

Использование функций, формул и вычислений в SAP BusinessObjects Web Intelligence

© 2011 SAP AG. Все права защищены. SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign и другие упомянутые здесь продукты и услуги SAP, а также соответствующие им логотипы являются торговыми марками или зарегистрированными торговыми марками SAP AG в Германии и других странах. Business Objects и логотип Business Objects, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius и другие продукты упомянутые здесь продукты и услуги Business Objects, а также соответствующие им логотипы являются торговыми марками или зарегистрированными торговыми марками Business Objects S.A. в США и других странах. Business Objects является компанией SAP. Все другие указанные продукты и услуги являются торговыми марками соответствующих компаний. Данные, содержащиеся в этом документе, предназначены только для информационных целей. Характеристики продуктов, поставляемых в разные страны, могут изменяться. Эти материалы могут быть изменены без предварительного уведомления. Материалы предоставлены компанией SAP AG и ее дочерними компаниями ("SAP Group") исключительно в информационных целях, без представления гарантии любого вида. SAP Group не несет ответственности за ошибки или пропуски в настоящих материалах. Все гарантии по продуктам и услугам SAP Group оговариваются в специальных гарантийных документах, которые прилагаются к соответствующим продуктам и услугам, имеющим гарантию. Ничто изложенное в данном документе не должно трактоваться как предоставление дополнительных гарантий

2011-05-06

Содержание

Глава 1	О данном руководстве	7
Глава 2	Использование стандартных и пользовательских вычислений	9
	Стандартные вычисления.....	10
	Использование формул для создания пользовательских вычислений.....	11
	Использование переменных для упрощения формул.....	12
	Работа с функциями.....	14
	Включение функций в ячейки.....	14
	Синтаксис функций.....	14
	Примеры функций.....	15
	Операторы функций и формул Web Intelligence.....	21
Глава 3	Описание контекстов вычисления	25
	Что из себя представляют контексты вычислений.....	26
	Входной контекст.....	27
	Выходной контекст.....	28
	Контексты вычислений по умолчанию.....	30
	Контексты по умолчанию в вертикальной таблице.....	31
	Контексты по умолчанию в горизонтальной таблице.....	33
	Контексты по умолчанию в кросс-таблице.....	33
	Контексты по умолчанию в разделе.....	34
	Контексты по умолчанию в разбиении.....	36
	Изменение контекста вычислений по умолчанию с расширенным синтаксисом.....	37
	Операторы расширенного синтаксиса.....	37

	Ключевые слова с расширенным синтаксисом в приложении Web Intelligence.....	41
Глава 4	Вычисление значений с помощью интеллектуальных мер	53
	Определение интеллектуальной меры.....	54
	Наборы группировок и интеллектуальные меры.....	54
	Принципы управления Web Intelligence наборами для группировки.....	55
	Интеллектуальные меры и область анализа.....	56
	Интеллектуальные меры и язык SQL.....	57
	Наборы группировок и оператор UNION.....	57
	Интеллектуальные меры и формулы.....	60
	Интеллектуальные меры и измерения, содержащие формулы.....	60
	Интеллектуальные меры в формулах.....	60
	Интеллектуальные меры и фильтры.....	61
	Интеллектуальные показатели и фильтры для измерений.....	61
	Интеллектуальные меры и фильтры детализации.....	62
	Интеллектуальные меры и вложенные фильтры OR.....	62
Глава 5	Функции, операторы и ключевые слова Web Intelligence	65
	Функции Web Intelligence.....	66
	Функции агрегирования.....	66
	Символьные функции.....	98
	Функции даты и времени.....	118
	Data Provider functions.....	134
	Функции Документа.....	149
	Логические функции.....	158
	Числовые функции.....	168
	Дополнительные.....	191
	Операторы функций и формул Web Intelligence.....	213
	Математические операторы.....	214

Содержание

Условные операторы.....	214
Логические операторы.....	215
Специальные операторы функций.....	218
Операторы расширенного синтаксиса.....	227
Ключевые слова с расширенным синтаксисом в приложении Web Intelligence.....	232
Ключевое слово блока.....	232
Ключевое слово "Тело".....	234
Ключевое слово "Разбиение".....	235
Ключевое слово Report.....	236
Ключевое слово "Раздел".....	238
Округление и обрезание чисел с помощью Web Intelligence.....	239

Глава 6 **Устранение ошибок в формулах Web Intelligence** **243**

Сообщения об ошибках формул и информационные сообщения.....	244
#COMPUTATION.....	244
#CONTEXT.....	244
#DATASYNC.....	245
#DIV/0.....	245
#EXTERNAL.....	246
#INCOMPATIBLE.....	246
#MULTIVALUE.....	246
#OVERFLOW.....	247
#PARTIALRESULT.....	247
#RANK.....	248
#RECURSIVE.....	248
#SECURITY.....	249
#SYNTAX.....	249
#TOREFRESH.....	249
#UNAVAILABLE.....	250
#ERROR.....	250

Содержание

Глава 7	Сравнение значений с помощью функций Web Intelligence	251
	Сравнение значений с помощью функции Previous.....	252
	Сравнение значений с помощью функции RelativeValue.....	252
	Измерения срезов и функция RelativeValue.....	254
	Измерения срезов и разделы.....	256
	Порядок измерений срезов.....	258
	Сортировка измерений срезов.....	261
	Использование функции RelativeValue в кросс-таблицах.....	263
Приложение А	Дополнительная информация	265
Указатель		269



О данном руководстве



1



В руководстве *Использование функций, формул и вычислений в SAP BusinessObjects Web Intelligence* представлена подробная информация о расширенных возможностях вычислений в Web Intelligence. В нем также представлен указатель синтаксиса функций и операторов Web Intelligence.

В данном руководстве представлены общие сведения, без ссылок на интерфейс Web Intelligence. Чтобы получить информацию о работе с функциями, связанными с вычислениями, в документах Web Intelligence (например, о добавлении в отчет переменной или формулы), см. интерактивную справку или следующие руководства: *Выполнение интерактивного анализа с помощью приложения SAP BusinessObjects Web Intelligence*, *Создание отчетов с помощью панели отчетов Java SAP BusinessObjects Web Intelligence* и *Создание отчетов с помощью SAP BusinessObjects Web Intelligence Rich Client*.

Для получения информации о создании пользовательских функций для Web Intelligence см. руководство *Элементы расширения SAP BusinessObjects Web Intelligence: создание пользовательских функций*.



Использование стандартных и пользовательских вычислений



Функции стандартных вычислений служат для выполнения быстрых вычислений с данными в отчетах Web Intelligence. Если возможности стандартных вычислений становятся недостаточными, можно использовать язык формул Web Intelligence для создания пользовательских вычислений.

Стандартные вычисления

Функции стандартных вычислений служат для выполнения быстрых вычислений с данными в отчетах Web Intelligence. Существуют следующие стандартные вычисления:

Вычисление	Описание
Сумма	Вычисляет сумму выбранных данных.
Количество	Подсчитывает все строки для объекта меры или подсчитывает отдельные строки для объекта измерения или сведений.
Average	Вычисляет среднее значение для данных.
Минимум	Отображает минимальное значение среди выбранных данных.
Максимум	Отображает максимальное значение среди выбранных данных.

Вычисление	Описание
Процентное соотношение	<p>Отображает выбранные данные в виде процента от суммарного значения. Результаты вычисления процентов отображаются в таблице в дополнительном столбце или дополнительной строке.</p> <p>Примечание: Значения в процентах вычисляются для выбранной меры по отношению к общим результатам для этой меры в таблице или в разрыве. Для вычисления процентного значения одной меры по отношению к другой мере необходимо создать формулу.</p>
По умолчанию	Применяет функцию агрегирования по умолчанию к стандартной мере или функцию агрегирования по базе данных для интеллектуальной меры.

При применении стандартных вычислений к столбцу таблицы результат вычислений отображается в столбце в нижнем колонтитуле. Если с одним столбцом выполняется несколько вычислений, Web Intelligence добавляет нижний колонтитул для каждого из результатов вычислений.

Использование формул для создания пользовательских вычислений

Пользовательские вычисления позволяют добавлять дополнительные вычисления к отчету поверх основных объектов и стандартные вычисления, доступные в Web Intelligence.

Чтобы добавить пользовательское вычисление, необходимо создать формулу, которую Web Intelligence определит после запуска отчета. Формула может состоять из основных переменных отчета, функций, операторов и контекстов вычислений.

Пользовательское вычисление – это формула, которая состоит из объектов отчета, функций и операторов. В формулах содержится контекст вычисления, который может быть отображен при необходимости.

Пример: Отображение среднего дохода от каждой продажи

Если в отчете есть объекты "Доход от продаж" и "Проданное количество", необходимо добавить в отчет доход от каждой продажи. Для получения данного значения при вычислении `[Доход от продаж] / [Проданное количество]` выполняется деление дохода на количество проданных товаров, что дает величину дохода с каждого товара.

См. также

- *Что из себя представляют контексты вычислений*

Использование переменных для упрощения формул

Если формула сложная, можно использовать переменные, чтобы упростить ее. С помощью переменных можно разбить сложную формулу на управляемые части, чтобы облегчить ее чтение, а также сделать процесс создания формул менее подверженным ошибкам.

В формуле можно использовать ранее созданные переменные точно так же, как и остальные объекты отчета. Переменные отображаются в редакторе формул в папке "Переменные".

В формуле можно вводить имя переменной или перетаскивать переменную на панель инструментов "Формула", как и в случае с другими объектами отчета.

Пример: Создайте формулу, которая возвращает статистическую дисперсию

Дисперсия – это статистический термин. Дисперсия множества значений служит мерой рассеяния этих значений относительно их среднего. В приложении Web Intelligence есть функция `Var()`, которая вычисляет дисперсию за один шаг, но ручное вычисление дисперсии

дает хороший пример того, как можно упростить сложную формулу, используя переменные. Чтобы вычислить дисперсию вручную, необходимо выполнить следующие действия:

- вычислить среднее количество проданных товаров;
- вычислить разность между каждым количеством проданных товаров и средним значением, а затем возвести это значение в квадрат;
- сложить все полученные квадраты разностей;
- разделить итог на количество значений минус единица.

Предположим, есть отчет по количеству товаров, проданных за квартал, и в него требуется включить дисперсию. Если для упрощения не использовать переменные, формула будет выглядеть следующим образом:

```
Sum(((Quantity sold] - Average([Quantity sold] ForEach
[Quarter]) In Report)*([Quantity sold] -
Average([Quantity sold] ForEach [Quarter]) In Report))
In [Quarter])/(Count ([Quantity sold] ForEach
[Quarter]) - 1)
```

Данная формула безусловно, является громоздкой. С помощью переменных можно упростить ее до формулы,

```
Sum ([Difference Squared])/[Number of Observations] -
1)
```

что гораздо легче воспринимать. Такой упрощенный вариант формулы дает представление более высокого уровня о том, что вычисляется в формуле, вместо того чтобы глубоко вникать в сбивающие с толку детали. Затем можно проверить формулы для переменных, на которые есть ссылки в формулах более высокого уровня, что даст более полное представление об их структуре.

Например, формула ссылается на переменную Difference Squared, которая, в свою очередь, ссылается на переменную Average Sold. Проанализировав формулы для переменных Difference Squared и Average sold, можно исследовать иерархическую структуру формулы и тем самым лучше понять механизм ее работы.

Работа с функциями

В пользовательском вычислении содержатся только объекты отчетов, например [Доход от продаж]/[Количество продаж]. В вычислениях, кроме объектов отчета, также могут содержаться функции.

Функция получает ноль или несколько значений в качестве входных данных и возвращает выходные данные на основе этих значений. Например, функция `Sum` суммирует все значения в мере и выводит результат. Формула `Sum([Доход от продаж])` выводит итог доходов от продаж. В данном случае входные данные функции – это мера "Доход от продаж", а выходные данные функций – сумма всех мер "Доход от продаж".

См. также

- *Операторы функций и формул Web Intelligence*
- *Функции Web Intelligence*

Включение функций в ячейки

Текст в ячейках отчета всегда начинается со знака "=". Буквенный текст отображается в кавычках, а формулы – без кавычек. Например, формула `Average([Прибыль])` отображается в ячейке как `=Average([Прибыль])`. Текст "Средняя прибыль?" отображается как `"Средняя прибыль?"`.

В ячейке можно использовать только текст или смесь формул и текста, с помощью оператора "+". Если необходимо, чтобы перед средней прибылью в ячейке отображался текст "Средняя прибыль:", текст ячейки должен выглядеть следующим образом: `"Средняя прибыль: " + Average([Прибыль])`

Обратите внимание на пробел в конце текстовой строки, который позволяет отделить в ячейке текст от значения.

Синтаксис функций

Чтобы использовать функцию, необходимо знать ее имя, количество необходимых для нее входных переменных и типы данных этих

переменных. Также необходимо знать тип данных, который выводится функцией.

Например, функция Sum использует числовой объект в качестве входной переменной (например меру, отображающую доход от продаж) и выводит числовые данные (сумму всех значений объекта меры).

Ниже представлен синтаксис функции Abs:

```
num Abs (number)
```

Данный синтаксис говорит о том, что функция Abs использует только одно число в качестве входной переменной и возвращает число в качестве выходных данных.

При выборе функции в редакторе формул отображается ее синтаксис.

Примеры функций

Пример: Отображение входа приглашений с помощью функции UserResponse

В отчете содержатся сведения о доходах от продаж за год и квартал. Объект состояния также отображается в данных отчета, хотя в самом отчете он не отображается. Когда пользователь выполняет отчет, ему выводится приглашение и он должен выбрать состояние. Необходимо отобразить состояние, выбранное в заголовке отчета. Если поставщик данных называется eFashion, а текст в приглашении звучит "Выберите состояние", формула для заголовка выглядит следующим образом:

```
"Quarterly Revenues for " + UserResponse (
  "eFashion"; "Choose a State")
```

Отчет выглядит следующим образом:

Узловые доходы для Уолл-стрит

Year	Quarter	Sales revenue
2001	Q1	\$591 454
2001	Q2	\$541 469
2001	Q3	\$107 086
2001	Q4	\$133 366
2001	Total	\$782 220

Year	Quarter	Sales revenue
2002	Q1	\$534 267
2002	Q2	\$524 722
2002	Q3	\$539 973
2002	Q4	\$531 067
2002	Total	\$1 930 030

Year	Quarter	Sales revenue
2003	Q1	\$559 699
2003	Q2	\$554 724
2003	Q3	\$573 069
2003	Q4	\$539 917
2003	Total	\$1 930 030

Пример: Вычисление процентного соотношения с помощью функции Percentage

В Web Intelligence функция Percentage используется для вычисления процентного соотношения. С помощью данной функции вычисляется процентное соотношение числа по отношению к окружающему контексту. Например, в следующей таблице отображаются доходы по годам и по кварталам. В столбце процентного соотношения содержится формула Процентное соотношение ([Доход от продаж]).

Year	Quarter	Sales revenue	Percentage
2001	Q1	\$2 660 700	0,07
2001	Q2	\$2 279 003	0,06
2001	Q3	\$1 367 841	0,04
2001	Q4	\$1 788 580	0,05
2002	Q1	\$3 326 172	0,09
2002	Q2	\$2 840 651	0,08
2002	Q3	\$2 879 303	0,08
2002	Q4	\$4 186 120	0,12
2003	Q1	\$3 742 989	0,10
2003	Q2	\$4 006 718	0,11
2003	Q3	\$3 953 395	0,11
2003	Q4	\$3 356 041	0,09
		Сумма:	1

В данном случае формула вычисляет каждый доход как процентное соотношение общего дохода. Окружающий контекст – это общий доход; это единственная необходимая цифра дохода, кроме его разбиения по годам и кварталам в таблице.

Если отчет разделен на разделы по годам, окружающий контекст вне таблицы становится общим доходом в разделе.

2001

Year	Quarter	Sales revenue	Percentage
2001	Q1	\$2 660 700	0,33
2001	Q2	\$2 279 003	0,28
2001	Q3	\$1 367 841	0,17
2001	Q4	\$1 788 580	0,22
		Сумма:	1

Если ячейка "Процентное соотношение" помещена вне таблицы, но внутри раздела, окружающий контекст становится общим доходом. В данном случае функция Percentage вычисляет общий доход для раздела в качестве процентного соотношения.

2001	0,22
------	------

Year	Quarter	Sales revenue
2001	Q1	\$2 660 700
2001	Q2	\$2 279 003
2001	Q3	\$1 367 841
2001	Q4	\$1 788 580

2002	0,36
------	------

Year	Quarter	Sales revenue
2002	Q1	\$3 326 172
2002	Q2	\$2 840 651
2002	Q3	\$2 879 303
2002	Q4	\$4 186 120

Пример: Вычисление процентного соотношения с помощью функции Sum

Можно усилить контроль над контекстом, в котором процентное соотношение вычисляется с помощью функции Sum, вместо функции Percentage. Если разделить одно число во множестве чисел на сумму этих чисел, то можно получить процентное соотношение итога; например, формула [Доход от продаж]/Sum([Доход от продаж]) представляет доход от продаж в качестве процентного соотношения от общего дохода.

В следующей таблице в столбце "Процентное соотношение итога" есть формула:

```
[Sales revenue]/(Sum([Sales revenue] In Report))
```

в столбце "Процентное соотношение года" также есть формула:

```
[Sales revenue]/(Sum([Sales revenue] In Section))
```

2001

Year	Quarter	Sales revenue	Percentage of Total	Percentage of Year
2001	Q1	\$2 660 700	0,07	0,33
2001	Q2	\$2 279 003	0,06	0,28
2001	Q3	\$1 367 841	0,04	0,17
2001	Q4	\$1 788 580	0,05	0,22

В данных формулах используются ключевые слова Report и Section расширенного синтаксиса, чтобы с помощью функции Sum можно было вычислять общий доход и годовой доход соответственно.

См. также

- [Изменение контекста вычислений по умолчанию с расширенным синтаксисом](#)

Упрощение формулы для дисперсии с помощью переменных

Дисперсия – это статистический термин. Дисперсия множества значений служит мерой рассеяния этих значений относительно их среднего. В приложении Web Intelligence есть функция Var(), которая вычисляет дисперсию за один шаг, но ручное вычисление дисперсии дает хороший пример того, как можно упростить сложную формулу, используя переменные. Чтобы вычислить дисперсию вручную, необходимо выполнить следующие действия:

- вычислить среднее количество проданных товаров;
- вычислить разность между каждым количеством проданных товаров и средним значением, а затем возвести это значение в квадрат;
- сложить все полученные квадраты разностей;
- разделить итог на количество значений минус единица.

Предположим, есть отчет по количеству товаров, проданных за квартал, и в него требуется включить дисперсию. Если для упрощения не

использовать переменные, формула будет выглядеть следующим образом:

```
Sum(((Quantity sold) - Average(Quantity sold) ForEach
[Quarter]) In Report)*(Quantity sold) -
Average(Quantity sold) ForEach [Quarter]) In Report))
In [Quarter])/(Count (Quantity sold) ForEach [Quarter])
- 1)
```

безусловно, формула выглядит очень громоздкой.

Создание формулы для дисперсии

Создание формулы для дисперсии выполняется за несколько шагов. Эти несколько шагов можно свести к одному с помощью одной переменной. Необходимо создать следующие переменные:

- среднее количество проданных товаров;
- количество наблюдений (то есть количество отдельных значений количества проданных товаров);
- разность между наблюдением и средним значением, возведенная в квадрат;
- сумма этих квадратов разностей, разделенная на количество наблюдений минус единица

Формулы с переменными имеют следующий вид.

Переменная	Формула
Average Sold	Average([Quantity Sold] In ([Quarter])) In Report
Number of Observations	Count([Quantity Sold] In ([Quarter])) In Report
Difference Squared	Power((Quantity sold) - Average Sold);2)
Отклонение	Sum([Difference Squared] In ([Quarter]))/([Number of Observations] - 1)

Окончательная формула принимает вид

```
Sum ([Difference Squared])/[Number of Observations] - 1)
```

что гораздо легче воспринимать. Такой упрощенный вариант формулы дает представление более высокого уровня о том, что вычисляется в формуле, вместо того чтобы глубоко вникать в сбивающие с толку детали. Затем можно проверить формулы для переменных, на которые есть ссылки в формулах более высокого уровня, что даст более полное представление об их структуре.

Например, формула ссылается на переменную Difference Squared, которая, в свою очередь, ссылается на переменную Average Sold. Проанализировав формулы для переменных Difference Squared и Average sold, можно исследовать иерархическую структуру формулы и тем самым лучше понять механизм ее работы.

Операторы функций и формул Web Intelligence

Операторы связывают различные элементы формул. Формулы могут содержать математические, условные, логические, специальные функциональные операторы или операторы расширенного синтаксиса.

Математические операторы

Математические операторы знакомы по арифметическим вычислениям, выполняемым ежедневно. Это операторы сложения (+), вычитания (-), умножения (*), деления (/), которые позволяют выполнять математические операции в формуле. В формуле [Доход продаж] - [Издержки при продажах] содержится математический оператор вычитания.

Примечание:

При использовании с символьными строками оператор "+" ставится оператором конкатенации строк. А это значит, что он объединяет символьные строки. Например, формула "Иван" + "Иванов" возвращает значение "Иван Иванов".

Условные операторы

Условные операторы определяют тип сравнения двух значений.

Оператор	Описание
=	Равно
>	Больше
<	Меньше
>=	Больше или равно
<=	Меньше или равно
<>	Не равно

Условные операторы используются с функцией If, например:

```
If [Revenue]>10000 Then "High" Else "Low"
```

которая возвращает значение "Высокая" для всех строк, в которых указанная прибыль больше или равна 10000, и значение "Низкая" для всех остальных строк.

Логические операторы

В приложении Web Intelligence используются следующие логические операторы: And, Or, Not, Between и Inlist. Логические операторы используются в булевых выражениях, возвращающих значения Истина или Ложь.

Контекстные операторы

Контекстные операторы образуют часть расширенного синтаксиса вычислений. Расширенный синтаксис позволяет определять, какие измерения будут учитываться формулой или мерой при вычислении.

Специальные операторы функций

Некоторые функции Web Intelligence могут использовать определенный набор операторов в качестве аргументов. Например, функция `Previous` может использовать оператор `Self`.

Аргументы всех функций заключаются в скобки: `)` и `(`. В функциях, работающих с несколькими параметрами, для их разделения используется знак `;`.



Описание контекстов вычисления



Что из себя представляют контексты вычислений

Контекст вычисления – это данные, которые учитывает вычисление, чтобы сгенерировать результат. В Web Intelligence это значит, что значение, предоставленное мерой, определяется с помощью измерения, которое используется для вычисления меры.

В отчете содержится два вида объектов.

- Измерения представляют бизнес-данные, которые генерируют цифры. Филиалы магазина, года или регионы – это примеры данных измерений. Например, филиал магазина, год или регион могут генерировать доход. Можно говорить о доходе по магазину, году или региону.
- Меры – это числовые данные, сгенерированные с помощью данных измерений. Примеры мер: доход и количество продаж. Например, можно говорить о количестве продаж в определенном магазине.

Меры также можно генерировать с помощью комбинаций данных измерений. Например, можно говорить о доходе, сгенерированном в определенном магазине в 2005 году.

Контекст вычисления меры состоит из двух компонентов:

- измерение или список измерений, которые определяют значение меры
- часть данных измерения, которая определяет значение меры

Контекст вычисления состоит из двух компонентов:

- Контекст ввода
- Контекст вывода

См. также

- *Входной контекст*
- *Выходной контекст*

Входной контекст

Входной контекст меры или формулы – это список измерений, которые подаются на вычисление.

Список измерений во входном контексте помещается внутри скобок функции, которая выводит значение. Список измерений также должен заключаться в скобки (даже если он содержит только одно измерение), а измерения должны разделяться точками с запятой.

Пример: Определение входного контекста

В отчете с разделами "Год" и блоком в каждом разделе со столбцами "Заказчик" и "Доход" входные контексты будут следующими.

Часть отчета	Входной контекст
Верхние колонтитулы разделов и нижние колонтитулы блоков	Год
Строки блока	"Год", "Заказчик"

Другими словами, в верхних колонтитулах разделов и нижних колонтитулах блоков выводится прибыль по годам, а в каждой строке блока – прибыль по годам и заказчику (прибыль, полученная от данного заказчика за рассматриваемый год).

При явном указании в формуле входные контексты будут следующими:

```
Sum ([Revenue] In ([Year]))
```

```
Sum ([Revenue] In ([Year]; [Customer]))
```

Таким образом, измерения во входном контексте оказываются внутри круглых скобок той функции (в данном случае – Sum), для которой указан входной контекст.

Выходной контекст

С помощью выходного контекста формула выводит значение, если она находится в нижнем колонтитуле блока с разбиением.

Пример: Определение выходного контекста

В следующем отчете отображена прибыль годам и по кварталам, с разбиением по году и минимальной прибылью, вычисленной по годам.

Year	Quarter	Sales revenue
2001	Q1	\$2 660 699,50
2001	Q2	\$2 279 003,00
2001	Q3	\$1 367 840,70
2001	Q4	\$1 788 580,40
2001		
	Минимум:	\$1 367 840,70

Year	Quarter	Sales revenue
2002	Q1	\$3 326 172,20
2002	Q2	\$2 840 650,80
2002	Q3	\$2 879 303,00
2002	Q4	\$4 186 120,00
2002		
	Минимум:	\$2 840 650,80

Year	Quarter	Sales revenue
2003	Q1	\$3 742 988,90
2003	Q2	\$4 006 717,50
2003	Q3	\$3 953 395,30
2003	Q4	\$3 356 041,10
2003		
	Минимум:	\$3 356 041,10

Что необходимо сделать, чтобы отобразить минимальный доход по годам в блоке с разбиением? Это можно осуществить, определив выходной контекст в формуле. В этом случае данная формула выглядит следующим образом.

```
Min ([Revenue]) In ([Year])
```

То есть выходной контекст отображается после круглых скобок функции, чей выходной контекст указывает пользователь. В данном случае с помощью выходного контекста Web Intelligence вычисляет минимальный доход за каждый год.

Если добавить дополнительный столбец с данной формулой в блок, результат будет выглядеть следующим образом.

Year	Quarter	Sales revenue	Min by Year
2001	Q1	\$2 660 699,50	\$1 367 840,70
2001	Q2	\$2 279 003,00	\$1 367 840,70
2001	Q3	\$1 367 840,70	\$1 367 840,70
2001	Q4	\$1 788 580,40	\$1 367 840,70
2002	Q1	\$3 326 172,20	\$2 840 650,80
2002	Q2	\$2 840 650,80	\$2 840 650,80
2002	Q3	\$2 879 303,00	\$2 840 650,80
2002	Q4	\$4 186 120,00	\$2 840 650,80
2003	Q1	\$3 742 988,90	\$3 356 041,10
2003	Q2	\$4 006 717,50	\$3 356 041,10
2003	Q3	\$3 953 395,30	\$3 356 041,10
2003	Q4	\$3 356 041,10	\$3 356 041,10

Можно увидеть, что в столбце "Минимум за каждый год" содержатся данные о минимальных доходах, которые отображаются в нижних колонтитулах разбиения предыдущего отчета.

Обратите внимание, что в данном примере выходной контекст не указан, так как это контекст по умолчанию (год, квартал) для блока. Другими словами, с помощью выходного контекста Web Intelligence определяет, какую прибыль по годам и кварталам следует выводить. Если включить все указанные входные и выходные формулы, данная формула выглядит следующим образом.

```
Min ([Sales Revenue] In ([Year]; [Quarter])) In ([Year])
```

Проще говоря, с помощью данной формулы Web Intelligence "вычисляет доходы за каждый год и квартал, а потом выводит наименьшие объемы доходов каждый год"?

Что произойдет, если не определить выходной контекст в столбце "Минимум за год"? В этом случае данные цифры будут совпадать с цифрами в столбце "Доход от продаж". Причина Следует помнить, что

контекст по умолчанию в блоке содержит измерения в данном блоке. Минимальная прибыль за каждый год/квартал совпадает с обычной прибылью за каждый год/квартал, так как существует только один показатель прибыли для каждой комбинации "год/квартал".

Контексты вычислений по умолчанию

В зависимости от того, где поместить меру или формулу, Web Intelligence назначает контекст вычисления для меры по умолчанию.

Семантика мер изменяется динамически. Это означает, что значения, возвращенные мерой, зависят от измерений, с которыми она связана. Данная комбинация измерений представляет контекст вычисления.

Web Intelligence связывает контекст по умолчанию с мерой, в зависимости от того, где она размещается. С помощью расширенного синтаксиса можно изменить контекст по умолчанию. Другими словами, можно определить набор измерений, который используется для создания меры. Это и есть определение контекста вычисления.

Пример: Контексты по умолчанию в отчете

В данном примере описывается контекст вычисления мер по умолчанию в простом отчете. В отчете отображается доход, созданный покупателями. Отчет разбит на разделы по годам.

2005	Всего: 8000
------	-------------

Заказчик	Доход
Харрис	1000
Джонс	3000
Уолш	4000

Заказчик	Доход
Всего:	8000

Всего в отчете: 8000

В таблице, указанной выше, содержится контекст вычисления мер в следующем отчете.

Мера	Значение	Контекст
Всего в отчете	20000	Итоговое значение общего дохода в отчете
Итоговое значение в верхнем колонтитуле	8000	Год
Итого по всем заказчикам	1000, 3000, 4000	Год; Заказчик
Итоговое значение в нижнем колонтитуле	8000	Год

См. также

- [Что из себя представляют контексты вычислений](#)
- [Изменение контекста вычислений по умолчанию с расширенным синтаксисом](#)

Контексты по умолчанию в вертикальной таблице

Вертикальная таблица – это стандартная таблица отчета с верхними колонтитулами сверху, данными, которые идут сверху вниз, и нижними колонтитулами внизу. Контексты по умолчанию в нижней таблице.

Если вычисление в...	Входной контекст – это...	Выходной контекст – это...
Верхний колонтитул	Измерения и меры используются для генерирования содержимого блока.	Все данные агрегированы, функция вычисления возвращает единственное значение.
Содержимое блока	Измерения и меры используются для генерирования текущей строки.	Идентично входному контексту.
Нижний колонтитул	Измерения и меры используются для генерирования содержимого блока.	Все данные агрегированы, функция вычисления возвращает единственное значение.

