## Язык запросов

 ***Источники данных для запроса***

В качестве источника данных языка запросов выступают таблицы данных. Таблицы разделяются на два вида: реальные и виртуальные.

 Реальные таблицы хранятся в базе данных. Это объекты информационной базы. В запросах к реальным таблицам можно использовать вычисляемые поля, рассчитываемые на основе реальных полей таблиц.

 Виртуальные таблицы не хранятся в базе данных. При обращении к ним система автоматически собирает данные из реальных таблиц, а в тексте запроса могут быть указаны параметры построения такой выборки.

 Примеры реальных таблиц – справочники, документы, регистры и т.п.

 Примеры виртуальных таблиц – таблицы остатков, оборотов и остатков и оборотов регистров накопления.

 Подклассом реальных таблиц являются объектные таблицы. Например, Справочник.Товары, Справочник.Контрагенты. Каждая отдельная запись объектной таблицы хранит состояние отдельного объекта соответствующего типа. Поэтому у каждой объектной таблицы определено поле *Ссылка*, в котором хранится ссылка на текущую запись (например, элемент справочника или документ). Объектные таблицы могут быть иерархическими. Для них имеется поле *Родитель*, содержащее ссылку на запись, которой подчинена текущая запись таблицы.

 Полем таблицы может быть либо одиночное поле, либо вложенная таблица (Н/р табл. часть документа). Вложенной таблице в запросе соответствует поле типа *РезультатЗапроса* с заранее заданным набором колонок.

 Если значение поля в запросе не определено, оно имеет тип NULL.

***Предопределенные данные конфигурации***

 В тексте запроса можно использовать обращение к предопределенным данным конфигурации. К ним относятся системные перечисления, определенные в конфигурации перечисления, предопределенные элементы справочников и др.

 Для обращения к ним используется конструкция

Значение(ПредопределенныйЭлемент)

Например,

Значение(ВидДвиженияНакопления.Приход) – системное перечисление.

Значение(Перечисление.ВидЗанятия.Лекция) – перечисление, определенное в конфигураторе.

Значение(Справочники.Города.Москва) – предопределенный элемент справочника городов.

***Основные секции текста запроса***

 Текст запроса состоит из последовательности секций.

<Описание запроса>

[<Объединение запросов>]

[<Упорядочивание результатов>]

[АВТОУПОРЯДОЧИВАНИЕ]

[<Описание итогов>]

 <Описание запроса> – единственная обязательная секция в тексте запроса. Во многих случаях достаточно только этой секции. В ней объявляются источники данных запроса, поля выборки, группировки и т.д.

<Объединение запросов> – предназначена для объединения результатов выполнения нескольких запросов.

<Упорядочивание результатов> – позволяет задать условия упорядочивания строк в результате запроса.

АВТОУПОРЯДОЧИВАНИЕ – включает режим автоматического упорядочивания, предопределенного системой.

<Описание итогов> – задает набор итогов, которые рассчитываются в запросе.

***Секция описания запроса***

Секция состоит из последовательности взаимосвязанных разделов.

ВЫБРАТЬ [РАЗРЕШЕННЫЕ] [РАЗЛИЧНЫЕ] [ПЕРВЫЕ <Кол-во>]

 <Список полей выборки>

[ИЗ <СписокИсточников>]

[ГДЕ <УсловиеОтбора>]

[СГРУППИРОВАТЬ ПО <Поля группировки>]

[ИМЕЮЩИЕ <Условие отбора группировок>]

[ДЛЯ ИЗМЕНЕНИЯ [<Список таблиц верхнего уровня>]]

 РАЗРЕШЕННЫЕ – запрос выбирает только те записи, которые разрешены текущему пользователю, иначе он отработает с ошибкой при попытке выбора неразрешенных записей.

 РАЗЛИЧНЫЕ – в результат запроса не попадут повторяющиеся строки.

 ПЕРВЫЕ <Количество> – запрос выбирает указанное количество строк, учитывая заданные в нем правила упорядочивания. Например, 5 самых дорогих товаров.

 ДЛЯ ИЗМЕНЕНИЯ – блокировка считываемых в транзакции данных. Блокируемые данные становятся недоступными другим пользователям. В файл-серверном варианте блокируется таблица целиком, в клиент-серверном – только записи, обрабатываемые запросом.

 **Примеры простейших запросов**

Выбрать

Документ.Лекция.Ссылка

 Результат – перечень ссылок на все документы-лекции, зарегистрированные в информационной базе.

Выбрать Различные

Документ.РасходнаяНакладная.Контрагент

 Результат – список контрагентов, которым отпускались товары. Если опустить РАЗЛИЧНЫЕ, каждый контрагент попадет в результат запроса столько раз, сколько ему отпускался товар.

