

## Лабораторная работа № 6

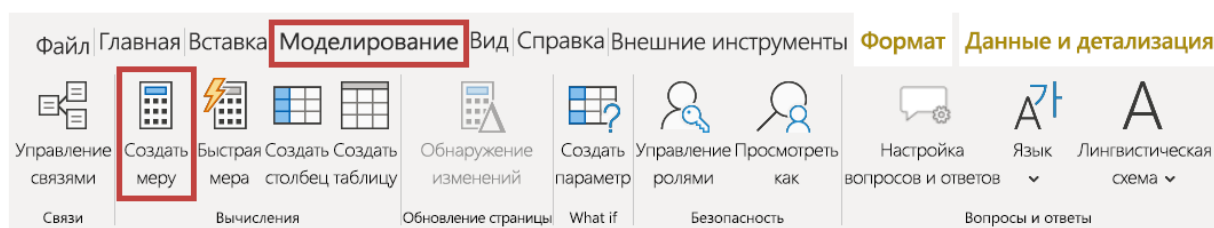
### Тема: «Основы создания мер в PowerBi с использованием DAX»

**Цель работы:** изучить основы работы с DAX в PowerBi.

#### Теоретическая справка

**Data Analysis Expressions**, сокращенно **DAX** – это язык запросов для Power Pivot, Power BI Desktop и SQL Server Analysis Services (SSAS). DAX представляет из себя коллекцию из более чем 200 функций, операторов и констант, которые можно использовать в формуле или выражении для вычисления и возврата одного или нескольких значений. DAX помогает создавать новые сведения из данных, уже имеющихся в модели.

**Мера** является базовым понятием в DAX и представляет собой выражение, которое позволяет рассчитать необходимый показатель на основании данных из модели. Пример использования мер, это расчет среднего, суммы, количества уникальных записей и пр. Меры вычисляются, когда вы взаимодействуете с отчетами, и не сохраняются в базу данных.



Пример меры, которая возвращает только уникальные записи из таблицы 'Сеансы', а именно из столбца [Идентификатор пользователя].

Пользователи :=

```
DISTINCTCOUNT ( 'Сеансы'[Идентификатор пользователя] )
```

**Вычисляемый столбец** также позволяет рассчитать показатели, но подсчет производится для каждой строки таблицы отдельно и результат сохраняется в отдельное поле (новый столбец таблицы). После создания подобного вычисляемого столбца его можно использовать наравне с остальными столбцами модели. Пример использования вычисляемого столбца – создание некоего столбца с ключами (уникальными идентификаторами записей, для связей с другими таблицами).

## «Визуальная аналитика», Киреев В.С.



**Формула DAX** всегда начинается с знака равенства (=). После знака равенства можно указать любое выражение, которое вычисляется скалярным или выражение, которое можно преобразовать в скаляр. следующие основные параметры:

- Скалярная константа или выражение, использующее скалярный оператор (+, -, \*, /, >=, <=, >, <, ...)
- Ссылки на столбцы или таблицы. Язык DAX всегда использует таблицы и столбцы в качестве входных данных для функций, никогда не массив или произвольный набор значений.
- Операторы, константы и значения, предоставляемые в рамках выражения.
- Результат функции и его обязательных аргументов. Некоторые функции DAX возвращают таблицу вместо скалярного и должны быть заключены в функцию, которая вычисляет таблицу и возвращает скаляр; Если таблица не является одним столбцом, одной строкой, то она рассматривается как скалярное значение.
- Большинство функций DAX требуют одного или нескольких аргументов, которые могут включать таблицы, столбцы, выражения и значения. Однако некоторые функции, такие как PI, не требуют каких-либо аргументов, но всегда требуют круглые скобки для указания аргумента NULL. Например, необходимо всегда вводить pi(), а не PI. Вы также можете вложить функции в другие функции.
- Выражения. Выражение может содержать любой или все из следующих: операторы, константы или ссылки на столбцы.

Каждый столбец и мера, добавляемая в существующую модель данных, должны принадлежать определенной таблице. При использовании таблицы или столбца в качестве входных данных для функции обычно необходимо *указать* имя столбца. Полное имя столбца — это имя таблицы, за которым следует имя столбца в квадратных скобках: например, "Продажи США"[Продукты].