Пример: Контексты по умолчанию в вертикальной таблице

В следующем отчете отображены контексты по умолчанию в вертикальной таблице.

Year	Quarter	Sales revenue	\$36387203
2001	Q1	\$2 660 700	\$2 660 699,50
2001	Q2	\$2 278 693	\$2 278 693,40
2001	Q3	\$1 367 841	\$1 367 840,70
2001	Q4	\$1 788 580	\$1 788 580,40
2002	Q1	\$3 326 172	\$3 326 172,20
2002	Q2	\$2 840 651	\$2 840 650,80
2002	Q3	\$2 879 303	\$2 879 303,00
2002	Q4	\$4 186 120	\$4 186 120,00
2003	Q1	\$3 742 989	\$3 742 988,90
2003	Q2	\$4 006 718	\$4 006 717,50
2003	Q3	\$3 953 395	\$3 953 395,30
2003	Q4	\$3 356 041	\$3 356 041,10
Сумма:		\$36 387 203	

Контексты по умолчанию в горизонтальной таблице

Горизонтальная таблица выглядит как вертикальная, повернутая на 90 градусов. Верхние колонтитулы отображаются слева, данные идут слева направо, а нижние колонтитулы отображаются справа. Контексты по умолчанию для горизонтальной таблицы выглядят так же, как и для вертикальной.

Контексты по умолчанию в кросс-таблице

В кросс-таблице отображаются данные в виде матрицы с мерами, которые отображаются на пересечении измерений. Контексты по умолчанию в кросс-таблице.

Вычисление находится в...	Входной контекст – это...	Выходной контекст – это...
Верхний колонтитул	Измерения и меры используются для генерирования содержимого блока.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.
Содержимое блока	Измерения и меры используются для генерирования содержимого блока.	Идентично входному контексту.
Нижний колонтитул	Измерения и меры используются для генерирования содержимого блока.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.
Нижний колонтитул VBody	Измерения и меры используются для генерирования текущего столбца.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.

Вычисление находится в...	Входной контекст – это...	Выходной контекст – это...
Нижний колонтитул HBody	Измерения и меры используются для генерирования текущей строки.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.
VFooter	Идентично нижнему колонтитулу.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.
HFooter	Идентично нижнему колонтитулу.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.

Пример: Контексты по умолчанию в кросс-таблице

В следующем отчете отображены контексты по умолчанию в кросс-таблице.

		FY2000	FY2000	FY2000	FY2000	1 115 730
		Q1	Q2	Q3	Q4	1 115 730
France	259 170	61 895	76 555	70 080	50 640	259 170
US	856 560	196 831	189 886	234 574	235 269	856 560
Сумма:	1 115 730	258 726	266 441	304 654	285 909	1 115 730

Контексты по умолчанию в разделе

Раздел состоит из верхнего колонтитула, текста сообщения и нижнего колонтитула. Контексты по умолчанию в разделе:

Вычисление находится в...	Входной контекст – это...	Выходной контекст – это...
Тело	Измерения и меры в отчете отфильтрованы, чтобы ограничить данные до данных раздела.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.

Пример: Контексты по умолчанию в разделе

В следующем отчете отображены контексты по умолчанию в кросс-таблице.

2001 8 096 123,60

Quarter	Sales revenue	
Q1	\$2 660 700	8 096 123,60
Q2	\$2 279 003	8 096 123,60
Q3	\$1 367 841	8 096 123,60
Q4	\$1 788 580	8 096 123,60
Сумма:	\$8 096 123,60	

2002 13 232 246

Quarter	Sales revenue	
Q1	\$3 326 172	13 232 246,00
Q2	\$2 840 651	13 232 246,00
Q3	\$2 879 303	13 232 246,00
Q4	\$4 186 120	13 232 246,00
Сумма:	\$13 232 246,00	

2003 15 059 142,80

Quarter	Sales revenue	
Q1	\$3 742 989	15 059 142,80
Q2	\$4 006 718	15 059 142,80
Q3	\$3 953 395	15 059 142,80
Q4	\$3 356 041	15 059 142,80
Сумма:	\$15 059 142,80	

Контексты по умолчанию в разбиении

Разбиение состоит из верхнего колонтитула, текста сообщения и нижнего колонтитула. Контексты по умолчанию в разбиении:

Вычисление находится в...	Входной контекст – это...	Выходной контекст – это...
Верхний колонтитул	Текущий экземпляр разбиения.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.
Нижний колонтитул	Текущий экземпляр разбиения.	Все данные агрегируются, затем функция вычисления возвращает единственное значение.

Пример: Контексты по умолчанию в разбиении

В следующем отчете отображены контексты по умолчанию в разбиении:

Year	Quarter	\$8096123
	Q1	\$2 660 700
	Q2	\$2 279 003
	Q3	\$1 367 841
	Q4	\$1 788 580
2001		
	Сумма:	\$8 096 124

Year	Quarter	\$13232246
	Q1	\$3 326 172
	Q2	\$2 840 651
	Q3	\$2 879 303
	Q4	\$4 186 120
2002		
	Сумма:	\$13 232 246

Изменение контекста вычислений по умолчанию с расширенным синтаксисом

В расширенном синтаксисе используются операторы контекста, которые добавляются к формуле или мере для указания контекста вычислений. Контекст меры или формулы состоит из входного и выходного контекста.

Операторы расширенного синтаксиса

Контексты ввода и вывода задаются явным образом при помощи операторов контекста. В следующей таблице перечислены операторы контекста:

Оператор	Описание
In	Определяет подробный список измерений для использования в контексте.
ForEach	Добавляет измерения в контекст по умолчанию
ForAll	Убирает измерения из контекста по умолчанию

Операторы ForAll и ForEach могут оказаться полезными, когда в контексте по умолчанию содержится множество измерений. Часто может быть проще добавлять или удалять измерения из контекста с помощью операторов ForAll и ForEach, чем создавать список с использованием In.

Оператор In Context

Оператор In Context определяет измерения в контексте в явной форме.

Пример: Использование оператора In Context для определения измерений в контексте

В этом примере у нас есть отчет, в котором отображается "Год" и "Доход с продаж". В поставщике данных также содержится объект "Квартал", однако это измерение в блок не включено. Вместо этого, требуется включить дополнительный столбец для отображения максимального значения дохода за каждый квартал каждого года. Отчет выглядит следующим образом:

Year	Sales revenue	Max Quarterly Revenue
2001	\$8 096 123,60	\$2 660 699,50
2002	\$13 232 246,00	\$4 186 120,00
2003	\$15 059 142,80	\$4 006 717,50

Узнать откуда берутся значения в столбце "Максимальный доход за квартал" в этом блоке можно просмотрев этот блок в паре с блоком, содержащим измерение "Квартал":

Year	Quarter	Sales revenue
	Q1	\$2660700
	Q2	\$2279003
	Q3	\$1367841
	Q4	\$1788580
2001		
	Макс.:	2660699.5

Year	Quarter	Sales revenue
	Q1	\$3326172
	Q2	\$2840651
	Q3	\$2879303
	Q4	\$4186120
2002		
	Макс.:	4186120

Year	Quarter	Sales revenue
	Q1	\$3742989
	Q2	\$4006718
	Q3	\$3953395
	Q4	\$3356041
2003		
	Макс.:	4006717.5

Столбец "Максимальный доход за квартал" содержит максимальные показатели дохода за каждый год. Например, K4 содержит значение максимального дохода в 2002 году, таким образом, "Максимальный доход за квартал" в строке 2002 года отобразит K4.

Использование оператора In, формула для "Максимального дохода за квартал"

```
Max ([Sales Revenue] In ([Year];[Quarter])) In ([Year])
```

По этой формуле Web Intelligence рассчитывает максимальных доход с продаж для каждой пары "год/квартал", а затем выводит полученное значение по годам.

Примечание:

Так как по умолчанию контекстом для вывода является "Год", особым образом указывать контекст вывода в этой формуле не требуется.

Контекстный оператор ForEach

Оператор ForEach добавляет в контекст измерения.

Пример: Использование оператора ForEach для добавления измерений в контекст

В следующей таблице показан максимальный доход по каждому кварталу в отчете, который содержит измерение "Квартал", не включенное в блок:

Год	Доход от продаж	Максимальный квартал- ный доход
2001	8 096 123,60	2 660 699,50
2002	13 232 246,00	4 186 120,00
2003	15 059 142,80	4 006 717,50

Для столбца "Максимальный доход за квартал" можно создать формулу, которая не содержит оператор ForEach:

```
Max ([Sales Revenue] In ([Year];[Quarter])) In ([Year])
```

Контекстный оператор "ForEach" позволяет добиться того же результата с помощью следующей формулы:

```
Max ([Sales Revenue] ForEach ([Quarter])) In ([Year])
```

Причина: Измерение "Год" является в блоке контекстом ввода по умолчанию. При использовании оператора "ForEach" измерение "Квартал" добавляется в контекст, в результате чего получается контекст ввода ([Год];[Квартал]).

Контекстный оператор ForAll

Контекстный оператор ForAll удаляет из контекста измерения.

Пример: Использование оператора ForAll для удаления измерений из контекста

Есть отчет, в котором отображается год, квартал и доход от продаж, и необходимо добавить столбец, в котором отображается суммарный доход по каждому году, как показано в следующем блоке:

Year	Quarter	Sales revenue	Yearly Total
2001	Q1	\$2 660 700	\$8 096 124
2001	Q2	\$2 279 003	\$8 096 124
2001	Q3	\$1 367 841	\$8 096 124
2001	Q4	\$1 788 580	\$8 096 124
2002	Q1	\$3 326 172	\$13 232 246
2002	Q2	\$2 840 651	\$13 232 246
2002	Q3	\$2 879 303	\$13 232 246
2002	Q4	\$4 186 120	\$13 232 246
2003	Q1	\$3 742 989	\$15 059 143
2003	Q2	\$4 006 718	\$15 059 143
2003	Q3	\$3 953 395	\$15 059 143
2003	Q4	\$3 356 041	\$15 059 143

Для суммирования доходов по годам контекстом ввода должен быть (Год); по умолчанию это (Год; Квартал). Поэтому необходимо удалить из контекста ввода квартал, указав в формуле ForAll ([Квартал]), что будет выглядеть следующим образом:

```
Sum([Sales Revenue] ForAll ([Quarter]))
```

Следует отметить, что для получения такого же результата можно использовать оператор "In". В данном случае формула будет выглядеть следующим образом:

```
Sum([Sales Revenue] In ([Year]))
```

В этой версии формулы в качестве контекста явно указан год, а не удаляется квартал, чтобы оставить год.

Ключевые слова с расширенным синтаксисом в приложении Web Intelligence

Ключевые слова с расширенным синтаксисом – это форма условного обозначения, которая позволяет обращаться к измерениям с расширенным синтаксисом без явного указания этих измерений. Это позволяет обеспечить создание неустаревающих отчетов; если в формулах не содержатся жестко запрограммированные ссылки на

измерения, то они будут оставаться действительными даже при добавлении измерений в отчет или удалении измерений из отчета.

Предусмотрено пять ключевых слов расширенного синтаксиса: Отчет, Раздел, Разбиение, Блок и Тело.

Ключевое слово Report

В следующей таблице приведены данные, на которые ссылается ключевое слово Report в зависимости от его местонахождения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Все данные в отчете
Разбиение по блокам (верхний или нижний колонтитул)	Все данные в отчете
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Все данные в отчете
За пределами всех блоков или разделов	Все данные в отчете

Пример: Ключевое слово Report

В отчете содержатся сведения о доходах от продаж за год, и квартал. В отчете есть столбец "Сумма по отчету", в котором отображается сумма всех доходов из отчета.

Year	Quarter	Sales revenue	Report Total
2001	Q1	\$2 660 700	36 387 512.4
2001	Q2	\$2 279 003	36 387 512.4
2001	Q3	\$1 367 841	36 387 512.4
2001	Q4	\$1 788 580	36 387 512.4
2002	Q1	\$3 326 172	36 387 512.4
2002	Q2	\$2 840 651	36 387 512.4
2002	Q3	\$2 879 303	36 387 512.4
2002	Q4	\$4 186 120	36 387 512.4
2003	Q1	\$3 742 989	36 387 512.4
2003	Q2	\$4 006 718	36 387 512.4
2003	Q3	\$3 953 395	36 387 512.4
2003	Q4	\$3 356 041	36 387 512.4

Формула для столбца "Сумма по отчету": Sum([Доходы от продаж]) In Report. Без ключевого слова Report в данном столбце воспроизводились бы цифры из столбца "Доходы от продаж", поскольку в нем использовался бы контекст вывода по умолчанию ([Год];[Квартал]).

Ключевое слово "Раздел"

В следующей таблице описываются данные, на которые ссылается ключевое слово "Раздел" в зависимости от его расположения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Все данные в разделе
Разбиение по блокам (верхний или нижний колонтитул)	Все данные в разделе
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Все данные в разделе
За пределами всех блоков или разделов	Не применимо

Пример: Ключевое слово "Раздел"

В отчете отображаются объекты "Год", "Квартал" и "Доход с продаж".

2001

Quarter	Sales revenue	Section Total
Q1	\$2 660 700	8 095 814
Q2	\$2 278 693	8 095 814
Q3	\$1 367 841	8 095 814
Q4	\$1 788 580	8 095 814

В этом отчете создан раздел на основе объекта "Год". В столбце "Итог раздела" записана формула:

Sum ([Sales Revenue]) In Section

Цифра в столбце "Итог раздела" обозначает совокупный доход за 2001 год, поскольку разбиение раздела выполнено в объекте "Год". При отсутствии ключевого слова "Раздел" в этом столбце будут дублироваться цифры из столбца "Доход с продаж", поскольку будет использоваться контекст вывода по умолчанию ([Год];[Квартал]).

Ключевое слово "Разбиение"

В следующей таблице описываются измерения, на которые ссылается ключевое слово "Разбиение" в зависимости от его расположения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Данные в части блока, разделенные разбиением
Разбиение по блокам (верхний или нижний колонтитул)	Данные в части блока, разделенные разбиением
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Не применимо

Если расположено в...	Ссылается на эти данные...
За пределами всех блоков или разделов	Не применимо

Пример: Ключевое слово "Разбиение"

В отчете содержатся сведения о доходах от продаж за год, и квартал.

Year	Quarter	Sales revenue	Break Total
2001	Q1	\$2 660 700	8 096 123.6
	Q2	\$2 279 003	8 096 123.6
	Q3	\$1 367 841	8 096 123.6
	Q4	\$1 788 580	8 096 123.6
2001			

В отчете содержится разбиение по объекту "Год". В столбце "Итог разбиения" записана формула:

```
Sum ([Sales Revenue]) In Break
```

При отсутствии ключевого слова "Разбиение" в этом столбце будут дублироваться цифры из столбца "Доход с продаж", поскольку будет использоваться контекст вывода по умолчанию ([Год];[Квартал]).

Ключевое слово блока

В следующей таблице описаны измерения, на которые ссылается ключевое слово блока, в зависимости от его местоположения в отчете; ключевое слово блока часто включает в себя те же данные, что и ключевое слово раздела. Вся разница заключается в том, что Блок принимает во внимание фильтры блока, а Раздел нет.

Если расположено в...	Ссылается на эти данные...
Блок	Данные во всем блоке, без учета разрывов, но с учетом фильтров

Если расположено в...	Ссылается на эти данные...
Разбиение по блокам (верхний или нижний колонтитул)	Данные во всем блоке, без учета разрывов, но с учетом фильтров
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Не применимо
За пределами всех блоков или разделов	Не применимо

Пример: Ключевое слово блока

В отчете содержатся сведения о доходах от продаж за год, и квартал. В этом отчете создан раздел на основе объекта "Год". Фильтры блока опускают показатели за третий и четвертый кварталы.

2001

Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$2 660 700	\$2 469 851.25	\$8 096 123.60
Q2	\$2 279 003	\$2 469 851.25	\$8 096 123.60
Сумма:	4 939 702.5		

2002

Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$3 326 172	\$3 083 411.50	\$13 232 246.00
Q2	\$2 840 651	\$3 083 411.50	\$13 232 246.00
Сумма:	6 166 823		

2003

Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$3 742 989	\$3 874 853.20	\$15 059 142.80
Q2	\$4 006 718	\$3 874 853.20	\$15 059 142.80
Сумма:	7 749 706.4		

В столбце "В среднем за год" присутствует формула.

```
Average ([Sales revenue] In Section)
```

Формула также присутствует в столбце "В среднем за первое полугодие".

```
Average ([Sales revenue]) In Block
```

Можно проследить за тем, как ключевое слово блока учитывает фильтр блока.

Ключевое слово "Тело"

В следующей таблице описываются измерения, на которые ссылается ключевое слово "Тело" в зависимости от его расположения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Данные в блоке
Разбиение по блокам (верхний или нижний колонтитул)	Данные в блоке
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Данные в разделе
За пределами всех блоков или разделов	Данные в отчете

Пример: Ключевое слово "Тело"

В отчете отображаются объекты "Год", "Квартал" и "Доход с продаж" с разбиением по объекту "Год". В отчете содержится раздел, основанный на объекте "Год", и разбиение по объекту "Квартал".

Год	Квартал	Доходы с продаж	Тело
2001	K1	2 660 700	2 660 699,5
	K2	2 279 003	2 279 003
	K3	1 367 841	1 367 840,7
	K4	1 788 580	1 788 580,4
2001		8 096 123,6	

В строке "Тело" записана формула

```
Sum ([Sales Revenue]) In Body
```

Итоги в столбце "Тело" и "Доход с продаж" одинаковы, поскольку ключевое слово "Тело" относится к данным в блоке. Если удалить объект "Месяц", цифры в столбце "Блок" изменятся для обеспечения соответствия цифрам в столбце "Доход с продаж". Если формулу разместить в нижнем колонтитуле отчета, то будет вычисляться совокупный доход для блока.

Использование ключевых слов для обобщения отчетов

Ключевые слова расширенного синтаксиса защищают отчет от изменений. При явном определении данных (измерение определяется с помощью операторов In, ForEach или ForAll) отчеты могут возвращать непредвиденные данные, если добавлять или удалять измерения. Это показано в следующем примере.

Пример: Использование ключевого слова Report для отображения процентного соотношения

В данном примере в блоке содержатся объекты "Год", "Квартал" и "Доход от продаж". Необходимо отобразить доходы за каждый год и квартал, а также процентное соотношение общего дохода в отчете, который представляет каждый индивидуальный доход, как показано ниже.

Year	Quarter	Sales revenue	Percentage of Total
2001	Q1	\$2 660 700	7,31
2001	Q2	\$2 279 003	6,26
2001	Q3	\$1 367 841	3,76
2001	Q4	\$1 788 580	4,92
2002	Q1	\$3 326 172	9,14
2002	Q2	\$2 840 651	7,81
2002	Q3	\$2 879 303	7,91
2002	Q4	\$4 186 120	11,50
2003	Q1	\$3 742 989	10,29
2003	Q2	\$4 006 718	11,01
2003	Q3	\$3 953 395	10,86
2003	Q4	\$3 356 041	9,22
		Сумма:	100

Формула для столбца "Процентное соотношение общего дохода" выглядит следующим образом.

```
([Sales revenue]/(Sum([Sales revenue]) In Report)) * 100
```

В блоке ключевое слово Report включает в себя все данные в отчете, поэтому данная формула выглядит следующим образом.

```
([Sales revenue]/Sum([Sales revenue] ForAll ([Year];[Quarter]))) * 100
```

С помощью данной формулы Web Intelligence удаляет объекты "Год" и "Квартал" в выходном контексте; другими словами, чтобы вычислить общий доход, так как в отчете нет других измерений. Затем в формуле каждый доход делится на общую сумму, чтобы найти его процентное соотношение от общего дохода.

Хотя в данной ситуации можно использовать оператор ForAll, предпочтительнее использовать ключевое слово Report. Причина Что произойдет, если добавить измерение "Месяц" в отчет позже? С помощью версии формулы, которая используется в ключевом слове Report, каждое процентное соотношение вычисляется правильно, но версия, которая определяет измерения "Год" и "Квартал", сейчас не является верной.

3 | Описание контекстов вычисления

Изменение контекста вычислений по умолчанию с расширенным синтаксисом

Year	Quarter	Month	Sales revenue	Percentage of Total
2001	Q1	1	\$1 003 541,20	26,13
2001	Q1	2	\$630 073,20	29,97
2001	Q1	3	\$1 027 085,10	27,12
2001	Q2	4	\$895 259,80	28,10
2001	Q2	5	\$865 615,10	24,30
2001	Q2	6	\$517 818,50	21,77
2001	Q3	7	\$525 903,50	20,42
2001	Q3	8	\$173 756,40	11,11
2001	Q3	9	\$668 180,80	16,45
2001	Q4	10	\$655 206,40	18,04
2001	Q4	11	\$484 024,20	18,55
2001	Q4	12	\$649 349,80	21,01
2002	Q1	1	\$1 335 401,90	34,77
2002	Q1	2	\$609 012,80	28,97
2002	Q1	3	\$1 381 757,50	36,49
2002	Q2	4	\$1 068 308,90	33,53
2002	Q2	5	\$1 081 884,80	30,38
2002	Q2	6	\$690 457,10	29,03
2002	Q3	7	\$801 954,70	31,14
2002	Q3	8	\$581 093,50	37,15
2002	Q3	9	\$1 496 254,80	36,84
2002	Q4	10	\$1 545 871,80	42,57
2002	Q4	11	\$1 081 915,30	41,47
2002	Q4	12	\$1 558 332,90	50,43
2003	Q1	1	\$1 501 366,70	39,09
2003	Q1	2	\$863 451,90	41,07
2003	Q1	3	\$1 378 170,30	36,39
2003	Q2	4	\$1 222 329,40	38,37
2003	Q2	5	\$1 614 147,30	45,32
2003	Q2	6	\$1 170 240,80	49,20
2003	Q3	7	\$1 247 313,50	48,44
2003	Q3	8	\$809 365,40	51,74
2003	Q3	9	\$1 896 716,40	46,70
2003	Q4	10	\$1 430 300,10	39,39
2003	Q4	11	\$1 043 098,80	39,98
2003	Q4	12	\$882 642,20	28,56
			Сумма:	1 200

В чем причина? Проблема заключается в следующем.


```
Sum ([Sales Revenue] ForAll ([Year];[Quarter]))
```

Когда измерения "Год" и "Квартал" были единственными измерениями в отчете, это было эквивалентно общей сумме всех доходов. Как только было добавлено измерение "Месяц", данное выражение удалило объекты "Год" и "Квартал" из выходного контекста, но объект "Месяц" остался.


Теперь формула включает "разбиение" по месяцу. Другими словами, в каждой строке, где у объекта "Месяц" значение 1, данное выражение означает "общий доход всех месяцев со значением 1s". В каждой строке, где у объекта "Месяц" значение 2, данное выражение означает "общий доход всех месяцев со значением 2s". В результате процентное соотношение не соответствует ожидаемому.

3 | Описание контекстов вычисления

Изменение контекста вычислений по умолчанию с расширенным синтаксисом



Вычисление значений с
помощью
интеллектуальных мер



4

Определение интеллектуальной меры

«Интеллектуальные меры» – это меры, значения которых вычисляются базой данных (реляционной или OLAP), лежащей в основе юниверса Web Intelligence, а не самим приложением Web Intelligence. Мера определяется в юниверсе интеллектуальной, когда ее данные агрегируются способом, который не поддерживается Web Intelligence.

Для возврата значений интеллектуальных мер Web Intelligence генерирует запрос для вычисления меры во всех контекстах вычисления, необходимых в отчете. При редактировании отчета эти контексты могут изменяться. В результате при каждом обновлении данных после изменения нужных контекстов Web Intelligence изменяет запрос.

Интеллектуальные меры ведут себя не так, как классические, поддерживающие основной набор функций агрегирования (Max, Min, Count, Sum, Average), которые Web Intelligence может вычислить во всех контекстах без помощи со стороны базы данных. Например, при создании запроса, содержащего измерения [Страна] и [Регион] и меру [Доход] (которая вычисляет сумму дохода), Web Intelligence исходно отображает страну, регион и доход в блоке. Если после этого из блока удалить регион, Web Intelligence сможет вычислить суммарный доход по каждой стране посредством сложения доходов по всем регионам страны.

Контексты вычислений представлены в генерируемых Web Intelligence запросах «наборами для группировки».

Наборы группировок и интеллектуальные меры

«Набор для группировки» представляет собой набор измерений, который генерирует результат для меры. Если Web Intelligence возвращает данные для интеллектуальной меры, созданный SQL включает в себя наборы для группировки для всех агрегирований данной меры, включенных в отчет.

Пример: Наборы для группировки в запросе

Запрос содержит измерения [Страна], [Регион], [Город] и интеллектуальная мера [Доход]. Эти объекты предполагают, что для вычисления дохода во всех возможных контекстах будут использованы следующие наборы группировок:

- Суммарное значение интеллектуальной меры
- значение интеллектуальной меры по (Страна, Регион, Город)
- значение интеллектуальной меры по (Страна, Город)
- значение интеллектуальной меры по (Город)
- значение интеллектуальной меры по (Регион, Город)
- значение интеллектуальной меры по (Регион)
- значение интеллектуальной меры по (Страна, Регион)
- значение интеллектуальной меры по (Страна)

Web Intelligence извлекает наборы для группировки с помощью оператора UNION в запросе. Если база данных не поддерживает UNION, Web Intelligence выполняет объединение самостоятельно.

Web Intelligence обновляет наборы для группировки согласно контекстам вычисления, необходимым для отчета, которые могут измениться в ответ на изменения в структуре отчета.

Принципы управления Web Intelligence наборами для группировки

При первом создании и выполнении запроса, включая интеллектуальные меры, Web Intelligence содержит наборы для группировки, необходимые для вычисления интеллектуальных мер на самом подробном уровне, подразумеваемом объектами запроса. Web Intelligence всегда включает в себя данный набор для группировки в SQL-запросе.

Например, при создании запроса, содержащего измерения [Страна], [Регион] и [Город] и интеллектуальную меру [Доход], Web Intelligence включает в себя набор для группировки (Страна, Регион, Город) в созданном SQL. данный набор для группировки всегда отображается в SQL. Web Intelligence добавляет и удаляет другие наборы для группировки в ответ на изменения отчета.

При удалении измерения [Город] из блока Web Intelligence потребуется набор для группировки (Страна, Регион), чтобы возвращать значения дохода. Данный набор для группировки еще не доступен в SQL-запросе, поэтому Web Intelligence отображает #TOREFRESH в ячейках [Доход]. При обновлении данных Web Intelligence может заменить #TOREFRESH на значения дохода.

Затем при замене измерения [Город] в блоке набор для группировки (Страна, Регион) больше не потребуется. Web Intelligence удаляет его из SQL-запроса и не учитывает его значения при следующем обновлении данных.

При каждом обновлении данных отчета Web Intelligence обновляет SQL-запрос, чтобы учитывать или не учитывать наборы для группировки, согласно контекстам вычислений, необходимым для отчета.

В определенных ситуациях Web Intelligence не может отобразить значение интеллектуальной меры. В данном случае Web Intelligence отображает #UNAVAILABLE в ячейках мер.

Интеллектуальные меры и область анализа

При создании запроса с областью анализа Web Intelligence создает изначальный набор для группировки, содержащий объекты результатов, но не объекты области. Web intelligence не создает все возможные наборы для группировки из комбинации объектов результатов и объектов области.

Пример: Запрос с областью анализа и интеллектуальная мера

Запрос содержит объекты результата [Страна] и [Доход]. В области анализа содержатся измерения [Регион] и [Город]. При выполнении запроса Web Intelligence извлекает набор для группировки (Страна) и отображает [Страна] и [Доход] в блоке.

Интеллектуальные меры и язык SQL

Наборы группировок и оператор UNION

Некоторые базы данных поддерживают наборы для группировки с оператором `GROUPING SETS`. Web Intelligence использует множество наборов результатов и оператор `UNION`, чтобы смоделировать эффект `GROUPING SETS`.

Пример: Наборы для группировки, извлеченные с помощью оператора `UNION`

В данном примере описывается запрос, содержащий измерения [Страна], [Регион], [Город] и интеллектуальную меру [Доход].

Примечание:

Для простоты, интеллектуальная мера вычисляет сумму. На практике интеллектуальная мера не требуется для данного агрегирования, поскольку универсы Web Intelligence поддерживают функцию `Сумма`.

При первом выполнении запроса набор для группировки представляет собой следующее: Страна, Регион, Город. Весь SQL-запрос возвращает данный набор для группировки, и необходимости в операторе `UNION` в SQL нет.

Если удалить измерение [Город] из таблицы, Web Intelligence потребуется набор для группировки (Страна, регион) для отображения дохода (отображается как `#TOREFRESH`). После обновления данных SQL выглядит следующим образом:

```
SELECT
  SELECT
    0 AS GID,
    country.country_name,
    region.region_name,
    NULL,
    sum(city.revenue)
FROM
  country,
  region,
  city
WHERE
```

```
( country.country_id=region.country_id )
AND ( region.region_id=city.region_id )
GROUP BY
    country.country_name,
    region.region_name
UNION
SELECT
    1 AS GID,
    country.country_name,
    region.region_name,
    city.city_name,
    sum(city.revenue)
FROM
    country,
    region,
    city
WHERE
    ( country.country_id=region.country_id )
    AND ( region.region_id=city.region_id )
GROUP BY
    country.country_name,
    region.region_name,
    city.city_name
```

Каждый набор группировок представлен оператором `SELECT` и имеет свой собственный идентификатор (столбец `GID`). В наборы для группировки, которые не содержат полного набора измерений, входят пустые столбцы (`SELECT ' '`), поскольку каждый оператор `SELECT` в запросе, включая `UNION`, должен иметь одинаковое количество столбцов.