Выбрать Первые 5

Справочник.Номенклатура.Наименование,

Справочник.Номенклатура.ЗакупочнаяЦена

Упорядочить По

Справочник.Номенклатура.ЗакупочнаяЦена Убыв

 Результат – выборка 5 самых дорогих позиций номенклатуры, упорядоченных по убыванию.

 **Список полей выборки** задается следующей конструкцией

<Выражение>[.<ГруппаПолей>] |

 ПУСТАЯТАБЛИЦА ([<Псевдоним, …>])

[КАК <ПсевдонимПоля>], …

 Выражение – ссылка на поле исходной таблицы или некоторое выражение. Группа полей указывается в том случае, если выражение задает вложенную таблицу. Если для вложенной таблицы не указать группу полей – будут обрабатываться все ее поля.

 ПсевдонимПоля – может быть назначен полю выборки. Псевдоним можно использовать в конструкциях УПОРЯДОЧИТЬ ПО, ИТОГИ и при обработке результатов запроса с целью более короткой записи.

 Пример псевдонима.

Выбрать

Справочник.Номенклатура.Наименование Как Товар,

Справочник.Номенклатура.Родитель.Наименование Как Группа

Из

Справочник.Номенклатура

 Результат – получим таблицу из 2 полей Товар и Группа:

|  |  |
| --- | --- |
| Товар | Группа |
| Консультации | Услуга |
| Ноутбук | Товар |
| … |  |

 Если бы псевдонимов не было, поля таблицы назывались бы Наименование и Наименование1.

 Поле в списке выбора может ссылаться на вложенную таблицу источника данных запроса (табл. часть документа). В этом случае после выполнения запроса оно будет содержать вложенный результат запроса, сформированный на основе таблицы-источника. По умолчанию во вложенный результат включаются все поля вложенной таблицы. Однако можно перечислить группу полей вложенной таблицы при помощи конструкции <ГруппаПолей>:

(<Выражение> [КАК <ПсевдонимПоля>], …)

 В простейшем случае выражение – ссылка на поле вложенной таблицы.

 Пример.

 В запросе необходимо выбрать данные по лекциям с перечнем присутствующих студентов.

Выбрать

Документ.Лекция.Ссылка как Лекция,

Документ.Лекция.Присутствующие.(Группа, Студент Как ФИО, Присутствие)

 Результат – таблица след. вида

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Лекция | Присутствующие |   | Группа | ФИО | Присутствие |
| Лекция 1 | ТаблицаЗначений | А-57 | Иванов | Нет |
| … |  | … |  |  |

 Если в запросе используется объединение, и в одной из частей присутствует вложенная таблица, а в других – нет, необходимо дополнить список выбора полями – пустыми вложенными таблицами. Это делается при помощи конструкции ПУСТАЯТАБЛИЦА.

 Пример.

Выбрать

Ссылка.Номер, ПУСТАЯТАБЛИЦА.(Ном, Тов, Кол) Как Состав

Из

Документ.РасходнаяНакладная

Объединить Все

Выбрать

Ссылка.Номер, Состав.(НомерСтроки, Номенклатура, Количество)

Из

Документ.РасходнаяНакладная

 ***Описание источников запроса***

 Конструкция ИЗ предназначена для описания источников данных для запроса. Эта конструкция может быть опущена, если все источники данных идентифицированы при объявлении полей выборки.

 ИЗ <СписокИсточников>

<СписокИсточников>:

<Источник>, <Источник>,…

<Источник>:

<Описание источника> [<Перечень соединений>]

<Описание источника>:

<Таблица> [КАК <Псевдоним таблицы>]

<Таблица>:

<ИмяТаблицы> [(Параметры)] | (<ОписаниеЗапроса>)

Если в качестве источника данных выступает таблица информационной базы, описание источника содержит имя таблицы. Если источником является виртуальная таблица, могут быть указаны параметры ее формирования. Все параметры подробно описаны в разделе Встроенный язык – Работа с запросами – Таблицы запросов синтакс-помошника, встроенного в систему.

 Для нескольких источников могут быть указаны правила соединений, которые задают порядок объединения данных этих таблиц.

 Рассмотрим пример. При выборе из 2 источников в запросе для каждой записи первого источника формируется комбинация записей из второго источника.

Выбрать

Контрагенты.Ссылка Как Контрагент,

ТипыЦен.Ссылка Как ТипыЦен

Из

Справочник.Контрагенты Как Контрагенты

Справочник.ТипыЦен

Результат – таблица, в которой для каждого контрагента перечислены все типы цен.

|  |  |
| --- | --- |
| Контрагент | ТипыЦен |
| Вист-Дон | Оптовая |
| Вист-Дон | Закупочная |
| Вист-Дон | Розничная |
| Санрайз | Оптовая |
| Санрайз | Закупочная |
| Санрайз | Розничная |
| … |  |

 Обычно такой результат смысла не имеет и требуется указать ряд дополнительных условий, задающих какие из записей нужно включить в результат запроса. Один из вариантов ограничения количества записей является задание правил соединения таблиц.

