

**1. Вычислить (Eval)** - предназначена для вычисления выражения в контексте некоторой группировки. Функция доступна на платформе. Вместо неё рекомендуется использовать функцию `ВычислитьВыражение`.

## Синтаксис:

Вычислить(Выражение, Группировка, ТипРасчета)

### Параметры:

- **Группировка** (Строка). Содержит имя группировки, в контексте которой необходимо вычислить выражение. В случае если группировка не определена, вычисление будет выполнено в контексте текущей группировки. В случае если в качестве имени группировки будет использован пустой текст, то вычисление будет выполняться в контексте родительской группировки. В остальных случаях вычисление будет выполняться в контексте родительской группировки с таким именем. Например:

Сумма(Продажи.СуммаОборот) / Вычислить("Сумма(Продажи.СуммаОборот)", "ОбщийИтог").

В данном примере в результате получится отношение суммы по полю "Продажи.СуммаОборот" записи группировки к су

- **ТипРасчета** (Строка). В случае если данный параметр имеет значение "ОбщийИтог", выражение будет вычисляться для "Группировка", значения будут вычисляться для текущей групповой записи группировки.

## 2. ВычислитьВыражение (EvalExpression)

## Синтаксис:

ВычислитьВыражение(Выражение, Группировка, ТипРасчета, Начало, Конец, Сортировка, ИерархическаяСортировка, ОбработкаОдинаковыхЗна

### Параметры:

- **Выражение** (Строка). Содержит вычисляемое выражение;
- **Группировка** (Строка). Содержит имя группировки, в контексте которой необходимо вычислить выражение. В случае если вычисление будет выполнено в контексте текущей группировки. В случае если в качестве имени группировки будет использовано слово "Итого", то значение будет равно общему итогу. В остальных случаях вычисление будет выполняться в контексте родительской группировки с таким именем.
- **ТипРасчета** (Строка). В случае если данный параметр имеет значение "ОбщийИтог", выражение будет вычисляться для "Итого". Если значение параметра не равно "ОбщийИтог", то значения будут вычисляться для текущей групповой записи группировки. Если параметр имеет значение "ПерваяЗапись", то значение будет вычисляться только для первой записи группировки. Если параметр имеет значение "ВсеЗаписи", то значение будет вычисляться для каждой записи группировки по ресурсам, выражение будет вычислено для первой групповой записи исходной группировки. При вычислении значения для групповых записей, не являющихся группировками по ресурсам, функция вычисляется так же, как бы она вычислялась для группировки. Например, при генерации макета компоновки данных при выводе в макет поля – ресурса, по которому выполнена функция ВычислитьВыражение, с указанным параметром "ГруппировкаНеРесурса". Для остальных ресурсов в группировке значение будет равно значению для первой записи. Если параметр имеет значение "Иерархия", то выражение нужно вычислять для родительской иерархической записи, если таковая имеется. Компоновщик макета при генерации выражения для поля "% в группе иерархии" генерирует выражение, содержащее функцию ВычислитьВыражение, с параметрами: имя поля, тип расчета, имя группировки, имя ресурса. Например, для выражения ресурса, вычисляемого для текущей группировки с типом вычисления "Иерархия".

Строка, содержащая одно из:

