**Новые функции Excel, основанные на LAMBDA**

В декабре 2020 года Microsoft [анонсировал](https://www.microsoft.com/en-us/research/blog/lambda-the-ultimatae-excel-worksheet-function/) функцию LAMBDA, которая позволяет определять пользовательские функции, написанные на языке формул Excel. Я недавно привел примеры того, как работает [LAMBDA](https://baguzin.ru/wp/?p=24759). Но Microsoft на этом не остановился, и в июле 2021 г. [объявил](https://insider.office.com/sk-sk/blog/new-lambda-functions-available-in-excel) о создании новых функций, основанных на LAMBDA:

[MAP](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-map-48006093-f97c-47c1-bfcc-749263bb1f01): принимает в качестве аргумента массив, и возвращает массив того же размера, обработав каждый элемент с помощью функции LAMBDA.

[REDUCE](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-reduce-42e39910-b345-45f3-84b8-0642b568b7cb): реализует цикл обработки элементов массива, применяя функцию LAMBDA к каждому значению и возвращает одно накопленное значение.

[SCAN](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-scan-d58dfd11-9969-4439-b2dc-e7062724de29): реализует цикл обработки элементов массива, применяя функцию LAMBDA к каждому значению и возвращает массив того же размера, содержащий все промежуточные значения.

[MAKEARRAY](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-makearray-b80da5ad-b338-4149-a523-5b221da09097): возвращает массив с заданным числом строк и столбцов, применяя функцию LAMBDA для вычисления элементов массива.

[BYCOL](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-bycol-58463999-7de5-49ce-8f38-b7f7a2192bfb): возвращает массив в виде строки, один результат для каждого столбца массива.

[BYROW](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-byrow-2e04c677-78c8-4e6b-8c10-a4602f2602bb): возвращает массив в виде столбца, один результат для каждой строки массива.

[ISOMITTED](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-isomitted-831d6fbc-0f07-40c4-9c5b-9c73fd1d60c1): проверяет, отсутствует ли значение, и возвращает ИСТИНА, в противном случае –ЛОЖЬ.

### Функция MAP

Цель – создать массив того же размера, что и исходный, преобразовав каждый элемент с помощью функции LAMBDA. Синтаксис:

=MAP(массив1; [массив2]; [массив#]; LAMBDA)

массив1 – массив, который нужно сопоставить (обязательный);

массив2 – еще один массив для сопоставления (необязательный);

массив# – еще массивы для сопоставления (необязательный);

LAMBDA – функция LAMBDA, которая должна быть последним аргументом и включать столько параметров, сколько передано массивов (обязательный).

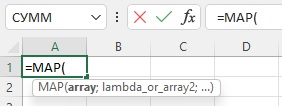


Рис. 1. Синтаксис функции MAP

#### Квадраты целых чисел, превышающих определенное значение

Следующая формула принимает массив А1:С2 и возвращает массив D4:F5. Если элемент в исходном массиве не больше 4, возвращается его значение, если более – квадрат значения:

=MAP(A1#;LAMBDA(a;ЕСЛИ(a>4;a^2;+a)))

Изображение выглядит как стол

Автоматически созданное описание

Рис. 2. Квадраты целых чисел, превышающих определенное значение

По поводу использования +a в качестве третьего аргумента функции ЕСЛИ() на сайте Microsoft и в Инете сказано, что обозначение +a используется для принудительного преобразования из ссылки на диапазон в массив значений. Без знака + функция MAP может вернуть ошибку #ВЫЧИСЛ! Этот шаг иногда необходим при работе с формулами массива, которые разливаются в диапазон. Я поэкспериментировал, но мне не удалось получить ошибку #ВЫЧИСЛ!. Так что запись

=MAP(A1#;LAMBDA(a;ЕСЛИ(a>4;a^2;a)))

… тоже работает. Возможно, ошибка существовала в более ранних версиях, а сейчас устранена.