Если добавляется новый блок, содержащий [Страна] и [Доход], в отчет, Web Intelligence потребуются набор для группировки (Страна). Теперь созданный SQL включает в себя три следующих набора для группировки:

```
SELECT
    0 AS GID,
    country.country_name,
    region.region_name,
    NULL,
    sum(city.revenue)
FROM
    country,
    region,
    city
```

```

WHERE
    ( country.country_id=region.country_id )
    AND ( region.region_id=city.region_id )
GROUP BY
    country.country_name,
    region.region_name
UNION
SELECT
    1 AS GID,
    country.country_name,
    NULL,
    NULL,
    sum(city.revenue)
FROM
    country,
    city,
    region
WHERE
    ( country.country_id=region.country_id )
    AND ( region.region_id=city.region_id )
GROUP BY
    country.country_name
UNION
SELECT
    2 AS GID,
    country.country_name,
    region.region_name,
    city.city_name,
    sum(city.revenue)
FROM
    country,
    region,
    city
WHERE
    ( country.country_id=region.country_id )
    AND ( region.region_id=city.region_id )
GROUP BY
    country.country_name,
    region.region_name,
    city.city_name

```

Интеллектуальные меры и формулы

Интеллектуальные меры и измерения, содержащие формулы

Если формула или переменная отображается как измерение в контексте вычисления интеллектуальной меры, и формула определяет набор для группировки, необходимый для меры, Web Intelligence не может отобразить значения для интеллектуальной меры. В этой ситуации Web Intelligence не может по формуле определить набор для группировки.

Например, отчет содержит переменную, Semester, с формулой

```
If [Quarter] = "Q1" or [Quarter] = "Q2" Then "H1" Else "H2"
```

При размещении в блоке переменная **Полугодие** возвращает следующий результат:

Полугодие	Доход
H1	#UNAVAILABLE
H2	#UNAVAILABLE

Интеллектуальные меры в формулах

Web Intelligence может вернуть значение для интеллектуальной меры, если интеллектуальная мера включена в формулу, даже если для формулы требуется контекст вычисления, отличный от контекста, подразумеваемого положением формулы.

Например, отчет содержит следующий блок:

Страна	Регион	Доход
США	Север	10 000
США	Юг	15 000
США	Восток	14 000
США	Запад	12000

Если добавить дополнительный столбец в таблицу с формулой

```
[Revenue] ForAll ([Region])
```

Web Intelligence сначала возвратит #TOREFRESH, поскольку для формулы необходим набор для группировки (Страна). Формула исключает регионы из вычислений. При обновлении данных Web Intelligence добавляет набор для группировки (Страна) в запрос и отображает значения меры.

Интеллектуальные меры и фильтры

Интеллектуальные показатели и фильтры для измерений

Если фильтр применяется к измерению, от которого зависит интеллектуальное значение, но измерение не отображается явно в контексте вычисления меры, Web Intelligence не может вернуть значение для интеллектуальной меры и отображает #UNAVAILABLE.

Данная ситуация возникает, поскольку Web Intelligence не может вычислить влияние фильтра на значения меры. Единственным способом определить влияние является применение фильтра к запросу. Это порождает риск воздействия на другие отчеты, в основе которых лежит этот же запрос. В результате Web intelligence не применяет фильтр на уровне запроса.

Пример: Интеллектуальная мера и фильтр для измерения

Запрос содержит измерения [Страна] и [Регион] и интеллектуальную меру [Доход]. [Страна] и [Доход] отображаются в блоке. Если применяется фильтр, ограничивающий значения [Регион] параметрами "юго-восток" или "юго-запад", Web Intelligence отображает #UNAVAILABLE в ячейках [Доход].

Интеллектуальные меры и фильтры детализации

Как правило, Web Intelligence не может вернуть значения для интеллектуальных мер, если фильтр применен к измерению, которое влияет на вычисление меры. Измерения, отфильтрованные с помощью фильтров детализации, являются исключением из данного правила.

Пример: Фильтр детализации, влияющий на интеллектуальную меру

Блок содержит объекты [Страна] и [Доход]. Перейдите к [Страна], и Web Intelligence отобразит [Регион], [Доход] в блоке и переместит фильтр на [Страна] на панель детализации.

Для этого Web Intelligence добавляет набор для группировки (Страна, Регион) в запрос и извлекает все его данные, затем фильтрует их, чтобы вывести на экран только те регионы, которые содержатся в подвергающейся детализации стране. Системе Web Intelligence не требуется добавлять фильтр на уровне запроса, чтобы отфильтровать регионы по стране.

Интеллектуальные меры и вложенные фильтры OR


Вложенные фильтры OR (ИЛИ), в которых хотя бы одно из отфильтрованных измерений не отображается в блоке, порождают ошибку #UNAVAILBLE для интеллектуальной меры в блоке.

Пример:


- Фильтр [Страна] = "США" ИЛИ [Страна] = "Франция" порождает ошибку #UNAVAILABLE для интеллектуальной меры, использующей измерение [Страна], если [Страна] не отображается в блоке.
 - Фильтр [Страна] = "США" ИЛИ [Год] = 2008 порождает ошибку #UNAVAILABLE для интеллектуальной меры, использующей измерения [Страна] и [Год], если [Страна] или [Год] не отображаются в блоке.
-

4 | Вычисление значений с помощью интеллектуальных мер

Интеллектуальные меры и фильтры



Функции, операторы и ключевые слова Web Intelligence



Функции Web Intelligence

В Web Intelligence функции подразделяются на следующие категории:

Категория	Описание
Агрегирование	Объединяют данные (например, суммируя или усредняя набор значений)
Символьные	Оперируют со строками символов
Дата и время	Возвращают данные даты или времени
Документ	Возвращают данные о документе
Поставщик данных	Возвращает данные о поставщике данных документа
Логические	Возвращают значения True или False
Числовой	Возвращают числовые данные
Разное	Функции, которые не попадают ни в одну из указанных выше категорий

Функции агрегирования

Average

Описание

Возвращает среднее значение показателя

Группа функций

Агрегирование

Синтаксис

```
num Average (measure [ ; IncludeEmpty ] )
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
IncludeEmpty	Включать в вычисление пустые строки	Ключевое слово	Нет (Пустые строки по умолчанию исключаются)

Примечания

- С функцией `Average` можно использовать операторы контекста расширенного синтаксиса.
- В качестве второго аргумента функции можно использовать `IncludeEmpty`. Если указать этот аргумент, то при вычислении функция будет учитывать пустые (нулевые) строки.

Примеры

Если для меры [Доходы от продаж] заданы значения 41569, 30500, 40000 и 50138, функция `Average ([Доходы от продаж])` возвращает 405552.

См. также

- [Оператор IncludeEmpty](#)

Count

Описание

Возвращает количество значений в измерении или мере

Группа функций

Агрегирование

Синтаксис

```
integer
Count (dimension|measure [; IncludeEmpty] [; Distinct|All])
```

Данные ввода

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Измерение или мера	Да
IncludeEmpty	Включает пустые значения в расчет	Ключевое слово	Нет
Distinct All	Включает только точные значения (по умолчанию для измерений) или все значения (по умолчанию для мер) в расчет	Ключевое слово	Нет

Примечания

- Можно использовать операторы контекста расширенного синтаксиса с функцией Count.
- В качестве второго аргумента функции можно использовать IncludeEmpty. С этим аргументом функция учитывает пустые строки при расчете.

- Параметр `Distinct/All` является дополнительным. Если этот параметр не указан, то по умолчанию значения следующие:
`Distinct` для измерений и `All` для мер.

Примеры

`Count ("Тест")` возвращает 1

`Count ([City];Distinct)` возвращает 5, если в списке городов есть 5 различных городов, даже если в списке более 5 строк из-за повторов.

`Count ([City];All)` возвращает 10, если в списке городов есть 10 городов, даже если некоторые из них повторяются.

`Count ([City];IncludeEmpty)` возвращает 6, если существует 5 городов и одна пустая строка в списке городов.

См. также

- [Оператор IncludeEmpty](#)
- [Операторы Distinct/All](#)

First

Описание

Возвращает первое значение из множества данных

Группа функций

Агрегирование

Синтаксис

```
input_type First (измерение | мера)
```

Входные данные

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Измерение или мера	Да

Примечания

- При нахождении в нижнем колонтитуле разрыва функция `First` возвращает первое значение в разрыве.
- При нахождении в нижнем колонтитуле раздела функция `First` возвращает первое значение в разделе.

Примеры

При размещении в нижнем колонтитуле таблицы функция `First ([Доход])` возвращает первое значение [Доход] в этой таблице.

Интерполяция

Описание

Вычисляет пустые значения мер путем интерполяции

Группа функций

Числовой

Синтаксис

```
num
Interpolation (measure [ ; PointToPoint | Linear ] [ ; NotOnBreak ] [ ; Row | Col ] )
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
PointToPoint	Метод интерполяции <ul style="list-style-type: none"> PointToPoint – двухточечная интерполяция Linear – линейная регрессия с интерполяцией методом наименьших квадратов 	Ключевое слово	Нет (PointToPoint – параметр по умолчанию)
NoBreak	Предотвращает сброс вычисления функции при нахождении разбиений блоков или разделов.	Ключевое слово	Нет
Row Col	Устанавливает направление вычисления	Ключевое слово	Нет

Примечания

- Функция `Interpolation` особенно необходима при создании линейного графика с использованием меры, в которой содержатся отсутствующие значения. При использовании этой функции обеспечивается непрерывное построение графика без получения разорванных линий и точек.
- При использовании линейной регрессии с интерполяцией методом наименьших квадратов отсутствующие значения вычисляются с помощью линейного выражения $f(x) = ax + b$, которое позволяет получить линию, наиболее приближенную ко всем доступным значениям меры.
- При двухточечной интерполяции отсутствующие значения получаются путем вычисления значений линейной функции $f(x) = ax + b$, проходящей через две соседние с отсутствующими значениями точки.
- От порядка сортировки меры зависят значения, полученные с помощью функции `Interpolation`.

- Применение сортировки или ранжирования в формуле с функцией `Interpolation` недопустимо.
- В списке значений содержится только одно значение, и это значение используется в функции `Interpolation` для получения всех отсутствующих значений.
- Фильтры, примененные к интерполированной мере, могут повлиять на значения, выведенные функцией `Interpolation`, в зависимости от значений, к которым применяются фильтры.

Примеры

`Interpolation([Значение])` возвращает следующие отсутствующие значения при использовании метода интерполяции "точка-точка" по умолчанию:

День	Значение	Interpolation([Значение])
Понедельник	12	12
Вторник	14	14
Среда		15
Четверг	16	16
Пятница		17
Суббота		18
Воскресенье	19	19

См. также

- [Оператор Linear](#)
- [Оператор PointToPoint](#)

Последний

Описание

Возвращает последнее значение в измерении или мере

Группа функций

Агрегирование

Синтаксис

```
input_type Last (dimension|measure)
```

Ввод

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Измерение или мера	Да

Примечания

- При нахождении в нижнем колонтитуле разрыва функция Last возвращает последнее значение в разрыве.
- При нахождении в нижнем колонтитуле раздела функция Last возвращает последнее значение в разделе.

Примеры

При размещении в нижнем колонтитуле таблицы функция First ([Доход]) возвращает первое значение [Доход] в этой таблице.

Max

Описание

Возвращает самое большое значение в измерении или мере

Группа функций

Агрегирование

Синтаксис

```
input_type Max(dimension|measure)
```

Ввод

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Измерение или мера	Да

Примечания

Можно использовать контекстные операторы расширенного синтаксиса с функцией `Max`.

Примеры

Если переменная "Доходы с продаж" имеет значения 3 000, 60 034 и 901 234, **Max([Доходы с продаж])** возвращает 901 234.

Если переменная размера "Город" имеет значения "Абердин" и "Лондон", **Max ([Город])** возвращает "Лондон".

Median

Описание

Возвращает медиану (среднее число) меры

Группа функций

Агрегирование

Синтаксис

```
число Median(measure)
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примечания

Если в наборе чисел четное количество значений, `Median` берет среднее от двух значений из середины.

Примеры

`Median ([Доход])` возвращает 971 444, если [Доход] имеет значения 835 420, 971 444 и 147 966.

Min

Описание

Возвращает наименьшее значение в измерении или мере

Группа функций

Агрегирование

Синтаксис

```
any_type Min (измерение | мера)
```

Входные данные

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Измерение или мера	Да

Примечания

С функцией `Min` можно использовать контекстные операторы расширенного синтаксиса.

Примеры

Если мера "Доход с продаж" содержит значения 3000, 60034 и 901234, функция `Min ([Доход от продаж])` возвращает 3000.

Если измерение "Город" содержит значения "Абердин" и "Лондон", функция `Min ([Город])` возвращает "Абердин".

Mode

Описание

Возвращает наиболее часто встречающееся значение в наборе данных

Группа функций

Агрегирование

Синтаксис

```
input_type Mode (dimension|measure)
```

Данные ввода

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Мера	Да

Примечания

- Функция `Mode` возвращает пустое значение, если во множестве данных не содержится значения, встречающегося чаще других значений.

Примеры

Функция `Mode ([Доход])` возвращает значение 200, если объект [Доход] имеет значения 100, 200, 300, 200.

`Mode ([Страна])` возвращает наиболее часто встречающееся значение в объекте [Страна].

Percentage

Описание

Выражает значение меры в процентном соотношении его внедренного контекста

Группа функций

Агрегирование

Синтаксис

```
num Percentage (measure [ ; Break ] [ ; Row | Col ] )
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
Разрыв	Счета для разбиений таблиц	Ключевое слово	Нет
Row Col	Устанавливает направление вычисления	Ключевое слово	Нет

Примеры

В следующей таблице для столбца "Процентное соотношение" задана формула `Percentage ([Доход от продаж])`

Год	Доход с продаж	Процентное соотношение
2001	1000	10
2002	5000	50
2003	4000	40
Сумма:	10000	100

По умолчанию присвоенный контекст – это мера итогового значения в таблице. Можно задать в функции учет разбиения таблицы с использованием дополнительного аргумента `Break`. В этом случае внедренный контекст по умолчанию становится разделом таблицы.

В следующей таблице для столбца "Процентное соотношение" задана формула `Percentage ([Доход от продаж]; Break)`

Год	Квартал	Доход с продаж	Процентное соотношение
2001	K1	1000	10

	K2	2000	20
	K3	5000	50
	K4	2000	20
2001	Сумма:	10000	100

Год	Квартал	Доход с продаж	Процентное соотношение
2002	K1	2000	20
	K2	2000	20
	K3	5000	50
	K4	1000	10
2002	Сумма:	10000	100

Функцию `Percentage` можно применять к строкам или столбцам; это можно сделать явно с помощью дополнительного аргумента `Row|Col`. Например, в следующей кросс-таблице столбец "Процентное соотношение" описан формулой `Percentage ([Доход с продаж] ; Row)`

	K1	Процентное соотношение	K2	Процентное соотношение	K3	Процентное соотношение	K4	Процентное соотношение
2001	1000	10	2000	20	5000	50	2000	20
2002	2000	20	2000	20	5000	50	1000	10

Percentile

Описание

Возвращает процентиль nth меры

Группа функций

Числовой

Синтаксис

```
num Percentile (measure; percentile)
```

Ввод

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
процентиль	Процентное значение, выраженное десятичной дробью	Число	Да

Примечания

N-й процентиль – это число, которое больше или равно n% чисел из набора. N% выражается в форме 0,n.

Примеры

Если [мера] содержит множество значений (10;20;30;40;50), то `Percentile ([мера]; 0,3)` возвращает 22, что больше или равно 30% чисел из множества.

Product

Описание

Умножает значения меры

Группа функций

Агрегирование

Синтаксис

```
num Product (measure)
```

Ввод

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примеры

`Product ([Мера])` возвращает 30, если [Мера] имеет значения 2, 3, 5.

RunningAverage

Описание

Возвращает скользящее среднее значение меры

Группа функций

Агрегирование

Синтаксис

```
num  
RunningAverage (measure [;Row|Col] [;IncludeEmpty] [;reset_dims])
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
Row Col	Устанавливает направление вычисления	Ключевое слово	Нет
IncludeEmpty	Включает пустые значения в расчет	Ключевое слово	Нет
reset_dims	Сбрасывает вычисление на указанные измерения	Список измерений	Нет

Примечания

- С функцией `RunningAverage` можно использовать операторы контекста расширенного синтаксиса.
- Направление расчета можно задать операторами `Row` и `Col`.
- Если применить сортировку к мере, на которую ссылается функция `RunningAverage`, в приложении Web Intelligence сначала применяется сортировка к мере, а затем вычисляется скользящее среднее.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.
- Измерения сброса в наборе должны разделяться двоеточиями.
- Функция `RunningAverage` не производит автоматический сброс среднего значения после разбиения блока или нового раздела.

Примеры

`RunningAverage ([Доход])` возвращает результаты, приведенные в следующей таблице:

Страна	Курорт	Доход	Промежуточное среднее
США	Hawaiian Club	1 479 660	835 420
США	Bahamas Beach	971 444	1 225 552

Франция	French Riviera	835 420	1 095 508
---------	----------------	---------	-----------

`RunningAverage ([Доход] ; ([Страна]))` возвращает результаты, приведенные в следующей таблице:

Страна	Курорт	Доход	Промежуточное среднее
США	Hawaiian Club	1 479 660	835 420
США	Bahamas Beach	971 444	1 225 552
Франция	French Riviera	835 420	835 420

См. также

- [Оператор IncludeEmpty](#)
- [Операторы Row/Col](#)

RunningCount

Описание

Возвращает текущий счет набора чисел

Группа функций

Агрегирование

Синтаксис

```
num
RunningCount (dimension|measure [; Row|Col] [; IncludeEmpty] [; reset_dims])
```

Данные ввода

Параметр	Описание	Тип	Обязательное
<code>dimension measure</code>	Любое измерение или мера	Измерение или мера	Да
<code>Row Col</code>	Устанавливает направление вычисления	Ключевое слово	Нет
<code>IncludeEmpty</code>	Включает пустые значения в расчет	Ключевое слово	Нет
<code>reset_dims</code>	Сбрасывает вычисление на указанные измерения	Список измерений	Нет

Примечания

- Контекстные операторы расширенного синтаксиса можно использовать с функцией `RunningCount`.
- Направление расчета можно задать операторами `Row` и `Col`.
- Если применить сортировку к мере, на которую ссылается функция `RunningCount`, Web Intelligence сначала применяет сортировку к мере, а затем вычисляет текущий счет.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.
- Измерения сброса в наборе должны разделяться двоеточиями.
- Функция `RunningCount` не производит автоматического сброса счетчика после разбиения по блокам или создания нового раздела.

Примеры

`RunningCount ([Доход])` возвращает эти результаты в следующей таблице:

Страна	Курорт	Доход	Промежуточное количество
США	Hawaiian Club	1 479 660	1
США	Bahamas Beach	971 444	2
Франция	French Riviera	835 420	3

RunningCount ([Доход]; ([Страна])) возвращает эти результаты в следующей таблице:

Страна	Курорт	Доход	Промежуточное количество
США	Hawaiian Club	1 479 660	1
США	Bahamas Beach	971 444	2
Франция	French Riviera	835 420	1

См. также

- Оператор IncludeEmpty
- Операторы Row/Col
- Оператор IncludeEmpty
- Оператор IncludeEmpty

RunningMax

Описание

Возвращает промежуточный максимум измерения или меры

Группа функций

Агрегирование

Синтаксис

```
input_type
RunningMax (dimension|measure [;Row|Col] [;reset_dims])
```

Данные ввода

Параметр	Описание	Тип	Обязательное
dimension measure	Любое измерение или мера	Измерение или мера	Да
Row Col	Устанавливает направление вычисления	Ключевое слово	Нет
reset_dims	Сбрасывает вычисление на указанные измерения	Список измерений	Нет

Примечания

- С помощью функции `RunningMax` можно использовать операторы контекста расширенного синтаксиса.
- Направление вычисления можно задать с помощью операторов `Row` и `Col`.
- Если применить сортировку к мере, на которую ссылается функция `RunningMax`, Web Intelligence сначала применяет сортировку к мере, а затем вычисляет промежуточный максимум.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.
- Измерения сброса в наборе должны разделяться двоеточиями.
- Функция `RunningMax` не производит автоматического сброса максимума после разбиения блока или создания нового раздела.

Примеры

`RunningMax ([Доход])` возвращает результаты, приведенные в следующей таблице:

Страна	Курорт	Доход	Текущий максимум

Франция	French Riviera	835 420	835 420
США	Bahamas Beach	971 444	971 444
США	Hawaiian Club	1 479 660	1 479 660

См. также

- Оператор *IncludeEmpty*
- Операторы *Row/Col*

RunningMin

Описание

Возвращает промежуточный минимум измерения или меры

Группа функций

Агрегирование

Синтаксис

```
input_type RunningMin (измерение | мера; [Row|Col]; [reset_dims])
```

Входные данные

Параметр	Описание	Тип	Обязатель- ный
<code>measure</code>	Любое измерение или ме- ра	Измерение или мера	Да
<code>Row Col</code>	Устанавливает направле- ние вычисления	Ключевое сло- во	Нет
<code>reset_dims</code>	Сбрасывает вычисление на указанные измерения	Список изме- рений	Нет

Примечания

- С функцией `RunningMin` можно использовать контекстные операторы расширенного синтаксиса.
- Направление расчета можно задать операторами `Row` и `Col`.
- Если применить сортировку к мере, на которую ссылается функция `RunningMin`, в приложении Web Intelligence сначала применяется сортировка к мере, а затем вычисляется промежуточный минимум.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.
- Измерения сброса в наборе должны разделяться двоеточиями.
- Функция `RunningMin` не производит автоматический сброс минимума после разбиения блока или нового раздела.

Примеры

`RunningMin ([Доход])` возвращает результаты в виде таблицы, представленной ниже:

Страна	Курорт	Доход	Текущий макси- мум
Франция	French Riviera	835 420	835 420
США	Bahamas Beach	971 444	835 420
США	Hawaiian Club	1 479 660	835 420

См. также

- *Оператор IncludeEmpty*
- *Операторы Row/Col*

RunningProduct

Описание

Возвращает промежуточный продукт меры

Группа функций

Агрегирование

Синтаксис

```
num RunningProduct (measure [;Row|Col] [;reset_dims])
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
Row Col	Устанавливает направление вычисления	Ключевое слово	Нет
reset_dims	Сбрасывает вычисление на указанные измерения	Список измерений	Нет

Примечания

- С помощью функции `RunningProduct` можно использовать контекстные операторы расширенного синтаксиса.
- Направление вычисления можно задать операторами `Row` и `Col`.
- Если применить сортировку к мере, на которую ссылается функция `RunningProduct`, Web Intelligence сначала применяет сортировку к мере, а затем вычисляет промежуточный максимум.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.

- Измерения сброса в наборе должны разделяться двоеточиями.
- Функция `RunningProduct` не производит автоматического сброса продукта после разбиения блока или создания нового раздела.

Примеры

`RunningProduct ([Количество гостей])` возвращает результаты, приведенные в следующей таблице:

Страна происхождения	Город	Количество гостей	Промежуточное произведение
Япония	Кобе	6	6
Япония	Осака	4	24
США	Чикаго	241	5 784

`RunningProduct ([Количество гостей]; ([Страна происхождения]))` возвращает результаты, приведенные в следующей таблице:

Страна происхождения	Город	Количество гостей	Промежуточное произведение
Япония	Кобе	6	6
Япония	Осака	4	24
США	Чикаго	241	5 784

См. также

- [Оператор IncludeEmpty](#)
- [Операторы Row/Col](#)

RunningSum

Описание

Возвращает текущую сумму измерения

Группа функций

Агрегирование

Синтаксис

```
num RunningSum(measure [;Row|Col] [;reset_dims])
```

Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да
Row Col	Устанавливает направление вычисления	Ключевое слово	Нет
reset_dims	Сбрасывает вычисление на указанные измерения	Список измерений	Нет

Примечания

- Функцию `RunningSum` можно использовать с контекстными операторами расширенного синтаксиса.
- Направление вычислений можно задать с помощью операторов `Row` и `Col`.
- Если к мере, на которую ссылается функция `RunningSum` применить сортировку, Web Intelligence сначала применяет сортировку, а затем рассчитывает промежуточную сумму.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.
- Измерения сброса в наборе должны разделяться двоеточиями.
- Функция `RunningSum` не производит автоматический сброс суммы после разбиения блока или создания нового раздела.

Пример

`RunningSum ([Доход])` возвращает результаты, приведенные в следующей таблице:

Страна	Курорт	Доход	Промежуточная сумма
Франция	French Riviera	835 420	835 420
США	Bahamas Beach	971 444	1 806 864
США	Hawaiian Club	1 479 660	3 286 524

`RunningSum ([Доход] ; ([Страна]))` возвращает результаты, приведенные в следующей таблице:

Страна	Курорт	Доход	Промежуточная сумма
Франция	French Riviera	835 420	835 420
США	Bahamas Beach	971 444	971 444
США	Hawaiian Club	1 479 660	2 451 104

См. также

- Оператор *IncludeEmpty*
- Операторы *Row/Col*

StdDev

Описание

Возвращает стандартное отклонение меры

Группа функций

Агрегирование

Синтаксис

```
num StdDev (мера)
```

Входные данные

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примечания

Среднеквадратичное отклонение – это мера статистической дисперсии во множестве чисел. Она вычисляется следующим образом:

- поиск среднего значения во множестве чисел
- вычитание среднего значения из каждого числа в множестве и возведение разности в квадрат
- сложение всех возведенных в квадрат разностей
- деление полученной суммы на (*количество чисел во множестве* – 1).
- вычисление квадратного корня из результата.

Примеры

Если мера содержит набор значений (2, 4, 6, 8), StdDev (мера) возвращает 2,58.

См. также

- [Var](#)

StdDevP

Описание

Возвращает стандартное отклонение по совокупности для меры

Группа функций

Агрегирование

Синтаксис

```
num StdDevP (measure)
```

Ввод

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примечания

Среднеквадратичное отклонение по совокупности – это мера статистического разброса множества значений. Она вычисляется следующим образом:

- поиск среднего значения в наборе чисел;
- вычитание среднего значения из каждого числа в наборе и возведение разности в квадрат;
- сложение всех возведенных в квадрат разностей;
- полученная сумма делится на (*количество чисел во наборе*).
- вычисление квадратного корня из результата.

Функцию StdDevP можно использовать с операторами контекста расширенного синтаксиса.

Примеры

Если мера содержит набор значений (2, 4, 6, 8) StdDevP ([мера]) вернет 2,24.

Sum

Описание

Возвращает сумму меры

Группа функций

Агрегирование

Синтаксис

```
num Sum(measure)
```

Ввод

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примечания

Можно использовать операторы контекста расширенного синтаксиса с функцией Sum.

Примеры

Если для меры "Доходы от продаж" заданы значения 2000, 3000, 4000 и 1000, функция Sum ([Доходы от продаж]) возвращает 10000.

Var

Описание

Возвращает отклонение для меры

Группа функций

Агрегирование

Синтаксис

```
num Var (measure)
```

Ввод

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примечания

Дисперсия – это мера статистического отклонения для множества чисел. Она вычисляется следующим образом:

- поиск среднего значения во множестве чисел
- вычитание среднего значения из каждого числа в множестве и возведение разности в квадрат
- сложение всех возведенных в квадрат разностей
- деление полученной суммы на (*количество чисел во множестве* – 1).

Дисперсия – это возведенное в квадрат значение среднеквадратичного отклонения.

Можно использовать операторы контекста расширенного синтаксиса с функцией Var.

Примеры

Если мера содержит набор значений (2, 4, 6, 8), `Var ([мера])` возвращает 6,67.

См. также

- [StdDev](#)

VarP

Описание

Возвращает отклонение по совокупности для меры

Группа функций

Агрегирование

Синтаксис

```
num VarP (measure)
```

Ввод

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Примечания

Дисперсия генеральной совокупности – это мера статистического отклонения для множества чисел. Она вычисляется следующим образом:

- поиск среднего значения во множестве чисел
- вычитание среднего значения из каждого числа в множестве и возведение разности в квадрат
- сложение всех возведенных в квадрат разностей
- полученная сумма делится на (*количество чисел во множестве*)

Дисперсия генеральной совокупности – это возведенное в квадрат значение среднеквадратичного отклонения генеральной совокупности.

Можно использовать контекстные операторы расширенного синтаксиса с функцией `VarP`.

Примеры

Если `мера` содержит набор значений (2, 4, 6, 8), `VarP([мера])` возвращает 5.

См. также

- `StdDevP`

Символьные функции

Asc

Описание

Возвращает код ASCII для символа

Группа функций

Символьные

Синтаксис

```
int Asc(строка)
```

Входные данные

Параметр	Описание	Тип	Обязательный
строка	Любая строка	Строка	Да

Примечания

Если строка содержит несколько символов, функция возвращает код ASCII для первого символа в строке.

Примеры

`Asc ("A")` возвращает 65.

`Asc ("ab")` возвращает 97.

`Asc ([Страна])` возвращает 85, если значение измерения "Страна" равно "US".

Char

Описание

Возвращает символ, связанный с кодом ASCII

Группа функций

Символьные

Синтаксис

```
string Char(ascii_code)
```

Ввод

Параметр	Описание	Тип	Требуется
ascii_code	Код ASCII	Число	Да

Примечания

Если **число** десятичное, данная функция игнорирует десятичную часть.

Например

s

Char(123) возвращает "{".

Concatenation

Описание

Соединяет две символьных строки

Группа функций

Символьные

Синтаксис

```
строка Concatenation(первая_строка; вторая_строка)
```

Ввод

Параметр	Описание	Тип	Требуется
первая_строка	Первая строка	Строка	Да
вторая_строка	Вторая строка	Строка	Да

Примечания

Для соединения строк также можно использовать оператор "+".

"Первый" + "Второй" возвращает "Первый Второй".

"Первый" + "Второй" + "Третий" возвращает "Первый Второй Третий".

Примеры

Concatenation ("Первый"; "Второй") возвращает "Первый Второй".

Concatenation ("Первый"; Concatenation ("Второй"; "Третий"))
 возвращает "Первый Второй Третий".

Fill

Описание

Создает строку путем повторения строки n раз

Группа функций

Символьные

Синтаксис

```
string Fill(repeating_string; num_repeats)
```

Входные данные

Параметр	Описание	Тип	Обязательный
repeating_string	Повторяющаяся строка	Строка	Да
num_repeats	Количество повторений	Число	Да

Примеры

Fill ("Нью-Йорк"; 2) возвращает "Нью-Йорк Нью-Йорк".