ВНУТРЕННЕЕ СОЕДИНЕНИЕ <ОписИсточн> ПО <УсловиеСоед>

ЛЕВОЕ ВНЕШНЕЕ СОЕДИНЕНИЕ

ПРАВОЕ ВНЕШНЕЕ СОЕДИНЕНИЕ

ПОЛНОЕ ВНЕШНЕЕ СОЕДИНЕНИЕ

<ОписИсточн> – ранее рассмотренное описание исходной таблицы.

<УсловиеСоед> – условие, в соответствии с которым необходимо соединить данные двух таблиц.

Соединяемые источники не равнозначны между собой и результат иногда зависит от того, какая таблица указана первой (до кл. слова СОЕДИНЕНИЕ), а какая второй (после этого слова).

 Для иллюстрации соединений используем 2 простые таблицы.

Компании

|  |  |
| --- | --- |
| Наименование | Телефон |
| Вист-Дон |  |
| Санрайз | 4-14-50-12 |

Контакты

|  |  |  |
| --- | --- | --- |
| Наименование | Телефон | Компания |
| Иванов Иван Иванович | 223-322 |  |
| Петров Петр Петрович |  | Санрайз |
| Сидоров Сидор Сидорович | 4-14-50-12 | Вист-Дон |

 Внутреннее соединение означает, что из обеих исходных таблиц в результат запроса необходимо включить только те записи, которые соответствуют заданному условию соединения. Остальные записи в результат не попадают.

 Пример.

Выбрать

Контакты.Наименование Как Контакт

Компании.Наименование Как Компания,

Из

Справочник.Контакты Как Контакты

 Внутреннее Соединение Справочник.Компании Как Компании

 По Контакты.Компания=Компании.Ссылка

 Результат

|  |  |
| --- | --- |
| Контакт | Компания |
| Петров Петр Петрович | Санрайз |
| Сидоров Сидор Сидорович | Вист-Дон |

 Левое внешнее соединение означает, что в результат запроса надо включить комбинации записей из обеих исходных таблиц, которые соответствуют условию соединения. Но в результат также следует включить еще и записи из первого источника данных, для которых не найдено соответствующих условию записей из второго источника.

 Т.е. в результат попадут все записи из первого источника, которые будут соединены с записями из второго источника при совпадении заданного условия соединения.

 Пример

Выбрать

Контакты.Наименование Как Контакт

Компании.Наименование Как Компания,

Из

Справочник.Контакты Как Контакты

 Левое Соединение Справочник.Компании Как Компании

 По Контакты.Компания=Компании.Ссылка

 Результат –

|  |  |
| --- | --- |
| Контакт | Компания |
| Иванов Иван Иванович | NULL |
| Петров Петр Петрович | Санрайз |
| Сидоров Сидор Сидорович | Вист-Дон |

 Правое внешнее соединение означает, что в результат запроса надо включить комбинации записей из обеих исходных таблиц, которые соответствуют условию соединения. Но в результат также следует включить еще и записи из второго источника данных, для которых не найдено соответствующих условию записей из первого источника.

 Т.е. в результат попадут все записи из второго источника, которые будут соединены с записями из первого источника при совпадении заданного условия соединения.

 Пример.

Выбрать

Контакты.Наименование Как Контакт

Компании.Наименование Как Компания,

Из

Справочник.Контакты Как Контакты

 Правое Соединение Справочник.Компании Как Компании

 По Контакты.Телефон=Компании.Телефон

 Результат –

|  |  |
| --- | --- |
| Контакт | Компания |
| Сидоров Сидор Сидорович | Санрайз |
| NULL | Вист-Дон |

 Полное внешнее соединение означает, что в результат запроса надо включить комбинации записей из обеих исходных таблиц, которые соответствуют условию соединения. Кроме того в результат также следует включить еще и записи из обеих таблиц, для которых не найдено соответствий.

 Т.е. в результат попадут все записи из обеих источников. Они будут соединены друг с другом при выполнении условия соединения.

Выбрать

Контакты.Наименование Как Контакт

Компании.Наименование Как Компания,

Из

Справочник.Компании Как Компании

 Полное Соединение Справочник.Контакты Как Контакты

 По Контакты.Телефон=Компании.Телефон

 Результат

|  |  |
| --- | --- |
| Контакт | Компания |
| NULL | Вист-Дон |
| Сидоров Сидор Сидорович | Санрайз |
| Иванов Иван Иванович | NULL |
| Петров Петр Петрович | NULL |

 Если источнику данных задать псевдоним, то в дальнейшем к этому источнику можно будет обращаться только через этот псевдоним (и никак иначе). Такое обращение является наглядным, а в некоторых случаях и единственно возможным.