Типы объектов	Примеры	Комментарий
Имя таблицы	Продажи	Если имя таблицы не содержит пробелы или другие специальные символы, имя не должно быть заключено в кавычки.
Имя таблицы	"Продажи Канады"	Если имя содержит пробелы, вкладки или другие специальные символы, заключите имя в одинарные кавычки.
Полное имя столбца	Продажи[сумма]	Имя таблицы предшествует имени столбца, а имя столбца заключено в квадратные скобки.

## «Визуальная аналитика», Киреев В.С.

Типы объектов	Примеры	Комментарий
Полное имя меры	<b>Продажи[прибыль]</b>	Имя таблицы предшествует имени меры, а имя меры заключено в квадратные скобки. В определенных контекстах всегда требуется полное имя.
Неквалифицированное имя столбца	<b>[Сумма]</b>	Неквалифицированное имя — это только имя столбца в квадратных скобках. Контексты, в которых можно использовать некавалифицированное имя, включают формулы в вычисляемый столбец в той же таблице или в функцию агрегирования, которая сканирует ту же таблицу.
Полный столбец в таблице с пробелами	<b>"Канада Продажи"[Qty]</b>	Имя таблицы содержит пробелы, поэтому он должен быть окружен одними кавычками.

Синтаксис, необходимый для каждой функции, и тип операции, который он может выполнять, сильно зависит от функции. Формулы и выражения DAX не могут изменять или вставлять отдельные значения в таблицы.

Тип оператора	Символ и использование
Оператор круглых скобок	() порядок приоритета и группирование аргументов
Арифметические операторы	+ (добавление) - (вычитание/знак) * (умножение) / (деление) ^ (экспонентация)
Операторы сравнения	= (равно) > (больше чем); < (меньше чем); >= (больше или равно); <= (меньше или равно). <> (не равно)
Оператор объединения текста	& (объединение)
Операторы логики	&& (и)    (или)

**Контекст** - описывает среду, в которой вычисляется формула DAX. Существует два типа контекста: контекст строки и контекст фильтра. Контекст строки представляет "текущую строку" и используется для вычисления формул и выражений вычисляемых столбцов, используемых итераторами таблиц. Контекст фильтра используется для оценки мер, и он представляет фильтры, применяемые непосредственно к столбцам модели и фильтрам, распространяемым связями модели.

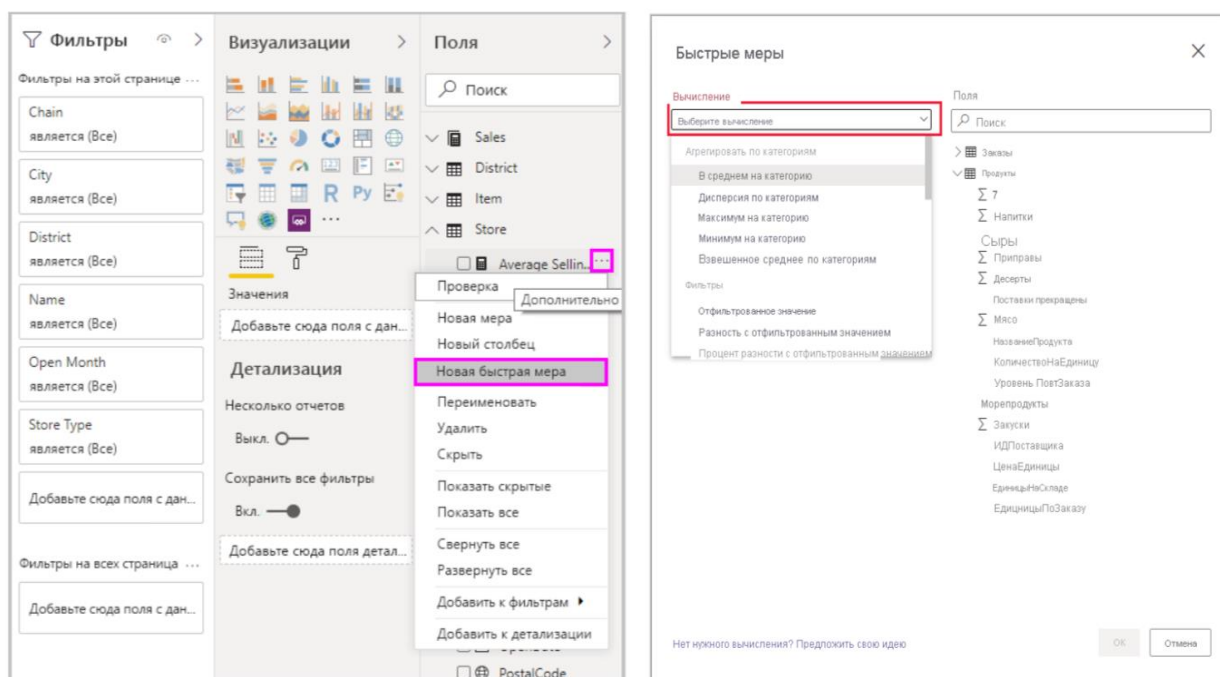
Рассмотрим сложную меру, рассчитывающую кол-во новых пользователей, которая, содержит несколько вложенных функций:

Новые пользователи :=

## «Визуальная аналитика», Киреев В.С.

```
COUNTROWS (
    FILTER (
        CALCULATETABLE (
            ADDCOLUMNS (
                VALUES ( 'Сеансы'[Идентификатор пользователя] );
                "Дата первого сеанса"; CALCULATE (
                    MIN ( 'Сеансы'[Дата] )
                )
            )
        );
        ALL ( 'Параметры дат' )
    );
    CONTAINS (
        VALUES ( 'Параметры дат'[Дата] );
        'Параметры дат'[Дата]; [Дата первого сеанса]
    )
)
```

**Быстрая мера** выполняет набор команд анализа данных (DAX) «под капотом», а затем представляет результаты, используемые в отчете. Для их создания не нужно писать DAX, это делается на основе входных данных, предоставляемых в диалоговом окне. Большое преимущество быстрых мер заключается в том, что они показывают формулу DAX, реализующую меру. При выборе быстрой меры на панели "Поля" появится строка формулы DAX, созданная Power BI для реализации меры.



Быстрые меры доступны только в том случае, если можно изменить модель. Одним из исключений является работа с некоторыми динамическими подключениями. Поддерживаются табличные динамические подключения SSAS. При работе в режиме DirectQuery нельзя создавать быстрые меры аналитики времени. Функции DAX, используемые в этих быстрых мерах, влияют на производительность при переводе в инструкции T-SQL, которые отправляются в источник данных.

## «Визуальная аналитика», Киреев В.С.

### Таблицы для хранения мер

Если в вашем отчете несколько страниц с несвязанными вычислениями, имеет смысл создать отдельную таблицу для них.

Общий

Имя

К поставке ОЗ с пов.коэф.

Начальная таблица

Меры OZON

Описание

Введите описание

Синонимы

к поставке ОЗ с пов.коэф.

Папка отображения

Факты;Проценты

Поиск

- #OZON Поставки
  - Моя мера
- Меры OZON
  - Проценты
    - % выкупа ОЗ
    - % комиссии ОЗ
    - % логистика от вывода ОЗ
    - % скидки ОЗ
    - К поставке ОЗ с пов.коэф.
  - Факты
    - Дата окончания остатка ОЗ
    - К поставке ОЗ
    - К поставке ОЗ с пов.коэф.
    - Количество заказов OZ
    - Остатка хватит на
    - Потребность в товаре ОЗ

Когда один показатель может выражаться и в процентах, и числом, используйте префиксы "\$, #, %, Δ" в названиях мер. Тогда вы точно будете знать в каком формате выдается вычисление. При работе с большим количеством расчетов отклонений по месяцам/неделям/дням, рационально не только создавать папки для измерений, но и применять в их названии буквенные коды:

Префикс	Назначение	Пример
DoD	Отклонение по дням	<b>DoD_Выручка</b> = DIVIDE( CALCULATE(_Finance[!Выручка], 'Calendar'[Сегодня]=1), CALCULATE(_Finance[!Выручка], 'Calendar'[Вчера]=1) )
WoW	Отклонение по неделям	<b>WoW_Выручка</b> = DIVIDE( CALCULATE(_Finance[!Выручка], 'Calendar'[Текущая неделя]=1), CALCULATE(_Finance[!Выручка], 'Calendar'[Прошлая неделя]=1) )
MoM	Отклонение по месяцам	<b>MoM_Выручка</b> = DIVIDE( CALCULATE(_Finance[!Выручка], 'Calendar'[Текущий месяц]=1),

**«Визуальная аналитика», Киреев В.С.**

<b>Префикс</b>	<b>Назначение</b>	<b>Пример</b>
		<code>CALCULATE(_Finance[!Выручка], 'Calendar'[Прошлый месяц]=1) )</code>
<b>YoY</b>	Отклонение по годам	<b>YoY_Выручка =</b> <code>DIVIDE(     CALCULATE(_Finance[!Выручка],     'Calendar'[Текущий год]=1),     CALCULATE(_Finance[!Выручка],     'Calendar'[Прошлый год]=1) )</code>
<b>T</b>	Измерение за сегодня	<b>T_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Сегодня]=1)</code>
<b>W</b>	Измерение за текущую неделю	<b>W_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Текущая неделя]=1)</code>
<b>M</b>	Измерение за текущий месяц	<b>M_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Текущий месяц]=1)</code>
<b>Y</b>	Измерение за текущий год	<b>Y_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Текущий год]=1)</code>
<b>AT</b>	Измерение за вчера	<b>AT_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Вчера]=1)</code>
<b>AW</b>	Измерение за прошлую неделю	<b>AW_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Прошлая неделя]=1)</code>
<b>AM</b>	Измерение за прошлый месяц	<b>AM_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Прошлая выручка]=1)</code>
<b>AY</b>	Измерение за прошлый год	<b>AY_Выручка =</b> <code>CALCULATE(_Finance[!Выручка], 'Calendar'[Прошлый год]=1)</code>
<b>@</b>	Текстовое значение	<b>@_Количество_заказов =</b> <code>"Кол-во заказов за пр. месяц: "&amp; FORMAT(     '_Delivery'[#_Количество заказов], "#,###")</code>

## «Визуальная аналитика», Киреев В.С.

Префикс	Назначение	Пример
#	Наиболее используемые меры	#_Выручка = SUM('uni vf Продажи'[DishSumInt])

### Самостоятельное задание

1. В Power BI Desktop выберите "Открыть файл>" и загрузите pbix файл с данным к лабораторной работе.
2. Создайте меру среднего значения поля SalesAmount .в таблице Sales и визуализируйте ее с помощью столбиковой диаграммы.
  - 2.1. Добавьте разрез RegionCountryName из таблицы Geography
3. Создайте меру Net Sales, очищающую сумму выручки от суммы скидок клиентам Sales[DiscountAmount] и суммы возвратов клиентов sales[ReturnAmount].
4. Визуализируйте с помощью столбиковой диаграммы общие и чистые продажи на одном графике в разрезе RegionCountryName
5. Добавьте срез для дальнейшего фильтрации чистых объемов продаж и продаж по календарю, используя поле "Год" из таблицы "Календарь".
  - 5.1. На основе этой меры сделайте новую, показывающую отклонение чистой выручки, по месяцам
  - 5.2. На основе этой меры сделайте новую, показывающую отклонение чистой выручки по годам
6. Создайте меру, вычисляющую какие продукты имеют самую высокую чистую сумму продаж за единицу продаж. Используйте sales[SalesQuantity].
7. Визуализируйте полученную меру с помощью Treemap диаграммы, используя поле "Категория продукта".
8. Создайте меру ежегодного процентного прироста чистой выручки внутри категории продукта и региона.
  - 8.1. На основе этой меры сделайте новую, показывающую отклонение прироста чистой выручки от прошлого месяца
  - 8.2. На основе этой меры сделайте новую, показывающую отклонение прироста чистой выручки от прошлого года
9. Создайте меру, показывающую категории продукта, приносящие 80%, 15% и 5% общей выручки, в рамках ABC –анализа.
10. Создайте таблицу и папки для своих мер, и переименуйте меры в соответствии с рекомендуемыми префиксами.