FormatDate

Описание

Форматирует дату в соответствии с указанным форматом

Группа функций

Символьные

Синтаксис

```
строка FormatDate (дата ; format_string)
```

Ввод

Параметр	Описание	Тип	Обязатель- ный
дата	Дата в формате	Дата	Да
format_string	Формат, который будет применен к числу	Строка	Да

Примечания

- Формат вывода зависит от формата даты, примененного к ячейке.
- Форматирование цвета строк (например: [Красный], [Синий] и т. д.) нельзя применять к FormatDate.

Примеры

FormatDate (CurrentDate () ; "dd/MM/yyyy") возвращает "15/12/2005", если текущая дата – 15 декабря 2005 года.

FormatNumber

Описание

Форматирует число согласно указанному формату

Группа функций

Символьные

Синтаксис

```
string FormatNumber(число;format_string)
```

Входные данные

Параметр	Описание	Тип	Обязательный
число	Число для форматирования	Число	Да
format_string	Формат, который будет применен к числу	Строка	Да

Примечания

- Формат вывода зависит от числового формата ячейки.
- Строки цветового форматирования (например, [Красный], [Голубой] и так далее) применить к функции FormatNumber невозможно.

Примеры

FormatNumber([Доход];"#,##.00") возвращает 835 420,00, если [Доход] равен 835 420.

HTMLEncode

Описание

Применяет к строке правила преобразования HTML

Группа функций

Символьные

Синтаксис

```
строка HTMLEncode (html)
```

Ввод

Параметр	Описание	Тип	Обязательный
html	Строка HTML	Строка	Да

Примеры

HTMLEncode ("http://www.businessobjects.com") возвращает "http%3A%2F%2Fwww%2Ebusinessobjects%2Ecom".

InitCap

Описание

Делает заглавной первую букву строки

Группа функций

Символьные

Синтаксис

```
строка InitCap (строка)
```


Ввод

Параметр	Описание	Тип	Обязатель- ный
строка	Строка для преобразова- ния в заглавные буквы	Строка	Да

Примеры

InitCap("мы исходим из той очевидной истины") возвращает "Мы исходим из той очевидной истины".

Left

Описание

Возвращает крайние левые символы строки

Группа функций

Символьные

Синтаксис

```
строка Left(string;num_chars)
```

Ввод

Параметр	Описание	Тип	Обязатель- ный
строка	Строка ввода	строка	Да
num_chars	Количество символов слева, которые следует вернуть	число	Да

Примеры

Left([Страна];2) возвращает "Фр", если [Страна] – "Франция"

LeftPad

Описание

Заполняет строку слева другой строкой

Группа функций

Символьные

Синтаксис

```
string LeftPad(padded_string;length;left_string)
```

Ввод

Параметр	Описание	Тип	Обязатель- ный
padded_string	Исходная строка	Строка	Да
длина	Длина выходной строки	Число	Да
left_string	Строка для добавления слева от padded_string	Строка	Да

Примечания

- Если длина меньше, чем общая длина left_string и padded_string, left_string сокращается.
- Если длина меньше или равна длине padded_string, функция возвращает padded_string.
- Если длина больше, чем общая длина padded_string и left_string, left_string повторяется или частично повторяется достаточное количество раз, чтобы заполнить длину.

Примеры

LeftPad("Йорк"; 8; "Нью-") возвращает "Нью-Йорк"

LeftPad("Йорк"; 6; "Нью") возвращает "Нью-Йорк"

`LeftPad("Йорк";11;"Нью")` возвращает "Нью Нью-Йорк"

`LeftPad("Нью";2;"Йорк")` возвращает "Нью".

LeftTrim

Описание

Удаляет начальные пробелы из строки

Группа функций

Символьные

Синтаксис

```
string LeftTrim(trimmed_string)
```

Входные данные

Параметр	Описание	Тип	Требуемая
trimmed_string	Строка для усечения	Строка	Да

Примеры

Функция `LeftTrim([Страна])` возвращает "Франция", если [Страна] имеет значение " Франция".

Length

Описание

Возвращает число символов в строке

Группа функций

Символьные

Синтаксис

```
int Length(строка)
```

Входные данные

Параметр	Описание	Тип	Обязательный
строка	Входная строка	Строка	Да

Примеры

Функция `Length([Фамилия])` возвращает 6, если для объекта [Фамилия] задано значение "Иванов".

Lower

Описание

Преобразует строку в нижний регистр

Группа функций

Символьные

Синтаксис

```
string Lower(строка)
```

Входные данные

Параметр	Описание	Тип	Обязательный
строка	Строка, преобразуемая в нижний регистр	Строка	Да

Примеры

Функция `Lower("Нью-Йорк")` возвращает "нью-йорк".

Match

Описание

Определяет соответствие строки шаблону

Группа функций

Символьные

Синтаксис

```
bool Match(test_string;pattern)
```

Ввод

Параметр	Описание	Тип	Требуемая
test_string	Строка для проверки соответствия образцу текста	строка	Да
образец	Строка	образца текста	Да

Примечания

- Образец содержать символы подстановки "*" (заменяет набор любых символов) или "?" (заменяет один символ).

Примеры

Match([Страна]; "Ф*") возвращает значение True, если [Страна] – "Франция".

Match([Страна]; "?Ш?") возвращает значение True, если [Страна] – США.

Match("Нью-Йорк"; "П*") возвращает значение False

Pos

Описание

Возвращает начальную позицию текстового шаблона в строке

Группа функций

Символьные

Синтаксис

```
int Pos(test_string;pattern)
```

Ввод

Параметр	Описание	Тип	Требуемая
test_string	Строка для проверки вхождения в нее образца текста	строка	Да
образец	Строка	образца текста	Да

Примечания

- Если образец встречается больше одного раза, Pos возвращает позицию первого экземпляра.

Примеры

Pos ("Нью-Йорк"; "Нью") возвращает 1

Pos ("Нью-Йорк, Нью-Йорк"; "Нью") возвращает 1.

Pos ("Нью-Йорк"; "Йорк") возвращает 5.

Replace

Описание

Заменяет часть строки другой строкой

Группа функций

Символьные

Синтаксис

строка `Replace(replace_in;replaced_string;replace_with)`

Ввод

Параметр	Описание	Тип	Требуемый
replace_in	Строка, в которой заменяется текст	строка	Да
replaced_string	Заменяемый текст	строка	Да
replace_with	Текст, который заменяет replaced_string	строка	Да

Примеры

`Replace("Нью-Йорк";"Орк";"орк")` возвращает "Нью-Йорк".

Right

Описание

Возвращает самые правые символы из строки

Группа функций

Символьные

Синтаксис

```
string Right(строка; num_chars)
```

Входные данные

Параметр	Описание	Тип	Обязатель- ный
строка	Любая строка	строка	Да
num_chars	Количество символов справа, которые следует вернуть	число	Да

Примеры

Функция `Right ([Страна] ; 2)` возвращает "ия", если объект [Страна] имеет значение "Франция".

RightPad

Описание

Заполняет строку справа другой строкой

Группа функций

Символьные

Синтаксис

```
string RightPad(padded_string; length; right_string)
```


Ввод

Параметр	Описание	Тип	Обязатель- ный
<code>padded_string</code>	Исходная строка	Строка	Да
длина	Длина выходной строки	Число	Да
<code>right_string</code>	Строка для добавления справа от <code>padded_string</code>	Строка	Да

Примечания

- Если длина меньше, чем общая длина `right_string` и `padded_string`, `right_string` сокращается.
- Если длина меньше или равна длине `padded_string`, функция возвращает `padded_string`.
- Если длина больше, чем общая длина `padded_string` и `right_string`, `right_string` повторяется или частично повторяется достаточное количество раз, чтобы заполнить всю длину строки.

Примеры

`RightPad("Нью-"; 8; "Йорк")` возвращает "Нью-Йорк"

`RightPad("Нью-"; 6; "Йорк")` возвращает "Нью-Йо"

`RightPad("Нью-"; 11; "Йорк")` возвращает "Нью-ЙоркЙор"

`RightPad("Нью-"; 2; "Йорк")` возвращает "Нью".

RightTrim

Описание

Удаляет пробелы на конце строки

Группа функций

Символьные

Синтаксис

```
строка RightTrim(trimmed_string)
```

Ввод

Параметр	Описание	Тип	Требуемая
trimmed_string	Строка для усечения	Строка	Да

Примеры

RightTrim([Страна]) возвращает "Франция", если [Страна] – "Франция ".

Substr

Описание

Возвращает часть строки

Группа функций

Символьные

Синтаксис

```
строка SubStr(строка; начало; длина)
```

Входные данные

Параметр	Описание	Тип	Обязательный
строка	Любая строка	Строка	Да
начало	Начальная позиция извлеченной строки	Число	Да
длина	Длина извлеченной строки	Число	Да

Примеры

SubStr ("Великобритания"; 1; 5) возвращает "Велик".

SubStr ("Великобритания"; 7; 7) возвращает "Британия".

Trim

Описание

Удаляет в начале и конце строки

Группа функций

Символьные

Синтаксис

```
строка Trim(trimmed_string)
```

Ввод

Параметр	Описание	Тип	Обязательный
строка	Строка для удаления	Строка	Да

Примеры

`Trim (" Великобритания ")` возвращает "Великобритания".

Upper

Описание

Преобразует строку символов в верхний регистр

Группа функций

Символьные

Синтаксис

```
string Upper(string)
```

Вводимые данные

Параметр	Описание	Тип	Обязательный
строка	Строка для преобразования	Строка	Да

Примеры

`Upper ("Нью-Йорк")` возвращает "НЬЮ-ЙОРК".

UrlEncode

Описание

Применяет к строке правила кодировки URL

Группа функций

Символьные

Синтаксис

```
строка UrlEncode (html)
```

Входные данные

Параметр	Описание	Тип	Обязатель- ный
html	URL-адрес, который необходимо закодировать	Строка	Да

Примеры

UrlEncode ("http://www.businessobjects.com") возвращает "http%3A%2F%2Fwww%2Ebusinessobjects%2Ecom".

WordCap

Описание

Перевод первых букв всех слов в строке в верхний регистр

Группа функций

Символьные

Синтаксис

```
строка WordCap(string)
```

Ввод

Параметр	Описание	Тип	Обязательный
строка	Строка, которую необходимо перевести в верхний регистр	Строка	Да

Примеры

WordCap("Доход с продаж за март") возвращает "Доход С Продаж За Март".

Функции даты и времени

CurrentDate

Описание

Возвращает текущую дату, отформатированную в соответствии с национальными установками

Группа функций

Дата и время

Синтаксис

```
дата CurrentDate()
```

Примеры

CurrentDate() возвращает 10 сентября 2002 года, если дата – 10 сентября 2002 года.

CurrentTime

Описание

Возвращает текущее время, отформатированное в соответствии с национальными установками

Группа функций

Дата и время

Синтаксис

```
время CurrentTime ()
```

Примеры

CurrentTime возвращает 11:15, если текущее время 11:15.

DayName

Описание

Возвращает название дня в дате

Группа функций

Дата и время

Синтаксис

```
строка DayName (дата)
```

Ввод

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примеры

DayName ([Дата резервирования]) возвращает "Суббота", если дата в [Дата резервирования] – 15 декабря 2001 года (которая приходится на субботу).

Примечание

Дата на входе должна быть представлена переменной. Прямое указание даты, например DayName ("07/15/2001"), не допускается.

DayNumberOfMonth

Описание

Возвращает номер дня в месяце

Группа функций

Дата и время

Синтаксис

```
int DayNumberOfMonth (дата)
```


Входные данные

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примеры

`DayNumberOfMonth ([Дата резервирования])` возвращает 15, если в [Дата сохранения] стоит 15 декабря 2001.

DayNumberOfWeek

Описание

Возвращает номер дня в неделе

Группа функций

Дата и время

Синтаксис

целое число `DayNumberOfWeek (дата)`

Ввод

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примечания

Первым днем недели в Web Intelligence считается понедельник.

Примеры

`DayNumberOfWeek ([Дата резервирования])` возвращает 1, если дата в [Дата резервирования] – 2 мая 2005 года (понедельник).

DayNumberOfYear

Описание

Возвращает номер дня в году

Группа функций

Дата и время

Синтаксис

```
int DayNumberOfYear (дата)
```

Входные данные

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примеры

DayNumberOfYear ([Дата резервирования]) возвращает 349, если в качестве значения параметра [Дата резервирования] установлено 15 декабря 2001.

DaysBetween

Описание

Возвращает количество дней между двумя датами

Группа функций

Дата и время

Синтаксис

```
int DaysBetween(first_date;last_date)
```

Входные данные

Параметр	Описание	Тип	Обязательный
first_date	Первая дата	Дата	Да
last_date	Последняя дата	Дата	Да

Примеры

DaysBetween([Дата продажи]; [Дата выставления счета])
 возвращает 2, если [Дата продажи] – 15 декабря 2001 г., а [Дата
 выставления счета] – 17 декабря 2001 г.

LastDayOfMonth

Описание

Возвращает дату последнего дня в месяце

Группа функций

Дата и время

Синтаксис

```
дата LastDayOfMonth(дата)
```

Ввод

Параметр	Описание	Тип	Обязательный
дата	Любая дата в месяце	Дата	Да

Примеры

`LastDayOfMonth([Дата продажи])` возвращает 31 декабря 2005 года, если [Дата продажи] равна 11 декабря 2005 года.

LastDayOfWeek

Описание

Возвращает дату последнего дня недели

Группа функций

Дата и время

Синтаксис

```
date LastDayOfWeek(дата)
```

Входные данные

Параметр	Описание	Тип	Обязательный
дата	Любая дата недели	Дата	Да

Примечания

Первым днем недели в Web Intelligence считается понедельник.

Примеры

Функция `LastDayOfWeek([Дата продажи])` возвращает 15 мая 2005 года (воскресенье), если [Дата продажи] имеет значение 11 мая 2005 года.

Month

Описание

Возвращает название месяца в дате

Группа функций

Дата и время

Синтаксис

```
string Month(дата)
```

Входные данные

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примеры

Функция `Month([Дата резервирования])` возвращает "Декабрь", когда [Дата резервирования] – 15 декабря 2005 г.

MonthNumberOfYear

Описание

Возвращает номер месяца в дате

Группа функций

Дата и время

Синтаксис

```
int MonthNumberOfYear (дата)
```

Входные данные

Параметр	Описание	Тип	Обязательный
дата	Любая дата в году	Дата	Да

Пример

Функция `MonthNumberOfYear ([Дата бронирования])` возвращает 12, если [Дата бронирования] – 15 декабря 2005 г.

MonthsBetween

Описание

Возвращает количество месяцев между двумя датами

Группа функций

Дата и время

Синтаксис

```
int MonthsBetween (first_date; last_date)
```

Входные данные

Параметр	Описание	Тип	Обязательный
first_date	Первая дата	Дата	Да
last_date	Последняя дата	Дата	Да

Примеры

Функция `MonthsBetween([Дата продажи]; [Дата выставления счета])` возвращает 1, если [Дата продажи] имеет значение 2 декабря 2005 г., а [Дата выставления счета] имеет значение 2 января 2006 г.

Quarter

Описание

Возвращает номер квартала в дате

Группа функций

Дата и время

Синтаксис

```
int Quarter(дата)
```

Входные данные

Параметр	Описание	Тип	Обязательный
дата	Любая дата в квартале	Дата	Да

Примеры

Формула `Quarter([Дата резервирования])` возвращает значение 4, если дата в переменной [Дата резервирования] равна 15 декабря 2005 года.

RelativeDate

Описание

Возвращает дату, отстоящую от другой даты

Группа функций

Дата и время

Синтаксис

```
дата RelativeDate(start_date; num_days)
```

Ввод

Параметр	Описание	Тип	Обязательный
start_date	Начальная дата	Дата	Да
num_days	Количество дней от начальной даты	Число	Да

Примечания

Параметр num_days может быть отрицательным, чтобы возвращать дату, ранее чем start_date.

Примеры

RelativeDate[Дата резервирования]; 2) возвращает 17 декабря 2005 года, если для объекта [Дата резервирования] установлено значение 15 декабря 2005 года.

RelativeDate[Дата резервирования]; -3) возвращает 9 января 2007 года, если [Дата резервирования] – 12 января 2007 года.

TimeDim

Описание

Измерение времени `TimeDim` позволяет строить оси времени на основе объекта юниверса типа данных. Измерение `TimeDim` возвращает данные для дат, задаваемых первым параметром, за периоды времени, задаваемые вторым параметром. Для периодов, не содержащих данных, возвращается первый день периода. Это позволяет построить полную ось для любого заданного периода. Такой подход гарантирует следующее:

- На оси сохраняется естественный порядок дат и времени (сначала идут самые старые объекты, за ними более новые).
- На оси будут представлены все периоды, заключенные между минимальной и максимальной датой текущего контекста.

Группа функций

Дата и время

Синтаксис

```
TimeDim([Date Type]; Period Type)
```

Данные ввода

Параметр	Описание	Тип	Обязателен
Тип даты	Объект даты для отчета, например InvoiceDate.	Дата	Да
Тип периода	Период для результатов. Может иметь следующие значения: <ul style="list-style-type: none"> • DayPeriod • MonthPeriod • QuarterPeriod • YearPeriod Если значение не выбрано, по умолчанию используется значение DayPeriod.	Дата	Дополнительно

Приведенные выше функции необходимо использовать совместно со следующими функциями:

- DayName
- DayNumberOfMonth
- DayNumberOfWeek
- DayNumberOfYear
- Month
- MonthNumberOfYear
- Квартал
- Год
- FormatDate

Пример

В первой таблице приведены данные, относящиеся к датам. В приведенных ниже примерах запросов показан порядок интерпретации результатов.

Дата счета-фактуры	Доход
1/3/00	31607
1/8/00	31244
7/3/00	38154

Запрос `DayName (TimeDim ([Invoice Date] ; QuarterPeriod)` возвращает значения за каждый день из приведенной выше таблицы.

Дата счета-фактуры	Доход
1/3/00	31607
1/8/00	31244
4/1/00	
7/3/00	38154

Необходимо отформатировать результаты функции `timedim` с помощью функции `Quarter`. Результаты, возвращаемые функцией `Quarter` (Q1, Q2...), формируют следующую таблицу результатов:

Дата счета-фактуры	Доход
K1	62851
K2	
K3	38154

ToDate

Описание

Возвращает символьную строку, отформатированную в соответствии с форматом даты

Группа функций

Дата и время

Синтаксис

```
date ToDate (date_string; format)
```

Ввод

Параметр	Описание	Тип	Обязательный
date_string	Дата для форматирования	строка	Да
формат	Формат даты	строка	Да

Примеры

ToDate ("15/12/2002"; "dd/MM/yyyy") возвращает 15/12/2002.

Week

Описание

Возвращает номер недели в году

Группа функций

Дата и время

Синтаксис

```
int Week (дата)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примеры

`Week ([Дата бронирования])` возвращает 1, когда [Дата бронирования] – 4 января 2004 г. (первая неделя 2004 года).

Year

Описание

Возвращает год в дате

Группа функций

Дата и время

Синтаксис

```
int Year(date)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
дата	Входная дата	Дата	Да

Примеры

`Year ([Дата записи])` возвращает 2005, если дата [Дата записи] – 15 декабря 2005.

Data Provider functions

Connection

Описание

Возвращает параметры соединения базы данных, используемого поставщиком данных

Группа функций

Источник данных

Синтаксис

```
string Connection(dp)
```

Входные данные

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- Имя поставщика данных необходимо указывать в квадратных скобках.
- По соображениям безопасности вывод этой функции не содержит имя хоста базы данных, а также имя и пароль пользователя.

DataProvider

Описание

Возвращает имя поставщика данных, в котором содержится объект отчета

Группа функций

Источник данных

Синтаксис

```
строка DataProvider (obj)
```

Ввод

Параметр	Описание	Тип	Обязательный
об.	Объект отчета	Объект отчета	Да

Примеры

`DataProvider([Общий доход])` возвращает "Продажи", если мера [Общий доход] содержится в поставщике данных "Продажи".

DataProviderKeyDate

Описание

Возвращает ключевую дату поставщика данных

Группа функций

Источник данных

Синтаксис

```
дата DataProviderKeyDate (dp)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- Имя поставщика данных необходимо указывать в квадратных скобках.
- Возвращенная ключевая дата форматируется в соответствии с языковым стандартом документа.

Примеры

`DataProviderKeyDate ([Продажи])` возвращает 3 августа 2007 года, если ключевая дата для поставщика данных продаж – 3 августа 2007 года.

DataProviderKeyDateCaption

Описание

Возвращает заголовок ключевой даты поставщика данных

Группа функций

Источник данных

Синтаксис

```
string DataProviderKeyDateCaption (dp)
```


Входные данные

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

`DataProviderKeyDateCaption([Продажи])` возвращает "Текущая календарная дата", если в заголовке ключевой даты поставщика данных "Продажи" стоит "Текущая календарная дата".

DataProviderSQL

Описание

Возвращает SQL, созданный поставщиком данных

Группа функций

Источник данных

Синтаксис

```
string DataProviderSQL(dp)
```

Входные данные

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

`DataProviderSQL([Запрос 1])` возвращает **"SELECT country.country_name FROM country"**, если код SQL для поставщика данных указан как **"SELECT country.country_name FROM country"**.

DataProviderType

Описание

Возвращает тип поставщика данных

Группа функций

Источник данных

Синтаксис

```
строка DataProviderType (dp)
```

Ввод

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- `DataProviderType` возвращает "Юниверс" для поставщиков данных юниверса или "Персональные данные" для поставщиков персональных данных.
- Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

`DataProviderType ([Продажи])` возвращает "Юниверс", если поставщик данных "Продажи" основан на юниверсе.

IsPromptAnswered

Описание

Определяет, был ли получен ответ на приглашение

Группа функций

Источник данных

Синтаксис

```
bool IsPromptAnswered([dp;]prompt_string)
```

Данные ввода

Параметр	Описание	Тип	Обязатель- ный
dp	Поставщик данных, со- держащий приглашение	Поставщик данных	Нет
prompt_string	Текст приглашения	Строка	Да

Примечания

Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

Функция `IsPromptAnswered("Выберите город")` возвращает значение `True`, если на запрос на ввод данных с текстом "Выберите город", был получен ответ.

Функция `IsPromptAnswered ([Продажи]; "Выберите город")` возвращает значение `True`, если на запрос на ввод данных с текстом "Выберите город" в поставщике данных [Продажи] был получен ответ.

LastExecutionDate

Описание

Возвращает дату последнего обновления поставщика данных

Группа функций

Источник данных

Синтаксис

```
data LastExecutionDate(dp)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- Если в отчете содержится только один поставщик данных, параметр dp можно пропустить
- Имя поставщика данных необходимо указывать в квадратных скобках.
- Можно использовать функцию `DataProvider` для предоставления ссылки на поставщика данных.

Примеры

`LastExecutionDate([Запрос продаж])` возвращает "4/3/2002", если поставщик данных "Запрос продаж" последний раз обновлялся 4 марта 2002 года.

[См. также](#)

- [DataProvider](#)

LastExecutionDuration

Описание

Возвращает время последнего обновления поставщика данных в секундах

Группа функций

Источник данных

Синтаксис

```
num LastExecutionDuration(dp)
```

Входные данные

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

Функция `LastExecutionDuration([Продажи])` возвращает 3, если поставщик данных "Продажи" затратил 3 секунды на возврат данных во время его последнего запуска.

LastExecutionTime

Описание

Возвращает время последнего обновления поставщика данных

Группа функций

Источник данных

Синтаксис

```
time LastExecutionTime(dp)
```

Входные данные

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- Если в отчете содержится только один поставщик данных, параметр dp можно опустить.
- Функцию `DataProvider` можно использовать для указания ссылки на поставщик данных.
- Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

Функция `LastExecutionTime([Запрос продаж])` возвращает "14:48:00", если последнее обновление поставщика данных "Запрос продаж" производилось в 14:48:00.

[См. также](#)

- [DataProvider](#)

NumberOfDataProvider

Описание

Возвращает количество поставщиков данных в отчете

Группа функций

Источник данных

Синтаксис

```
int NumberOfDataProviders()
```

Примеры

Функция `NumberOfDataProviders()` возвращает 2, если в отчете указаны два поставщика данных.

NumberOfRows

Описание

Возвращает количество строк в поставщике данных

Группа функций

Источник данных

Синтаксис

```
int NumberOfRows(dp)
```

Ввод

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- Имя поставщика данных необходимо указывать в квадратных скобках.
- Можно использовать функцию `DataProvider` для предоставления ссылки на поставщика данных.

Примеры

`NumberOfRows([Запрос 1])` возвращает 10, если в поставщике данных "Запрос 1" содержится 10 строк.

См. также

- [DataProvider](#)

RefValueDate

Описание

Возвращает справочные данные, используемые для отслеживания данных

Группа функций

Источник данных

Синтаксис

```
дата RefValueDate ( )
```

Примеры

Функция RefValueDate () возвращает значение 15 декабря 2008 г., если опорная дата – 15 декабря 2008 г.

RefValueUserReponse

Описание

Возвращает ответ на запрос на ввод данных, когда опорные данные были текущими данными

Группа функций

Источник данных

Синтаксис

```
string RefValueUserResponse ([dp;]prompt_string[;Index])
```

Данные ввода

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Нет
prompt_string	Текст приглашения	Строка	Да
Index	Указывает функции на то, что необходимо возвращать основные ключи базы данных для значений приглашения	Ключевое слово	Нет

Примечания

- Эта функция возвращает пустую строку, если отслеживание данных не включено.
- Имя поставщика данных необходимо указывать в квадратных скобках.
- Для указания ссылки на поставщик данных можно использовать функцию `DataProvider`.
- При выборе более одного значения для ответа на приглашение функция возвращает строку, состоящую из списка значений (или основных ключей, если указан оператор `Index`), разделенных между собой точками с запятой.

Примеры

`RefValueUserResponse("Какой город?")` возвращает "Лос-Анджелес", если в поле "Какой город?" было указано "Лос Анджелес" запрос на ввод данных в момент времени, когда опорные данные были текущими.

`RefValueUserResponse([Запрос продаж]; "Какой город?")` возвращает "Лос-Анджелес", если в поле "Какой город?" было указано "Лос Анджелес" запрос на ввод данных в поставщике данных "Запрос продаж" в момент времени, когда опорные данные были текущими.

UniverseName

Описание

Возвращает имя юниверса, на котором основан поставщик данных

Группа функций

Источник данных

Синтаксис

```
строка UniverseName (dp)
```

Ввод

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Да

Примечания

- В приложении Web Intelligence выполняется автоматическое обновление имен поставщиков данных в формуле. Если в приведенном выше примере присвоить имя поставщику данных K1, формула будет записана как `UniverseName (K1)`.
- Имя поставщика данных необходимо указывать в квадратных скобках.
- Можно использовать функцию `DataProvider` для предоставления ссылки на поставщика данных.

Примеры

`UniverseName ([Запрос 1])` возвращает "eFashion", если поставщик данных основан на юниверсе eFashion.

См. также

- [DataProvider](#)

UserResponse

Описание

Возвращает ответ на запрос

Группа функций

Источник данных

Синтаксис

```
string UserResponse ([dp;]prompt_string[;Index])
```

Данные ввода

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Нет
prompt_string	Текст приглашения	Строка	Да
Index	Указывает функции на то, что необходимо возвращать основные ключи базы данных для значений приглашения	Ключевое слово	Нет

Примечания

- Имя поставщика данных необходимо указывать в квадратных скобках.
- Функцию `DataProvider` можно использовать для указания ссылки на поставщика данных.
- При выборе нескольких значений для ответа на приглашение функция возвращает строку, состоящую из списка значений (или основных ключей, если указан оператор `Index`), разделенных между собой точкой с запятой.

Примеры

`UserResponse("Какой город?")` возвращает "Лос-Анджелес", если в поле запросов "Какой город?" было указано "Лос-Анджелес".

`UserResponse([Запрос продаж]; "Какой город?")` возвращает "Лос-Анджелес", если в поле "Какой город?" было указано "Лос Анжелес" приглашения в поставщике данных "Запрос продаж".

`UserResponse([Запрос продаж]; "Какой город?"; Индекс)` возвращает 23, если в поле запросов "Какой город" было указано "Лос-Анджелес" в поставщике данных "Запрос продаж", а первичный ключ значения "Лос-Анджелес" в базе данных – 23.

Функции Документа

DocumentAuthor

Описание

Возвращает регистрационное имя создателя документа в приложении InfoView

Группа функций

Документ

Синтаксис

```
строка DocumentAuthor()
```

Примеры

`DocumentAuthor()` возвращает "gkn", если регистрационное имя автора документа – "gkn".

DocumentCreationDate

Описание

Возвращает дату создания документа

Группа функций

Документ

Синтаксис

```
дата DocumentCreationDate ()
```

Примеры

`DocumentCreationDate ()` возвращает 15 декабря 2008 года, если документ был создан 15 декабря 2008 года.

DocumentCreationTime

Описание

Возвращает время создания документа.

Группа функций

Документ

Синтаксис

```
время DocumentCreationTime ()
```

Примеры

`DocumentCreationTime ()` возвращает 11:15, если документ создан в 11:15.

DocumentDate

Описание

Возвращает дату последнего сохранения документа

Группа функций

Документ

Синтаксис

```
дата DocumentDate ()
```

Примеры

`DocumentDate ()` возвращает 8 августа 2005 года, если документ был в последний раз сохранен 8 августа 2005 года.

DocumentName

Описание

Возвращает имя документа

Группа функций

Документ

Синтаксис

```
строка DocumentName ()
```

Примеры

`DocumentName ()` возвращает "Отчет о продажах", если документ назван "Отчет о продажах".

DocumentPartiallyRefreshed

Описание

Определяет, обновлен ли документ частично.

Группа функций

Документ

Синтаксис

```
логическое значение DocumentPartiallyRefreshed()
```

Примечания

DocumentPartiallyRefreshed() возвращает логическое значение, которое можно использовать в функции If.

Примеры

DocumentPartiallyRefreshed() возвращает значение True, если документ обновлен частично.

