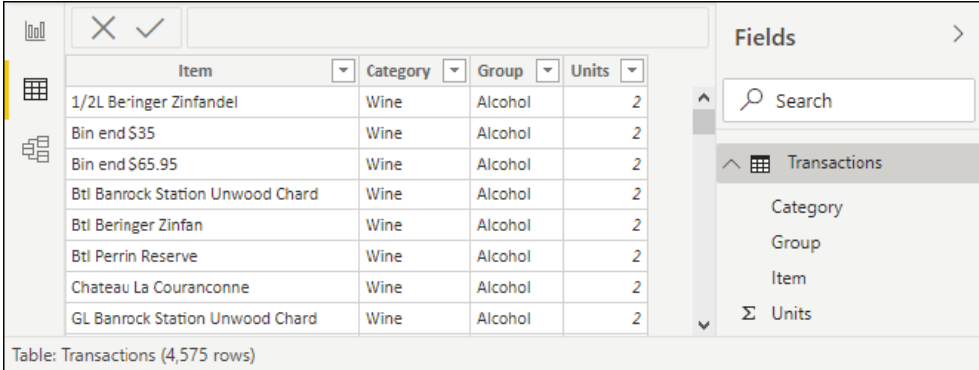
Загрузка запроса в Power BI

Единственным существенным отличием процесса *загрузки запроса в Power BI* по сравнению с Excel является название кнопки. Итак, для загрузки данных в Power BI сделайте следующее:

- перейдите на вкладку **Главная** (Home) в Power Query;
- нажмите на кнопку **Закорыть и применить** (Close & Apply).

Как и в случае с Excel, в этот момент Power Query применит шаги ко всему набору данных в источнике. Также при работе с Power BI данные будут загружены не на рабочий лист, а в модель данных. После загрузки вы увидите свою таблицу в следующих секциях:

- на панели **Поля** (Fields), расположенной в правой части окна;
- на левой вкладке **Данные** (Data), как показано на рис. 1.15;
- на вкладке **Модель** (Model).



Item	Category	Group	Units
1/2L Beringer Zinfandel	Wine	Alcohol	2
Bin end \$35	Wine	Alcohol	2
Bin end \$65.95	Wine	Alcohol	2
Btl Banrock Station Unwood Chard	Wine	Alcohol	2
Btl Beringer Zinfan	Wine	Alcohol	2
Btl Perrin Reserve	Wine	Alcohol	2
Chateau La Couranconne	Wine	Alcohol	2
GL Banrock Station Unwood Chard	Wine	Alcohol	2

Table: Transactions (4,575 rows)

Рис. 1.15. Таблица Transactions, загруженная в Power BI Desktop

Тогда как Excel показывает суммарное количество загруженных строк на панели **Запросы и подключения**, Power BI этого не делает. Эту информацию вы можете получить в Power BI, переключившись на вкладку **Данные** путем выбора таблицы из правой панели **Поля**. После этого в нижней левой части экрана появится информация о загруженных строках.



Примечание. В отличие от Excel, Power BI сортирует данные после их загрузки по первому столбцу. Чтобы скопировать такое поведение в Excel, вам придется явным образом добавить шаг с сортировкой перед загрузкой данных.

Обновление запросов

Со временем, узнавая Power Query все лучше и лучше, вы обнаружите, что при помощи этого инструмента можно обрабатывать и очищать исходные данные гораздо более эффективно по сравнению с классическими методами при работе в Excel. Но истинная магия Power Query проявляется тогда, когда вы понимаете, что можете *обновлять созданные ранее запросы* при изменении данных, поступающих на вход. Это позволит прогнать обновленные данные через все шаги преобразования и загрузить новые данные в место назначения. И что еще лучше – сделать это можно очень легко:

- в Excel: перейдите на вкладку **Данные** (Data) и нажмите на кнопку **Обновить все** (Refresh All);
- в Power BI: перейдите на вкладку **Главная** (Home) и нажмите на кнопку **Обновить** (Refresh).

После этого вам останется подождать, пока Power Query считает обновленные данные из файла, обработает их и загрузит в таблицу Excel или модель данных.

И если Power BI информирует пользователя о происходящем при помощи специального диалогового окна, то в Excel бывает трудно понять, что в текущий момент происходит. В строке состояния, находящейся в нижней части окна Excel, будет выведена соответствующая информация, но это не самый очевидный способ оповещения. Лучше всего ориентироваться на панель **Запросы и подключения**, на которой появится индикатор выполнения загрузки данных. На рис. 1.16 показано, как внешне выглядит процесс обновления информации из источника в Excel и Power BI соответственно.