 Пример

Выбрать

Товар.Наименование,

Товар.Родитель

Из

Справочник.Номенклатура Как Товар

 В списке источников могут фигурировать вложенные таблицы – табличные части справочников и документов.

 Пример.

Выбрать

ЛекцияПрисут.Ссылка Как Лекция,

ЛекцияПрисут.Группа,

ЛекцияПрисут.Студент,

ЛекцияПрисут.Присутствие

Из

Документ.Лекция.Присутствующие Как ЛекцияПрисут

 Результат

|  |  |  |  |
| --- | --- | --- | --- |
| Лекция | Группа | Студент | Присутствие |
| Лекция 1 | А-57 | Иванов И.И | Нет |
| Лекция1 | А-57 | Петров П.П. | Да |
| … |  |  |  |
| Лекция2 | … |  |  |

 В списке источников запроса в качестве таблицы-источника может использоваться вложенный запрос. В этом случае описание источника содержит описание вложенного запроса, которое ничем не отличается от обычного. Использование вложенного запроса ничем не отличается от использования таблицы информационной базы. В качестве полей такого источника доступны все поля выборки вложенного запроса.

 Пример.

Выбрать

ВложенныйЗапрос.Сссылка Как Лекция

ВложенныйЗапрос.Группа,

ВложенныйЗапрос.Студент,

ВложенныйЗапрос.Присутствие

Из

 (Выбрать

 ЛекцияПрисут.Ссылка,

 ЛекцияПрисут.Группа,

 ЛекцияПрисут.Студент,

 ЛекцияПрисут.Присутствие

 Из

 Документ.Лекция.Присутствующие Как ЛекцияПрисут) Как ВложенныйЗапрос

 Результат этого запроса полностью совпадает с результатом предыдущего.

 ***Фильтрация результатов запроса***

 Конструкция ГДЕ позволяет задать условие отбора данных из таблиц-источников запроса. Запрос будет обрабатывать только записи, удовлетворяющие заданному условию.

 При записи условия допускается использовать логические операции И, ИЛИ, НЕ.

 Пример. Выбрать все предметы из справочника предметов, по которым итогом является экзамен.

ВЫБРАТЬ

 Предметы.Наименование

ИЗ

 Справочник.Предметы КАК Предметы

ГДЕ

 Предметы.Итог=Значение(Перечисление.ИтогПоПредмету.Экзамен)

 В условии отбора допускается использовать встроенные функции языка запросов: Подстрока, Год, Квартал, Месяц, ДеньГода, День, Неделя, ДеньНедели, Час, Минута, Секунда, НачалоПериода, КонецПериода, ДобавитьКДате, РазностьДат, ТипЗначения, Представление, ЕстьNULL. Описание функций – в синтакс-помошнике.

 Эти же функции можно использовать в списке полей выборки.

***Группировка результатов запроса***

 Результат запроса может быть сгруппирован с помощью агрегатных функций, указанных в списке полей выборки. Строки результата запроса будут содержать значения вычисленных агрегатных функций, рассчитанные для записей исходных таблиц с одинаковым значением поля, по которому ведется группировка.

 Список полей, по которым ведется группировка указывается в конструкции

СГРУППИРОВАТЬ ПО <Поле1>[,<Поле2>,…]

 Пример. Выборка статистики по продаже товаров: максимальная, минимальная и средняя цена по расходным накладным

Выбрать

 Накладная.Номенклатура,

 Среднее (Накладная.Цена) Как Среднее,

 Максимум (Накладная.Цена) Как Максимум,

 Минимум (Накладная.Цена) Как Минимум

Из

 Документ.РасходнаяНакладная.Состав Как Накладная

Сгруппировать По

 Накладная.Номенклатура

Результат.

|  |  |  |  |
| --- | --- | --- | --- |
| Номенклатура | Среднее | Максимум | Минимум |
| Товар1 | Ср.цена1 | Макс.цена1 | Мин.цена1 |
| … |  |  |  |

 Если в запросе присутствуют агрегатные функции, а конструкция Сгруппировать По отсутствует, то все записи группируются в единственную строку.

 В качестве агрегатных функций используются 5 функций.

Сумма (ЧисловоеПоле) – сумма значений по полю.

Среднее (ЧисловоеПоле) – среднее значений числового поля.

Минимум (Поле) – минимальное значение поля (в т.ч. и нечислового).

Максимум (Поле) – максимальное значение поля (в т.ч. и нечислового).

Количество( \* | [Различные] Поле) – количество значений поля. \* – общее количество, Различные – количество неповторяющихся значений.

 Пример.