DocumentTime

Описание

Возвращает время последнего сохранения документа

Группа функций

Документ

Синтаксис

```
время DocumentTime()
```


Примечания

Формат возвращаемого времени зависит от формата ячейки.

Например

`DocumentTime()` возвращает 15:45, если документ был последний раз сохранен в 15:45.

DrillFilters

Описание

Возвращает фильтры детализации, примененные к документу или объекту в режиме детализации

Группа функций

Документ

Синтаксис

```
строка DrillFilters(obj|separator)
```

Ввод

Параметр	Описание	Тип	Обязательный
об.	Объект отчета	Объект отчета	Необходим obj или separator
разделитель	Разделитель фильтров детализации	Строка	Необходим obj или separator

Примечания

- Можно вставить `DrillFilters` напрямую, вставив ячейку `DrillFilters`, без необходимости ввода формулы вручную.
- Если объект не указан, функция возвращает значения всех фильтров детализации, примененных к документу.

Примеры

`DrillFilters()` возвращает "США", если в документе применен фильтр детализации, допускающий в объекте [Страна] только значение "США".

`DrillFilters()` возвращает "США – 1999", если в документе применен фильтр, допускающий в объекте [Страна] только значение "США" и в объекте [Год] только значение "1999"

`DrillFilters("/")` возвращает "США / 1999", если в документе применены фильтры, допускающие в объекте [Страна] только значение "США" и в объекте [Год] только значение "1999"

`DrillFilters ([Квартал])` возвращает "К3", если в документе применен фильтр детализации, допускающий в объекте [Квартал] только значение "К3"

PromptSummary

Описание

Возвращает текст запроса и ответ пользователя на все запросы на ввод данных в документе

Группа функций

Документ

Синтаксис

```
string PromptSummary()
```

Примеры

`QuerySummary()` возвращает сведения о всех запросах на ввод данных в документе.

Примерный вывод:

```
Enter Quantity Sold: 5000  
Enter value(s) for State (optional): California,
```

```
Texas, Utah
Enter Customer (optional):
```

QuerySummary

Описание

Возвращает информацию о запросах в документе

Группа функций

Документ

Синтаксис

```
строка QuerySummary([dp])
```

Ввод

Параметр	Описание	Тип	Обязательный
dp	Поставщик данных	Поставщик данных	Нет

Примечания

- Имя поставщика данных необходимо указывать в квадратных скобках.

Примеры

`QuerySummary()` возвращает сведения о всех запросах в документе.

`QuerySummary([Запрос 1])` возвращает сведения о запросах, основанных на поставщике данных [Запрос 1].

Пример вывода:

```
Query 1:
Universe: eFashion
Last execution time: 1s
```

```

NB of rows: 34500
Result objects: State, Year, Sales Revenue
Scope of analysis: State, City, Year, Quarter,
Month
Filters:
  (State inlist{"US";"France";}
  And (Sales Revenue Greater Than 1000000
  Or Sales Revenue Less Than 10000))

Query 2:
Source file: D:\Data\dataacar.xls
Result objects: State, Year, Sales Revenue
  
```

ReportFilter

Описание

Возвращает фильтры, примененные к объекту или отчету

Группа функций

Документ

Синтаксис

```
string ReportFilter(объект)
```

Входные данные

Параметр	Описание	Тип	Обязательный
об.	Объект отчета	Объект отчета	Да

Примеры

Функция `ReportFilter([Страна])` возвращает значение "США", если для объекта "Страна" задан фильтр отчета, которым допускаются только значения "США".

ReportFilterSummary

Описание

Возвращает общие сведения о фильтрах отчета в документе или отчете

Группа функций

Документ

Синтаксис

```
string ReportFilterSummary(report_name)
```

Данные ввода

Параметр	Описание	Тип	Обязательное
report_name	Имя отчета	Строка	Нет

Примечания

Если параметр `report_name` опущен, функция `ReportFilterSummary` возвращает общие сведения обо всех фильтрах отчетов в документе.

Примеры

Функция `ReportFilterSummary()` возвращает сведения обо всех фильтрах отчетов в документе.

Функция `ReportFilterSummary("Report1")` возвращает сведения обо всех фильтрах отчетов в отчете "Отчет1".

Пример вывода:

```
Filters on Report1:
    (Sales Revenue Greater Than 1000000
    Or (Sales Revenue Less Than 3000))
Filters on Section on City:
    (City InList{"Los Angeles";"San Diego";})
Ranking Filter:
```

```
(Top 10 & Bottom 10 [Customer] Based on [Sales  
Revenue] (Count))
```

Логические функции

Even

Описание

Определяет, является ли число четным

Группа функций

Логические

Синтаксис

```
bool Even (число)
```

Входные данные

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примечания

- Функция **Even** возвращает булево значение, которое можно использовать в функции **If**.
- Если функция **Even** размещена в столбце, в приложении Web Intelligence возвращаемое значение преобразуется в целое число (1=истина; 0=ложь). Это число можно отформатировать с помощью формата булева числа.

Примеры

Even (4) возвращает значение True.

Even (3) возвращает значение False.

`Even (23, 2)` возвращает значение `False`.

`Even (-4)` возвращает значение `True`.

`Even (-2, 2)` возвращает значение `False`.

IsDate

Описание

Определяет, является ли значение датой

Группа функций

Логические

Синтаксис

```
bool IsDate (obj)
```

Ввод

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- `IsDate` возвращает булево значение, которое можно использовать в функции `If`.
- Если поместить функцию `IsDate()` в столбец, приложение Web Intelligence преобразует введенное значение в целое число (1=истина; 0=ложь). Потом это число можно форматировать с использованием формата булевых чисел.

Примеры

`IsDate ([Reservation Date])` возвращает значение `True`, если [Дата бронирования] является датой.

If(IsDate([Дата бронирования]) Then "Дата" Else "Не дата"
возвращает "Дата", если [Дата бронирования] – это дата.

См. также

- *If...Then...Else*

IsError

Описание

Определяет, возвращает ли объект ошибку

Группа функций

Логические

Синтаксис

```
bool IsError(объект)
```

Входные данные

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- Функция **IsError** возвращает булево значение, которое можно использовать в функции **If**.
- Если поместить функцию **IsError** непосредственно в столбец, в приложении Web Intelligence возвращаемое значение будет преобразовано в целое. Для этого числа можно использовать формат булева числа.

Примеры

Функция **IsError([Доход])** возвращает значение **False**, если переменная **[Доход]** не возвращает ошибку.

Функция `IsError([Среднее число гостей])` возвращает значение `True`, если переменная `[Среднее число гостей]` возвращает ошибку деления на ноль (`#DIV/0`).

Функция `If IsError([Среднее число гостей]) Then "Ошибка" Else "Нет ошибок"` возвращает значение "Ошибка", если переменная `[Среднее число гостей]` возвращает ошибку деления на ноль (`#DIV/0`).

См. также

- [If...Then...Else](#)

IsLogical

Описание

Определяет, является ли значение булевым

Группа функций

Логические

Синтаксис

```
bool IsLogical(объект)
```

Входные данные

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- Функция `IsLogical` возвращает булево значение, которое можно использовать в функции `If`.
- Если поместить функцию `IsLogical` непосредственно в столбец, в приложении Web Intelligence возвращаемое значение будет преобразовано в целое. Для этого числа можно использовать формат булева числа.

Примеры

Функция `IsLogical(IsString([Страна]))` возвращает значение `True`.

Функция `IsLogical([Страна])` возвращает значение `False`, если страна возвращает какой-либо тип данных, отличный от булева.

Функция `If IsLogical(IsDate([Страна])) Then "Булев" Else "Не булев"` возвращает "Булев".

См. также

- [If...Then...Else](#)

IsNull

Описание

Определяет, является ли значение неопределенным

Группа функций

Логические

Синтаксис

```
bool IsNull(obj)
```

Ввод

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- `IsNull` возвращает логическое значение, которое можно использовать в функции `If`.

- Если поместить `IsNull` прямо в столбец, Web Intelligence преобразует возвращаемое значение в целое. Для этого числа можно использовать параметр булевого числа.

Примеры

`IsNull ([Доход])` возвращает значение `False`, если переменная `[Доход]` не является неопределенным значением.

`IsNull ([Гостей в среднем])` возвращает значение `True`, если переменная `[Гостей в среднем]` не является неопределенным значением.

См. также

- [If...Then...Else](#)

IsNumber

Описание

Определяет, является ли значение числом

Группа функций

Логические

Синтаксис

```
bool IsNumber (obj)
```

Ввод

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- `IsNumber` возвращает булево значение, которое можно использовать в функции `If`.

- Если вставить функцию `IsNumber` непосредственно в столбец, Web Intelligence преобразует возвращаемое значение в целое. Позже это число можно форматировать с использованием формата булева числа.

Примеры

`IsNumber([Доход])` возвращает значение `True`, если переменная `[Доход]` является числом.

`IsNumber([Имя клиента])` возвращает значение `False`, если переменная `[Имя клиента]` не является числом.

`If IsNumber([Имя клиента]) Then "Число" Else "Не число"` возвращает "Не число", если переменная `[Имя клиента]` не является числом.

См. также

- [*If...Then...Else*](#)

IsString

Описание

Определяет, является ли значение строкой

Группа функций

Логические

Синтаксис

```
bool IsString(объект)
```

Входные данные

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- Функция `IsString` возвращает булево значение, которое можно использовать в функции `If`.
- Если функция `IsString` размещена в столбце, в приложении Web Intelligence возвращаемое значение преобразуется в целое число. Для этого числа можно использовать формат булева числа.

Примеры

Функция `IsString([Доход])` возвращает значение `False`, если переменная `[Доход]` не является строкой.

Функция `IsString([Имя покупателя])` возвращает значение `True`, если переменная `[Имя покупателя]` является строкой.

Функция `If IsString([Имя покупателя]) Then "Строка" Else "Не строка"` возвращает значение "Строка", если переменная `[Имя покупателя]` является строкой.

См. также

- *If...Then...Else*

IsTime

Описание

Определяет, является ли переменная переменной времени

Группа функций

Логические

Синтаксис

```
bool IsTime (obj)
```

Ввод

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

- `IsTime` возвращает булево значение, которое можно использовать в функции `If`.
- Если поместить `IsTime` прямо в столбец, Web Intelligence преобразует возвращаемое значение в целое. Для этого числа можно использовать параметр булевого числа.

Примеры

`IsTime ([Срок резервации])` возвращает значение `True`, если переменная `[Срок резервации]` является переменной времени. .

`IsTime ([Гостей в среднем])` возвращает значение `False`, если переменная `[Гостей в среднем]` не является переменной времени.

`If IsTime ([Гостей в среднем]) Then "Время" Else "Не время"` возвращает "Не время", если переменная `[Гостей в среднем]` не является переменной времени.

См. также

- [*If...Then...Else*](#)

Odd

Описание

Определяет, является ли число нечетным

Группа функций

Логические

Синтаксис

```
bool Odd(number)
```

Ввод

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примечания

- Odd возвращает булево значение, которое можно использовать в функции If.
- Если поместить Odd непосредственно в столбец, Web Intelligence преобразует возвращаемое значение в целое. Для этого числа можно использовать форматирование булевого числа.
- Odd опускает дробные части десятичных чисел.

Примеры

Odd(5) возвращает значение True.

Odd(4) возвращает значение False.

Odd(23.2) возвращает значение True.

Odd(24.2) возвращает значение True.

Odd(-23.2) возвращает значение True.

Odd(-24.2) возвращает значение True.

См. также

- [If...Then...Else](#)

Числовые функции

Abs

Описание

Возвращает абсолютное значение числа

Группа функций

Числовой

Синтаксис

```
num Abs (number)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примеры

Abs (25) возвращает 25.

Abs (-11) возвращает 11.

Ceil

Описание

Возвращает число, округленное до ближайшего целого числа

Группа функций

Числовой

Синтаксис

```
num Ceil (число)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примеры

Ceil (2.4) возвращает 3.

Ceil (3.1) возвращает 4.

Ceil (-3.1) возвращает -3.

Cos

Описание

Возвращает косинус угла

Группа функций

Числовой

Синтаксис

```
num Cos (угол)
```

Входные данные

Параметр	Описание	Тип	Обязательный
угол	Угол в радианах	Число	Да

Примеры

`Cos(180)` возвращает -0.6.

EuroConvertFrom

Описание

Конвертирует сумму в евро в другую валюту

Группа функций

Числовой

Синтаксис

```
число EuroConvertFrom(euro_amount; curr_code; round_level)
```

Ввод

Параметр	Описание	Тип	Обязательный
euro_amount	Сумма в евро	Число	Да
curr_code	Код ISO для целевой валюты	Строка	Да
round_level	Количество десятичных разрядов, до которых округляется результат	Число	Да

Примечания

Код валюты должен соответствовать коду одной из 12 валют ЕС, курс которых был зафиксирован по отношению к евро перед их отменой в

январе 2002 года. Если это не выполняется, функция возвращает ошибку: #ERROR. Валюты:

BEF	Бельгийский франк
DEM	Немецкая марка
GRD	Греческая драхма
ESP	Испанская песета
FRF	Французский франк
IEP	Ирландский фунт
ITL	Итальянская лира
LUF	Люксембургский франк
NLG	Голландский гульден
ATS	Австрийский шиллинг
PTS	Португальский эскудо
FIM	Финская марка

Примеры

`EuroConvertFrom(1000;"FRF";2)` возвращает 6559,57.

`EuroConvertFrom(1000;"FRF";1)` возвращает 6559,60.

`EuroConvertFrom(1000.04;"DEM";2)` возвращает 1955,83.

`EuroConvertFrom(1000.04;"DEM";1)` возвращает 1955,80.

См. также

- [Округление и обрезание чисел с помощью Web Intelligence](#)

EuroConvertTo

Описание

Преобразует сумму в евро

Группа функций

Числовой

Синтаксис

```
num EuroConvertTo (noneuro_amount; curr_code; round_level)
```

Входные данные

Параметр	Описание	Тип	Обязатель- ный
noneuro_amount	Сумма в валюте, отличной от евро	Число	Да
curr_code	Код ISO валюты, отличной от евро	Строка	Да
round_level	Количество десятичных разрядов, до которых округляется результат	Число	Да

Пример

EuroConvertTo (6559; "FRF"; 2) возвращает 999,91.

EuroConvertTo (6559; "FRF"; 1) возвращает 999,90.

EuroConvertTo (1955; "DEM"; 2) возвращает 999,58.

EuroConvertTo (1955; "DEM"; 1) возвращает 999,60.

Примечание

Код валюты должен соответствовать коду одной из 12 валют ЕС, курс которых был зафиксирован по отношению к евро перед их отменой в январе 2002 года. Если это не выполняется, функция возвращает ошибку: #ERROR. Валюты:

BEF	Бельгийский франк
DEM	Немецкая марка
GRD	Греческая драхма
ESP	Испанская песета
FRF	Французский франк
IEP	Ирландский фунт
ITL	Итальянская лира
LUF	Люксембургский франк
NLG	Голландский гульден
ATS	Австрийский шиллинг
PTS	Португальский эскудо
FIM	Финская марка

См. также

- [Округление и обрезание чисел с помощью Web Intelligence](#)

EuroFromRoundError

Описание

Возвращает ошибку округления при преобразовании из евро

Группа функций

Числовой

Синтаксис

```
num EuroFromRoundError(euro_amount;curr_code;round_level)
```

Входные данные

Параметр	Описание	Тип	Обязатель- ный
euro_amount	Сумма в евро	Число	Да
curr_code	Код ISO для целевой валюты	Строка	Да
round_level	Количество десятичных разрядов, до которых округляется результат	Число	Да

Вывод

Ошибка округления при вычислениях

Примеры

`EuroFromRoundErr(1000;"FRF";2)` возвращает 0. (Нет разницы между неокругленным преобразованием и преобразованием, округленным до 2 десятичных разрядов.)

`EuroFromRoundErr(1000;"FRF";1)` возвращает 0,03. (Неокругленное преобразование – 6559,57. Преобразование, округленное до 1 десятичного разряда, равно 6559,60. Ошибка округления составляет 0,03.)

`EuroFromRoundErr(1000;"DEM";2)` возвращает 0. (Нет разницы между неокругленным преобразованием и преобразованием, округленным до 2 десятичных разрядов.)

`EuroFromRoundErr(1000;"DEM";1)` возвращает -0,01. (Неокругленное преобразование – 1955,83. Преобразование, округленное до 1 десятичного разряда, 1995,80. Ошибка округления составляет -0,03.)

Примечание

Код валюты должен соответствовать коду одной из 12 валют ЕС, курс которых был зафиксирован по отношению к евро перед их отменой в

январе 2002 года. Если это не выполняется, функция возвращает ошибку: #ERROR. Валюты:

BEF	Бельгийский франк
DEM	Немецкая марка
GRD	Греческая драхма
ESP	Испанская песета
FRF	Французский франк
IEP	Ирландский фунт
ITL	Итальянская лира
LUF	Люксембургский франк
NLG	Голландский гульден
ATS	Австрийский шиллинг
PTS	Португальский эскудо
FIM	Финская марка

См. также

- [Округление и обрезание чисел с помощью Web Intelligence](#)

EuroToRoundError

Описание

Возвращает ошибку округления при конвертации в евро

Группа функций

Числовой

Синтаксис

число

```
EuroToRoundError (noneuro_amount; curr_code; round_level)
```

Данные ввода

Параметр	Описание	Тип	Обязатель- ный
noneuro_amount	Сумма в валюте, отличной от евро	Число	Да
curr_code	Код ISO валюты, отличной от евро	Строка	Да
round_level	Количество десятичных разрядов, до которых округляется результат	Число	Да

Примеры

`EuroToRoundErr (6559; "FRF"; 2)` возвращает 0. (Нет разницы между неокругленной конвертацией и конвертацией, округленной до 2 десятичных знаков.)

`EuroToRoundErr (6559; "FRF"; 1)` возвращает -0.01. (Неокругленная конвертация – 999,91. Конвертация, округленная до 1 десятичного знака – 999,90. Ошибка округления – -0,01).

`EuroToRoundErr (1955; "DEM"; 2)` возвращает 0. (Нет разницы между неокругленной конвертацией и конвертацией, округленной до 2 десятичных знаков.)

`EuroToRoundErr (1955; "DEM"; 1)` возвращает 0,02. (Неокругленная конвертация – 999,58. Конвертация, округленная до 1 десятичного знака – 999,60. Ошибка округления – 0,02).

Примечание

Код валюты должен соответствовать коду одной из 12 валют ЕС, курс которых был зафиксирован по отношению к евро перед их отменой в январе 2002 года. Если это не выполняется, функция возвращает ошибку: #ERROR. Валюты:

BEF	Бельгийский франк
-----	-------------------

DEM	Немецкая марка
GRD	Греческая драхма
ESP	Испанская песета
FRF	Французский франк
IEP	Ирландский фунт
ITL	Итальянская лира
LUF	Люксембургский франк
NLG	Голландский гульден
ATS	Австрийский шиллинг
PTS	Португальский эскудо
FIM	Финская марка

См. также

- [Округление и обрезание чисел с помощью Web Intelligence](#)

Exp

Описание

Возвращает значение экспоненциальной функции (число e , возведенное в указанную степень)

Группа функций

Числовой

Синтаксис

```
num Exp(степень)
```

Входные данные

Параметр	Описание	Тип	Обязательный
степень	Степень	Число	Да

Примечания

Экспоненциальная функция вычисляется путем возведения константы e (2,718...) в степень.

Примеры

$\text{Exp}(2, 2)$ возвращает 9,03.

Fact

Описание

Возвращает факториал числа

Группа функций

Числовой

Синтаксис

```
int Fact(число)
```

Входные данные

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примечания

Факториал числа — это произведение всех целых чисел от 1 до этого числа.

Примеры

Fact (4) возвращает 24.

Fact (5, 9) возвращает 120.

Floor

Описание

Возвращает число, округленное до ближайшего целого числа

Группа функций

Числовой

Синтаксис

```
целое число Floor (число)
```

Ввод

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Пример

Floor (24.4) возвращает 24.

Ln

Описание

Возвращает натуральный логарифм числа

Группа функций

Числовой

Синтаксис

```
num Ln (число)
```

Входные данные

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примеры

Функция Ln (10) возвращает 2.

Log

Описание

Возвращает логарифм числа по указанному основанию

Группа функций

Числовой

Синтаксис

```
num Log (number; base)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да
основание	Основание логарифма	Число	Да

Примеры

Log (125; 5) возвращает 3.

Log10

Описание

Возвращает логарифм по основанию 10 для числа

Группа функций

Числовой

Синтаксис

```
num Log10 (number)
```

Ввод

введенное_число	Число
-----------------	-------

Примеры

Log10 (100) возвращает 2.

Mod

Описание

Возвращает остаток от деления двух чисел

Группа функций

Числовой

Синтаксис

```
num Mod (dividend;divisor)
```

Ввод

Параметр	Описание	Тип	Обязательный
делимое	Делимое	Число	Да
делитель	Делитель	Число	Да

Примеры

`Mod (10; 4)` возвращает 2.

`Mod (10, 2; 4, 2)` возвращает 1,8

Power

Описание

Возвращает число, возведенное в степень

Группа функций

Числовой

Синтаксис

```
num Power (число; степень)
```

Входные данные

Параметр	Описание	Тип	Обязательный
число	Число, возводимое в степень	Число	Да
степень	Степень	Число	Да

Пример

Формула `Power (10; 2)` возвращает 100.

Rank

Описание

Ранжирует меру по измерениям

Группа функций

Числовой

Синтаксис

```
int
Rank (measure; [ranking_dims] [; Top|Bottom] [; reset_dims])
```

Данные ввода

Параметр	Описание	Тип	Обязателен
мера	Мера для ранжирования	Мера	Да
ranking_dims	Измерения, используемые для ранжирования меры	Список измерений	Нет
Первые/последние	Устанавливает порядок ранжирования: <ul style="list-style-type: none"> Top – по убыванию Bottom – по возрастанию 	Ключевое слово	Нет (Top по умолчанию)
reset_dims	Измерения, которые сбрасывают ранжирование	Список измерений	Нет

Примечания

- Если не указаны измерения для ранжирования, Web Intelligence использует контекст вычисления по умолчанию, чтобы вычислить ранжирование.

- Измерения необходимо всегда помещать в круглые скобки, даже если в списке ранжирования или измерений сброса присутствует только одно измерение.
- Указываемые наборы ранжирования или измерений сброса должны разделяться точкой с запятой.
- По умолчанию ранжирование сбрасывается при переходе к разделу или разрыву блока.

Примеры

В следующей таблице ранжирование задается функцией `Rank ([Доход]; ([Страна]))`:

	Доход	Ранжирование
Франция	835 420	2
США	2 451 104	1

В следующей таблице ранжирование задается функцией `Rank ([Доход]; ([Страна]); Bottom)`. Аргумент `Bottom` означает, что ранжирование мер происходит в порядке убывания.

Страна	Доход	Ранжирование
Франция	835 420	1
США	2 451 104	2

В следующей таблице ранжирование задается функцией `Rank ([Доход]; ([Страна]; [Курорт]))`:

Страна	Курорт	Доход	Ранжирование
Франция	French Riviera	835 420	3
США	Bahamas Beach	971 444	2
США	Hawaiian Club	1 479 660	1

В следующей таблице ранжирование задается функцией `Rank ([Доход]; ([Страна]; [Год]); ([Страна]))`. Ранжирование сбрасывается на измерении страны.

Страна	Год	Доход	Ранжирование
Франция	FY1998	295 940	1
Франция	FY1999	280 310	2
Франция	FY2000	259 170	3
США	FY1998	767 614	3
США	FY1999	826 930	2
США	FY2000	856 560	1

См. также

- *Операторы Bottom/Top*

Round

Описание

Округляет число

Группа функций

Числовой

Синтаксис

```
num Round (number;round_level)
```

Ввод

Параметр	Описание	Тип	Обязатель- ный
число	Округляемое число	Число	Да
round_level	Количество десятичных знаков, до которого округляется число	Число	Да

Примеры

Round (9.44; 1) возвращает 9.4.

Round (9.45; 1) возвращает 9.5.

Round (9.45; 0) возвращает 9.

Round (9.45; -1) возвращает 10.

Round (4.45; -1) возвращает 0.

См. также

- [Округление и обрезание чисел с помощью Web Intelligence](#)

Sign

Описание

Возвращает знак числа

Группа функций

Числовой

Синтаксис

```
int Sign (number)
```

Ввод

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Примечания

Знак возвращает 1, если число является отрицательным, 0 если число равно нулю, 1 если число является положительным.

Примеры

Знак (3) возвращает 1.

Знак (27.5) возвращает 1.

Sin

Описание

Возвращает синус угла

Группа функций

Числовой

Синтаксис

```
num Sin (angle)
```

Ввод

Параметр	Описание	Тип	Обязательный
угол	Угол в радианах	Число	Да

Пример

Sin (234542) возвращает -0,116992.

Sqrt

Описание

Возвращает квадратный корень из числа

Группа функций

Числовой

Синтаксис

```
num Sqrt (number)
```

Ввод

Параметр	Описание	Тип	Обязательный
число	Любое число	Число	Да

Например

`Sqrt (25)` возвращает 5.

Tan

Описание

Возвращает тангенс угла

Группа функций

Числовой

Синтаксис

```
num Tan (angle)
```

Ввод

Параметр	Описание	Тип	Обязательный
угол	Угол в радианах	Число	Да

Примеры

`Tan (90)` возвращает -2.

ToNumber

Описание

Преобразует строку в число

Группа функций

Числовой

Синтаксис

```
num ToNumber(string)
```

Ввод

Параметр	Описание	Тип	Обязательный
строка	Число в качестве строки	Строка	Да

Примечания

Если строка не является числом, функция `ToNumber` возвращает `#ERROR`.

Примеры

`ToNumber ("45")` возвращает 45.

Truncate

Описание

Обрезает число

Группа функций

Числовой

Синтаксис

```
num Truncate (number; truncate_level)
```

Ввод

Параметр	Описание	Тип	Обязательный
число	Округляемое число	Число	Да
truncate_level	Количество знаков после запятой, до которого обрезается число	Число	Да

Примечания

Например

Truncate (3,423; 2) возвращает 3,42.

См. также

- [Округление и обрезание чисел с помощью Web Intelligence](#)

Дополнительные

BlockName

Описание

Возвращает имя блока

Группа функций

Разное

Синтаксис

```
строка BlockName()
```

Примеры

BlockName() возвращает "Блок 1", если он помещен в блок под названием "Блок 1".

ColumnNumber

Описание

Возвращает порядковый номер столбца

Группа функций

Разное

Синтаксис

```
int ColumnNumber()
```

Примеры

`ColumnNumber()` возвращает 2, если данная формула помещена во второй столбец таблицы.

CurrentUser

Описание

Возвращает параметры входа в InfoView для текущего пользователя

Группа функций

Разное

Синтаксис

```
строка CurrentUser()
```

Примеры

`CurrentUser()` возвращает "gkn", если текущее регистрационное имя пользователя InfoView – "gkn".

ForceMerge

Описание

Включает синхронизированные измерения в вычисления мер, если эти измерения не содержатся в контексте вычисления меры

Группа функций

Разное

Синтаксис

```
num ForceMerge(measure)
```


Данные ввода

Параметр	Описание	Тип	Обязателен
показатель	Любой показатель	Показатель	Да

Вывод

Результат вычисления с учетом синхронизированных измерений

Примечания

- ForceMerge возвращает значение #НЕОДНОЗНАЧНО, если функция применяется к интеллектуальной мере, поскольку набора для группировки, необходимого для вычисления интеллектуальной меры, не существует.
- Функция ForceMerge в приложении Web Intelligence аналогична функции Multicube приложения BusinessObjects/Desktop Intelligence.

Примеры

ForceMerge ([Доход]) возвращает значение объекта [Доход] с учетом всех синхронизированных измерений, которые не отображаются в одном блоке с мерой [Доход].

GetContentLocale

Описание

Возвращает языковой стандарт данных, содержащихся в документе (языковой стандарт документа)

Группа функций

Разное

Синтаксис

```
строка GetContentLocale()
```

Примечания

Языковой стандарт используется для форматирования данных в документе Web Intelligence.

Примеры

Функция `GetContentLocale()` возвращает "fr_FR", если в качестве языкового стандарта документа установлен "Французский (Франция)".

GetDominantPreferredViewingLocale

Описание

Возвращает основной языковой стандарт в группе предпочтительного языкового стандарта для просмотра, выбранного пользователем

Группа функций

Разное

Синтаксис

```
string GetDominantPreferredViewingLocale()
```

Примечания

- Каждая группа связанных языковых стандартов имеет основной языковой стандарт, используемый в качестве базового для всех остальных языковых стандартов в данной группе. Например, английский (США) ("en_US") является основным языковым стандартом в группе стандартов английского языка. Английский (Новая Зеландия) также входит в эту группу.
- В *Руководстве к диспетчеру переводов* перечислены все основные предпочтительные языковые стандарты для просмотра.

Примеры

Функция `GetDominantPreferredViewingLocale` возвращает "en_US", если в качестве предпочтительного языкового стандарта для просмотра установлен "Английский (Новая Зеландия)".

См. также

- [GetPreferredViewingLocale](#)

GetLocale

Описание

Возвращает пользовательский языковой стандарт, применяемый для форматирования интерфейса Web Intelligence (языковой стандарт продукта)

Группа функций

Разное

Синтаксис

```
строка GetLocale ()
```

Примечания

Языковой стандарт продукта – это языковой стандарт, используемый для отображения пользовательского интерфейса Web Intelligence (например, элементов меню и текста кнопок).

Примеры

Функция `GetLocale ()` возвращает "en_US", если в качестве пользовательского языкового стандарта продукта установлен "Английский (США)".

GetLocalized

Описание

Возвращает локализованную строку в соответствии с предпочтительным языковым стандартом для просмотра, выбранным пользователем.

Синтаксис

```
string GetLocalized(string[,comment])
```

Данные ввода

Параметр	Описание	Тип	Обязательный
строка	Строка для перевода	строка	Да
comment	Комментарий, добавляемый переводчиками	строка	Нет

Примечания

- Параметр `string` может быть строкой в любой формуле Web Intelligence (например, в ячейке, сообщении сигнализатора или определении переменной).
- При создании отчета можно использовать параметр `comment` для ввода дополнительной информации, помогающей переводчикам перевести строку. Этот комментарий появляется вместе со строкой в инструменте "Диспетчер переводов", используемом переводчиками для перевода отчетов Web Intelligence.
- Каждая пара строка + комментарий создает отдельную строку для перевода в инструменте "Диспетчер переводов". В результате `GetLocalized("Итого по продукту"; "Максимум 20 символов")` и `GetLocalized("Итого по продукту"; "Использовать не более 20 символов")` могут возвращать различные переводы.

Примеры

Функция `GetLocalized("Итого для всех продуктов")` возвращает перевод на французский язык строки "Итого для всех продуктов", если в качестве предпочтительного языкового стандарта для просмотра установлен "fr_FR".

Функция `GetLocalized("Итого для всех продуктов", "Старайтесь не использовать больше 20 символов")` возвращает перевод на немецкий язык строки "Итого для всех продуктов", если в качестве предпочтительного языкового стандарта для просмотра установлен "de_DE". Эта функция также сообщает переводчику отчета, что не

рекомендуется использовать больше 20 символов при переводе этой строки.

См. также

- [GetPreferredViewingLocale](#)

GetPreferredViewingLocale

Описание

Возвращает пользовательский предпочтительный языковой стандарт для просмотра данных в документе (предпочтительный языковой стандарт для просмотра)

Группа функций

Разное

Синтаксис

```
string GetPreferredViewingLocale()
```

Примеры

Функция `GetPreferredViewingLocale` возвращает "en_US", если в качестве предпочтительного языкового стандарта для просмотра установлен "Английский (США)".

См. также

- [GetLocalized](#)
- [GetDominantPreferredViewingLocale](#)

If...Then...Else

Описание

Возвращает значение в зависимости от того, является ли значение выражения истиной или ложью

Группа функций

Разное

Синтаксис

```
If bool_value Then true_value [Else false_value]
```

Данные ввода

Параметр	Описание	Тип	Требуемое
bool_value	Логическое значение	Логическое значение	Да
true_value	Возвращаемое значение, если bool_value имеет значение True	Любой	Да
false_value	Возвращаемое значение, если bool_value имеет значение False	Любой	Да, если Else включено

Примечания

- true_value и false_value могут смешивать типы данных.
- Можно использовать операторы логических выражений And, Between, InList, Or и Not с функцией If.
- Также можно использовать вложенные условия If, задействовав конструкцию ElseIf. Синтаксическая конструкция:

```
If test_value Then true_value [Else false_value|ElseIf  
test_value Then true_value [Else  
false_value...]]
```

- Web Intelligence также поддерживает синтаксическую конструкцию If(bool_value;true_value;false_value).

Примеры

If [Доход с продаж]>1000000 Then "Высокий доход" возвращает "Высокий доход" только для тех рядов, где значение дохода превышает 1 000 000.

If [Доход с продаж] >1000000 Then "Высокий доход" Else [Доход] возвращает "Высокий доход" для всех рядов, где значение дохода превышает 1 000 000, для всех остальных рядов просто выводится значение дохода.

If [Доход с продаж]>1000000 Then "Высокий доход" Else "Низкий доход" возвращает "Высокий доход" для всех рядов, где значение дохода превышает 1 000 000 и "Низкий доход" для рядов, где значение дохода не достигает 1 000 000.

If [Доход с продаж]>1000000 Then "Высокий доход" ElseIf [Доход с продаж] > 800000 Then "Средний доход" Else "Низкий доход" возвращает "Высокий доход" для рядов, где значение дохода превышает 1 000 000, "Средний доход" для рядов, где значение дохода находится в пределах 800 000 и 1 000 000 и "Низкий доход" во всех остальных случаях.

См. также

- *If*
- *Оператор ""и""*
- *Оператор Between*
- *Оператор Inlist*
- *Оператор ""или""*
- *Оператор отрицания*

If

Описание

Возвращает значение в зависимости от того, является ли значение выражения истиной или ложью

Группа функций

Разное

Синтаксис

```
If (bool_value; true_value; false_value)
```

Данные ввода

Параметр	Описание	Тип	Требуемое
bool_value	Логическое значение	Логическое значение	Да
true_value	Возвращаемое значение, если bool_value имеет значение True	Любой	Да
false_value	Возвращаемое значение, если bool_value имеет значение False	Любой	Да

Примечания

- В параметрах true_value и false_value можно смешивать типы данных
- Условия If можно вкладывать в другие условия путем замены параметра false_value дополнительными условиями If.