- "Первая" ("First"). Необходимо получать первую запись группировки. После слова в скобках можно указывать выражение группировки. Получаемое значение должно целым быть числом, больше нуля. Например, Первая(3) – получение третьей группировки, то считается, что записей нет. Например, если записей 3, а требуется получить Первую(4), то считается, что запись не найдена.
- "Последняя" ("Last"). Необходимо получить последнюю запись группировки. После слова в скобках можно указывать выражение группировки. Получаемое значение должно целым быть числом, больше нуля. Например, Последняя(3) – получение третьей группировки, то считается, что записей нет. Например, если записей 3, а требуется получить Последнюю(4), то считается, что запись не найдена.
- "Предыдущая" ("Previous"). Необходимо получить предыдущую запись группировки. После слова в скобках можно указывать выражение группировки. Получаемое значение должно целым быть числом, больше нуля. Например, Предыдущая(3) – получение третьей группировки, то считается, что записей нет. Например, если записей 3, а требуется получить Предыдущую(4), то считается, что запись не найдена.
- "Следующая" ("Next"). Необходимо получить следующую запись группировки. После слова в скобках можно указывать выражение группировки. Получаемое значение должно целым быть числом, больше нуля. Например, Следующая(3) – получение третьей группировки, то считается, что записей нет. Например, если записей 3, а требуется получить Следующую(4), то считается, что запись не найдена.
- "Текущая" ("Current"). Необходимо получить текущую запись. При получении для итога по группировке получается первая запись.
- "ОграничивающееЗначение" ("BoundaryValue"). Необходимо получить запись по указанному значению. После слова: значения которого нужно начинать фрагмент, первого поля упорядочивания. В качестве записи будет получаться первая запись, соответствующая указанному значению. Например, если в качестве поля упорядочивания используется поле Период, и оно имеет значение 01.02.2010, то будет получена запись с датой 01.02.2010.

- **Конец**. Указывает, до какой записи нужно продолжать фрагмент, в котором рассчитывать агрегатное выражение. Строки
  - "Первая" ("First").
  - "Последняя" ("Last").
  - "Предыдущая" ("Previous").
  - "Следующая" ("Next").
  - "Текущая" ("Current").
  - "ОграничивающееЗначение" ("BoundaryValue").
- **Сортировка**. Строка, в которой перечисляются выражения, разделенные запятыми, в направлении которых нужно упорядочивается также как и у группировки, для которой вычисляется выражение. После каждого выражения можно указать упорядочивания по убывания, Автоупорядочивание, для упорядочивания ссылочных полей по полям, по которым нужен Автоупорядочивание может использоваться как со словом Возр, так и со словом Убыв.
- **ИерархическаяСортировка**. Аналогично сортировке. Применяется для упорядочивания иерархических записей. Если в соответствии с упорядочиванием, указанным в параметре Сортировка.
- **ОбработкаОдинаковыхЗначенийПорядка**. Строка, содержащее одно из:
  - "Вместе" ("Together") - обозначает, что для определения предыдущей и следующей записей используется последовательность записей.
  - "Отдельно" ("Separately") - обозначает, что предыдущая и следующие записи определяются на основании значений вычислений. Например, если полученная последовательность упорядочена по дате:
    1. 01 января 2001 Иванов М. 10
    2. 02 января 2001 Петров С. 20
    3. 02 января 2001 Сидоров Р. 30
    4. 03 января 2001 Петров С. 40При использовании обработки одинаковых значений порядка "Отдельно", предыдущей к записи 3 будет запись 2, а при использовании "Вместе" будет запись 2, а для "Вместе" - записи 2 и 3. Таким образом, сумма по текущей записи для "Отдельно" параметров Начало и Конец нельзя указывать смещение для позиций "Первая", "Последняя", "Предыдущая", "Следующая".

## Пример:

Получение отношения суммы по полю "Продажи.СуммаОборот" записи группировки к сумме того же поля во всей компоновке:  
Сумма(Продажи.СуммаОборот) / ВычислитьВыражение("Сумма(Продажи.СуммаОборот)", "ОбщийИтог").

В данном примере вычисляется значение текущей иерархии:

```
Выбор
    Когда Уровень() > 0
        Тогда ВычислитьВыражение("Ссылка", , "Иерархия")
    Иначе Null
Конец
```

### Примеры применения функции СКД - ВычислитьВыражение

## Примечания:

Функция учитывает отбор группировок, но не учитывает иерархические отборы. Функция не может применяться к группировкам, если в выражении используется выражение ВычислитьВыражение("Сумма(СуммаОборот)", , "ОбщийИтог") > 1000. Если запись предшествует начальной, то считается, что записи для расчета детальных данных и расчета агрегатных функций (параметр Группировка имеет значение "ОбщийИтог") считается, что записи для расчета детальных данных и расчета функций функции ВычислитьВыражение, в случае если выражение упорядочивания содержит поля, которые не могут быть вычислены на NULL.