Выбрать

 Количество (\*) Как Всего,

 Количество (Различные Накладная.Номенклатура) Как Разных

Из

 Документ.РасходнаяНакладная.Состав Как Накладная

 Результат:

|  |  |
| --- | --- |
| Всего | Разных |
| 215 | 87 |

 На значение агрегатных функций могут быть наложены условия отбора. Эти условия допускаются только в специальной конструкции

ИМЕЮЩИЕ УсловиеОтбора

 Пример. Выборка товаров, которых продали больше 20 штук.

Выбрать

 Накладная.Номенклатура,

 Сумма(Накладная.Количество) Как Количество

Из

 Документ.РасходнаяНакладная.Состав Как Накладная

Сгруппировать По

 Накладная.Номенклатура

Имеющие

 Сумма(Накладная.Количество)>20

***Объединение запросов***

 Язык запросов позволяет объединить результаты нескольких запросов в одну таблицу. Главным требованием объединения запросов является одинаковый состав полей. Имена полей берутся из первого запроса.

 Объединение выполняется при помощи конструкции

<Описание запроса>

Объединить [Все]

 <Описание запроса>

Объединить [Все]

 <Описание запроса>

 При объединении одинаковые строки из 2 таблиц попадают в конечную таблицу как 1 строка. ВСЕ – включать в конечную таблицу все одинаковые строки.

 Пример. Построить таблицу прихода и расхода товаров по 2 документам: ПриходнаяНакладная и РасходнаяНакладная.

Выбрать

 Приход.Номенклатура Как Товар,

Сумма (Приход.Количество) Как Поступило,

Сумма (0) Как Отпущено

Из

 Документ.ПриходнаяНакладная.Состав Как Приход

Сгруппировать По

 Приход.Номенклатура

Объединить

Выбрать

 Расход.Номенклатура,

 Сумма(0),

 Сумма(Расход.Количество)

Из

 Документ.РасходнаяНакладная.Состав Как Расход

Сгруппировать По

 Расход.Номенклатура

Результат–

|  |  |  |
| --- | --- | --- |
| Товар | Поступило | Отпущено |
| Товар1 | 55 | 32 |
| … |  |  |

 ***Упорядочивание результатов запроса***

 Для упорядочивания таблицы-результата запроса предназначена конструкция

УПОРЯДОЧИТЬ ПО

<Выражение> ВОЗР | УБЫВ | ИЕРАРХИЯ | ИЕРАРХИЯ УБЫВ, …

 <Выражение> – условие упорядочивания, которое является либо ссылкой на некоторое поле, либо является выражением с использованием 5 агрегатных функций. Выражение упорядочивания может не присутствовать в списке выбора. Для справочников можно использовать упорядочивание по иерархии справочника.

 Пример.

Выбрать

 Справочник.Номенклатура.Наименование

Упорядочить По

 Справочник.Номенклатура.Наименование Иерархия

Результат.

Процессоры

 Intel

 Core 2 Duo 8500

 …

 AMD

 Phenom X4

 …

Память

 …

 ***Расчет итогов запроса***

 При выполнении запроса можно вычислить итоги и вставить их в таблицу-результат как отдельную строку. Итоги бывают 2 видов: общий итог по всей таблице и итоги по контрольным точкам (некоторым выражениям). Итоги по контрольным точкам вычисляются для каждого значения заданного выражения.

ИТОГИ

 <Агрегатная функция> | <Выражение> [КАК <Псевдоним поля>],

 …

ПО [ОБЩИЕ]

<Выражение>

[[ТОЛЬКО] ИЕРАРХИЯ] | ПЕРИОДАМИ(Секунда | Минута | Час | День | Неделя | Месяц | Квартал | Год | Декада | Полугодие, НачальнаяДата, КонечнаяДата),

…

 Агрегатные функции – те, которые нужно рассчитать в итогах.

Общие – задает формирование итоговой строки по всей таблице.

Конструкция ПО задает перечень контрольных точек для расчета итогов, указанных через запятую. Если контрольная точка ссылается на иерархический справочник, итоги можно рассчитать по иерархии (ключевое слово ИЕРАРХИЯ).

 Пример.

ВЫБРАТЬ

 Документ.Номенклатура КАК Номенклатура,

 Документ.Количество КАК Количество,

 Документ.Ссылка.Номер,

 Документ.Ссылка.Контрагент

ИЗ

 Документ.РасходнаяНакладная.Состав КАК Документ

УПОРЯДОЧИТЬ ПО

 Документ.Номенклатура

ИТОГИ

 Сумма(Количество)

ПО

 Номенклатура ИЕРАРХИЯ

Результат.

|  |  |  |  |
| --- | --- | --- | --- |
| Номенклатура | Количество | Номер | Контрагент |
| Ноутбуки | 50 |  |  |
|  Acer | 15 |  |  |
|  Acer Aspire.. | 2 | 00001 | Вист-Дон |
|  Acer Aspire.. | 3 | 00002 | Санрайз |
|  Samsung | 12 |  |  |
| … |  |  |  |

 Если использовать конструкцию ТОЛЬКО ИЕРАРХИЯ, будут вычисляться итоги только по иерархии без расчета итогов в контрольных точках.

