

Лабораторная работа № 4. Учетные объекты. Запросы.

Цель лабораторной работы № 4: освоение процесса создания запросов, работа с учетными объектами 1С.

Задачи работы – приобретение навыков:

- создания учетных объектов;
- создания запросов.

Учетные объекты

В «1С:Предприятие» принята следующая модель: деятельность организации разбивается на набор операций. Для регистрации факта осуществления хозяйственной операции создается документ. Т.е. документы содержат первичную информацию о совершенной хозяйственной операции. Фактом принятия данных документа к учету является проведение документа. Таким образом, получается, что данные, принятие к учету, находятся в регистрах.

Все регистры имеют ряд общих черт:

1. Записи регистров не имеют объектной природы.
2. Одной из важных характеристик любого регистра является состав его измерений, ресурсов, реквизитов.
3. У каждого регистра есть некие «специфические умения».

Регистры накопления используются в системе для накопления информации о наличии и движении средств – товарных, денежных и других. Они позволяют очень быстро получать информацию об остатках, оборотах, остатках с оборотами на интересующие даты, за указанные периоды.

Различают два вида регистров накопления:

1. Регистры остатков.
2. Регистры оборотов.

На регистрах накопления решают задачи оперативного учета.

Ведение бухгалтерского учета в системе «1С:Предприятие» обеспечивают объекты конфигурации «Планы счетов» и «Регистры бухгалтерии». Средства системы позволяют организовывать учет по нескольким планам счетов, при этом для каждого плана счетов может строиться произвольная иерархия субсчетов большой вложенности.

Планом счетов называется совокупность синтетических счетов, предназначенных для группировки информации о хозяйственной деятельности предприятия. Информация, накапливаемая на таких синтетических счетах, позволяет получить полную картину состояния средств предприятия.

Дополнительные аналитические разрезы определяются в плане видов характеристик (так называемое «Субконто»). Максимальное количество аналитических разрезов определяется в самом плане счетов.

Для отражения в бухгалтерском учете информации о хозяйственных операциях в системе «1С:Предприятие» используются регистры бухгалтерии.

В конфигурации может быть определено несколько (или один) планов счетов, несколько регистров бухгалтерии. Но каждый регистр бухгалтерии связан только с одним планом счетов.

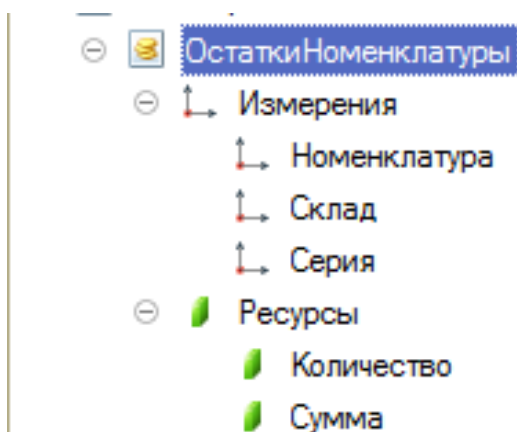
Регистр бухгалтерии может не иметь измерений, реквизитов, но обязательно у него должен быть хоть один ресурс (например, «сумма проводки»).

Для решения расчетных задач (расчет заработной платы) используются такие объекты, как планы видов расчета и регистры расчета.

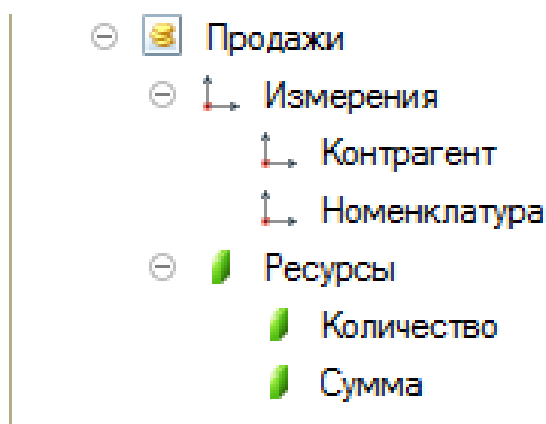
Планы видов расчета содержат перечень видов расчета.

Регистр расчета – это объект конфигурации, который позволяет организовать учет результатов вычислений, осуществляемых с некоторой периодичностью, тесно связанных друг с другом по некоторым правилам и взаимно влияющих друг на друга в пределах определенного периода.