## 3. ВычислитьВыражениеСГруппировкойМассив (EvalExpressionWithGrouping)

Элемент которого содержит результат вычисления выражения для группировки по указанному полю.

## Синтаксис:

ВычислитьВыражениеСГруппировкойМассив (Выражение, ВыраженияПолейГруппировки, ОтборЗаписей, ОтборГруппировок)

## Параметры:

- **Выражение** (Строка) - выражение, которое нужно вычислить. Например, "Сумма(СуммаОборот)";
- **ВыраженияПолейГруппировки** - выражения полей группировки, перечисленные через запятую. Например, "Контрагент.Имя";
- **ОтборЗаписей** - выражение, применяемое к детальным записям. Например, "ПометкаУдаления = Ложь". Если в данной компоновке данных возникнет ошибка;
- **ОтборГруппировок** - отбор, применяемый к групповым записям. Например: "Сумма(СуммаОборот) > &Параметр1".

## Пример:

Максимум(ВычислитьВыражениеСГруппировкойМассив ("Сумма(СуммаОборот)", "Контрагент"));

Компоновщик макета при генерации макета преобразовывает параметры функции в термины полей макета компоновки да НаборДанных.Контрагент.

Компоновщик макета при генерации выражений для вывода пользовательского поля, в выражении которого присутствует выражение таким образом, чтобы данные выводить представления и данные были упорядочены.

Например, для пользовательского поля с выражением:

ВычислитьВыражениеСГруппировкойМассив ("Сумма(СуммаОборот)", "Контрагент")

компоновщик макета сгенерирует для вывода следующее выражение:

СоединитьСтроки(Массив(Упорядочить(ВычислитьВыражениеСГруппировкойТаблицаЗначений ("Представление(Сумма(НаборДанных.СуммаОборот))

## 4. ВычислитьВыражениеСГруппировкойТаблицаЗначений (Evall

возвращает таблицу значений, каждый элемент которого содержит результат вычисления выражения для группировки по

## Синтаксис:

ВычислитьВыражениеСГруппировкойТаблицаЗначений (Выражение, ВыраженияПолейГруппировки, ОтборЗаписей, ОтборГруппировок)

## Параметры:

- Выражение** (Строка) - выражение, которое нужно вычислить. В строке может быть перечислено несколько выражений ключевое слово КАК и имя колонки таблицы значений. Например: "Контрагент, Сумма(СуммаОборот) Как ОбъемПродаж
- ВыраженияПолейГруппировки** - выражения полей группировки, перечисленные через запятую. Например, "Контр
- ОтборЗаписей** - выражение, применяемое к детальным записям. Например, "ПометкаУдаления = Ложь". Если в данно компоновки данных возникнет ошибка;
- ОтборГруппировок** - отбор, применяемый к групповым записям. Например: "Сумма(СуммаОборот) > &Параметры".

## Пример:

ВычислитьВыражениеСГруппировкойТаблицаЗначений ("Контрагент КАК Контрагент, Сумма(СуммаОборот) Как ОбъемПродаж", "Контрагент")

Результатом работы данной функции будет таблица значений с колонками Контрагент и ОбъемПродаж, в которых будут н: Компоновщик макета при генерации макета преобразовывает параметры функции в термины полей макета компоновки да НаборДанных.Контрагент.

Например, пользовательского поля с выражением:

ВычислитьВыражениеСГруппировкойТаблицаЗначений ("Контрагент, Сумма(СуммаОборот)", "Контрагент")

Компоновщик макета сгенерирует для вывода следующее выражение:

СоединитьСтроки(ПолучитьЧасть(Упорядочить(ВычислитьВыражениеСГруппировкойТаблицаЗначений ("НаборДанных.Контрагент, НаборДанных.Ко

Представление(НаборДанных.СуммаОборот), НаборДанных.ПолеУпорядочивание", "НаборДанных.Контрагент"), "5, 1, 3"), "2, 4"))

## 5. Уровень (Level) - функция предназначена для получения текущего уровня записи.

## Синтаксис:

Уровень()

## Пример:

Уровень()

## 6. НомерПоПорядку (SequenceNumber) - получить следующий порядковый номер.

## Синтаксис:

НомерПоПорядку()

## Пример:

НомерПоПорядку()

## 7. НомерПоПорядкуВГруппировке (SequenceNumberInGrouping) .

Пример:  
НомерПоПорядкуВГруппировке()

**8. Формат (Format)** - получить отформатированную строку переданного значения.

Синтаксис:  
Формат(Значение, ФорматнаяСтрока)

Параметры:

- **Значение** - выражение, которое требуется отформатировать;
- **ФорматнаяСтрока** - форматная строка задается в соответствии с форматной строкой 1С:Предприятие.