 Пример. Для предыдущего запроса используем ТОЛЬКО ИЕРАРХИЯ. Результат –

|  |  |  |  |
| --- | --- | --- | --- |
| Номенклатура | Количество | Номер | Контрагент |
| Ноутбуки | 50 |  |  |
|  Acer | 15 |  |  |
|  Samsung | 12 |  |  |
|  Lenovo | 23 |  |  |
| Мониторы | 78 |  |  |
| … |  |  |  |

 Конструкция ПЕРИОДАМИ используется тогда, когда итоги рассчитываются по полю типа Дата и в результат запроса требуется включить все даты (или временные интервалы) заданного периода. Такое действие называется дополнением дат.

 По умолчанию запрос собирает данные только по тем периодам, для которых в информационной базе имеются ненулевые значения. Однако в некоторых случаях требуется получить данные по всем датам (периодам), как с ненулевыми, так и с нулевыми значениями. Например, в целях составления отчета об активности покупателей по дням (месяцам, часам и т.п.). В таких случаях и используется дополнение дат.

 Если в конструкции не указать начальную и конечную даты, будут использованы первая и последняя даты из выборки данных, обрабатываемых запросом.

 Пример. Составить запрос, который вычисляет количество продаж по часам заданного дня с группировкой по клиентам.

Выбрать

 РасходнаяНакладная .Контрагент,

 НачалоПериода(РасходнаяНакладная.Дата, Час) Как Период,

 Количество(РасходнаяНакладная.Ссылка) Как КоличествоПродаж

Из

 Документ.РасходнаяНакладная Как РасходнаяНакладная

Сгруппировать По

 РасходнаяНакладная.Контрагент,

 НачалоПериода(РасходнаяНакладная.Дата, Час)

Итоги

 Сумма(КоличествоПродаж)

По

 Период Периодами(Час, ДатаВремя(2011,6,28), ДатаВремя(2011,6,28))

Результат.

|  |  |  |
| --- | --- | --- |
| Контрагент | Период | КоличествоПродаж |
|  | 28.06.2011 00:00 | 0 |
|  | 28.06.2011 01:00 | 0 |
|  | … |  |
| Вист-Дон | 28.06.2011 08:00 | 4 |
| Санрайз | 28.06.2011 08:00 | 6 |
|  | 28.06.2011 09:00 | 25 |
| Димир | 28.06.2011 09:00 | 12 |
| … |  |  |

 Ключевое слово ОБЩИЕ позволяет добавить в таблицу общий итог, вычисленный по всем записям таблицы.

 Пример. Добавим в предыдущий запрос Общие. В таблицу будет добавлена итоговая строка в начале таблицы.

|  |  |  |
| --- | --- | --- |
| Контрагент | Период | КоличествоПродаж |
|  |  | 175 |
|  | 28.06.2011 00:00 | 0 |
| … |  |  |

 Если итоги используются совместно с группировкой и для итогов не указан список агрегатных функций, он будет автоматически формироваться из агрегатных функций группировок, заданных в списке полей выборки. При объединении запросов список агрегатных функций берется из первого запроса.

 Пример. Сгруппировать данные по продажам по номенклатуре и покупателям и вычислить общее количество продаж по номенклатуре и покупателям.

Выбрать

 Документ.Номенклатура Как Номенклатура,

 Документ.Ссылка.Контрагент Как Контрагент,

 Сумма(Документ.Количество) Как Количество

Из

 Документ.РасходнаяНакладная.Состав Как Документ

Сгруппировать По

 Документ.Номенклатура,

 Документ.Ссылка.Контрагент

Итоги

По

 Номенклатура,

 Контрагент

Результат.

|  |  |  |
| --- | --- | --- |
| Номенклатура | Контрагент | Количество |
| Ноутбук Acer Aspire-2534 |  | 52 |
| Ноутбук Acer Aspire-2534 | Вист-Дон | 22 |
| Ноутбук Acer Aspire-2534 | Санрайз | 23 |
| Ноутбук Acer Aspire-2534 | Мир ПК | 7 |
| Ноутбук Samsung GT-1201 |  | 76 |
| Ноутбук Samsung GT-1201 | Магистр | 21 |
| Ноутбук Samsung GT-1201 | Бакалавр | 22 |
| Ноутбук Samsung GT-1201 | Инженер | 23 |
| Ноутбук Samsung GT-1201 | Мир ПК | 10 |
| … |  |  |

***Параметры запроса***

 В тексте запроса в качестве параметра можно использовать либо константу, либо значение внешней переменной.