Создайте регистр накопления «ОстаткиНоменклатуры». Вид регистра «Остатки». Состав измерений и ресурсов следующий:



Создайте регистр накопления «Продажи». Вид регистра «Обороты». Состав измерений и ресурсов следующий:



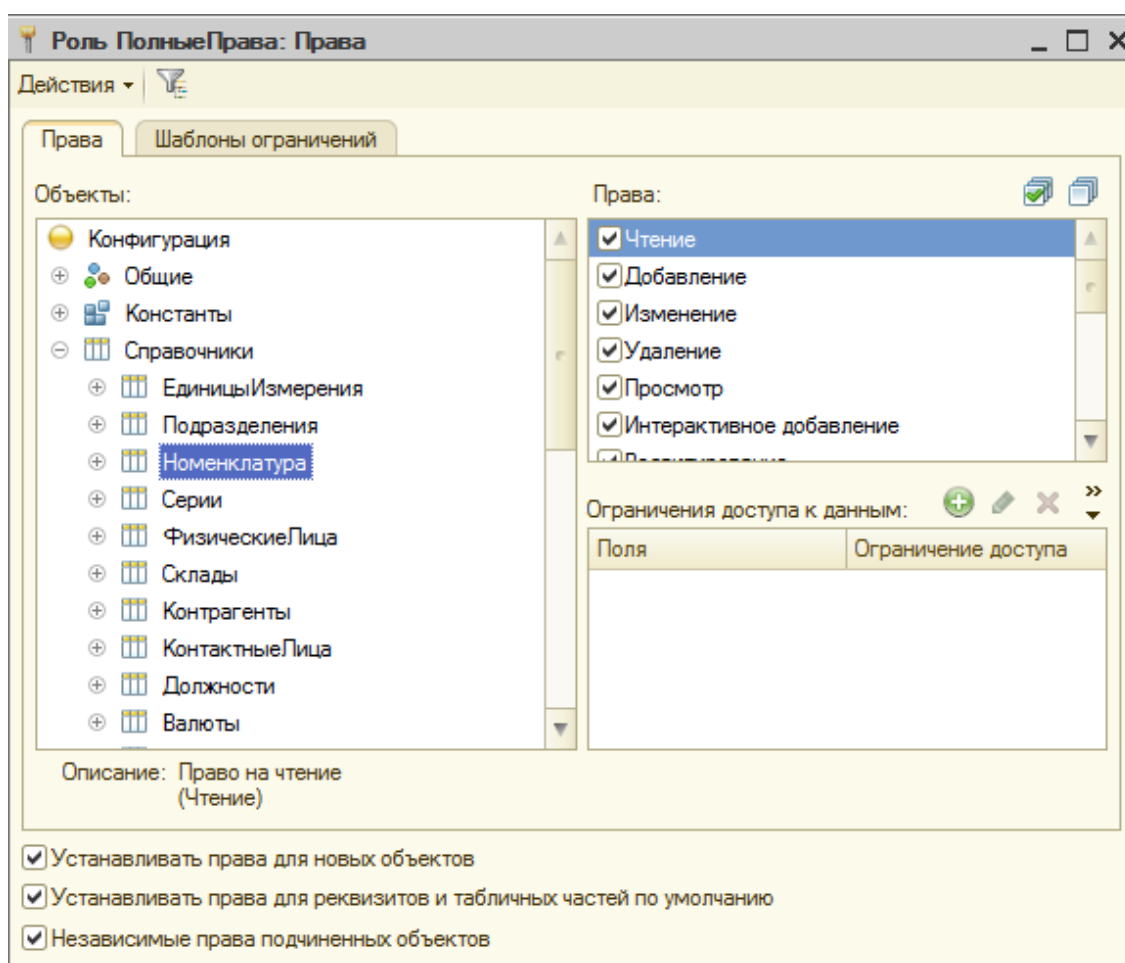
Таким же образом, как и для регистра сведений, настройте движения документов «Поступление товаров» и «Продажа товаров». Проверьте работу механизмов на практике.

Элементы администрирования

Уже создано некоторое количество объектов конфигурации. Еще раз взглянем на роли.