#### Применение логического И к двум массивам

Следующая формула в качестве аргументов принимает два массива – столбцы Элемент1 и Элемент2, и возвращает ИСТИНА, только если оба значения в одной строке = ИСТИНА:

=MAP(Таблица1[Элемент1];Таблица1[Элемент2];LAMBDA(a;b;И(a;b)))

Изображение выглядит как стол

Автоматически созданное описание

Рис. 3. Массив, основанный на логическом И

#### Фильтрация строк, отвечающих нескольким условиям

Изображение выглядит как стол

Автоматически созданное описание

Рис. 4. Вывод элементов массива по критериям

Следующая формула фильтрует таблицу А1:В11, оставляя только строки, которые соответствуют критериям отбора:

=ФИЛЬТР(

Таблица3;

MAP(

Таблица3[Размер];

Таблица3[Цвет];

LAMBDA(

разм;

цвет;

И(разм=E2;цвет=E3)

)

)

)

#### Удаление не цифр

В ячейке А2 я создал случайный массив из цифр, прописных и строчных букв английского алфавита и некоторых специальных символов:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 5. Массив случайных символов

Проанализируем формулу

=СИМВОЛ(СЛМАССИВ(12;6;48;122;ИСТИНА))

Функция СЛМАССИВ() создает массив из 12 строк и 6 столбцов целых чисел (аргумент ИСТИНА) в диапазоне от 48 до 122.

Изображение выглядит как текст, устройство

Автоматически созданное описание

Рис. 6. Таблица ASCII кодов

Функция СИМВОЛ() выводит символ по его ASCII коду. Функция СЛМАССИВ() волатильна, т.е. её значения меняются с любым изменением на листе Excel. Можно принудительно запускать пересчет, нажимая кнопку F9.

Создадим новый массив, который вернет только цифры:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 7. Функция MAP возвращает массив чисел

Несколько слов о формуле

=MAP(ЗНАЧЕН(A2#);LAMBDA(a;ЕСЛИ(ЕЧИСЛО(a);+a;"")))

Функция MAP() принимает массив А2#, обернутый в функцию ЗНАЧЕН(). Функция СИМВОЛ() возвращает текст, даже, если это цифра. ЕЧИСЛО() проверяет является символ числом. Если да, то возвращается сам символ +а, если нет – пустая строка "".

#### Когда использовать MAP

Механизм динамических массивов в Excel может выполнять ряд операций с массивами и без использования MAP. Например, следующие формулы возвращают один и тот же результат:

=массив+1

=MAP(массив; LAMBDA(a;a+1))

В таких кейсах использование MAP не оправдано. Однако есть случаи, когда MAP может быть полезна:

* Для выполнения логических тестов с функциями И, ИЛИ.
* Для выполнения операций агрегирования нескольких массивов. Например,

=MAP(rng1;rng2;rng3; LAMBDA(a;b;c;МАКС(a;b;c)))

* Чтобы заставить функцию разлиться, чего в противном случае не будет.

Функция MAP возвращает массив результатов. Если вы хотите обрабатывать каждый элемент в массиве по отдельности, но получить единый агрегированный результат, вам нужна функция REDUCE.

Изображение выглядит как стол

Автоматически созданное описание

Рис. 8. Максимальные значения трех массивов

#### Фильтрация таблицы по условию, рассчитываемому налету

Пусть у нас есть данные многолетних наблюдений температуры в Москве:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 9. Среднемесячные температуры в Москве по годам

Наша цель – оставить только те строки таблицы, которые соответствуют среднегодовой температуре > 6°C… и представить все данные по шкале Фаренгейта (извините за некоторую искусственность примера). С этим преобразованием справится следующая формула:

=MAP(

ФИЛЬТР(

Таблица4;

BYROW(

Таблица4;

LAMBDA(

год;

СРЗНАЧ(год)>S2)

);

"нет данных"

);

LAMBDA(

темп;

ЕСЛИ(

ЕЧИСЛО(темп);

ПРЕОБР(темп;"C";"F");

Темп

)

)

)

Чтобы номера годов не участвовали в вычислениях среднего значения по строке я предварительно преобразовал их в текст с помощью формулы ТЕКСТ(год;"0000").

Изображение выглядит как стол

Автоматически созданное описание

Рис. 10. В правом массиве отфильтрованы строки, соответствующие среднегодовой температуре > 6°C; температура указана по шкале Фаренгейта

Проанализируем формулу. Функция MAP() имеет два аргумента: массив (рис. 11а) и LAMBDA (рис. 11б).

