**Глава 24. Динамический календарь в Power Query**

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

[Предыдущая глава](http://baguzin.ru/wp/?p=21433) [Содержание](http://baguzin.ru/wp/?p=20852#content) Следующая глава

С помощью Power Query довольно легко настроить полностью динамический календарь в Excel. Благодаря идеям [главы 23](http://baguzin.ru/wp/?p=21433) вы сможете управлять им на основе значений в двух ячейках Excel. Вам предстоит пройти четыре шага:

1. Добавить в Excel таблицу параметров для хранения границ календаря.
2. Подключить функцию *fnGetParameter* для передачи границ в Power Query.
3. Создать базовый столбец календаря с датами.
4. Добавить в календарь дополнительные столбцы (месяц, квартал, день недели, …).



Рис. 24.1. Таблица параметров

### Добавление таблицы параметров

Создайте новую книгу Excel, а в ней *Таблицу*, как показано на рис. 24.1. *Таблица* должна называться *Parameters*. Заголовки столбцов должны иметь те же имена, что и на рисунке. Стартовая дата календаря = 01.01.2014. Конечная дата = последней дате текущего месяца, и вычисляется по формуле =КОНМЕСЯЦА(СЕГОДНЯ();0). Синтаксис функции: КОНМЕСЯЦА(нач\_дата;число\_месяцев), где число\_месяцев – количество месяцев до или после даты "нач\_дата". Положительное значение аргумента "число\_месяцев" означает будущие даты; отрицательное значение – прошедшие даты. Если вы используете 0, функция вернет последний день текущего месяца. Если вы укажете -1, это будет конец предыдущего месяца. 1 соответствует последней дате следующего месяца.

### Функция *fnGetParameter*

Откройте файл *fnGetParameter.txt.* Скопируйте его содержимое в буфер. Вернитесь в книгу Excel, пройдите по меню *Данные* –> *Получить данные* –> *Из других источников* –> *Пустой запрос*. В редакторе Power Query перейдите на вкладку *Главная* –> *Расширенный редактор*. Выделите весь код в окне и нажмите Ctrl+V –> *Готово*. Переименуйте запрос *fnGetParameter*.



Рис. 24.2. Функция *fnGetParameter* готова к работе

### Построение структуры календаря

Продолжим. Создайте новый запрос не выходя из редактора Power Query. *Главная* –> *Создать источник* –> *Другие источники* –> *Пустой запрос*. Переименуйте запрос *Calendar*. Самый простой способ начать строить календарь – создать простой список. Введите в строке формул: ={1..10}. Нажмите *Enter*. Вы создали список от одного до десяти:



Рис. 24.3. Теперь у вас есть простой список

Преобразуйте этот список в таблицу и посмотрите, что произойдет, если изменить тип данных –> *Дата*. Перейдите на вкладку *Средства для списков* –> *Преобразование* –> *В таблицу*. Оставьте параметры по умолчанию и нажмите Ok. Щелкните правой кнопкой мыши столбец *Column1* –> *Тип изменения* –> *Дата*. Щелкните правой кнопкой мыши столбец *Column1* –> *Переименовать* –> *Date*. Хотя это не тот диапазон дат, который вам нужен, вы получили фрагмент календаря:



Рис. 24.4. Календарь; правда, немного устаревший))

Теперь нужно вставить пару шагов в начале запроса, чтобы извлечь даты начала и окончания, а затем передать эти даты в список вместо "от 1 до 10". Перейдите на вкладку *Главная* –> *Расширенный редактор*. После строки *let* добавьте две строки кода M:

startdate = fnGetParameter("Start Date"),

enddate = fnGetParameter("End Date"),

Не забудьте про запятые в конце каждой строки. Нажмите *Готово*. Убедитесь, что оба шага возвращают «правильные» даты:



Рис. 24.5. *Start Date* возвращает 01.01.2014

Замените числа 1 и 10 переменными. Выберите шаг *Источник*. Измените формулу:

= {startdate..enddate}

Нажать *Enter*. К сожалению, возвращается ошибка:



Рис. 24.6. Что не так?

Это сообщение об ошибке звучит не вполне понятно. Лучше бы Power Query сказал: «Невозможно использовать оператор .. (две точки), так как он может располагаться только между двумя числами». Хотя даты в Excel это целые числа, Power Query их таковыми не воспринял. Преобразуйте даты в числа явным образом внутри Power Query. Вернитесь в *Расширенный редактор*. Дополните строки с определением переменных:

startdate = Number.From(fnGetParameter("Start Date")),

enddate = Number.From(fnGetParameter("End Date")),

Нажмите *Готово*. Перейдите на шаг *startdate*. Дата отражается в виде целого числа 41640. Перейдите к последнему шагу запроса, вы увидите таблицу, начинающуюся с 1 января 2014 года:



Рис. 24.7. Календарь готов к использованию

### Добавление столбцов календаря

Выберите столбец *Date* –> *Добавление столбца* –> *Дата* –> *Год* –> *Год.* Выберите столбец *Date* –> *Добавление столбца* –> *Дата* –> *Квартал* –> *Квартал года.* Выберите столбец *Date* –> *Добавление столбца* –> *Дата* –> *Месяц* –> *Месяц.* Выберите столбец *Date* –> *Добавление столбца* –> *Дата* –> *День* –> *День недели.*



Рис. 24.8. В календарь добавлены часто используемые столбцы

### Функции даты в Excel и Power Query



Рис. 24.9. Примеры функций Excel



Рис. 24.10. Сравнение функций в Excel и Power Query

### Сложение дат

Вы будете разочарованы, если попытаетесь взять дату и добавить к ней число. Даже несмотря на то, что даты – это числа, Power Query не может неявно преобразовать тип данных, как это умеет делать Excel. Поэтому в Power Query существует коллекция функций для добавления дат:



Рис. 24.11. Функции сложения дат в Excel и Power Query

### Даты как текст

Для возврата дат в текстовом формате в Excel используют функцию ТЕКСТ(). В Power Query аналогичным целям служит функция Date.ToText(). С последней нужно держать ухо востро: не только имя функции чувствительно к регистру, но и ее параметры. Поскольку Power Query не русифицирован, то и текст он возвращает английский:



Рис. 24.12. Даты как текст

### Работа с ошибками преобразования даты

Многие из перечисленных выше функций требуют ввода типа данных *Дата* или *Дата и время*, и возвращают ошибку, если указан иной тип данных. Чтобы избежать этой ситуации, можно поместить столбец [Date] внутрь функции Date.From(). Например, так:

=Date.AddDays(Date.From([DateColumn]),1)