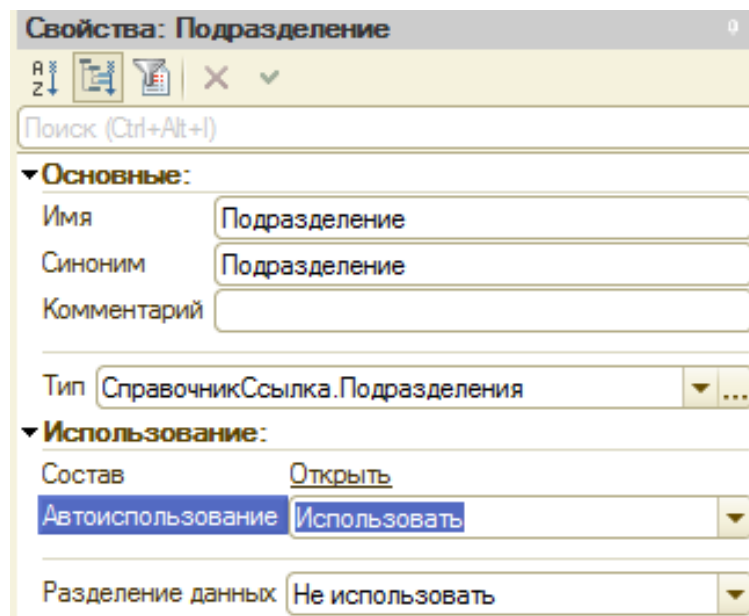
Права делятся на программные и интерактивные. С помощью ограничения доступа к данным можно определить доступ только к определенным записям таблицы (например, организовать доступ только к документам какой-либо группы контрагентов).

С точки зрения разделения доступа рассмотрим «механизм разделения данных», в основе которого лежат «общие реквизиты». Находятся эти объекты в ветви «Общие» конфигурации. С их помощью можно создавать реквизиты общие для всех объектов.



Права делятся на программные и интерактивные. С помощью ограничения доступа к данным можно определить доступ только к определенным записям таблицы.

Также существует «механизм разделения данных». В основе данного механизма лежат «общие реквизиты». Находятся эти объекты в ветви «Общие» конфигурации. Как следует из названия, с их помощью можно создавать реквизиты общие для всех объектов.



Свойства: Подразделение

Поиск (Ctrl+Alt+I)

▼ **Основные:**

Имя:

Синоним:

Комментарий:

Тип:

▼ **Использование:**

Состав:

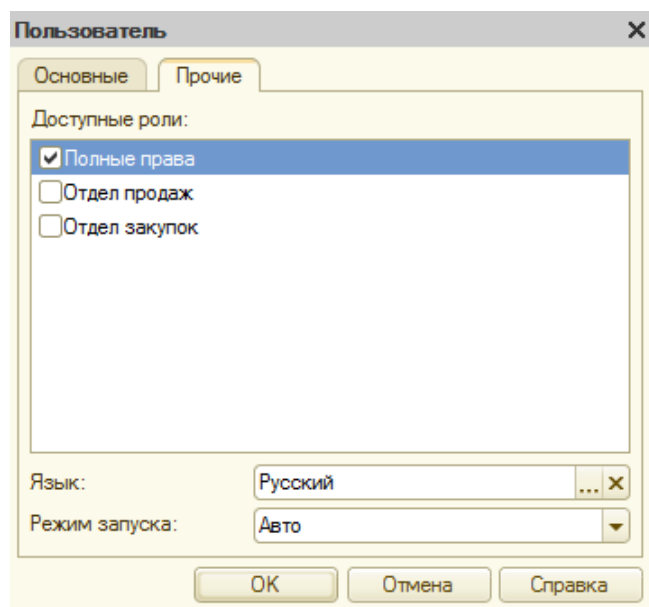
Автоиспользование:

Разделение данных:

Механизм разделения данных позволяет хранить данные нескольких независимых организаций в одной информационной базе. Т.е. в одной базе можно организовать несколько независимых информационных пространств. Они могут быть полностью изолированными или пересекаться по нужным объектам (например, может быть общий справочник валют, регистр сведений с курсами валют и т.д.).

Это становится возможным благодаря тому, что общие реквизиты объектов конфигурации можно использовать не только как «одинаковый реквизит, который есть у всех объектов», но и как идентификатор того, что данные относятся к какой-то одной из нескольких независимых областей.

После настройки ролей их нужно «раздать». Список пользователей открывается при выполнении команды «Администрирование/Пользователи». В открывшемся окне можно указать параметры авторизации и отметить доступные данному пользователю роли.