Пример:  
Формат(РасходныеНакладные.СуммаДок, "ЧДЦ=2")

**9. НачалоПериода (BeginOfPeriod)** - функция предназначена для выделения определенной д

Синтаксис:  
НачалоПериода(Дата, ТипПериода)

Параметры:

- **Дата** (Дата). Заданная дата;
- **ТипПериода** (Строка). Содержит одно из значений: Минута; Час; День; Неделя; Месяц; Квартал; Год; Декада; Полуто

Пример:  
НачалоПериода(ДатаВремя(2002, 10, 12, 10, 15, 34), "Месяц")  
Результат: 01.10.2002 0:00:00

**10. КонецПериода (EndOfPeriod)** - функция предназначена для выделения определенной даты

Синтаксис:  
КонецПериода(Дата, ТипПериода)

Параметры:

- **Дата** (Дата). Заданная дата;
- **ТипПериода** (Строка). Содержит одно из значений: Минута; Час; День; Неделя; Месяц; Квартал; Год; Декада; Полуто

Пример:  
КонецПериода(ДатаВремя(2002, 10, 12, 10, 15, 34), "Неделя")  
Результат: 13.10.2002 23:59:59

**11. ДобавитьКДате (DateAdd)** - функция предназначена для прибавления к дате некоторой велич

Синтаксис:  
ДобавитьКДате(Выражение, ТипУвеличения, Величина)

Параметры:

- **Выражение** (Дата). Исходная дата;
- **ТипУвеличения** (Строка). Содержит одно из значений: Минута; Час; День; Неделя; Месяц; Квартал; Год; Декада; Пол
- **Величина** (Число). На сколько необходимо увеличить дату, дробная часть игнорируется.

Пример:  
ДобавитьКДате(ДатаВремя(2002, 10, 12, 10, 15, 34), "Месяц", 1)

**12. РазностьДат (DateDifference)** - функция предназначена для получения разницы между двумя датами

**Синтаксис:**

РазностьДат(Выражение1 , Выражение2 , ТипРазности)

**Параметры:**

- **Выражение1** (Дата). Вычитаемая дата;
- **Выражение2** (Дата). Исходная дата;
- **ТипРазности** (Строка). Содержит одно из значений: Секунда; Минута; Час; День; Месяц; Квартал; Год.

**Пример:**

РАЗНОСТЬДАТ(ДАТАВРЕМЯ(2002, 10, 12, 10, 15, 34),  
ДАТАВРЕМЯ(2002, 10, 14, 9, 18, 06), "ДЕНЬ")

Результат: 2

**13. Подстрока (Substring)** - данная функция предназначена для выделения подстроки из строки.

**Синтаксис:**

Подстрока(Строка, Позиция, Длина)

**Параметры:**

- **Строка** (Строка). Строка, из которой выделяют подстроку;
- **Позиция** (Число). Позиция символа, с которого начинается выделяемая из строки подстрока;
- **Длина** (Число). Длина выделяемой подстроки.

**Пример:**

ПОДСТРОКА(Контрагенты.Адрес, 1, 4)

**14. ДлинаСтроки (StringLength)** - функция предназначена для определения длины строки.

**Синтаксис:**

ДлинаСтроки(Строка)

**Параметр:**

- **Строка** (Строка). Строка, длину которой определяют.

**Пример:**

Строка(Контрагенты.Адрес)

**15. Год (Year)** - данная функция предназначена для выделения года из значения типа Дата.

**Синтаксис:**

Год(Дата)

**Параметр:**

- **Дата** (Дата). Дата, по которой определяют год.