 В качестве констант можно использовать значения типов Булево, Число, Строка, Дата, НЕОПРЕДЕЛЕНО и NULL.

 При этом значения типа Дата задаются при помощи конструкции:

ДатаВремя(ГГГ,ММ,ДД,ЧЧ,ММ,СС).

 Значение внешней переменной указывается с использованием амперсанда: &.

 Пример. Составить запрос, который вычисляет количество продаж по часам введенного пользователем дня с группировкой по клиентам.

Выбрать

 РасходнаяНакладная .Контрагент,

 НачалоПериода(РасходнаяНакладная.Дата, Час) Как Период,

 Количество(РасходнаяНакладная.Ссылка) Как КоличествоПродаж

Из

 Документ.РасходнаяНакладная Как РасходнаяНакладная

Сгруппировать По

 РасходнаяНакладная.Контрагент,

 НачалоПериода(РасходнаяНакладная.Дата, Час)

Итоги

 Сумма(КоличествоПродаж)

По

 Период Периодами(Час, &ПараметрДата, &ПараметрДата)

## Обработка результатов запроса

 Выполнение и обработка результатов запроса производится в 4 этапа.

1. Создание объекта типа Запрос, в котором задается текст запроса.

2. Выполнение запроса, формирование таблицы данных и получение ссылки на таблицу-результат запроса.

3. Открытие выборки строк таблицы с заданием порядка обхода таблицы.

4. Перебор и обработка строк таблицы в соответствии с заданным порядком обхода.

***Создание и выполнение запроса***

 Создание объекта типа Запрос выполняется следующей конструкцией.

Запрос = Новый Запрос(“Выбрать … текст запроса …

 | Продолжение текста запроса

 | …

 | Завершение текста запроса”);

 Выполнение запроса и получение ссылки на таблицу результат:

РезультатЗапроса = Запрос.Выполнить();

 Если текст запроса оказался неверным, возникает исключительная ситуация и формируется ее описание, в котором указывается номер ошибочной строки. Эту информацию можно получить в обработчике исключения функциями ИнформацияОбОшибке и ОписаниеОшибки.

***Выборка результатов запроса***

 Выполняется при помощи метода

Выбрать(ПорядокОбходаТаблицыЗапроса)

 Для иллюстрации способов обхода таблицы используем запрос с иерархическими результатами количества продаж.

Запрос = Новый.Запрос(“

| Выбрать

| РасходнаяНакладнаяСостав.Номенклатура Как Номенклатура,

| РасходнаяНакладнаяСостав.Количество Как Количество

| Из

| Документ.РасходнаяНакладная.Состав Как РасходнаяНакладнаяСостав

| Упорядочить По

| РасходнаяНакладнаяСостав.Номенклатура

| Итоги

| Сумма(Количество)

| По

| Номенклатура Иерархия”);

РезультатЗапроса = Запрос.Выполнить();

 Результат ­ – таблица. Номера не присутствуют в таблице, а заданы в целях демонстрации.

|  |  |  |
| --- | --- | --- |
| № | Номенклатура | Количество |
| 1 | Ноутбуки | 150 |
| 2 |  ASUS | 55 |
| 3 |  ASUS-1 | 12 |
| 4 |  ASUS-2 | 15 |
| 5 |  ASUS-3 | 7 |
| 6 |  ASUS-4 | 21 |
| 7 |  Samsung | 95 |
| 8 |  Samsung-1 | 25 |
| 9 |  Samsung-2 | 70 |
| 10 | Процессоры | 220 |
| 11 |  AMD | 80 |
| 12 |  AMD | 15 |
| 13 |  AMD | 20 |
| 14 |  AMD | 35 |
| 15 |  AMD | 10 |
| 16 |  Intel | 140 |
| 17 |  Intel | 40 |
| 18 |  Intel | 80 |
| 19 |  Intel | 20 |

**Линейный порядок обхода**

 При линейном порядке записи таблицы перебираются в соответствии с их порядком в таблице. Для этого метод Выбрать вызывается без параметров:

РезультатЗапроса.Выбрать() или с параметром

РезультатЗапроса.Выбрать(ОбходРезультатаЗапрса.Прямой)

 При данном порядке обход выполняется по номерам строк 1, 2, …19.

**Иерархический обход**

 При таком обходе перебираются только записи, находящиеся на одном уровне иерархии:

РезультатЗапроса.Выбрать(ОбходРезультатаЗапрса.ПоГруппировкамСИерархией)

 В таблице-примере такой обход задает обход только записей 1 и 10, т.к. только они находятся на одном уровне. Для обхода записей, находящихся на более низшем уровне иерархии, требуется снова открыть выборку. Такая выборка для строки 1 будет перебирать строки 2 и 7, а для строки 10 – 11 и 16. Соответственно для строк 2,7,11 и 16 также можно открыть выборку, которая будет перебирать подчиненные им по иерархии строки.