Пользователь

Основные | Прочие

Доступные роли:

- ☒ Полные права
- ☐ Отдел продаж
- ☐ Отдел закупок

Язык:

Режим запуска:

ОК | Отмена | Справка

Для получения архива всей базы можно использовать команду главного меню «Администрирование/Выгрузить информационную базу». Сформируется файл с расширением «dt». Это полная выгрузка данных, пользователей и всех настроек, которые были сделаны.

Обратная операция делается с помощью команды «Администрирование/Загрузить информационную базу».

Запросы

Таблицы в «1С:Предприятие 8» подразделяются на два класса:

1. Реальные таблицы – «хранятся» в базе данных. В случае использования реальной таблицы могут присутствовать вычисляемые поля, значения которых рассчитываются как функция нескольких разных полей.

2. Виртуальные таблицы – в базе данных не «хранятся». При обращении к информации виртуальных таблицы система автоматически «собирает» информацию из реальных таблиц для выполнения запроса. Виртуальная таблица может быть параметризована.

Отдельный подкласс таблиц образует так называемые объектные таблицы. Эти таблицы предназначены для хранения состояния объектов системы, таких как справочники, документы и т.д. В таких таблицах присутствует поле «Ссылка».

Любая таблица состоит из набора полей. В качестве поля таблицы может фигурировать:

1. Обычное поле (содержащее какое-либо значение, либо значение типа «Null»).

2. Вложенная таблица.

Основное отличие обычного поля от вложенной таблицы состоит в том, что в рамках одной записи обычному полю соответствует одно единственное значение, а вложенной таблице соответствует значение типа «РезультатЗапроса» с заранее заданным набором колонок.

Иллюстрация данного понятия.

Дата	Время	Контрагент	№	Товар	Кол-во	Сумма
			Товар			
01,01,16	11:00	ООО «Вектор»	1	Шампунь	5	100
			2	Ножницы	3	200
02,02,16	11:51	ООО «Марина»	1	Краска	10	200

Поле может содержать значение одного типа, может содержать значения нескольких типов, при этом для конкретной задачи поле содержит значение одного типа.

Структура запроса

Для выполнения запроса (получения выборки данных) необходимо составить текст запроса. Текст запроса – это конструкция, в соответствии с которой должен быть выполнен запрос.

После составления текста запроса его необходимо выполнить. По окончании выполнения полученный результат необходимо «обойти» (разобрать результат запроса).

Для формирования текста запроса существует специализированный язык запросов. Он определяет используемые синтаксические конструкции, структуру запроса.

Текст запроса состоит из следующих секций:

1. Описание запроса.
2. Объединение запросов.
3. Упорядочивание результатов или Автоупорядочивание.
4. Описание итогов.

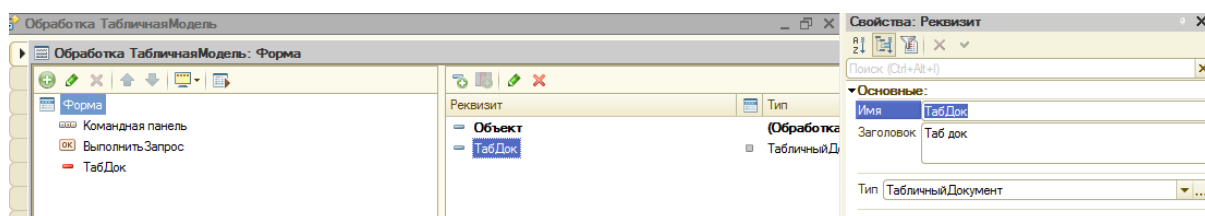
Из всех секций обязательно наличие только описания запроса. Данная секция имеет следующую структуру:

Выбрать [Различные] [Первые <Количество>
<Список полей выборки>
[Из <Список источников>]
[Где <Условие отбора>]
[Сгруппировать По <Поля группировки>]
[Имеющие <Условия отбора>]
[Для измерения [[Of]<Список таблиц верхнего уровня>]]

В рамках данной секции обязательно только наличие «Выбрать» и указание полей выборки. Все остальные структурные элементы могут опускаться.