Изображение выглядит как текст

Автоматически созданное описание

Рис. 11. Два аргумента функции MAP

Первый аргумент фильтрует исходную таблицу, т.е., оставляет только строки с годами, соответствующими условию, второй – преобразует значения элементов отфильтрованного массива.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 12. Три аргумента функции ФИЛЬТР

Функция ФИЛЬТР() имеет три аргумента: *массив* – ссылается на таблицу без строки заголовка (Таблица4); *включить* – какие строки таблицы оставить (функция BYROW); *если пусто* – что вывести, если строк после фильтрации не осталось.

Функция BYROW() подробнее будет рассмотрена ниже. Она принимает в качестве первого аргумента массив (Таблица4), и обрабатывает его с помощью функции LAMBDA(год;СРЗНАЧ(год)>S2). BYROW() возвращает массив в виде одного столбца с количеством строк, равным числу строк в Таблица4. Массив, возвращаемый BYROW(), содержит значения ИСТИНА, если СРЗНАЧ(год)>S2, или ЛОЖЬ, если условие не выполнено.

Второй аргумент функции MAP преобразует значения элементов отфильтрованного массива. Преобразование выполняется внутри функции LAMBDA.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 13. Два аргумента функции LAMBDA

Первый аргумент функции LAMBDA – параметр (*темп*). Этот параметр кодирует (ссылается на) массив, возвращаемый функции ФИЛЬТР(). Параметр *темп* обрабатывается функцией ЕСЛИ.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 14. Три аргумента функции ЕСЛИ

Первый аргумент функции ЕСЛИ() – *лог\_выражение* – проверяет, является ли элемент массива (отфильтрованной Таблица4) числом. Если да, то возвращается *значение\_если\_истина* – ПРЕОБР(темп;"C";"F"). Именно здесь происходит преобразование градусов Цельсия в Фаренгейта. Если проверяемый элемент массива не является числом (а в Таблица4 таковыми будут только года), то возвращается сам элемент.

Вуаля))

### Функция REDUCE

Смысл функции – обработать массив, чтобы вернуть одно число. Синтаксис:

=REDUCE([начальное\_значение]; массив; LAMBDA(а; b; вычисление))

[*начальное\_значение*] – задает начальное значение для аккумулятора (необязательный аргумент);

*массив* – массив, каждый элемент которого нужно обработать с помощью функции LAMBDA;

LAMBDA – функция обработки элементов массива. Поскольку функция REDUCE имеет два параметра – *начальное\_значение* и *массив*, функция LAMBDA также включает два параметра, ссылающиеся на *начальное\_значение* и *массив:*

* а – аккумулятор, накапливает значение, возвращаемое в качестве конечного результата функцией REDUCE; сначала аккумулятор получает *начальное\_значение*, а затем изменяется с каждым циклом обработки *массива*.
* b – ссылается на *массив* (другими словами – на каждый элемент *массива)*.

Третий аргумент функции LAMBDA – вычисление – задает порядок обработки элементов *массива.*

#### Суммирование квадратов значений

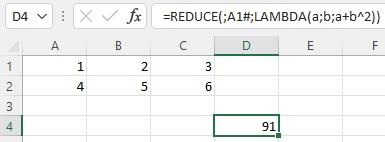


Рис. 15. Суммирование квадратов элементов массива

Разберем формулу

=REDUCE(;A1#;LAMBDA(a;b;a+b^2))

Функция REDUCE имеет три аргумента. Первый – initial\_value – необязательный (рис. 16а). Иногда его можно опустить, иногда следует явно указать ноль (см. пример 3). Иногда нужно присвоить значение, отличное от нуля, чтобы избежать деления или умножения на ноль. Второй аргумент – array – обязательный (рис. 16б). Это массив, вычисления по которому вернет одно значение. В нашем примере аргумент задается динамическим массивом А1#, который разлит в диапазоне А1:С2. Третий аргумент – function – обязательный. Он задает функцию обработки массива. По определению здесь может использоваться только функция LAMBDA. Параметр *a* соответствует начальному значению (равному нулю), *b* – массиву А1#. В третьем аргументе функции LAMBDA происходит суммирование квадратов всех элементов массива А1#.