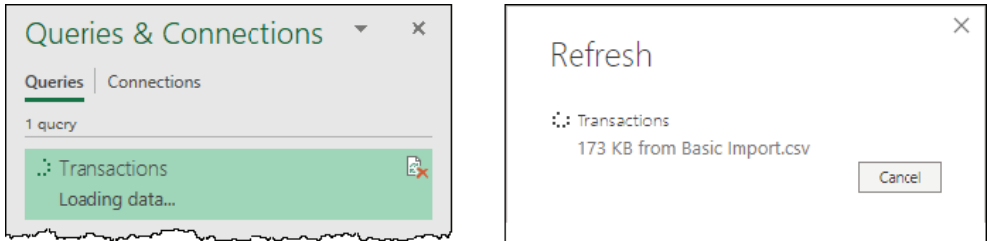


Рис. 1.16. Процесс загрузки данных в Excel (слева) и Power BI Desktop (справа)

После окончания загрузки Excel покажет в этом заголовке количество обработанных строк, а в Power BI, как мы уже сказали, эту информацию можно получить на вкладке **Данные**.

Этот метод прекрасно работает с файлами данных, которые обновляются на регулярной основе. Вне зависимости от того, является ли источник файлом Excel, обновляемым сразу несколькими людьми, или файлом CSV, который вы загружаете в конце месяца и кладете поверх старого, вы можете сделать полное обновление данных всего в один клик.

Редактирование запросов

Хотя обновлять данные одним нажатием мышки – это здорово, иногда требуется, например, перед обновлением изменить источник данных. Допустим, наш запрос был нацелен на обработку и загрузку данных из файла с названием *Jan.CSV*, в котором хранятся данные за январь. Но в следующем месяце вы получаете новый файл с именем *Feb.CSV*. Очевидно, что в этом случае одного щелчка мыши вам явно не хватит, поскольку это приведет к обновлению лишь январских данных и не затронет новую информацию,

содержащуюся в файле *Feb.CSV*. Что нам нужно сделать, так это изменить путь к файлу с исходными данными перед выполнением обновления. А это значит, что нам необходимо отредактировать имеющийся у нас запрос. Для этого нужно вернуться в редактор Power Query. А как это сделать – зависит от используемого вами инструмента.

Запуск редактора Power Query в Power BI

В Power BI открыть редактор Power Query проще простого. Все, что вам нужно, – это перейти на вкладку **Главная** (Home) и нажать на кнопку **Преобразование данных** (Transform Data), как показано на рис. 1.17. Это приведет к открытию окна редактора Power Query, в котором вы можете изменить существующие запросы и даже создать новые.

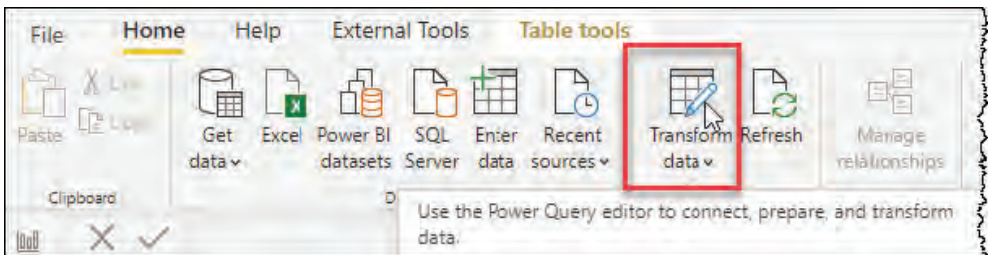


Рис. 1.17. Кнопка преобразования данных в Power BI

Запуск редактора Power Query в Excel

В Excel запустить редактор Power Query можно тремя способами, два из которых полагаются на активную панель **Запросы и подключения**. К сожалению, при запуске нового экземпляра Excel эту панель необходимо открывать вручную, и это может сбивать пользователей с толку. Поскольку в наше время львиная доля решений в Excel так или иначе связана с Power Query, первым делом после открытия Excel вы делаете активной эту панель. А сделать это можно четырьмя способами:

- перейти на вкладку **Данные** (Data) и нажать на кнопку **Запросы и подключения** (Queries & Connections);
- перейти на вкладку **Данные** (Data), нажать на кнопку **Получить данные** (Get Data) и выбрать пункт **Запустить редактор Power Query** (Launch Power Query Editor);
- открыть панель **Запросы и подключения** (Queries & Connections), щелкнуть правой кнопкой мыши по запросу и выбрать пункт **Изменить** (Edit);
- открыть панель **Запросы и подключения** (Queries & Connections) и дважды щелкнуть мышью по запросу.