```
If(bool_value;true_value;If(bool_value;true_value;false_value|If...))
```

- В приложении Web Intelligence также поддерживается синтаксическая конструкция If...Then...Else.

Примеры

Функция If([Доход с продаж]>1000000;"Высокий доход";"Низкий доход") возвращает "Высокий доход" для всех строк, где значение дохода превышает 1 000 000, и "Низкий доход" для тех строк, где значение дохода не достигает 1 000 000.

Функция If([Доход с продаж]>1000000;"Высокий доход";[Доход]) возвращает "Высокий доход" для всех строк, где значение дохода превышает 1 000 000, для всех остальных строк просто выводится значение дохода.

См. также

- [If...Then...Else](#)

LineNumber

Описание

Возвращает номер строки в таблице

Группа функций

Разное

Синтаксис

```
int LineNumber()
```

Примечания

Нумерация строк таблицы начинается с заголовка, который является строкой за номером 1.

Примеры

LineNumber() возвращает 2, если функция отображается на второй строке в таблице.

NameOf

Описание

Возвращает имя объекта

Группа функций

Разное

Синтаксис

```
string NameOf(объект)
```

Входные данные

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примечания

В приложении Web Intelligence функция `NameOf` используется в заголовках столбцов и строк в отчетах.

Примеры

Функция `NameOf ([Дата бронирования])` возвращает "Дата бронирования".

NoFilter

Описание

Игнорирует фильтры при вычислении значения

Группа функций

Разное

Синтаксис

```
input_type NoFilter(obj[;All|Drill])
```

Данные ввода

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да
All Drill	<ul style="list-style-type: none"> Нет указанного ключевого слова – игнорировать отчет и блокировать фильтры All – игнорировать все фильтры Drill – игнорировать фильтры отчета и фильтры детализации 	Ключевое слово	Нет

Примечания

- `NoFilter(obj;Drill)` не работает в режиме детализации запроса, т. к. фильтры детализации добавляются в запрос, но не применяются для фильтра данных отчета.
- Если работа в режиме детализации завершена, а фильтры детализации остались активны, они становятся фильтрами отчета и могут изменять значения любых выражений, на которые распространяется действие функции `NoFilter(obj;Drill)`.

Примеры

Если поместить функцию `NoFilter(Sum([Доходы с продаж]))` в нижний колонтитул блока, она возвращает суммарный объем доходов с продаж по всем рядам блока, даже если применяемый фильтр исключает некоторые ряды из блока.

Функция `NoFilter(Sum([Доходы с продаж]);All)` возвращает общий объем доходов с продаж для всех стран, включая Францию, несмотря на то, что применяемый фильтр исключает Францию из отчета.

Функция `NoFilter(Sum([Доходы с продаж]);Drill)` возвращает общий объем доходов с продаж для всех стран, игнорируя при этом фильтр детализации по размеру страны.

NumberOfPages

Описание

Возвращает количество страниц в отчете

Группа функций

Разное

Синтаксис

```
integer NumberOfPages ()
```

Примеры

Примеры `NumberOfDataPages ()` возвращает 2, если в отчете содержатся две страницы.

Page

Описание

Возвращает номер текущей страницы в отчете.

Группа функций

Разное

Синтаксис

```
целое Page ()
```

Например

`Page ()` возвращает 2, если отображается на второй странице отчета.

Previous

Описание

Возвращает предыдущее значение объекта

Группа функций

Разное

Синтаксис

```
input_type  
Previous (dimension|measure|Self[;reset_dims][;offset][;NotNull])
```

Данные ввода

Параметр	Описание	Тип	Обязательный
измерение measure Self	Измерение или мера, чье предыдущее значение возвращает функция, или ключевое слово Self	Измерение, мера или ключевое слово	Да
reset_dims	Список измерений, используемых для сброса вычислений	Список измерений	Нет
смещение	Указывает значение измерения или меры, то есть строки со смещением, предшествующие текущей строке	Целое	Нет (по умолчанию 1)
NotNull	Указывает функции на то, что необходимо возвращать первое ненулевое значение, начиная со смещения	Ключевое слово	Нет

Примечания

- По умолчанию смещение равно 1. Previous ([Revenue]; 1) и Previous ([Revenue]) функционально одно и то же.

- В случае включения аргумента `NoNull`, Web Intelligence возвращает первое ненулевое значение объекта, начиная с ячейки строк смещения, предшествующих текущей строке и отсчитанных в обратном направлении.
- С функцией `Previous` (Предыдущее) можно использовать контекстные операторы расширенного синтаксиса
- Оператор `SELF` позволяет обращаться к предыдущему значению ячейки, если ее содержимое не является объектом отчета.
- Измерения нужно всегда помещать в круглые скобки, даже если в списке измерений сброса присутствует только одно измерение.
- Измерения сброса в наборе должны разделяться двоеточиями.
- Web Intelligence применяет функцию `Previous` после применения всех фильтров отчетов, разделов и блоков.
- Нельзя применять фильтры к формуле, в которой используется функция `Previous`.
- Web Intelligence применяет функцию `Previous` после применения всех сортировок.
- Нельзя применять сортировку к формуле, в которой используется функция `Previous`.
- Если функция `Previous` применяется к мере, а мера возвращает неопределенное значение, функция `Previous` возвращает неопределенное значение, даже если в предыдущей строке возвращается значение.
- При помещении за пределами верхнего или нижнего колонтитула разрыва функция `Previous` игнорирует разрывы.
- Если функция `Previous` помещена в нижний колонтитул разрыва, то она возвращает значение из предыдущего экземпляра нижнего колонтитула.
- Web Intelligence сбрасывает `Previous` в каждом разделе отчета.
- При использовании в кросс-таблице, функция `Previous` не обрабатывает последнее значение в строке как предыдущее значение для первого значения в следующей строке.

Примеры

`Previous([Country];1)` возвращает следующие значения в следующей таблице:

Страна	Доход	Предыдущий
--------	-------	------------

США	5 000 000	
Великобритания	2 000 000	США
Франция	2 100 000	Великобритания

`Previous ([Revenue])` возвращает следующие значения в следующей таблице:

Страна	Доход	Предыдущий
США	5 000 000	
Великобритания	2 000 000	5 000 000
Франция	2 100 000	2 000 000

`Previous ([Revenue]; ([Country])` возвращает следующие значения в следующей таблице:

Страна	Регион	Доход	Предыдущий
США	Север	5 000 000	
	Юг	7 000 000	5 000 000
Великобритания	Север	3 000 000	
	Юг	4 000 000	3 000 000

`Previous ([Revenue])` возвращает следующие значения в следующей кросс-таблице:

	2004	Предыдущий	2005	Предыдущий
США	5 000 000		6 000 000	5 000 000

Великобритания	2 000 000		2 500 000	2 000 000
Франция	3 000 000		2 000 000	3 000 000

`Previous ([Revenue])` возвращает следующие значения в следующей таблице с разделением по [Странам]:

Страна	Регион	Доход	Предыдущий
США	Север	5 000 000	
	Юг	7 000 000	5 000 000
США		12 000 000	

Страна	Регион	Доход	Предыдущий
Великобритания	Север	3 000 000	7 000 000
	Юг	4 000 000	3 000 000
Великобритания		7 000 000	12 000 000

`Previous ([Revenue]) ; 2 ; NoNull)` возвращает следующие значения в следующей таблице:

Год	Квартал	Доход	Предыдущий
2008	K1	500	
2008	K2		
2008	K3	400	500
2008	K4	700	500

2008	K1	300	400
2008	K2		700
2008	K3		300
2008	K4	200	300

`2*Previous(Self)` возвращает последовательность 2, 4, 6, 8, 10...

См. также

- *Сравнение значений с помощью функции Previous*
- *Оператор Self*

RefValue

Описание

Возвращает опорное значение для объекта отчета, когда включено отслеживание данных

Группа функций

Разное

Синтаксис

```
input_type RefValue(obj)
```

Примеры

`RefValue([Самый эффективный регион])` возвращает "Юго-запад", если значение переменной [Самый эффективный регион] в опорных данных равно "Юго-запад".

`RefValue([Доход])` возвращает 1000, если значение меры [Доход] в опорных данных равно 1000.

RelativeValue

Описание

Возвращает предыдущие или последующие значения объекта

Группа функций

Разное

Синтаксис

```
input_type
RelativeValue (measure|detail;slicing_dims;offset)
```

Данные ввода

Параметр	Описание	Тип	Обязательный
measure detail	Любая мера или деталь измерения в блоке	Мера или деталь	Да
slicing_dims	Измерения, которые служат контекстом вычислений	Список измерений	Да
смещение	Указывает значение меры или детали, которая является строкой смещения, удаленной из текущей строки	Целое	Да

Примечания

- Объект должен быть мерой или деталью измерения, доступной в блоке.
- Порядок сортировки списка значений измерений срезов используется для определения выходных данных функции.

Порядок сортировки определяется двумя факторами: сортировки, которые применяются для измерений срезов, и порядок, в котором измерения срезов перечислены в функции.

- Измерение, которое используется как основной элемент раздела может быть указано как измерение срезов.
- Все измерения срезов должны быть представлены в блоке или в заголовке раздела блока, где находится функция. Если измерение срезов затем удаляется из блока, функция возвращает ошибку #COMPUTATION.
- Если смещение превышает количество строк в списке значений измерения срезов, функция возвращает нулевое значение.
- RelativeValue невозможно использовать рекурсивно.
- Необходимо всегда заключать измерения в круглые скобки, даже если в списке измерений срезов есть только одно измерение.

Примеры

Столбец RelativeValue в таблице ниже содержит следующую формулу.

```
RelativeValue ([Revenue]; ([Year]); -1)
```

Год	Квартал	Продавец	Доход	RelativeValue
2007	K1	Смит	1000	
2007	K2	Джонс	2000	
2007	K3	Уилсон	1500	
2007	K4	Харрис	3000	
2008	K1	Смит	4000	1000
2008	K2	Джонс	3400	2000
2008	K3	Уилсон	2000	1500
2008	K4	Харрис	1700	3000

Для получения подробной информации о RelativeValue см. главу "Сравнение значений с помощью функций Web Intelligence" руководства *Использование функций, формул и вычислений в SAP BusinessObjects Web Intelligence* или см. ссылку в конце этого раздела.

См. также

- [#COMPUTATION](#)
- [Сравнение значений с помощью функции RelativeValue](#)

ReportName

Описание

Возвращает имя отчета

Группа функций

Разное

Синтаксис

```
строка ReportName ()
```

Примеры

Функция `ReportName ()` возвращает значение "Отчет о продажах", если находится в отчете под названием "Отчет о продажах".

RowIndex

Описание

Возвращает номер строки

Группа функций

Разное

Синтаксис

```
integer RowIndex ()
```

Примечания

- Нумерация строк начинается с 0.
- Функция `RowIndex` возвращает значение `#MULTIVALUE` при ее размещении в верхнем или нижнем колонтитуле таблицы.

Примеры

Функция `RowIndex` возвращает 0, если расположена в первой строке таблицы.

UniqueNameOf

Описание

Возвращает уникальное имя объекта

Группа функций

Разное

Синтаксис

строка `UniqueNameOf (obj)`

Ввод

Параметр	Описание	Тип	Обязательный
объект	Любой объект отчета	Объект отчета	Да

Примеры

`UniqueNameOf ([Дата сохранения])` возвращает "Дата сохранения".

Операторы функций и формул Web Intelligence

Операторы связывают различные элементы формул. Формулы могут содержать математические, условные, логические, специальные функциональные операторы или операторы расширенного синтаксиса.

Математические операторы

Математические операторы знакомы по арифметическим вычислениям, выполняемым ежедневно. Это операторы сложения (+), вычитания (-), умножения (*), деления (/), которые позволяют выполнять математические операции в формуле. В формуле [Доход продаж] – [Издержки при продажах] содержится математический оператор вычитания.

Примечание:

При использовании с символьными строками оператор "+" ставится оператором конкатенации строк. А это значит, что он объединяет символьные строки. Например, формула "Иван" + "Иванов" возвращает значение "Иван Иванов".

Условные операторы

Условные операторы определяют тип сравнения двух значений.

Оператор	Описание
=	Равно
>	Больше
<	Меньше
>=	Больше или равно
<=	Меньше или равно
<>	Не равно

Условные операторы используются с функцией If, например:

```
If [Revenue]>10000 Then "High" Else "Low"
```

которая возвращает значение "Высокая" для всех строк, в которых указанная прибыль больше или равна 10000, и значение "Низкая" для всех остальных строк.

Логические операторы

В приложении Web Intelligence используются следующие логические операторы: And, Or, Not, Between и Inlist. Логические операторы используются в булевых выражениях, возвращающих значения Истина или Ложь.

Оператор ""и""

Описание

Оператор And связывает булевы значения. Если все булевы значения, связанные оператором And, возвращают значение True, то комбинация всех значений также будет возвращать значение True.

Синтаксис

```
bool_value And bool_value [And bool_value...]
```

Примеры

Функция If [Курорт] = "Багамский пляж" And [Доход]>100000 Then "Высокий доход с Багамских островов" возвращает значение "Высокий доход с Багамских островов", если выполнено условие [Курорт] = "Багамский пляж" And [Доход]>100000.

Оператор ""или""

Описание

Оператор Or связывает булевы значения. Если одно любое булево значение, связанное оператором Or, возвращает значение True, совокупность всех значений также возвращает значение True.

Синтаксис

```
bool_value Or bool_value [Or bool_value...]
```

Примеры

If [Курорт] = "Багамский пляж" Or [Курорт]="Гавайский клуб" Then "США" Else "Франция" возвращает "США", если [Курорт]="Багамский пляж" или "Гавайский клуб" или, в противном случае, возвращает "Франция".

Оператор отрицания

Описание

Оператор Not возвращает значение, противоположное логическому.

Синтаксис

```
bool Not(bool_value)
```

Примеры

If Not([Страна] = "США") Then "Не США" возвращает "Не США", если [Страна] имеет значение, отличное от "США".

Оператор Between

Описание

Оператор Between определяет, заключено ли значение переменной между двумя значениями.

Синтаксис

```
bool Between(first_value;second_value)
```


Примечания

- Функция `Between` используется вместе с функцией `If` и оператором `Where`.
- Поскольку языковой стандарт форматирования документа может повлиять на порядок сортировки данных, изменение языкового стандарта может повлиять на результат, возвращаемый оператором `Between`. (Языковой стандарт форматирования документов устанавливается на вкладке **Предпочтительные параметры документов Web Intelligence** в InfoView.)

Примеры

Функция `If [Доход] Between(800000;900000) Then "Средний доход"` возвращает значение "Средний доход", если `[Доход]` заключен между 800000 и 900000.

Функция `[Доход от продаж] Between (10000;20000)` возвращает значение `True`, если доход от продаж находится между 10000 и 20000.

Функция `If ([Доход от продаж] Between (200000;500000) ; "Средний доход" ; "Низкий/Высокий доход")` возвращает значение "Средний доход", если `[Доход от продаж]` составляет 300000.

См. также

- [If...Then...Else](#)
- [Оператор Where](#)

Оператор Inlist

Описание

Оператор `Inlist` определяет, принадлежит ли значение списку значений.

Синтаксис

```
bool test_value Inlist(value_list)
```

Примечания

Именно комбинация `test_value + InList` возвращает булево значение, а не только один оператор `InList`.

Примеры

`If Not ([Страна] InList("Англия"; "Шотландия"; "Уэльс"))`
`Then "Не Британия" Else "Британия"` возвращает "Не Британия", если [Страна] не совпадает с "Англия", "Шотландия" или "Уэльс" или, в противном случае, возвращает "Британия".

`If [Курорт] InList("Багамский пляж"; "Гавайский клуб")`
`Then "Курорт США"` возвращает "Курорт США", если [Курорт] совпадает с "Багамский пляж" или "Гавайский клуб".

См. также

- *If...Then...Else*
- *Оператор Where*

Специальные операторы функций

Некоторые функции Web Intelligence могут использовать определенный набор операторов в качестве аргументов. Например, функция `Previous` может использовать оператор `Self`.

Аргументы всех функций заключаются в скобки: `)` и `(`. В функциях, работающих с несколькими параметрами, для их разделения используется знак `;`.

Оператор All

Оператор `All` указывает функции `NoFilter` игнорировать все фильтры, а также указывает функции `Count` подсчитывать все значения, в том числе повторяющиеся.

См. также

- *Count*
- *Операторы Distinct/All*

- *NoFilter*
- *Операторы All/Drill*

Операторы All/Drill

Описание

С помощью операторов All/Drill можно определить, какие фильтры игнорирует функция NoFilter.

- Не задано – NoFilter игнорирует фильтры отчета и блока
- All – NoFilter игнорирует все фильтры
- Drill – NoFilter игнорирует фильтры отчета и фильтры детализации

Операторы Bottom/Top

Описание

Операторы Bottom/Top указывают функции Rank, ранжировать в порядке убывания или возрастания.

- Top – ранжирует в порядке убывания
- Bottom – ранжирует в порядке возрастания

Примеры

Rank ([Доход] ; ([Страна]) ; Top ранжирует страны по доходу от высокого до низкого.

См. также

- *Rank*

Оператор Break

Описание

Оператор `Break` указывает функции `Percentage`, что необходимо учитывать разбиения таблиц.

Примеры

Формула `Percentage ([Доход])` возвращает следующий результат в данной таблице (процентные соотношения вычисляются на основе общего значения дохода в блоке):

Год	Квартал	Доход	Процентное соотношение
2005	K1	10000	10%
2005	K2	20000	20%
2006	K1	30000	30%
2006	K2	40000	40%

Формула `Percentage ([Доход]; Break)` возвращает следующий результат в данной таблице (процентные соотношения вычисляются на основе общего значения дохода в каждой части блока):

Год	Квартал	Доход	Процентное соотношение
2005	K1	10000	33,3%
2005	K2	20000	66,6%

2006	K1	30000	42.9%
2006	K2	40000	57.1%

См. также

- *Percentage*

Операторы Distinct/All

Операторы *Distinct/All* сообщают функции *Count*, каким образом следует подсчитывать число значений: учитывать только различные или учитывать все

Примеры

`Count([Доход]; Distinct)` возвращает 3, если [Доход] имеет значения (5;5;6;4)

`Count([Доход]; All)` возвращает 4, если [Доход] имеет значения (5;5;6;4)

См. также

- *Count*

Оператор IncludeEmpty

Описание

Оператор *IncludeEmpty* указывает некоторым функциям агрегирования на необходимость включать пустые значения в вычисления.

Примеры

Формула `Average([Доход]; IncludeEmpty)` возвращает 3 при следующих значениях переменной [Доход]: 5; 3; <пусто>; 4.

См. также

- [Average](#)
- [Count](#)
- [RunningAverage](#)
- [RunningCount](#)

Оператор Index

Описание

Оператор `Index` указывает функциям `UserResponse` и `RefValueUserResponse` возвращать основной ключ базы данных ответа на запрос ввода данных.

См. также

- [UserResponse](#)
- [RefValueUserResponse](#)

Оператор Linear

Описание

С помощью оператора `Линейный` можно использовать в функции `Интерполяция` линейную регрессию с интерполяцией методом наименьших квадратов для получения отсутствующих значений мер.

При использовании линейной регрессии с интерполяцией методом наименьших квадратов отсутствующие значения вычисляются с помощью линейного выражения $f(x) = ax + b$, которое позволяет получить линию, наиболее приближенную ко всем доступным значениям меры.

См. также

- [Интерполяция](#)

Оператор NoNull

Описание

С помощью оператора `NoNull` функция `Previous` игнорирует нулевые значения.

С помощью оператора `NoNull` функция `Previous` возвращает первое ненулевое значение объекта, обнаруженного при отсчете в обратном направлении, начиная со строк смещения ячеек перед текущей строкой.

См. также

- [Previous](#)

Оператор NotOnBreak

Описание

С помощью оператора `NotOnBreak` функция `Interpolation` игнорирует разбиения разделов и блоков.

См. также

- [Интерполяция](#)

Оператор PointToPoint

Описание

С помощью оператора `PointToPoint` функция `Interpolation` использует двухточечную интерполяцию для получения отсутствующих значений мер.

При двухточечной интерполяции отсутствующие значения получаются путем вычисления значений линейной функции $f(x) = ax + b$, проходящей через две соседние с отсутствующими значениями точки.

См. также

- [Интерполяция](#)

Операторы Row/Col

Описание

Операторы Row/Col устанавливают направление вычисления следующих функций: Percentage, RunningAverage, RunningCount, RunningMax, RunningMin, RunningProduct, RunningSum.

Примечания

При использовании оператора ROW Web Intelligence вычисляет каждое значение в строке в виде процента от общего значения всех строк во внедренном контексте. При использовании оператора COL Web Intelligence вычисляет каждое значение в столбце в виде процента от общего значения всех столбцов во внедренном контексте.

В кросс-таблице Web Intelligence по умолчанию вычисляет значение в каждой ячейке в виде процента от суммарного значения в кросс-таблице. При использовании оператора ROW Web Intelligence вычисляет значения в строках в виде процента от суммарного значения для строки. При использовании оператора COL Web Intelligence вычисляет значения в столбцах в виде процента от суммарного значения для столбца.

Примеры

В кросс-таблице Percentage ([Мера]) дает следующий результат:

Мера	Процентное соотношение	Мера	Процентное соотношение
100	10%	500	50%
200	20%	200	20%

Percentage ([Мера] ; ROW) дает следующий результат:

Мера	Процентное соотношение	Мера	Процентное соотношение
100	16,7%	500	83,3%
200	50%	200	50%

Percentage ([Мера] ; COL) дает следующий результат:

Мера	Процентное соотношение	Мера	Процентное соотношение
100	33,3%	500	83,3%
200	66,6%	200	16,7%

При использовании оператора ROW (по умолчанию) Web Intelligence вычисляет суммарный промежуточный результат по строкам. При использовании оператора COL Web Intelligence вычисляет суммарный промежуточный результат по столбцам.

В кросс-таблице функции RunningSum ([Measure]) и RunningSum ([Measure] ; Row) дают следующие результаты:

Мера	RunningSum	Мера	RunningSum
100	100	200	300
400	700	250	950

В кросс-таблице Percentage ([Мера]) дает следующий результат:

Мера	RunningSum	Мера	RunningSum
100	100	200	700
400	500	250	950

См. также

- [Percentage](#)
- [RunningAverage](#)
- [RunningCount](#)
- [RunningMax](#)
- [RunningMin](#)
- [RunningProduct](#)
- [RunningSum](#)

Оператор Self

Описание

Указывает функции Previous на предыдущую ячейку, если она не содержит объект отчета.

Примеры

$5 + \text{Previous}(\text{Self})$ возвращает последовательность 5, 10, 15, 20, 25, 30...

$1 + 0.5 * \text{Previous}(\text{Self})$ возвращает последовательность 1, 1.5, 1.75, 1.88...

См. также

- [Previous](#)

Оператор Where

Описание

Оператор Where ограничивает данные, используемые для вычисления меры.

Примеры

Формулой `Average ([Доход с продаж]) Where ([Страна] = "США")` вычисляется средний доход с продаж для страны "США".

Формулой `Average ([Доход с продаж]) Where ([Страна] = "США" Or [Страна] = "Франция")` вычисляется средний доход с продаж для страны "США" или "Франция".

Формулой `[Доход] Where (Not ([Страна] Inlist ("США"; "Франция")))` вычисляется доход для стран, отличных от "США" и "Франция".

Переменная [Высокий доход] описана формулой `[Доход] Where [Доход > 500000]`. При размещении в блок переменной [Высокий доход] отображаются только значения дохода, превышающие 500000. При помещении в нижний колонтитул в нижней части столбца [Высокий доход], формула `Average ([Высокий доход])` возвращает среднее значение всех доходов, превышающих 500000.

См. также

- Оператор `"и"`
- Оператор *Between*
- Оператор *Inlist*
- Оператор `"или"`
- Оператор отрицания

Операторы расширенного синтаксиса

Контексты ввода и вывода задаются явным образом при помощи операторов контекста. В следующей таблице перечислены операторы контекста:

Оператор	Описание
In	Определяет подробный список изменений для использования в контексте.

Оператор	Описание
ForEach	Добавляет измерения в контекст по умолчанию
ForAll	Убирает измерения из контекста по умолчанию

Операторы ForAll и ForEach могут оказаться полезными, когда в контексте по умолчанию содержится множество измерений. Часто может быть проще добавлять или удалять измерения из контекста с помощью операторов ForAll и ForEach, чем создавать список с использованием In.

Оператор In Context

Оператор In Context определяет измерения в контексте в явной форме.

Пример: Использование оператора In Context для определения измерений в контексте

В этом примере у нас есть отчет, в котором отображается "Год" и "Доход с продаж". В поставщике данных также содержится объект "Квартал", однако это измерение в блок не включено. Вместо этого, требуется включить дополнительный столбец для отображения максимального значения дохода за каждый квартал каждого года. Отчет выглядит следующим образом:

Year	Sales revenue	Max Quarterly Revenue
2001	\$8 096 123,60	\$2 660 699,50
2002	\$13 232 246,00	\$4 186 120,00
2003	\$15 059 142,80	\$4 006 717,50

Узнать откуда берутся значения в столбце "Максимальный доход за квартал" в этом блоке можно просмотрев этот блок в паре с блоком, содержащим измерение "Квартал":

Year	Quarter	Sales revenue
	Q1	\$2660700
	Q2	\$2279003
	Q3	\$1367841
	Q4	\$1788580
2001		
	Макс.:	2660699.5

Year	Quarter	Sales revenue
	Q1	\$3326172
	Q2	\$2840651
	Q3	\$2879303
	Q4	\$4186120
2002		
	Макс.:	4186120

Year	Quarter	Sales revenue
	Q1	\$3742989
	Q2	\$4006718
	Q3	\$3953395
	Q4	\$3356041
2003		
	Макс.:	4006717.5

Столбец "Максимальный доход за квартал" содержит максимальные показатели дохода за каждый год. Например, K4 содержит значение максимального дохода в 2002 году, таким образом, "Максимальный доход за квартал" в строке 2002 года отобразит K4.

Использование оператора In, формула для "Максимального дохода за квартал"