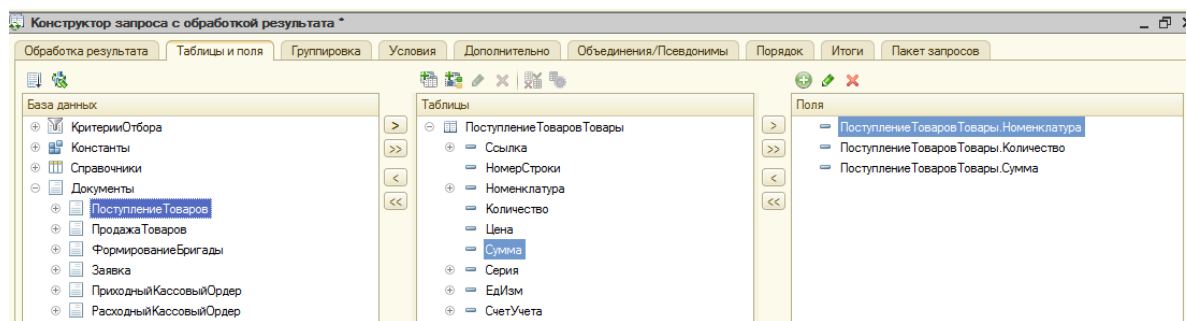
Использование конструктора запросов

Вернемся к обработке «Табличная модель». Добавим реквизит с именем «ТабДок» и типом «Табличный документ». «Перетащим» этот реквизит в дерево элементов формы.

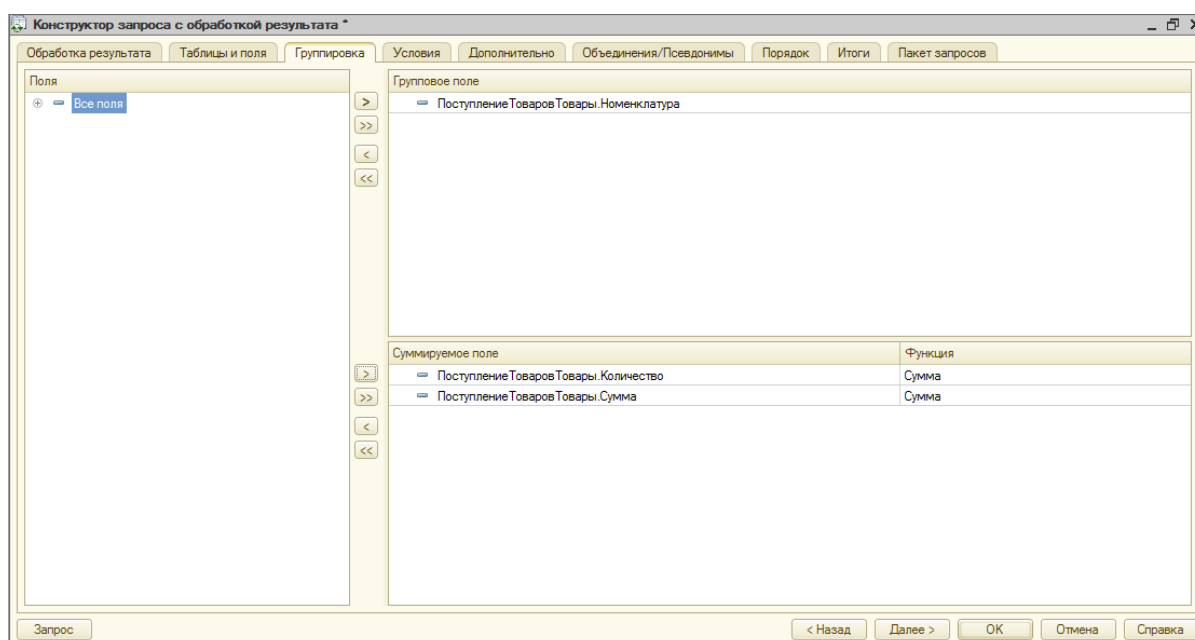


Для знакомства с механизмом запросов будем использовать конструктор запросов с обработкой результата, но тип обработки следует выбрать «Вывод в табличный документ».