Примечание. Поскольку мы в большинстве случаев предпочитаем оставлять панель **Запросы и подключения** открытой, чаще всего мы пользуемся последними двумя способами. Мы также любим шутить по этому поводу, говоря о том, что выбор способа зависит от того, хотите ли вы, чтобы на вашей мышке больше изнашивалась левая кнопка, или стремитесь к равномерному износу кнопок.

Просмотр шагов

Вернувшись в редактор Power Query, вы можете просмотреть любой из созданных шагов на панели **Примененные шаги** (Applied Steps). Выделяя шаг, вы будете видеть в области предварительного просмотра состояние данных на момент окончания действия выделенного шага.



Примечание. В окне предварительного просмотра используется технология кэширования данных. Если вы заметили, что данные в таблице утратили актуальность, или хотите убедиться, что в нее загружены последние сведения, вам необходимо вручную обновить информацию. Для этого можно нажать на кнопку **Обновить предварительный просмотр** (Refresh Preview) на вкладке **Главная** (Home) в редакторе Power Query, как показано на рис. 1.18.

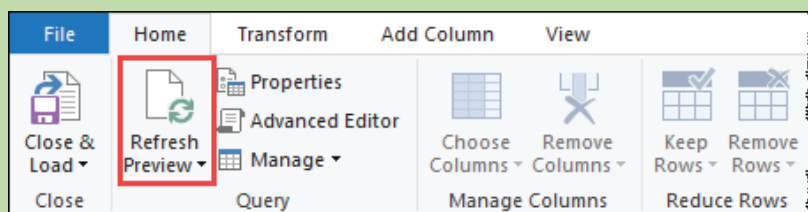


Рис. 1.18. Кнопка обновления предварительного просмотра в редакторе Power Query

Настройка шагов

После возвращения в редактор Power Query вы можете добавлять новые шаги, удалять существующие и даже редактировать шаги из состава запроса. Сейчас мы покажем, что нужно сделать, чтобы направить запрос на новый источник данных.



Предупреждение. Если вы откроете окончательные версии файлов для Excel или Power BI из этой главы (с пометкой *Complete*), то увидите, что обновление в них работать не будет. Причина в том, что шаг *Источник* (Source) в этих файлах указывает на исходный файл, находящийся на нашем компьютере. Следуя шагам из этого раздела, вы сможете выбрать в качестве источника собственные файлы.

Взгляните на шаги, представленные на рис. 1.19.

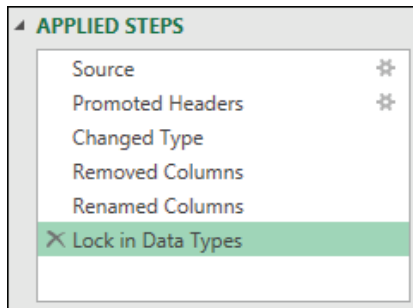



Рис. 1.19. Примененные шаги в запросе *Transactions*

На этом рисунке можно заметить кое-что очень важное, а именно *кнопки с шестеренками* справа от двух шагов, дающие возможность изменить шаги при помощи пользовательского интерфейса.

 **Примечание.** Как правило, если выполняемое действие в Power Query сопровождается показом диалогового окна, впоследствии созданный шаг будет отмечен шестеренкой, позволяющей настроить его параметры. Если действие не подразумевает ввода данных в диалоговом окне, шестеренка, скорее всего, не появится. Но из этого правила есть и исключения, и примером такого исключения является шаг с повышением заголовков.


Мы понимаем, что обращение к файлу с данными должно производиться где-то в начале списка шагов. К счастью, у первого шага *Источник* (Source) как раз есть шестеренка:

- выделите шаг *Источник* (Source);
- нажмите на кнопку с шестеренкой.

Откроется диалоговое окно, показанное на рис. 1.20, в котором вы можете настроить ключевые составляющие этого шага.

Нам необходимо изменить значение в первом поле диалогового окна – **Путь к файлу** (File Path). Давайте это сделаем:

- нажмите на кнопку **Обзор** (Browse);
- выберите файл *Ch01 Examples\New Data.csv*;
- нажмите на кнопку **ОК**, чтобы закрыть диалоговое окно.

 **Примечание.** Power Query прекрасно справился с задачей первоначальной настройки всех параметров этого шага, так что ничего другого нам менять не нужно. А что было бы, если бы он выбрал неправильный разделитель? Ничего страшного. В этом же окне настроек есть поле для указания разделителя, и вы можете изменить его при необходимости.