Если вы немного программируете, то воспринимайте аргумент a+b^2, как выражение внутри цикла: a = a + b^2.

Изображение выглядит как стол

Автоматически созданное описание

Рис. 16. Три аргумента функции REDUCE

#### Произведение значений, превышающих критерий

Задача – найти произведение всех элементов массива, значение которых превышает 50. Используем формулу:

=ЕСЛИ(

REDUCE(1;A2#;LAMBDA(a;b;ЕСЛИ(b>50;a\*b;a)))=1;

"нет значений более 50";

REDUCE(

1;

A2#;

LAMBDA(

a;

b;

ЕСЛИ(

b>50;

a\*b;

a

)

)

)

)

Изображение выглядит как стол

Автоматически созданное описание

Рис. 17. Произведение значений, превышающих критерий

Основное действие происходит в этом фрагменте:

REDUCE(

1;

A2#;

LAMBDA(

a;

b;

ЕСЛИ(

b>50;

a\*b;

a

)

)

)

Функция REDUCE имеет три аргумента: initial\_value = 1, array – A2#, function – LAMBDA(a;b;ЕСЛИ(b>50;a\*b;a)). LAMBDA принимает два параметра: а = 1 и b = массив A2#. Третий аргумент функции LAMBDA – расчет = ЕСЛИ(b>50;a\*b;a). Если элемент массива A2# > 50, умножаем значение элемента на произведение, полученное на предыдущем шаге. Иначе, оставляем произведение, полученное на предыдущем шаге без изменения. Переходим к следующему элементу.

Функция REDUCE() обёрнута во внешнее ЕСЛИ(). Здесь проверяется: если REDUCE = 1, значит не нашлось ни одного элемента > 50. Возвращается текст "нет значений более 50". Иначе возвращается значение REDUCE().

#### Сумма четных и нечетных значений

Изображение выглядит как стол

Автоматически созданное описание

Рис. 18. Сумма четных и нечетных значений

С кодом формулы предлагаю вам разобраться самостоятельно:

=REDUCE(0;A2#;LAMBDA(a;b;ЕСЛИ(ЕЧЁТН(b);a+b;a)))

#### Подсчет числа значений, удовлетворяющих критерию

Продолжим примеру со среднемесячными температурами в Москве. Подсчитаем, сколько месяцев за последовательные 10-летние периоды имели среднюю температуру ≥ 18°С:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 19. Число месяцев в десятилетии, начинающемся с указанного года, с температурой ≥ 18°С

=REDUCE(

0;

ДВССЫЛ("B"&(O5-1879)&":M"&(O5-1870));

LAMBDA(

a;

b;

ЕСЛИ(b>=$O$2;a+1;a)

)

)

Функция REDUCE имеет три аргумента: initial\_value = 0, array – ДВССЫЛ("B"&(O5-1879)&":M"&(O5-1870)), function – LAMBDA(a;b;ЕСЛИ(b>=$O$2;a+1;a)). Функция ДВССЫЛ() возвращает диапазон (массив), соответствующий году в столбце О. Например, в ячейке Р5 функция ДВССЫЛ() возвращает диапазон В2:М11. (Подробнее о ДВССЫЛ см. [Примеры использования функции ДВССЫЛ](http://baguzin.ru/wp/?p=7119).) LAMBDA принимает два параметра: а = 0 и b = массив, сформированный ДВССЫЛ(). Третий аргумент функции LAMBDA – расчет = ЕСЛИ(b>=$O$2;a+1;a). Если элемент массива ≥ $O$2, накопитель – *а* – увеличивается на 1, если нет, то *а* остается без изменения.

Справедливости ради отметим, что «старые» формулы Excel справятся с задачей даже лучше. Например, введите в ячейке Q5 формулу (см. рис. 19):

=СЧЁТЕСЛИ(ДВССЫЛ("B"&(O5-1879)&":M"&(O5-1870));">="&$O$2)

### Функция SCAN

Функция сканирует массив и возвращает промежуточные результаты. Синтаксис:

=SCAN([начальное\_значение]; массив; LAMBDA(а; b; вычисление))

[*начальное\_значение*] – задает начальное значение для аккумулятора (необязательный аргумент); если вы работаете с текстом, установите начальное значение "".