**Пример:**

ГОД(РасхНакл.Дата)

**16. Квартал (Quarter)** - данная функция предназначена для выделения номера квартала из значения тип

## Синтаксис:

Квартал(Дата)

## Параметр:

- Дата (Дата). Дата, по которой определяют квартал

## Пример:

КВАРТАЛ(РасхНакл.Дата)

**17. Месяц (Month)** - данная функция предназначена для выделения номера месяца из значения типа Дата. I

## Синтаксис:

Месяц(Дата)

## Параметр:

- Дата (Дата). Дата, по которой определяют месяц.

## Пример:

МЕСЯЦ(РасхНакл.Дата)

**18. ДеньГода (DayOfYear)** - данная функция предназначена для получения дня года из значения тип

## Синтаксис:

ДеньГода(Дата)

## Параметр:

- Дата (Дата). Дата, по которой определяют день года.

## Пример:

ДЕНЬГОДА(РасхНакл.Дата)

**19. День (Day)** - данная функция предназначена для получения дня месяца из значения типа Дата. День месяц

## Синтаксис:

День(Дата)

## Параметр:

- Дата (Дата). Дата, по которой определяют дня месяца.

## Пример:

ДЕНЬ(РасхНакл.Дата)

**20. Неделя (Week)** - данная функция предназначена для получения номера недели года из значения типа I

## Синтаксис:

Неделя(Дата)

## Параметр:

- Дата (Дата). Дата, по которой определяют номера недели.

## Пример:

**21. ДеньНедели (WeekDay)** - данная функция предназначена для получения дня недели из значения до 7(воскресенье).

Синтаксис:

ДеньНедели(Дата)

Параметр:

- Дата** (Дата). Дата, по которой определяют день недели .

Пример:

ДЕНЬНЕДЕЛИ(РасхНакл.Дата)

**22. Час (Hour)** - данная функция предназначена для получения часа суток из значения типа Дата. Час суток на

Синтаксис:

Час(Дата)

Параметр:

- Дата** (Дата). Дата, по которой определяют час суток.

Пример:

ЧАС(РасхНакл.Дата)

**23. Минута (Minute)** - данная функция предназначена для получения минуты часа из значения типа Дата

Синтаксис:

Минута(Дата)

Параметр:

- Дата** (Дата). Дата, по которой определяют минута часа.

Пример:

МИНУТА(РасхНакл.Дата)

**24. Секунда (Second)** - данная функция предназначена для получения секунды минуты из значения типа

Синтаксис:

Секунда(Дата)

Параметр:

- Дата** (Дата). Дата, по которой определяют секунды минуты.

Пример:

СЕКУНДА(РасхНакл.Дата)

**25. Выразить (Cast)** - данная функция предназначена для выделения типа из выражения, которое может с отличный от требуемого типа, будет возвращено значение NULL.

Синтаксис:

Выразить(Выражение, УказаниеТипа)

Параметры:

- **Выражение** - преобразуемое выражение;
- **УказаниеТипа** (Строка). Содержит строку типа. Например, "Число", "Строка" и т.п. Кроме примитивных типов данная попытка выразить к ссылке на указанную таблицу.

Пример:

Выразить(Данные.Реквизит1, "Число(10,3)")

**26. ЕстьNull (IsNull)** - данная функция возвращает значение второго параметра в случае, если значение первого параметра.

Синтаксис:

ЕстьNull(Выражение1, Выражение2)

Параметры:

- **Выражение1** - проверяемое значение;
- **Выражение2** - возвращаемое значение, если значение Выражение1 есть NULL.

Пример:

ЕстьNULL(Сумма(Продажи.СуммаОборот), 0)

**27. ACos** - вычисляет арккосинус в радианах.

Синтаксис:

ACos(Выражение)

Параметр:

- **Выражение** (Число). Значение косинуса (в диапазоне -1 ... 1), по которому определяется угол.

**28. ASin** - вычисляет арксинус в радианах.

Синтаксис:

ASin(Выражение)

Параметр:

- **Выражение** (Число). Значение синуса (в диапазоне -1 ... 1), по которому определяется угол.