```
Max ([Sales Revenue] In ([Year];[Quarter])) In ([Year])
```

По этой формуле Web Intelligence рассчитывает максимальных доход с продаж для каждой пары "год/квартал", а затем выводит полученное значение по годам.

Примечание:

Так как по умолчанию контекстом для вывода является "Год", особым образом указывать контекст вывода в этой формуле не требуется.

Контекстный оператор ForEach

Оператор ForEach добавляет в контекст измерения.

Пример: Использование оператора ForEach для добавления измерений в контекст

В следующей таблице показан максимальный доход по каждому кварталу в отчете, который содержит измерение "Квартал", не включенное в блок:

Год	Доход от продаж	Максимальный квартальный доход
2001	8 096 123,60	2 660 699,50
2002	13 232 246,00	4 186 120,00
2003	15 059 142,80	4 006 717,50

Для столбца "Максимальный доход за квартал" можно создать формулу, которая не содержит оператор ForEach:

```
Max ([Sales Revenue] In ([Year];[Quarter])) In ([Year])
```

Контекстный оператор "ForEach" позволяет добиться того же результата с помощью следующей формулы:

```
Max ([Sales Revenue] ForEach ([Quarter])) In ([Year])
```

Причина: Измерение "Год" является в блоке контекстом ввода по умолчанию. При использовании оператора "ForEach" измерение "Квартал" добавляется в контекст, в результате чего получается контекст ввода ([Год];[Квартал]).

Контекстный оператор ForAll

Контекстный оператор ForAll удаляет из контекста измерения.

Пример: Использование оператора ForAll для удаления измерений из контекста

Есть отчет, в котором отображается год, квартал и доход от продаж, и необходимо добавить столбец, в котором отображается суммарный доход по каждому году, как показано в следующем блоке:

Year	Quarter	Sales revenue	Yearly Total
2001	Q1	\$2 660 700	\$8 096 124
2001	Q2	\$2 279 003	\$8 096 124
2001	Q3	\$1 367 841	\$8 096 124
2001	Q4	\$1 788 580	\$8 096 124
2002	Q1	\$3 326 172	\$13 232 246
2002	Q2	\$2 840 651	\$13 232 246
2002	Q3	\$2 879 303	\$13 232 246
2002	Q4	\$4 186 120	\$13 232 246
2003	Q1	\$3 742 989	\$15 059 143
2003	Q2	\$4 006 718	\$15 059 143
2003	Q3	\$3 953 395	\$15 059 143
2003	Q4	\$3 356 041	\$15 059 143

Для суммирования доходов по годам контекстом ввода должен быть (Год); по умолчанию это (Год; Квартал). Поэтому необходимо удалить из контекста ввода квартал, указав в формуле ForAll ([Квартал]), что будет выглядеть следующим образом:

```
Sum([Sales Revenue] ForAll ([Quarter]))
```

Следует отметить, что для получения такого же результата можно использовать оператор "In". В данном случае формула будет выглядеть следующим образом:

```
Sum([Sales Revenue] In ([Year]))
```

В этой версии формулы в качестве контекста явно указан год, а не удаляется квартал, чтобы оставить год.

Ключевые слова с расширенным синтаксисом в приложении Web Intelligence

Ключевые слова с расширенным синтаксисом – это форма условного обозначения, которая позволяет обращаться к измерениям с расширенным синтаксисом без явного указания этих измерений. Это позволяет обеспечить создание неустаревающих отчетов; если в формулах не содержатся жестко запрограммированные ссылки на измерения, то они будут оставаться действительными даже при добавлении измерений в отчет или удалении измерений из отчета.

Предусмотрено пять ключевых слов расширенного синтаксиса: Отчет, Раздел, Разбиение, Блок и Тело.

Ключевое слово блока

В следующей таблице описаны измерения, на которые ссылается ключевое слово блока, в зависимости от его местоположения в отчете; ключевое слово блока часто включает в себя те же данные, что и ключевое слово раздела. Вся разница заключается в том, что Блок принимает во внимание фильтры блока, а Раздел нет.

Если расположено в...	Ссылается на эти данные...
Блок	Данные во всем блоке, без учета разрывов, но с учетом фильтров
Разбиение по блокам (верхний или нижний колонтитул)	Данные во всем блоке, без учета разрывов, но с учетом фильтров
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Не применимо
За пределами всех блоков или разделов	Не применимо

Пример: Ключевое слово блока

В отчете содержатся сведения о доходах от продаж за год, и квартал. В этом отчете создан раздел на основе объекта "Год". Фильтры блока опускают показатели за третий и четвертый кварталы.

2001

Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$2 660 700	\$2 469 851.25	\$8 096 123.60
Q2	\$2 279 003	\$2 469 851.25	\$8 096 123.60
Сумма:	4 939 702.5		

2002

Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$3 326 172	\$3 083 411.50	\$13 232 246.00
Q2	\$2 840 651	\$3 083 411.50	\$13 232 246.00
Сумма:	6 166 823		

2003

Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$3 742 989	\$3 874 853.20	\$15 059 142.80
Q2	\$4 006 718	\$3 874 853.20	\$15 059 142.80
Сумма:	7 749 706.4		

В столбце "В среднем за год" присутствует формула.

```
Average([Sales revenue] In Section)
```

Формула также присутствует в столбце "В среднем за первое полугодие".

```
Average([Sales revenue]) In Block
```

Можно проследить за тем, как ключевое слово блока учитывает фильтр блока.

Ключевое слово "Тело"

В следующей таблице описываются измерения, на которые ссылается ключевое слово "Тело" в зависимости от его расположения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Данные в блоке
Разбиение по блокам (верхний или нижний колонтитул)	Данные в блоке
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Данные в разделе
За пределами всех блоков или разделов	Данные в отчете

Пример: Ключевое слово "Тело"

В отчете отображаются объекты "Год", "Квартал" и "Доход с продаж" с разбиением по объекту "Год". В отчете содержится раздел, основанный на объекте "Год", и разбиение по объекту "Квартал".

Год	Квартал	Доходы с продаж	Тело
2001	K1	2 660 700	2 660 699,5
	K2	2 279 003	2 279 003
	K3	1 367 841	1 367 840,7
	K4	1 788 580	1 788 580,4
2001		8 096 123,6	

В строке "Тело" записана формула

```
Sum ([Sales Revenue]) In Body
```

Итоги в столбце "Тело" и "Доход с продаж" одинаковы, поскольку ключевое слово "Тело" относится к данным в блоке. Если удалить объект "Месяц", цифры в столбце "Блок" изменятся для обеспечения соответствия цифрам в столбце "Доход с продаж". Если формулу разместить в нижнем колонтитуле отчета, то будет вычисляться совокупный доход для блока.

Ключевое слово "Разбиение"

В следующей таблице описываются измерения, на которые ссылается ключевое слово "Разбиение" в зависимости от его расположения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Данные в части блока, разделенные разбиением
Разбиение по блокам (верхний или нижний колонтитул)	Данные в части блока, разделенные разбиением
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Не применимо

Если расположено в...	Ссылается на эти данные...
За пределами всех блоков или разделов	Не применимо

Пример: Ключевое слово "Разбиение"

В отчете содержатся сведения о доходах от продаж за год, и квартал.

Year	Quarter	Sales revenue	Break Total
2001	Q1	\$2 660 700	8 096 123.6
	Q2	\$2 279 003	8 096 123.6
	Q3	\$1 367 841	8 096 123.6
	Q4	\$1 788 580	8 096 123.6
2001			

В отчете содержится разбиение по объекту "Год". В столбце "Итог разбиения" записана формула:

```
Sum ([Sales Revenue]) In Break
```

При отсутствии ключевого слова "Разбиение" в этом столбце будут дублироваться цифры из столбца "Доход с продаж", поскольку будет использоваться контекст вывода по умолчанию ([Год];[Квартал]).

Ключевое слово Report

В следующей таблице приведены данные, на которые ссылается ключевое слово Report в зависимости от его местонахождения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Все данные в отчете
Разбиение по блокам (верхний или нижний колонтитул)	Все данные в отчете

Если расположено в...	Ссылается на эти данные...
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Все данные в отчете
За пределами всех блоков или разделов	Все данные в отчете

Пример: Ключевое слово Report

В отчете содержатся сведения о доходах от продаж за год, и квартал. В отчете есть столбец "Сумма по отчету", в котором отображается сумма всех доходов из отчета.

Year	Quarter	Sales revenue	Report Total
2001	Q1	\$2 660 700	36 387 512.4
2001	Q2	\$2 279 003	36 387 512.4
2001	Q3	\$1 367 841	36 387 512.4
2001	Q4	\$1 788 580	36 387 512.4
2002	Q1	\$3 326 172	36 387 512.4
2002	Q2	\$2 840 651	36 387 512.4
2002	Q3	\$2 879 303	36 387 512.4
2002	Q4	\$4 186 120	36 387 512.4
2003	Q1	\$3 742 989	36 387 512.4
2003	Q2	\$4 006 718	36 387 512.4
2003	Q3	\$3 953 395	36 387 512.4
2003	Q4	\$3 356 041	36 387 512.4

Формула для столбца "Сумма по отчету": Sum([Доходы от продаж]) In Report. Без ключевого слова Report в данном столбце воспроизводились бы цифры из столбца "Доходы от продаж", поскольку в нем использовался бы контекст вывода по умолчанию ([Год];[Квартал]).

Ключевое слово "Раздел"

В следующей таблице описываются данные, на которые ссылается ключевое слово "Раздел" в зависимости от его расположения в отчете:

Если расположено в...	Ссылается на эти данные...
Блок	Все данные в разделе
Разбиение по блокам (верхний или нижний колонтитул)	Все данные в разделе
Раздел (верхний колонтитул, нижний колонтитул или данные за пределами блока)	Все данные в разделе
За пределами всех блоков или разделов	Не применимо

Пример: Ключевое слово "Раздел"

В отчете отображаются объекты "Год", "Квартал" и "Доход с продаж".

2001

Quarter	Sales revenue	Section Total
Q1	\$2 660 700	8 095 814
Q2	\$2 278 693	8 095 814
Q3	\$1 367 841	8 095 814
Q4	\$1 788 580	8 095 814

В этом отчете создан раздел на основе объекта "Год". В столбце "Итог раздела" записана формула:

```
Sum ([Sales Revenue]) In Section
```

Цифра в столбце "Итог раздела" обозначает совокупный доход за 2001 год, поскольку разбиение раздела выполнено в объекте "Год". При отсутствии ключевого слова "Раздел" в этом столбце будут

дублироваться цифры из столбца "Доход с продаж", поскольку будет использоваться контекст вывода по умолчанию ([Год];[Квартал]).

Округление и обрезание чисел с помощью Web Intelligence

Некоторые функции Web Intelligence содержат параметр, определяющий, до какого уровня функция округляет или обрезает возвращаемое значение. Этот параметр принимает целое число, которое должно быть больше, меньше или равно 0.

Параметр	Описание
> 0	<p>Эта функция округляет или обрезает до десятичных знаков <параметра>.</p> <p>Примеры.</p> <p>Round(3,13; 1) возвращает 3,1</p> <p>Round(3,157; 2) возвращает 3,16</p>
0	<p>Эта функция округляет или обрезает до ближайшего целого числа</p> <p>Примеры.</p> <p>Truncate(3,7; 0) возвращает 3</p> <p>Truncate(4,164; 0) возвращает 4</p>
< 0	<p>Эта функция округляет или обрезает до ближайшего числа 10 (параметр = -1), 100 (параметр = -2), 1000 (параметр = -3) и так далее.</p> <p>Примеры.</p> <p>Round(123,76; -1) возвращает 120</p> <p>Round(459,9; -2) возвращает 500</p> <p>Truncate(1600; -3) возвращает 1000</p>

Примечание:

Числа представляются в виде двойных значений и имеют точность до шестнадцатой цифры.

См. также

- [Round](#)
- [Truncate](#)
- [EuroConvertTo](#)
- [EuroConvertFrom](#)
- [EuroFromRoundError](#)

- *EuroToRoundError*



Устранение ошибок в формулах Web Intelligence



Сообщения об ошибках формул и информационные сообщения

В некоторых случаях формулой Web Intelligence не может быть возвращено значение, поэтому появляется сообщение об ошибке или информационное сообщение, которое начинается с символа "#". Сообщение появляется в ячейке, в которой размещена формула.

#COMPUTATION

Сообщение об ошибке #COMPUTATION выводится в том случае, когда измерение со срезами, указанное в качестве параметра функции `RelativeValue`, более не доступно в контексте вычислений того блока, в котором эта функция расположена.

Сообщение об ошибке #COMPUTATION также связано с неправильным использованием контекстных операторов в формуле. Для получения дополнительной информации см. руководство *Использование функций, формул и вычислений в SAP BusinessObjects Web Intelligence*.

См. также

- [RelativeValue](#)

#CONTEXT

#CONTEXT отображается в мере, если мера имеет несуществующий контекст вычисления.

Сообщение об ошибке #CONTEXT относится к сообщениям об ошибках #INCOMPATIBLE и #DATASYNC, которые отображаются в измерениях в случае, когда блок содержит несуществующий контекст вычисления.

В случае #INCOMPATIBLE контекст не существует по причине несовместимости измерений; в случае #DATASYNC контекст не существует потому, что измерения получены из нескольких несинхронизированных поставщиков данных.

Пример: Несуществующий контекст вычисления в запросе

Если блок, в основе которого лежит юниверс "Island Resorts Marketing", содержит объекты "Год резервирования" и "Доход", сообщение об ошибке #CONTEXT отображается потому, что невозможно агрегировать доход по году резервирования. (Резервирования еще не сгенерировали ни один доход).

#DATASYNC

Сообщение #DATASYNC появляется при размещении измерения от другого поставщика данных в блок, в котором содержатся измерения от отличных поставщиков данных, и два поставщика данных не синхронизированы в объединенном измерении. Сообщение #DATASYNC появляется во всех измерениях в блоке, а сообщение #CONTEXT – в мерах.

Пример: Измерения от разных поставщиков данных в блоке

Если в отчете, основанном на Юниверсе Island Resorts Marketing, содержатся поставщики данных с объектами (Год, Доход) и (Квартал), в блоке с объектами Год, Квартал и Доход отображается сообщение #DATASYNC в столбцах Год и Квартал, поскольку два поставщика данных не синхронизированы в объединенном измерении.

#DIV/0

Сообщение #ДЕЛЕНИЕ/0 появляется, если в формуле выполняется попытка деления числа на ноль, что математически не представляется возможным. Ноль никогда не может являться делителем.

Пример: Определение дохода для одного объекта

В отчете отображаются доходы с продаж, число проданных объектов и доход, полученный с продажи одного объекта (который вычисляется путем деления дохода с продаж на число проданных объектов).

Для квартала отсутствует доход с продаж; в столбце дохода с продажи одного элемента появляется сообщение #ДЕЛЕНИЕ/0 для этого квартала, поскольку в формуле была предпринята попытка деления на ноль; т.е. деление дохода на нулевое число проданных объектов.

#EXTERNAL

Ошибка #EXTERNAL возникает в случае, когда формула ссылается на внешнюю функцию, недоступную в Web Intelligence.

#INCOMPATIBLE

#INCOMPATIBLE – сообщение о наличии несовместимых объектов в блоке.

Пример: Несовместимые объекты в запросе

Если блок, в основе которого лежит юниверс Island Resorts Marketing (Маркетинг островных курортов), содержит измерения "Год" и "Год бронирования", в столбцах, в которых содержатся эти измерения, отображается сообщение #INCOMPATIBLE, поскольку эти объекты несовместимы.

#MULTIVALUE

Сообщение #НЕОДНОЗНАЧНО появляется при размещении формулы, которая возвращает несколько значений в ячейку, для которой предусмотрен вывод только одного значения.

Пример: Многозначность в ячейке

В отчет с объектами "Страна", "Курорт" и "Доход" добавляется ячейка с формулой [Доход] ForEach ([Страна]). В этой ячейке отображается сообщение #НЕОДНОЗНАЧНО, поскольку для объекта "Страна" в отчете заданы два значения: "США" и "Франция".

В одной ячейке невозможно одновременно отобразить доходы для США и Франции. В ячейке с доходом, расположенной за пределами таблицы, можно только определенным образом агрегировать доходы, указанные в таблице (например путем сложения или нахождения среднего значения).

Если в отчете выполнено разбиение на разделы по объекту "Страна", формула является правильной, если помещена в раздел, поскольку в одном разделе задано только одно значение для объекта "Страна". Однако за пределами раздела формула будет возвращать #НЕОДНОЗНАЧНО

#OVERFLOW

Сообщение #ПЕРЕПОЛНЕНИЕ появляется, если в результате вычисления было получено слишком большое значение, которое не может быть обработано приложением Web Intelligence. Это значение в экспоненциальном представлении имеет вид 1,7E308 (1,7 с 307 нолями).

#PARTIALRESULT

Сообщение #PARTIALRESULT появляется, если приложению Web Intelligence не удалось извлечь все строки, связанные с объектом отчета.

Если сообщение #PARTIALRESULT появляется в отчете слишком часто и при наличии подходящих прав доступа, измените свойство запроса "Максимальное количество извлекаемых строк", чтобы разрешить приложению Web Intelligence извлечение дополнительных данных. Если отсутствуют права на изменение запроса, обратитесь к администратору Business Objects.

Если в отчете содержатся интеллектуальные меры, предпочтительнее настроить отображение сообщения #PARTIALRESULT, поскольку при использовании интеллектуальных мер в приложении Web Intelligence извлекаются большие объемы данных, чем при использовании классических мер.

#RANK

Сообщение об ошибке #RANK возникает при попытке ранжирования данных, в основе которых лежит объект, зависящий от порядка значений. (От порядка значений зависят объекты, использующие функцию Previous() или любую функцию промежуточного агрегирования.) Ранжирование приводит к тому, что эти объекты пересчитывают свои значения, в результате чего изменяется ранжирование – это приводит к возникновению циклической зависимости. Такая зависимость может возникать при использовании диалогового окна "Ранг" для создания ранжирования или при использовании функции Rank().

Пример: Ранжирование по промежуточному среднему предыдущих значений

Если выполняется попытка ранжирования блока по столбцу, который содержит функцию Previous() или любую функцию промежуточного среднего, ошибку #RANK возвращает весь блок.

#RECURSIVE

#RECURSIVE – сообщение о том, что Web Intelligence не удалось произвести расчет из-за наличия циклической зависимости.

Пример: Использование функции NumberOfPages()

Если поместить функцию NumberOfPages() в ячейку с установленными свойствами "Автоматический подбор высоты" или "Автоматический подбор ширины", Web Intelligence возвратит #RECURSIVE, так как в этом случае возникает циклическая зависимость. Web Intelligence необходимо знать точный размер отчета до того, как можно будет вернуть значение функции, но размер ячейки (который влияет на размер отчета) определяется содержимым ячейки.

#SECURITY

Ошибка #SECURITY возникает при попытке использования функции, для которой у пользователя нет прав защиты.

Пример: Использование функции DataProviderSQL()

Если пользователь, у которого нет прав просмотра поставщика данных SQL, размещает в ячейке функцию DataProviderSQL(), в ячейке отображается сообщение #SECURITY.

#SYNTAX

Ошибка #SYNTAX возникает в случае, когда формула ссылается на объект, который больше не существует в отчете.

Пример: Ссылка на несуществующий объект

Есть отчет, в котором исходно отображался доход от продаж по годам и кварталам, с дополнительным столбцом, в котором отображалась разница между доходом и средним доходом за год. Это значение представлено переменной "Разница со средним за год".

Если из отчета удалить переменную "Разница со средним за год", в столбце, в котором она находилась, возвращается ошибка #SYNTAX.

#TOREFRESH

#TOREFRESH – сообщение о том, что в ячейках, использующих интеллектуальные меры, возвращаемое ими значение не поддерживается. Такая ситуация имеет место, когда в поставщике данных недоступен «набор для группировки», содержащий значение.

Ошибка #TOREFRESH устраняется посредством обновления данных.


#UNAVAILABLE

Ошибка #UNAVAILABLE возникает в случае, когда Web Intelligence не может вычислить значение интеллектуальной меры.


Эта ситуация возникает в случае, когда Web Intelligence не может отобразить значения в отфильтрованной интеллектуальной мере без применения фильтра к запросу. Поскольку это порождает риск воздействия на другие отчеты, в основе которых лежит этот же запрос, Web Intelligence не применяет фильтр запроса.

#ERROR

#ERROR – сообщение об ошибке по умолчанию, которое отображается при возникновении ситуации, которой не присвоено никакое другое сообщение об ошибке.



Сравнение значений с
помощью функций Web
Intelligence



7



Сравнение значений с помощью функции *Previous*

Функция *Previous* возвращает предыдущее значение выражения для сравнения. Возвращенное значение зависит от макета отчета.

Для использования расширенных возможностей сравнения применяйте функцию *RelativeValue*. *RelativeValue* возвращает предыдущее или последующее значение выражения для сравнения. Возвращенное значение не зависит от макета отчета.

См. также

- *Previous*
- *RelativeValue*
- Сравнение значений с помощью функции *RelativeValue*

Сравнение значений с помощью функции *RelativeValue*

Функция *RelativeValue* возвращает значения выражения для сравнения. Функция возвращает данные значения независимо от макета отчета.

Во время использования функции *RelativeValue* необходимо указать следующие данные.

- Выражение, чье значение для сравнения необходимо найти (должно быть мерой или объектом-описанием измерения, доступными в блоке)
- Список «измерений срезов»
- Смещение

В данной функции используются измерения срезов, смещение и «измерения вложенных осей» (которые применяются измерениями срезов) для возврата значения для сравнения. Измерения вложенных осей – это все остальные измерения в контексте вычислений, не считая измерений срезов.

Выражаясь в общих чертах, функция `RelativeValue` возвращает значение выражения в строку, которая в списке значений измерений срезов является строкой смещения, удаленной из текущей строки, и в которой измерения вложенных осей такие же, как и в текущей строке.

Примечание:

Все измерения срезов должны находится в контексте вычислений блока, куда помещена функция. Если измерение среза впоследствии удалить, функция возвращает `#COMPUTATION`.

Пример:

В следующем примере в столбце `RelativeValue` содержится следующая формула.

```
RelativeValue ([Revenue]; ([Year]); -1)
```

- Выражение – [Доход];
- Измерение среза – [Год];
- Смещение равно минус единице (функция сразу возвращает предыдущее значение в списке).

Год	Квартал	Продавец	Доход	RelativeValue
2007	К1	Смит	1000	
2007	К2	Джонс	2000	
2007	К3	Уилсон	1500	
2007	К4	Харрис	3000	
2008	К1	Смит	4000	1000
2008	К2	Джонс	3400	2000
2008	К3	Уилсон	2000	1500
2008	К4	Харрис	1700	3000

Формула, выраженная бизнес-вопросом, указывает Web Intelligence, что необходимо возратить прибыль, полученную одним продавцом в таком же квартале предыдущего года.

С помощью формулы, выраженной вычислением при помощи слов, Web Intelligence возвращает значение [Прибыль] (выражение) в строку, где значение [Год] (измерение среза) является предыдущим значением списка значений объекта [Год], и где значения [Квартал] и [Продавец] (измерения вложенных осей) совпадают со значениями в текущей строке.

См. также

- *RelativeValue*

Измерения срезов и функция *RelativeValue*

В функции *RelativeValue* список значений измерений срезов используется для поиска сравнимой строки. Данная функция возвращает значение для сравнения с выражением, указанным в функции, находящейся за смещение строк от списка измерений срезов.

В результате порядок сортировки измерений срезов является решающим в определении выходных данных функции.

Пример: Несколько измерений срезов

В указанной ниже таблице столбец *RelativeValue* содержит следующую формулу:

```
RelativeValue ([Revenue]; ([Year]; [Quarter]); -1)
```

- Выражение – [Доход];
- Измерения срезов – ([Год];[Квартал]);
- Смещение равно минус единице (функция сразу возвращает предыдущее значение в списке).

Год	Квартал	Продавец	Доход	RelativeValue
2007	K1	Смит	1000	
2007	K2	Смит	2000	
2007	K3	Смит	1500	
2007	K4	Смит	3000*	

Год	Квартал	Продавец	Доход	RelativeValue
2007	K1	Джонс	4000	
2007	K2	Джонс	3400	
2007	K3	Джонс	2000	
2007	K4	Джонс	1700	
2008	K1	Смит	5000**	3000*
2008	K2	Смит	3000***	5000**
2008	K3	Смит	2700****	3000***
2008	K4	Смит	6800	2700****

Формула, выраженная бизнес-вопросом, указывает Web Intelligence, что необходимо отобразить прибыль, полученную одним продавцом в предыдущем квартале.

Формула, выраженная в виде вычисления в словах, указывает Web Intelligence, что необходимо возратить значение [Прибыль] в строку, где значения [Год] и [Квартал] представляют предыдущее значение в списке значений ([Год];[Квартал]), и где значение [Продавец] совпадает со значением в текущей строке.

Чтобы найти значение прибыли для сравнения, в Web Intelligence используется список значений измерений срезов.

Год	Квартал	
2007	K1	
2007	K2	
2007	K3	
2007	K4	*
2008	K1	**

Год	Квартал	
2008	K2	***
2008	K3	****
2008	K4	

Порядок сортировки в измерениях срезов определяет выходные данные функции. Знак "*" в таблицах отображает порядок сортировки.

См. также

- *RelativeValue*

Измерения срезов и разделы

Измерение среза может находиться в основной ячейке раздела отчета.

Пример:

В указанной ниже таблице столбец *RelativeValue* содержит следующую формулу:

```
RelativeValue ([Revenue]; ([Year]; [Quarter]); -1)
```

2007

Квартал	Продавец	Доход	RelativeValue
K1	Смит	1000	
K2	Смит	2000	
K3	Смит	1500	
K4	Смит	3000*	
K1	Джонс	4000	

Квартал	Продавец	Доход	RelativeValue
K2	Джонс	3400	
K3	Джонс	2000	
K4	Джонс	1700	

2008

Квартал	Продавец	Доход	RelativeValue
K1	Смит	5000**	3000*
K2	Смит	3000***	5000**
K3	Смит	2700****	3000***
K4	Смит	6800	2700****

Чтобы найти значение прибыли для сравнения, в Web Intelligence используется список значений измерений срезов.

Год	Квартал	
2007	K1	
2007	K2	
2007	K3	
2007	K4	*
2008	K1	**
2008	K2	***
2008	K3	****
2008	K4	

Порядок сортировки в измерениях срезов определяет выходные данные функции. Знак "*" в таблицах отображает порядок сортировки.

См. также

- *RelativeValue*

Порядок измерений срезов

Так как порядок сортировки списка значений измерений срезов определяет выходные данные функции *RelativeValue*, порядок, в котором указываются измерения срезов, влияет на выходные данные функции.

Пример: Порядок измерений срезов

В указанной ниже таблице столбец *RelativeValue* содержит следующую формулу:

```
RelativeValue ([Revenue]; ([Year]; [Quarter]); -1)
```

Год	Квартал	Продавец	Доход	RelativeValue
2007	K1	Смит	1000	
2007	K2	Смит	2000	
2007	K3	Смит	1500	
2007	K4	Смит	3000*	
2007	K1	Джонс	4000	
2007	K2	Джонс	3400	
2007	K3	Джонс	2000	
2007	K4	Джонс	1700	
2008	K1	Смит	5000**	3000*

Год	Квартал	Продавец	Доход	RelativeValue
2008	K2	Смит	3000***	5000**
2008	K3	Смит	2700****	3000***
2008	K4	Смит	6800	2700****

Формула, выраженная бизнес-вопросом, указывает Web Intelligence, что необходимо отобразить прибыль, полученную одним продавцом в предыдущем квартале.

Порядок сортировки измерений срезов выглядит следующим образом.

Год	Квартал	
2007	K1	
2007	K2	
2007	K3	
2007	K4	*
2008	K1	**
2008	K2	***
2008	K3	****
2008	K4	

Название функции изменяется на:

```
RelativeValue ([Revenue]; ([Quarter]; [Year]); -1)
```

Порядок сортировки измерений срезов становится следующим.

Квартал	Год	
K1	2007	*
K1	2008	**
K2	2007	***
K2	2008	****
K3	2007	*****
K3	2008	*****
K4	2007	*****
K4	2008	*****

Порядок сортировки оказывает следующее влияние на результат функции.

Год	Квартал	Продавец	Доход	RelativeValue
2007	K1	Смит	1000*	
2007	K2	Смит	2000***	
2007	K3	Смит	1500*****	
2007	K4	Смит	3000*****	
2007	K1	Джонс	4000	
2007	K2	Джонс	3400	
2007	K3	Джонс	2000	
2007	K4	Джонс	1700	
2008	K1	Смит	5000**	1000*

Год	Квартал	Продавец	Доход	RelativeValue
2008	K2	Смит	3000****	2000***
2008	K3	Смит	2700*****	1500*****
2008	K4	Смит	6800*****	3000*****

Формула, выраженная бизнес-вопросом, указывает Web Intelligence, что необходимо отобразить прибыль, полученную одним продавцом в предыдущем квартале.

Изменение в порядке сортировки измерения среза изменяет смысл формулы. Знак "*" в таблицах означает порядок сортировки

См. также

• [RelativeValue](#)

Сортировка измерений срезов

Так как порядок сортировки списка значений измерений срезов определяет выходные данные функции, сортировка, применяемая к измерению в измерениях срезов, влияет на выходные данные функции.

Пример: Пользовательская сортировка, применяемая к измерению среза

В указанной ниже таблице столбец *RelativeValue* содержит следующую формулу:

```
RelativeValue ([Revenue] ; ([Year] ; [Quarter]) ; -1)
```

Пользовательская сортировка (K1, K2, K4, K3) применяется к [Квартал], предоставляя следующие результаты для функции.

Год	Квартал	Продавец	Доход	RelativeValue
2007	K1	Смит	1000	
2007	K2	Смит	2000	
2007	K4	Смит	3000	
2007	K3	Смит	1500*	
2007	K1	Джонс	4000	
2007	K2	Джонс	3400	
2007	K4	Джонс	1700	
2007	K3	Джонс	2000	
2008	K1	Смит	5000**	1500*
2008	K2	Смит	3000***	5000**
2008	K4	Смит	6800****	3000***
2008	K3	Смит	2700	6800****

Отсортированный список измерений срезов выглядит следующим образом.

Год	Квартал	
2007	K1	
2007	K2	
2007	K4	
2007	K3	*
2008	K1	**
2008	K2	***
2008	K4	****
2008	K3	

Знак "*" в таблицах отображает порядок сортировки.

См. также

- [RelativeValue](#)

Использование функции RelativeValue в кросс-таблицах


Функция RelativeValue работает в кросс-таблицах точно так же, как и в вертикальных таблицах. Размещение данных в кросс-таблицах не влияет на выходные данные функции.

См. также


- [RelativeValue](#)

7 | Сравнение значений с помощью функций Web Intelligence

Сравнение значений с помощью функции `RelativeValue`



Дополнительная
информация



А



Источник информации	Местоположение
Информация о продуктах SAP BusinessObjects	http://www.sap.com
Справочный портал SAP	<p>Перейдите к http://help.sap.com/businessobjects/ и на боковой панели «BusinessObjects Overview» выберите All Products.</p> <p>Справочный портал SAP предоставляет доступ к актуальной документации по всем продуктам SAP BusinessObjects и их развертыванию. Можно загрузить документы в формате PDF или устанавливаемые HTML-библиотеки.</p> <p>Некоторые руководства находятся на веб-сайте SAP Service Marketplace и недоступны на справочном портале SAP. На данном портале перечислены эти руководства и даны соответствующие ссылки на SAP Service Marketplace. Клиенты, заключившие соглашение о техническом обслуживании, получают идентификатор авторизованного пользователя для доступа к этому веб-сайту. Для получения идентификатора обратитесь к представителю службы поддержки пользователей.</p>

Источник информации	Местоположение
SAP Service Marketplace	<p>http://service.sap.com/bosap-support > Документация</p> <ul style="list-style-type: none"> • Руководства по установке: https://service.sap.com/bosap-instguides • Примечания к выпуску: http://service.sap.com/releasenotes <p>На веб-сайте SAP Service Marketplace содержатся некоторые руководства по установке, модернизации, миграции и развертыванию, а также примечания к версиям и документация по поддерживаемым платформам. Клиенты, заключившие соглашение о техническом обслуживании, получают идентификатор авторизованного пользователя для доступа к этому веб-сайту. Для получения идентификатора обратитесь к представителю службы поддержки пользователей. Если вас перенаправили на веб-сайт SAP Service Marketplace со справочного портала SAP, с помощью меню в навигационной панели слева перейдите в категорию, содержащую нужные вам документы.</p>
Docupedia	<p>https://cw.sdn.sap.com/cw/community/docupedia</p> <p>Docupedia предоставляет дополнительные ресурсы документации, объединенную среду для создания контента и интерактивный канал обратной связи.</p>
Ресурсы разработчика	<p>https://bos.sdn.sap.com/</p> <p>https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary</p>
Статьи SAP BusinessObjects в сети сообщества SAP	<p>https://www.sdn.sap.com/irj/boc/businessobjects-articles</p> <p>Подобные статьи ранее назывались технической документацией.</p>

Источник информации	Местоположение
Примечания	https://service.sap.com/notes Эти примечания ранее назывались статьями базы знаний.
Форумы в сети сообщества SAP	https://www.sdn.sap.com/irj/scn/forums
Обучение	http://www.sap.com/services/education Мы можем предложить обучающий пакет, соответствующий вашим потребностям и предпочтительным формам обучения – от классических занятий в классах до специализированных курсов eLearning.
Интерактивная служба поддержки пользователей	http://service.sap.com/bosap-support На портале службы поддержки SAP представлены сведения о программах и услугах поддержки. Здесь также содержатся ссылки на самую разнообразную техническую информацию и множество файлов для загрузки. Клиенты, заключившие соглашение о техническом обслуживании, получают идентификатор авторизованного пользователя для доступа к этому веб-сайту. Для получения идентификатора обратитесь к представителю службы поддержки пользователей.
Консалтинговые услуги	http://www.sap.com/services/bysubject/businessobjectsconsulting Наши консультанты готовы оказывать вам поддержку на всех этапах – от начального анализа до развертывания системы. Советы экспертов можно найти, например, в темах, посвященных относительным и многомерным базам данных, возможностям подключения, средствам разработки баз данных и технологии встраивания с индивидуальными настройками.

Указатель

I

InfoView

отображение регистрационного имени
автора документов 149

S

SQL 137

и группирование множеств 54, 55
и наборы группировок 54
и обновление данных 55
оператор GROUPING SETS 57
оператор UNION 54, 57

T

TimeDim 129

U

URL-адреса

применение правил кодировки к 117

Б

блоки

отображение имен 191

булевы выражения

возвращение противоположного 216
связывание с помощью оператора And
215

булевы значения

определение 161

булевы значения ((*продолжение*))

проверка 197, 199

В

валюты

преобразование между европейскими
валютами 170, 172

вертикальные таблицы

контексты вычислений по умолчанию в
31

входной контекст

определенный 27

выходной контекст

определенные 28

вычисление абсолютного значения числа
168

вычисление дисперсии 96

вычисление дисперсии генеральной
совокупности 97

вычисление квадратного корня 188

вычисление косинуса 169

вычисление логарифма по основанию 10 181

вычисление логарифма по основанию n 180

вычисление максимального значения 73

вычисление медианы 74

вычисление минимального значения 75

вычисление моды 76

вычисление натурального логарифма 179

вычисление ошибок округления 173, 175

вычисление произведения 80

вычисление промежуточного количества 83

вычисление промежуточного максимума 85

вычисление промежуточного минимума 87

Указатель

вычисление промежуточного произведения 89
вычисление процентиля 80
вычисление процентного выражения 15, 77
вычисление синуса угла 187
вычисление скользящего среднего 81
вычисление среднего значения 66
вычисление среднеквадратического отклонения 92
вычисление среднеквадратического отклонения генеральной совокупности 94
вычисление суммы 95
вычисление тангенса 188
вычисление текущей суммы 91
вычисление факториала 178
вычисление экспоненциальной функции 177
вычисления
 пользовательские 10
 стандартные 10

Г

горизонтальные таблицы
 контексты вычислений по умолчанию в 33
группирование множеств
 управление в Web Intelligence 55

Д

данные
 обновление 55
даты
 вычисление относительных дат 128
 идентификация 159
 форматирование 102, 131
диаграммы
 отображение имен 191
дисперсия 96
дисперсия генеральной совокупности 97
документы
 отображение автора 149

документы *((продолжение))*
 отображение времени последнего сохранения 152
 отображение даты последнего сохранения 151
 отображение даты создания 150
 отображение имени 151
 просмотр запросов в 155

Е

евро
 преобразование в 172
 преобразование из 170
европейские валюты
 преобразование между 170, 172

З

завершающие пробелы
 удаление из строк символов 113
запросы
 просмотр сводки 155
значения
 округление 239
 подсчет 68, 83
 сравнение с помощью RelativeValue 252, 254, 256, 258, 261, 263
 сравнение с помощью функции Previous 252
 усечение 239
значения времени
 определение 165
значения меры
 интерполяция 70, 223

И

идентификация дат 159
идентификация чисел 163

- измерения
 - добавление к контексту вычисления 39, 230
 - и группирование множеств 60
 - и сообщение об ошибке #DATASYNC 245
 - и сообщение об ошибке #INCOMPATIBLE 246
 - измерения среза 252, 254
 - удаление из контекста вычисления 40, 230
 - указание в контексте вычислений 37, 228
- измерения среза 252
 - влияние порядка сортировки 254, 258, 261
 - и разделы 256
 - порядок 258
- интеллектуальные меры
 - и наборы группировок 54
 - и область анализа 56
 - и операторы контекста 60
 - и переменные 60
 - и сообщение об ошибке #PARTIALRESULT 247
 - и сообщение об ошибке #TOREFRESH 249
 - и сообщение об ошибке #UNAVAILABLE 250
 - и стандартные вычисления 10
 - и фильтры детализации 62
 - и формулы 60
 - невозможность вычисления 250
 - определенные 54
- интеллектуальные показатели
 - влияние фильтров на 61
- интерполяция значений меры 70, 223
- К**
 - квадратный корень 188
 - ключевое слово "Разбиение" 44, 235
 - ключевое слово Body 47, 234
 - ключевое слово Report 48
 - ключевое слово блока 45, 232
 - ключевое слово отчета 42, 236
 - ключевое слово раздела 43, 238
 - ключевые слова
 - Блок 45, 232
 - ключевые слова расширенного синтаксиса 41, 232
 - Отчет 42, 48, 236
 - Разбиение 44, 235
 - Раздел 43, 238
 - создание отчетов, общих с 48
 - Тело 47, 234
 - ключевые слова расширенного синтаксиса 41, 232
 - использование вместе с функцией Average 45, 232
 - использование вместе с функцией Sum 42, 43, 44, 47, 234, 235, 236, 238
 - создание отчетов, общих с 48
 - коды ASCII 98
 - возвращение символов, связанных с 99
 - конечные пробелы
 - удаление из строк символов 115
 - конкатенация строк символов 21, 100, 214
 - контекстные операторы 22
 - контексты вычислений
 - входной контекст 26, 27
 - выходной контекст 26, 28
 - изменение с расширенным синтаксисом 30
 - определенные 26
 - по умолчанию 30
 - контексты вычислений по умолчанию
 - в вертикальных таблицах 31
 - в горизонтальных таблицах 33
 - в кросс-таблицах 33
 - в разбиениях 36
 - в разделах 34
 - изменение с расширенным синтаксисом 37
 - контексты расчета
 - и интеллектуальные меры 60

Указатель

косинус 169
кросс-таблицы
и функция RelativeValue 263
контексты вычислений по умолчанию в
33

Л

логарифмы 179, 180, 181
логарифмы по основанию 10 181
логарифмы по основанию n 180
логические операторы 22, 215
логические функции
описание 66

М

максимальные значения
вычисление промежуточного максимума
85
математические операторы 21, 214
медиана 74
меры
возвращение последующих значений 210
возвращение предыдущих значений 210
и контексты вычислений по умолчанию
30
минимальные значения
вычисление промежуточного минимума
87
мода 76

Н

наборы группировок 54
и область анализа 56
и оператор UNION 57
определенные 54
пример управления 57
натуральные логарифмы 179

начальные пробелы
удаление из строк символов 107, 115
нечетные числа
определение 158, 166
номера страниц
отображение в отчетах 204

О

область анализа
и интеллектуальные меры 56
и наборы группировок 56
обновление данных 55
объединение строк символов 21, 100, 214
объекты
отображение имени 213
просмотр фильтров по 156
округление значений 239
округление чисел 168
оператор All 202, 218, 219, 221
оператор And 22, 215
оператор Between 22, 215, 216
оператор Bottom 183, 219
оператор Break 220
оператор Col 81, 85, 87, 89, 91, 224
оператор Distinct 221
оператор Drill 202, 219
оператор ForAll 40, 48, 230
оператор ForEach 39, 230
оператор In 37, 228
оператор IncludeEmpty 221
оператор Inlist 22, 215
оператор InList 217
оператор Linear 70, 222
оператор NoNull 205, 223
оператор Not 22, 215, 216
оператор NotOnBreak 70, 223
оператор Or 22, 215
оператор PointToPoint 70, 223
оператор Row 81, 85, 87, 89, 91, 224
оператор Self 205, 226
оператор Top 183, 219

- оператор UNION 57
- оператор Where 226
- оператор ИЛИ 215
- оператор Индекс 148
- операторы
 - All 202, 218, 219, 221
 - And 22, 215
 - Between 22, 215, 216
 - Bottom 183, 219
 - Break 220
 - Col 81, 83, 85, 87, 89, 91, 224
 - Distinct 221
 - Drill 202, 219
 - ForAll 40, 48, 230
 - ForEach 39, 230
 - In 37, 228
 - IncludeEmpty 83, 221
 - InList 22, 215, 217
 - Linear 70, 222
 - NotNull 205, 223
 - Not 22, 215, 216
 - NotOnBreak 70, 223
 - Or 22, 215
 - PointToPoint 70, 223
 - Row 81, 83, 85, 87, 89, 91, 224
 - Self 205, 226
 - Where 226
 - зависящие от функций 23, 218
 - зависящие от функций операторы 23, 218
 - ИЛИ 215
 - Индекс 148, 222
 - контекст 22
 - логический 22, 215
 - математические 21, 214
 - оператор Index 222
 - определенные 21, 213
 - Сверху 183, 219
 - условные 22, 214
- опорные данные
 - возвращение даты 145
- определение булевых значений 161
- определение значений времени 165
- определение нечетных чисел 158, 166
- определение ошибок 160
- определение пустых значений 162
- определение строк 164
- определение четных чисел 158, 166
- основной предпочтительный языковой
 - стандарт для просмотра
 - отображение 194
- отдельные ячейки
 - и сообщение об ошибке #MULTIVALUE 246
- отображение автора документов 149
- отображение времени последнего сохранения документа 152
- отображение даты последнего сохранения документа 151
- отображение даты создания документов 150
- отображение имен диаграмм 191
- отображение имен объектов отчетов 201
- отображение имен таблиц 191
- отображение имен юниверсов 147
- отображение количества страниц в отчете 204
- отображение номера строки 212
- отображение номеров столбцов 191
- отображение номеров страниц 204
- отображение основного предпочтительного языкового стандарта для просмотра 194
- отображение откликов на приглашения 148
- отображение предпочтительного языкового стандарта для просмотра 197
- отображение синтаксиса функций 14
- отображение языкового стандарта документа 193
- отображение языкового стандарта продукта 195
- отчеты
 - отображение имен объектов в 201
 - отображение количества страниц в 204
 - отображение номеров страниц в 204
 - просмотр имен 212

Указатель

ошибки

определение 160

ошибки округления 173, 175

П

переменные

и интеллектуальные меры 60

упрощение формул с 12, 19

подсчет значений 83

подсчет строк в таблицах 201

показатели

определение интеллектуальных мер 54

пользовательские вычисления 10

использование формул для построения
11

порядок сортировки

и измерения среза 254, 258, 261

поставщики данных 143

отображение имени юниверса 147

отображение количества строк в 144

просмотр кода SQL, сгенерированного
137

Предпочтительный языковой стандарт для

просмотра

отображение 197

преобразование в евро 172

преобразование из евро 170

приглашения

отображение откликов на 15, 145, 148

проверка булевых значений 197, 199

произведения

вычисление промежуточного

произведения 89

промежуточное количество 83

процентиль 80

пустые значения

определение 162

Р

разбиения

контексты вычислений по умолчанию в
36

разделы

и измерения среза 256

и функция RelativeValue 256

контексты вычислений по умолчанию в
34

разные функции

описание 66

ранжирование

и функции промежуточного агрегирования
248

ранжирование данных 183

расширенный синтаксис 22

изменение контекста вычислений по
умолчанию с 37

ключевое слово Block 45, 232

ключевое слово Body 47, 234

ключевое слово Break 44, 235

ключевое слово Report 42, 48, 236

ключевое слово Section 43, 238

оператор ForAll 37, 40, 227, 230

оператор ForEach 37, 39, 227, 230

оператор In 37, 227, 228

Редактор формул

отображение синтаксиса функций в 14

Руководство "Использование функций,
формул и вычислений"
сведения 8

С

свойства запроса

Максимальное количество извлекаемых
строк 247

символы

возвращение кодов ASCII 98

отображение из кодов ASCII 99

- символьные функции
 - описание 66
- синтаксис функций
 - как в Web Intelligence отображается 14
 - пример 14
- синус 187
- создание пользовательских вычислений с помощью формул 11
- сообщение об ошибке #COMPUTATION 210, 244
- сообщение об ошибке #CONTEXT 244, 245
- сообщение об ошибке #DATASYNC 244, 245
- сообщение об ошибке #ERROR 170, 172, 173, 175, 189, 250
- сообщение об ошибке #EXTERNAL 246
- сообщение об ошибке #INCOMPATIBLE 244, 245, 246
- сообщение об ошибке #MULTIVALUE 212, 246
- сообщение об ошибке #OVERFLOW 247
- сообщение об ошибке #PARTIALRESULT 247
 - и интеллектуальные меры 247
- сообщение об ошибке #RANK 248
- сообщение об ошибке #RECURSIVE 248
- сообщение об ошибке #SECURITY 249
- сообщение об ошибке #SYNTAX 249
- сообщение об ошибке #TOREFRESH 55, 60, 249
 - и интеллектуальные меры 249
- сообщение об ошибке #UNAVAILABLE 61, 250
 - и интеллектуальные меры 250
- сообщение об ошибке #ДЕЛЕНИЕ/0 245
- сообщения об ошибках
 - #COMPUTATION 210, 244
 - #CONTEXT 244, 245
 - #DATASYNC 244, 245
 - #DIV/0 245
 - #ERRPR 170, 172, 173, 175, 189, 250
 - #EXTERNAL 246
 - #INCOMPATIBLE 244, 245, 246
 - #MULTIVALUE 212, 246
- сообщения об ошибках ((продолжение))
 - #OVERFLOW 247
 - #PARTIALRESULT 247
 - #RANK 248
 - #RECURSIVE 248
 - #SECURITY 249
 - #SYNTAX 249
 - #TOREFRESH 55, 60, 249
 - #UNAVAILABLE 61, 250
- сообщения об ошибках в формулах 244
- сортировки
 - и измерения среза 254, 258, 261
- сравнение значений с помощью RelativeValue 252, 254, 256, 258, 261, 263
- сравнение значений с помощью функции Previous 252
- среднее
 - вычисление скользящего среднего 81
- среднеквадратическое отклонение 92
- среднеквадратическое отклонение генеральной совокупности 94
- стандартное вычисление функции Average 10
- стандартное вычисление функции Count 10
- стандартное вычисление функции Default 10
- стандартное вычисление функции Maximum 10
- стандартное вычисление функции Minimum 10
- стандартное вычисление функции Percentage 10
- стандартное вычисление функции Sum 10
- стандартные вычисления 10
- столбцы
 - отображение номеров 191
- страницы
 - отображение количества в отчетах 204
- строки
 - определение 164
 - отображение количества в поставщике данных 144
 - отображение номера строки 212

Указатель

строки ((*продолжение*))
 подсчет в таблицах 201
 подсчет всех значений 218
 подсчет различных значений 218
строки символов
 возврат самых левых символов 105
 возвращение самых правых символов 111
 вычисление длины 107
 замена частей 111
 заполнение другими строками 106, 112
 извлечение разделов из 114
 объединение/конкатенация 21, 100, 214
 перевод в верхний регистр 116
 перевод в верхний регистр всех первых букв в 117
 перевод в верхний регистр первых букв 117
 перевод в числа 189
 перевод первой буквы в верхний регистр 104
 повторение 101
 преобразование в нижний регистр 108
 применение правил кодировки URL к 117
 соответствие шаблону 109
 удаление конечных пробелов из 113, 115
 удаление начальных пробелов из 107, 115

T

таблицы
 вертикальная 31
 горизонтальная 33
 отображение имен 191
 подсчет строк в 201
тангенсы 188

У

умножение чисел 80, 89

упрощение формул с помощью переменных 12, 19
усечение значений 239
условие If...Then...Else 197
условные операторы 22, 214

Ф

факториал 178
фильтры
 блок 157
 детализация 219
 и интеллектуальные меры 62
 на измерения 61
 отображение всех значений 218
 отображение фильтров детализации 218
 отчет 156, 157, 219
 раздел 157
 фильтры детализации 62
фильтры детализации 219
 и интеллектуальные меры 62
 игнорирование 219
 отображение 218
фильтры для измерений
 влияние на интеллектуальные показатели 61
фильтры отчета 156, 219
 игнорирование 219
 просмотр сводки 157
форматирование чисел 103
формулы
 и интеллектуальные меры 60
 интеллектуальные меры в 60
 использование операторов в 21, 213
 создание пользовательских вычислений с помощью 11
 сообщения об ошибках, сгенерированные 244
 упрощение с переменными 12, 19
функции
 Abs 168
 Asc 98

функции ((продолжение))

Average 45, 66, 221, 232
 BlockName 191
 Ceil 168
 Char 99
 ColumnNumber 191
 Concatenation 100
 Cos 169
 Count 68, 218, 221
 CurrentDate 118
 CurrentTime 119
 CurrentUser 192
 DataProvider 134
 DataProviderKeyDate 135
 DataProviderKeyDateCaption 136
 DataProvidersSQL 137
 DataProviderType 138
 DayName 119
 DayNumberOfMonth 120
 DayNumberOfWeek 121
 DayNumberOfYear 122
 DaysBetween 122
 DocumentAuthor 149
 DocumentCreationDate 150
 DocumentCreationTime 150
 DocumentDate 151
 DocumentName 151
 DocumentPartiallyRefreshed 152
 DocumentTime 152
 DrillFilters 153
 EuroConvertFrom 170, 239
 EuroConvertTo 172, 239
 EuroFromRoundError 173, 239
 EuroToRoundError 175, 239
 Even 158
 Exp 177
 Fact 178
 Fill 101
 First 69
 Floor 179
 ForceMerge 192
 FormatDate 102

функции ((продолжение))

FormatNumber 103
 GetContentLocale 193
 GetDominantPreferredViewingLocale 194
 GetLocale 195
 GetLocalized 195
 GetPreferredViewingLocale 197
 HTMLEncode 104
 If 22, 199, 214
 InitCap 104
 IsDate 159
 IsError 160
 IsLogical 161
 IsNull 162
 IsNumber 163
 IsPromptAnswered 139
 IsString 164
 IsTime 165
 Last 73
 LastDayOfMonth 123
 LastDayOfWeek 124
 LastExecutionDate 140
 LastExecutionDuration 141
 Left 105
 LeftPad 106
 LeftTrim 107
 Length 107
 LineNumber 201
 Ln 179
 Log 180
 Log10 181
 Lower 108
 Match 109
 Max 39, 73, 230
 Median 74
 Min 75
 Mod 181
 Mode 76
 Month 125
 MonthNumberOfYear 125
 MonthsBetween 126
 NameOf 201

Указатель

функции ((продолжение))

NoFilter 202, 218, 219
NumberOfDataProviders 143
NumberOfPages 204, 248
NumberOfRows 144
Odd 166
Page 204
Percentage 15, 77, 220, 224
Percentile 80
Pos 110
Power 182
Previous 205, 223, 226, 248, 252
Product 80
PromptSummary 154
Quarter 127
QuerySummary 155
RefValue 209
RefValueDate 145
RefValueUserResponse 145, 222
RelativeDate 128
RelativeValue 210, 244, 252, 256, 258, 261, 263
Replace 111
ReportFilter 156
ReportFilterSummary 157
ReportName 212
Right 111
RightPad 112
RightTrim 113
Round 185, 239
RowIndex 212
RunningAverage 81, 221, 224
RunningCount 83, 221, 224
RunningMax 85, 224
RunningMin 87, 224
RunningProduct 89, 224
RunningSum 91, 224
Sign 186
Sin 187
Sqrt 188
StdDev 92
StdDevP 94

функции ((продолжение))

Substr 114
Sum 15, 40, 42, 43, 44, 47, 48, 95, 230, 234, 235, 236, 238
Tan 188
ToDate 131
ToNumber 189
Trim 115
Truncate 190, 239
UniqueNameOf 213
UniverseName 147
Upper 116
URLEncode 117
UserResponse 15, 222
Var 96
VarP 97
Week 132
WordCap 117
Year 133
включение в ячейки 14
Интерполяция 70, 222, 223
категории 66
определенные 14
ОтветПользователя 148
примеры 15
Ранжирование 183, 219
синтаксис функций 14
смесь с текстом в ячейках 14
соединение 134
функция DocumentCreationDate 150
функция DocumentCreationTime 150
функция GetPreferredViewingLocale 197
функция Interpolation 223
функция LastExecutionTime 142
функция Percentage 224
функция RunningAverage 224
функция RunningCount 224
функция RunningMax 224
функция RunningMin 87, 224
функция RunningProduct 224
функция RunningSum 224
функция Sum 95

- функции агрегирования
 - описание 66
- функции даты и времени
 - описание 66
- функции для поставщиков данных
 - описание 66
- функции документов
 - описание 66
- функции промежуточного агрегирования
 - и ранжирование 248
 - и сообщение об ошибке #RANK 248
- функция Abs 168
- функция Asc 98
- функция Average 66, 221
 - использование вместе с ключевыми словами расширенного синтаксиса 45, 232
- функция BlockName 191
- функция Ceil 168
- функция Char 99
- функция ColumnNumber 191
- функция Concatenation 100
- функция Connection 134
- функция Cos 169
- функция Count 68, 218, 221
- функция CurrentDate 118
- функция CurrentTime 119
- функция CurrentUser 192
- функция DataProvider 134
- функция DataProviderKeyDate 135
- функция DataProviderKeyDateCaption 136
- функция DataProviderSQL 137
- функция DataProviderType 138
- функция DayName 119
- функция DayNumberOfMonth 120
- функция DayNumberOfWeek 121
- функция DayNumberOfYear 122
- функция DaysBetween 122
- функция DocumentAuthor 149
- функция DocumentDate 151
- функция DocumentName 151
- функция DocumentPartiallyRefreshed 152
- функция DocumentTime 152
- функция DrillFilter 153
- функция EuroConvertFrom 170, 239
- функция EuroConvertTo 172, 239
- функция EuroFromRoundError 173, 239
- функция EuroToRoundError 175, 239
- функция Even 158
- функция Exp 177
- функция Fact 178
- функция Fill 101
- функция First 69
- функция Floor 179
- функция ForceMerge 192
- функция FormatDate 102
- функция FormatNumber 103
- функция GetContentLocale 193
- функция GetDominantPreferredViewingLocale 194
- функция GetLocale 195
- функция GetLocalized 195
- функция HTMLEncode 104
- функция If 22, 199, 214
- функция InitCap 104
- функция Interpolation 222, 223
- функция IsDate 159
- функция IsError 160
- функция IsLogical 161
- функция IsNull 162
- функция IsNumber 163
- функция IsPromptAnswered 139
- функция IsString 164
- функция IsTime 165
- функция Last 73
- функция LastDayOfMonth 123
- функция LastDayOfWeek 124
- функция LastExecutionDate 140
- функция LastExecutionDuration 141
- функция LastExecutionTime 142
- функция Left 105
- функция LeftPad 106
- функция LeftTrim 107
- функция Length 107

Указатель

- функция LineNumber 201
- функция Ln 179
- функция Log 180
- функция Log10 181
- функция Lower 108
- функция Match 109
- функция Max 73
 - использование с операторами контекста 39, 230
- функция Median 74
- функция Min 75
- функция Mod 181
- функция Mode 76
- функция Month 125
- функция MonthNumberOfYear 125
- функция MonthsBetween 126
- функция NameOf 201
- функция NoFilter 202, 218, 219
- функция NumberOfDataProviders 143
- функция NumberOfPages 204
 - и сообщение об ошибке #RECURSIVE 248
- функция NumberOfRows 144
- функция Odd 166
- функция Page 204
- функция Percentage 15, 77
- функция Percentile 80
- функция Pos 110
- функция Power 182
- функция Previous 205, 223, 226
 - и сообщение об ошибке #RANK 248
 - сравнение значений с помощью 252
- функция Product 80
- функция PromptSummary 154
- функция Quarter 127
- функция QuerySummary 155
- функция Rank 183, 219
- функция RefValue 209
- функция RefValueDate 145
- функция RefValueUserResponse 145, 222
- функция RelativeDate 128
- функция RelativeValue 210
 - и измерения среза 254, 256, 261
 - и кросс-таблицы 263
 - и разделы 256
 - и сообщение об ошибке #COMPUTATION 244
 - сравнение значений с помощью 252, 254, 256, 258, 261, 263
- функция Replace 111
- функция ReportFilter 156
- функция ReportFilterSummary 157
- функция ReportName 212
- функция Right 111
- функция RightPad 112
- функция RightTrim 113
- функция Round 185, 239
- функция RowIndex 212
- функция RunningAverage 81, 221
- функция RunningCount 83, 221
- функция RunningMax 85
- функция RunningProduct 89
- функция RunningSum 91
- функция Sign 186
- функция Sin 187
- функция Sqrt 188
- функция StdDev 92
- функция StdDevP 94
- функция Substr 114
- функция Sum 15
 - использование вместе с ключевыми словами расширенного синтаксиса 42, 43, 44, 47, 48, 234, 235, 236, 238
 - использование с операторами контекста 40, 230
- функция Tan 188
- функция ToDate 131
- функция ToNumber 189
- функция Trim 115
- функция Truncate 190, 239
- функция UniqueNameOf 213
- функция UniverseName 147
- функция Upper 116

функция URLEncode 117
 функция UserResponse 15, 148, 222
 функция Var 96
 функция VarP 97
 функция Week 132
 функция WordCap 117
 функция Year 133
 функция интерполяции 70

Ч

четные числа
 идентификация 158, 166
 числа
 возведение в степень 182
 вычисление модуля 181
 деление 181
 идентификация 163
 модуль 181
 округление 168, 185
 округление с недостатком 179
 округление чисел 185
 округление чисел с недостатком 179
 определение четности/нечетности 158, 166
 перевод строк символов в 189
 умножение 80, 89
 усечение 190
 усечение чисел 190
 форматирование 103

числовые функции
 описание 66

Э

экспоненциальная функция 177

Ю

юниверсы
 отображение имен 147

Я

Языковой стандарт документа
 отображение 193
 Языковой стандарт продукта
 отображение 195
 языковые стандарты
 отображение основного
 предпочтительного языкового
 стандарта для просмотра 194
 отображение языкового стандарта
 документа 193
 отображение языкового стандарта
 продукта 195
 ячейки
 включая текст в 14
 включение функций в ячейки 14