 Пример. Полный обход таблицы

Обход= ОбходРезультатаЗапрса.ПоГруппировкамСИерархией;

Выборка1=РезультатЗапроса.Выбрать(Обход);

Пока Выборка1.Следующий() Цикл

// Обработка строки 1-го уровня иерархии

Сообщить(“Товар”+Выборка.Наименование+” Количество”+Выборка.Количество);

//…

// Открываем обход 2-го уровня иерархии

 Выборка2=РезультатЗапроса.Выбрать(Обход);

 Пока Выборка2.Следующий() Цикл

 // Обработка строки 2-го уровня иерархии

 Сообщить(“Товар”+Выборка.Наименование+” Количество”+Выборка.Количество);

 //…

 // Открываем обход 3-го уровня иерархии

 Выборка3=РезультатЗапроса.Выбрать(Обход);

 Пока Выборка3.Следующий() Цикл

 // Обработка строки 3-го уровня иерархии

 Сообщить(“Товар”+Выборка.Наименование+” Количество”+Выборка.Количество);

 // …

 КонецЦикла

 КонецЦикла

КонецЦикла;

**Обход по группам**

Этот обход перебирает только записи, соответствующие группам иерархии. Выполняется методом

РезультатЗапроса.Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам)

 Данный обход будет перебирать записи с номерами 1,2,7,10,11,16. Для перебора подчиненным им по иерархии записям используется еще одна выборка с прямым порядком обхода.

 Пример. Перебор всей таблицы.

Обход= ОбходРезультатаЗапрса.ПоГруппировкам;

Выборка1=РезультатЗапроса.Выбрать(Обход);

Пока Выборка1.Следующий() Цикл

// Обработка строки группы

Сообщить(“Товар”+Выборка.Наименование+” Количество”+Выборка.Количество);

//…

// Открываем прямой обход подчиненных по иерархии записей

 Выборка2=РезультатЗапроса.Выбрать();

 Пока Выборка2.Следующий() Цикл

 // Обработка строк внутри группы

 Сообщить(“Товар”+Выборка.Наименование+” Количество”+Выборка.Количество);

 //…

 КонецЦикла

КонецЦикла;

**Методы перебора строк результата запроса**

 Для перебора записей таблицы с результатами запроса предназначены 3 метода.

1. Выборка.Следующий() – перейти к следующей записи в соответствии с порядком обхода выборки.

2. СледующийПоЗначениюПоля() – получить следующую записьсо значением в заданном поле, отличающимся от значения этого поля в текущей записи.

Пример.

Запрос = Новый Запрос(“

| Выбрать

| Док.Номенклатура,

| Док.Ссылка.Контрагент Как Контрагент,

| Док.Количество

| Из

| Документ.РасходнаяНакладная.Состав Как Док

| УпорядочитьПо

| Док.Номенклатура.Наименование,

| Контрагент”);

РезультатЗапроса=Запрос.Выполнить();

|  |  |  |  |
| --- | --- | --- | --- |
| № | Номенклатура | Контрагент | Количество |
| 1 | Ноутбук Acer Aspire-2534 | Магистр | 21 |
| 2 | Ноутбук Acer Aspire-2534 | Вист-Дон | 22 |
| 3 | Ноутбук Acer Aspire-2535 | Санрайз | 23 |
| 4 | Ноутбук Acer Aspire-2535 | Мир ПК | 7 |
| 5 | Ноутбук Samsung GT-1201 | Вист-Дон | 21 |
| 6 | Ноутбук Samsung GT-1201 | Магистр | 21 |
| 7 | Ноутбук Samsung GT-1205 | Бакалавр | 22 |
| 8 | Ноутбук Samsung GT-1205 | Инженер | 23 |
| 9 | Ноутбук Samsung GT-1205 | Мир ПК | 10 |
|  | … |  |  |

Выборка=РезультатЗапроса.Выбрать();

Пока Выборка.СледующийПоЗначениюПоля(“Номенклатура”) Цикл

//Здесь выбираются записи 1,3,5,7

 Пока Выборка.СледующийПоЗначениюПоля(“Контрагент”) Цикл

 // Здесь выбираются последовательно

 // 1,2

 // 3,4

 // 5,6

 // 7,8,9

 КонецЦикла

КонецЦикла

3. НайтиСледующий(Структура) – найти следующую строку с заданными в структуре значениями полей таблицы запроса.

**Обращение к полям таблицы**

 Обращение выполняется при помощи ссылки на строку таблицы, полученную при обходе выборке. В качестве атрибутов таблицы выступают поля этой таблицы. В качестве полей выступают псевдонимы полей выборки или имена по умолчанию (при отсутствии псевдонимов).

Пока Выборка.Следующий() Цикл

 // Обработка Выборка.ПолеТаблицы

 …

КонецЦикла