*массив* – массив, который нужно просканировать;

LAMBDA – функция сканирования массива; включает два параметра, ссылающиеся на *начальное\_значение* и *массив:*

* а – сканируемое значение, возвращаемое в качестве конечного результата функцией SCAN; сначала аккумулятор получает *начальное\_значение*, а затем изменяется с каждым циклом обработки *массива*.
* b – ссылается на *массив* (другими словами – на каждый элемент *массива)*.

Третий аргумент функции LAMBDA – вычисление – задает порядок обработки элементов *массива.*

#### Вычисление факториала

=SCAN(

1;

A1#;

LAMBDA(

a;

b;

a\*b

)

)

Изображение выглядит как стол

Автоматически созданное описание

Рис. 20. Вычисление факториала

#### Сцепление символов в массиве

=SCAN(

"";

A1:C2;

LAMBDA(

a;

b;

a&b

)

)

Изображение выглядит как стол

Автоматически созданное описание

Рис. 21. Сцепление символов в массиве

На этом примере видно, как работает накопитель – параметр *а* функции LAMBDA.

#### Специфика SCAN, REDUCE и MAP

Функция SCAN возвращает массив, содержащий промежуточные значения, созданные во время сканирования. SCAN можно использовать для вычислений, которые показывают промежуточные результаты. Функция SCAN возвращает массив того же размера, что и исходный. Подобно функции REDUCE, SCAN реализует цикл по всем элементам массива, выполняя вычисления для каждого элемента. Отличие в том, что REDUCE возвращает одно – финальное – значение, а SCAN возвращает все промежуточные значения. Ранее изученная функция MAP также сканирует массив, и возвращает массив того же размера, что и исходный. Отличие в том, что MAP обрабатывает каждый элемент массива как независимый, а SCAN последовательно накапливает результат.

#### Накопительный итог в Умной Таблице

Допустим, у нас есть продажи за 6 месяцев:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 22. Таблица продаж

… и вы хотите показать продажи накопительным итогом. Можно использовать следующую неуклюжую формулу:

=Ч(C1)+[@Продажи]

Изображение выглядит как стол

Автоматически созданное описание

Рис. 23. Сумма продаж накопительным итогом

В этой формуле смешаны ссылка на ячейку (C1) и структурированная ссылка ([@Sales]). А функция Ч() добавлена, чтобы текстовое значение вернуло 0, а не ошибку #ЗНАЧ!

Формула работает, если добавляются значения, но ломается при вставке значений:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 24. Формула ломается при вставке строк

Вот тут-то и приходит на помощь функция SCAN:

=SCAN(

0;

Sales[Продажи];

LAMBDA(

a;

b;

a+b

)

)

Изображение выглядит как стол

Автоматически созданное описание

Рис. 25. Накопительный итог

SCAN сканирует массив – столбец *Продажи* таблицы *Sales* (диапазон А1:В10), последовательно суммируя значения на основе LAMBDA. Затем SCAN возвращает массив результатов (диапазон D2:D10), который подстраивается под размер столбца *Продажи*. И дополнительные строки Таблицы, и вставка новых строк, отражаются корректно.

### Функция MAKEARRAY

Функция возвращает массив с заданным числом строк и столбцов на основе пользовательского вычисления LAMBDA. MAKEARRAY можно использовать для создания массивов с переменными размерами и значениями, вычисляемыми внутри функции. Синтаксис:

=MAKEARRAY(строки; столбцы; LAMBDA(r; c; вычисление))

строки – количество строк в массиве; должно быть больше нуля;

столбцы – количество столбцов в массиве; должно быть больше нуля;

LAMBDA – функция, используемая при создании значений массива; общее количество значений в массиве будет равно числу строк, умноженному на число столбцов. Функция LAMBDA включает два параметра, ссылающиеся на *строки* и *столбцы:*

* r – ссылается на число строк;
* с – ссылается на число столбцов.