**29. ATan** - вычисляет арктангенс в радианах.

Синтаксис:

ATan(Выражение)

Параметр:

- **Выражение** (Число). Значение тангенса, по которому определяется угол.

**30. Cos** - вычисляет косинус.

Синтаксис:

Cos(Выражение)

Параметр:



- **Выражение** (Число). Задается в радианах.

**31. Exp** - возведение числа e в степень.

Синтаксис:

Exp(Выражение)

Параметр:

- **Выражение** (Число). Значение степени.

**32. Log** - вычисляет натуральный логарифм.

Синтаксис:

Log(Выражение)

Параметр:

- **Выражение** (Число). Исходное число, больше 0.

**33. Log10** - вычисляет логарифм X по основанию 10.

Синтаксис:

Log10(Выражение)

Параметр:

- **Выражение** (Число). Исходное число, больше 0.

**34. Pow** - возведение в степень.

Синтаксис:

Pow(Основание, Показатель)

Параметры:

- **Основание** (Число). Основание операции возведения в степень.
- **Показатель** (Число). Показатель степени.

**35. Sin** - вычисляет синус.

Синтаксис:

Sin(Выражение)

Параметр:

- **Выражение** (Число). Задается в радианах.

**36. Sqrt** - вычисляет квадратный корень.

Синтаксис:

Sqrt(Выражение)

Параметр:

- **Выражение** (Число). Неотрицательное число.

**37. Tan** - вычисляет тангенс.

Синтаксис:

Tan(Выражение)

Параметр:

- Выражение** (Число). Значение синуса, по которому определяется угол.

**38. Окр (Round)** - округляет исходное число до нужной разрядности. Режим округления стандартный (1,5 как

Синтаксис:

Окр(Выражение, Разрядность)

Параметры:

- Выражение** (Число). Исходное число;
- Разрядность** (Число). Число знаков дробной части, до которых производится округление.

**39. Цел (Int)** - отсекает дробную часть числа.

Синтаксис:

Цел(Выражение)

Параметр:

- Выражение** (Число). Дробное число.

## 40. Функции общих модулей

Выражение механизма компоновки данных может содержать вызовы функций глобальных общих модулей конфигурации. требуется.

Пример:

СокращенноеНаименование(Документы.Ссылка, Документы.Дата, Документы.Номер)

В данном примере будет осуществлен вызов функции "СокращенноеНаименование" из общего модуля конфигурации. Отметим, что использование функций общих модулей разрешено только при указании соответствующего параметра проц Кроме того, функции общих модулей не могут быть использованы в выражениях пользовательских полей.

**41. Представление (Resentation)** - данная функция возвращает строковое представление перед возвращает само значение.

Если в качестве параметра используется массив или таблица значений, то функция возвращает строку, содержащую строки Если у какого либо элемента строковое представление пустое, то вместо его представления выводится строка "<Пустое зн

Пример:

Представление(Контрагент)

**42. Строка (String)** - данная функция преобразует переданное значение в строку.

Если в качестве параметра используется массив или таблица значений, то функция возвращает строку, содержащую строки Если у какого либо элемента строковое представление пустое, то вместо его представления выводится строка "<Пустое зн

Пример:

Строка(ДатаПродажи)

## 43. ЗначениеЗаполнено (ValueIsFilled)

Для значений NULL, Неопределено всегда возвращает Ложь.  
Для значений Булево всегда возвращает Истину.  
Для остальных типов возвращает Истину, если значение отличается от значения по умолчанию для данного типа.

Пример:

ЗначениеЗаполнено(ДатаДоставки)

## 44. УровеньВГруппировке (LevelInGroup)

- данная функция получает текущий уровень

Может быть использована для получения уровня вложенности записи в иерархической группировке.

Пример:

УровеньВГруппировке()

## 45. ТипЗначения(ValueType)

Синтаксис:

ТипЗначения(Выражение)

Параметр:

- Выражение** (Строка). Строковое значение типа.

Возвращает значение типа Тип, содержащее тип значения параметра функции.