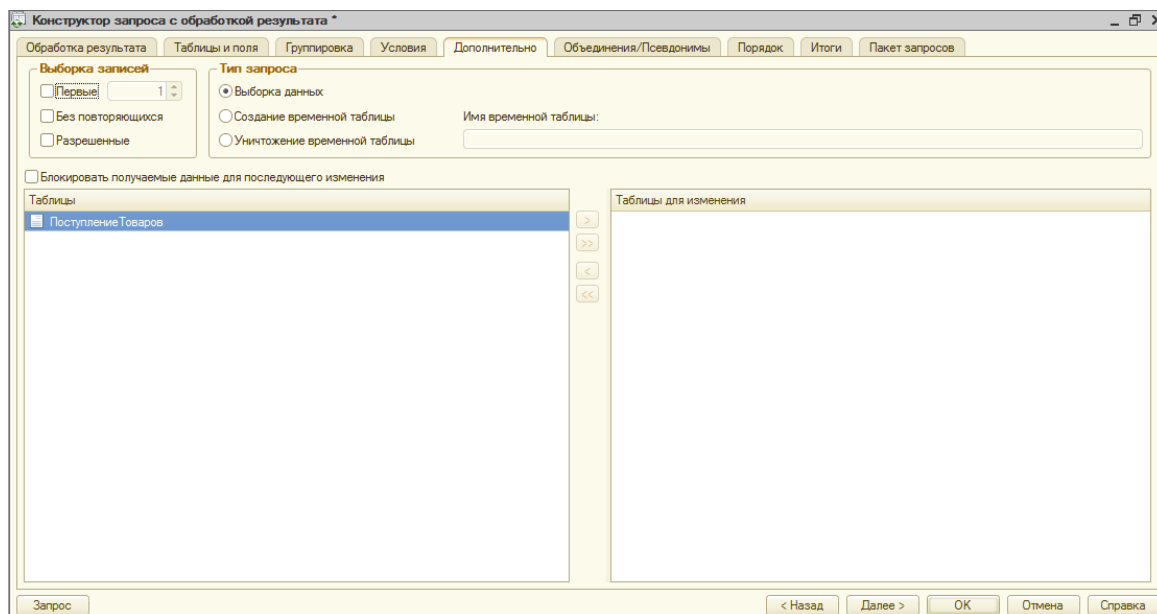
В качестве источника данных выберем табличную часть «Товары» документа «ПоступлениеТоваров».



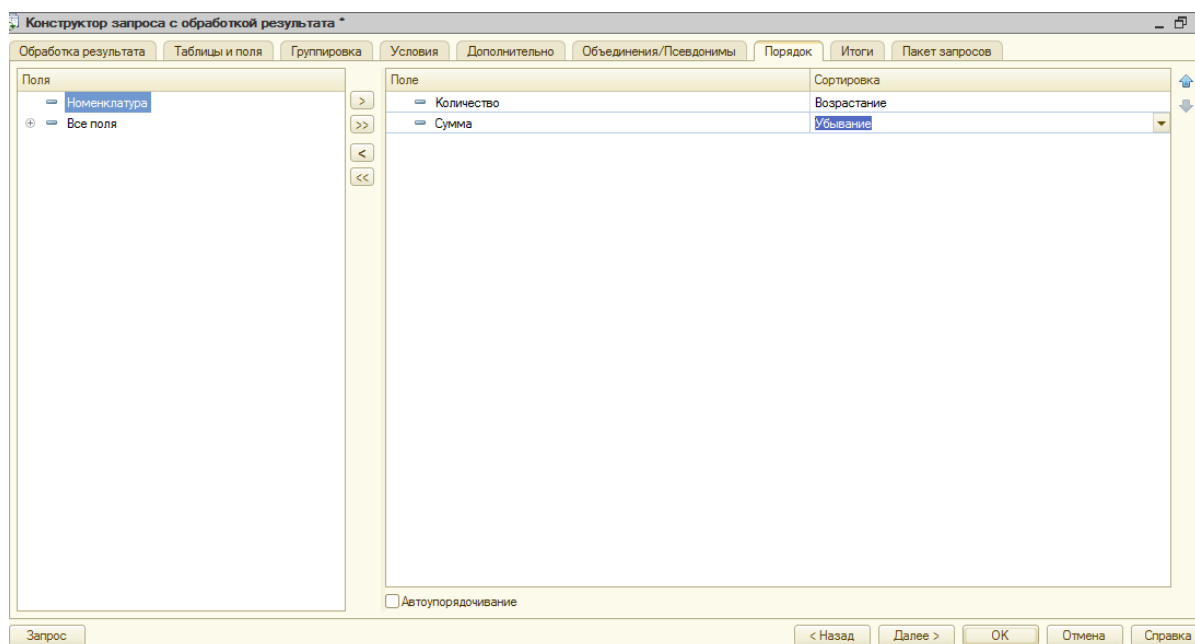
Группировки определим следующим образом



На закладке «Дополнительно» можно отметить ряд флагов (касающихся ключевого слова «Выбрать» языка запросов) и определить состав таблиц, предназначенных для изменения.