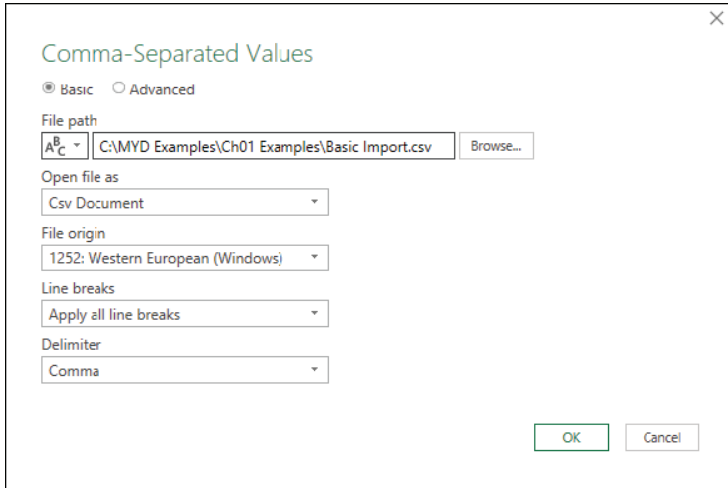



Рис. 1.20. Настройка шага *Источник*

Если данные в новом файле будут значительно отличаться, вы сразу заметите это в окне предварительного просмотра. Но в наших файлах данные похожи. Как в этом случае убедиться, что изменения вступили в силу? Дело усложняется еще и тем, что в наших файлах содержится более 999 строк, которые представлены в окне предварительного просмотра. Так что же делать? Загрузим данные!

 **Примечание.** Конечно, вы можете выделять каждый шаг в примененных шагах, чтобы убедиться, что все работает, но делать это нет необходимости. Поскольку наши данные обладают одинаковой структурой, все шаги будут применены без проблем, и не нужно проходить по ним всем.

- на вкладке **Главная** (Home) нажмите на кнопку **Закреть и загрузить** (Close & Load) для Excel или **Закреть и применить** (Close & Apply) для Power BI.

Данные будут загружены, и вы сможете проверить это в Excel на панели **Запросы и подключения**, как показано на рис. 1.21, или в Power BI на вкладке **Данные**.

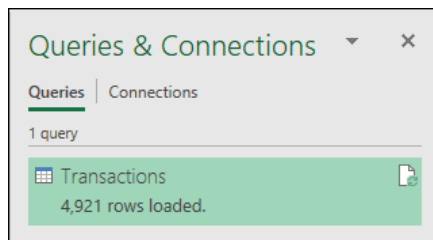


Рис. 1.21. Количество загруженных строк изменилось с 4575 на 4921

Влияние Power Query

Освоившись с Power Query, вы в какой-то момент поймете, что этот инструмент постепенно оказывает все большее влияние на ваш рабочий процесс. Давайте перечислим ключевые возможности этой технологии, известные нам на данный момент:

- с помощью Power Query можно подключаться к огромному количеству разнообразных источников данных;
- Power Query фиксирует все произведенные вами действия, создавая некое подобие скрипта;
- Power Query никогда не затрагивает исходные данные, что позволяет применять различные шаги и удалять их, если результат вас не устраивает;
- шаги в Power Query можно менять в будущем, когда исходные данные изменятся.

Как видите, возможности Power Query просто огромны. Представьте, что вы создали решение на основе скрипта Power Query по массовой очистке исходных данных и загрузке их на рабочий лист Excel. Данные в Excel в дальнейшем используются в качестве основы для отчетов и диаграмм. В прошлом при получении нового файла с информацией вам пришлось бы заново вручную производить очистку исходных данных, после чего копировать и вставлять их в таблицу.

С приходом Power Query вся эта лишняя работа становится не нужна. Вы просто нажимаете на кнопку **Обновить все** (Refresh All), и от вас больше ничего не требуется. Все действительно очень просто. И дело не только в скорости, но и в постоянстве производимых операций и практически полном исключении человеческого фактора, который зачастую становится источником ошибок.

Даже если вы не собираетесь пользоваться богатыми возможностями Power Query в отношении обновления данных, вы сможете применять широкий спектр инструментов, входящих в его состав, для очистки и обработки данных. Как вы узнаете, читая эту книгу, действительно сложные задачи по приведению исходных данных к надлежащему виду выполняются при помощи Power Query с поразительной легкостью и быстрой, что позволяет все силы сосредоточить на том, за что вам платят, а именно на анализе данных.

Также хочется отметить, что с Power Query вам не придется изучать отдельные инструменты ETL для Excel, Power BI и других программных продуктов. Power Query присутствует в Excel, Power BI Desktop, потоках данных Power BI, Power Automate и многих других пакетах и службах от Microsoft. При этом в компании четко дали понять, что возлагают большие надежды на эту технологию и планируют в будущем внедрять ее во все большем количестве продуктов. Изучать новый программный пакет всегда нелегко, но радует хотя бы то, что впоследствии вы сможете применять его на самых разных платформах.