Третий аргумент функции LAMBDA – вычисление – задает значения возвращаемого массива*.*

#### Таблица умножения

=MAKEARRAY(

9;

9;

LAMBDA(

r;

c;

r\*c

)

)

Изображение выглядит как текст, электроника

Автоматически созданное описание

Рис. 26. Таблица умножения

#### Массивы фиксированных и случайных значений

Элементы массива могут быть жестко заданными. Например…

=MAKEARRAY(2;3;LAMBDA(r;c;0))

… вернет прямоугольник 2х3 с шестью нулями.

Массив может состоять из букв. Например…

=MAKEARRAY(2;3;LAMBDA(r;c;"x"))

Более интересен вариант со случайными элементами массива. Например…

=MAKEARRAY(6;6;LAMBDA(r;c;СИМВОЛ(СЛУЧМЕЖДУ(65;90))))

… вернет массив 6х6 со случайно отобранными прописными английскими буквами:

Изображение выглядит как стол

Автоматически созданное описание

Рис. 27. Массив случайных букв

#### Визуализация адресов ячеек

=MAKEARRAY(

10;

10;

LAMBDA(

a;

b;

ОБЪЕДИНИТЬ(

;

;

СИМВОЛ(b+64);

a

)

)

)

Изображение выглядит как стол

Автоматически созданное описание

Рис. 28. Адреса ячеек

### Функция BICOL

Функция возвращает массив в виде строки, один результат для каждого столбца массива. Например, если исходный массив включает 3 столбцами, BYCOL вернет строку из 3 элементов. Синтаксис:

= BYCOL(массив; LAMBDA(а; вычисление))

*массив* – исходный массив;

LAMBDA – функция обработки массива; включает один параметр *а*, ссылающиеся на *массив.* Второй аргумент функции LAMBDA – вычисление – задает способ агрегирования столбцов *массива.*

#### Возврат максимального значения в каждом столбце

=BYCOL(A1#;LAMBDA(a;МАКС(a)))

Изображение выглядит как стол

Автоматически созданное описание

Рис. 29. Максимальное значение в каждом столбце массива

#### Сумма квадратов элементов массива отдельно по каждому столбцу

=BYCOL(A1#;LAMBDA(a;СУММКВ(a)))

Изображение выглядит как стол

Автоматически созданное описание

Рис. 30. Сумма квадратов по столбцу

### Функция BIROW

Функция возвращает массив в виде столбца, один результат для каждой строки массива. Например, если исходный массив включает 3 строки, BYROW вернет столбец из 3 элементов.

Изображение выглядит как стол

Автоматически созданное описание

Рис. 31. Примеры использования функции BYROW

Для расчета суммы по строке в ячейки I2 используется формула:

=BYROW(B2#;LAMBDA(a;СУММ(a)))

Для нахождения числа вхождений со значениями более 90 по строке в ячейке J2 используется формула:

=BYROW(B2#;LAMBDA(a;СЧЁТЕСЛИ(a;">"&90)))

Еще один метод использования функции BYROW был показан [выше](#_Фильтрация_таблицы_по).

Функция [ISOMITTED](https://support.microsoft.com/ru-ru/office/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F-isomitted-831d6fbc-0f07-40c4-9c5b-9c73fd1d60c1) на момент написания заметки отсутствовала в моей версии Excel.

### Использованные материалы

New LAMBDA functions available in Excel <https://insider.office.com/es-es/blog/new-lambda-functions-available-in-excel>

Excel MAP Function <https://exceljet.net/excel-functions/excel-map-function>

Excel REDUCE Function <https://exceljet.net/excel-functions/excel-reduce-function>

Excel SCAN Function <https://exceljet.net/excel-functions/excel-scan-function>

Excel MAKEARRAY Function <https://exceljet.net/excel-functions/excel-makearray-function>

Excel BYCOL Function <https://exceljet.net/excel-functions/excel-bycol-function>

Excel BYROW Function <https://exceljet.net/excel-functions/excel-byrow-function>

Liam Bastick and Chris Gross. Not Just Mary Can Have a Little LAMBDA / Excel Hits 500 <https://www.sumproduct.com/news/article/not-just-mary-can-have-a-little-lambda-excel-hits-500>