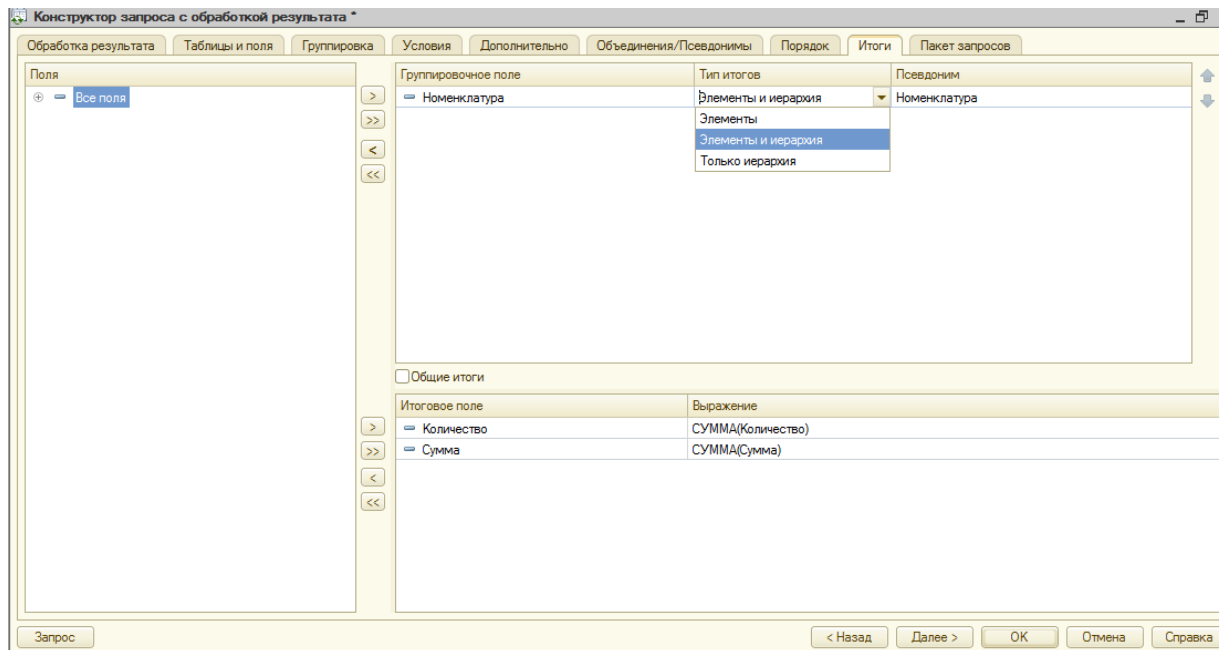


На закладке «Порядок» можно определить порядок сортировки записей в результате запроса.



Флаг «Автоупорядочивание» может устанавливаться для упорядочивания по полям ссылочного типа.

При определении раздела «Итоги» нужно учитывать, что в результате запроса появятся «дополнительные» итоговые записи. Вместе с этими записями результат запроса становится иерархическим.



Допустимо указание нескольких типов итогов:

1. Элементы (в выборке результата запроса присутствуют итоги по группировкам и детальные записи).
2. Иерархия (в выборке результата запроса в общем случае присутствуют итоговые записи по иерархии, итоговые записи по группировке, детальные записи).
3. Только иерархия (в выборке результата запроса в общем случае присутствуют итоговые записи по иерархии, детальные записи).

После нажатия на кнопку «ОК» конструктора, в модуле формы будет сформирован программный код следующего вида.

&НаКлиенте

Процедура ВыполнитьЗапрос(Команда)

 ВыполнитьЗапросНаСервере();

КонецПроцедуры

&НаСервере

Процедура ВыполнитьЗапросНаСервере()

//{{КОНСТРУКТОР_ЗАПРОСА_С_ОБРАБОТКОЙ_РЕЗУЛЬТАТА

 // Данный фрагмент построен конструктором.

 // При повторном использовании конструктора, внесенные вручную изменения будут утеряны!!!

Макет

Обработки.ТабличнаяМодель1.ПолучитьМакет("Макет1");

Запрос = Новый Запрос;

Запрос.Текст =

 "ВЫБРАТЬ

=

Номенклатура,		ПоступлениеТоваровТовары.Номенклатура	КАК
КАК Количество,		СУММА(ПоступлениеТоваровТовары.Количество)	
Сумма		СУММА(ПоступлениеТоваровТовары.Сумма)	КАК
		ИЗ	
ПоступлениеТоваровТовары		Документ.ПоступлениеТоваров.Товары	КАК
		СГРУППИРОВАТЬ ПО	
		ПоступлениеТоваровТовары.Номенклатура	
		УПОРЯДОЧИТЬ ПО	
		Количество,	
		Сумма УБЫВ	
		ИТОГИ	
		СУММА(Количество),	
		СУММА(Сумма)	
		ПО	
		Номенклатура ИЕРАРХИЯ";	

РезультатЗапроса = Запрос.Выполнить();

ОбластьЗаголовок = Макет.ПолучитьОбласть("Заголовок");	
ОбластьПодвал = Макет.ПолучитьОбласть("Подвал");	
ОбластьШапкаТаблицы	=
Макет.ПолучитьОбласть("ШапкаТаблицы");	
ОбластьПодвалТаблицы	=
Макет.ПолучитьОбласть("ПодвалТаблицы");	
ОбластьНоменклатураИерархия	=
Макет.ПолучитьОбласть("НоменклатураИерархия");	
ОбластьНоменклатура	=
Макет.ПолучитьОбласть("Номенклатура");	

ТабДок.Очистить();
ТабДок.Вывести(ОбластьЗаголовок);
ТабДок.Вывести(ОбластьШапкаТаблицы);
ТабДок.НачатьАвтогруппировкуСтрок();

ВыборкаНоменклатура	=
РезультатЗапроса.Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам);	

Пока ВыборкаНоменклатура.Следующий() Цикл

```

Если ВыборкаНоменклатура.ТипЗаписи() =
ТипЗаписиЗапроса.ИтогПоИерархии Тогда
    Область = ОбластьНоменклатураИерархия;
Иначе
    Область = ОбластьНоменклатура;
КонецЕсли;

Область.Параметры.Заполнить(ВыборкаНоменклатура);
ТабДок.Вывести(Область,
ВыборкаНоменклатура.Уровень());
КонецЦикла;

ТабДок.ЗакончитьАвтогруппировкуСтрок();
ТабДок.Вывести(ОбластьПодвалТаблицы);
ТабДок.Вывести(ОбластьПодвал);

```

```

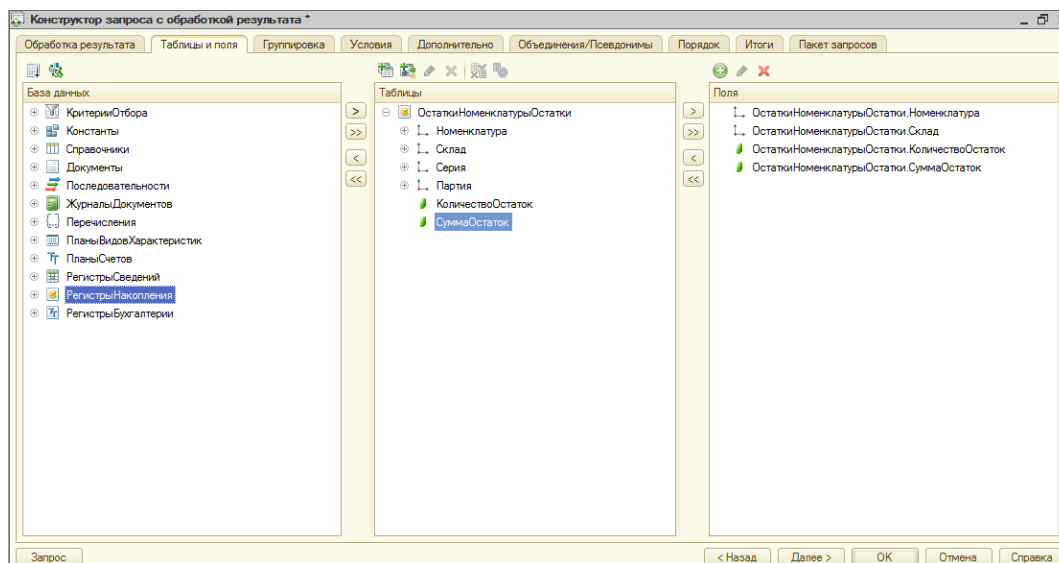
//}} КОНСТРУКТОР_ЗАПРОСА_С_ОБРАБОТКОЙ_РЕЗУЛЬТАТА
КонецПроцедуры

```

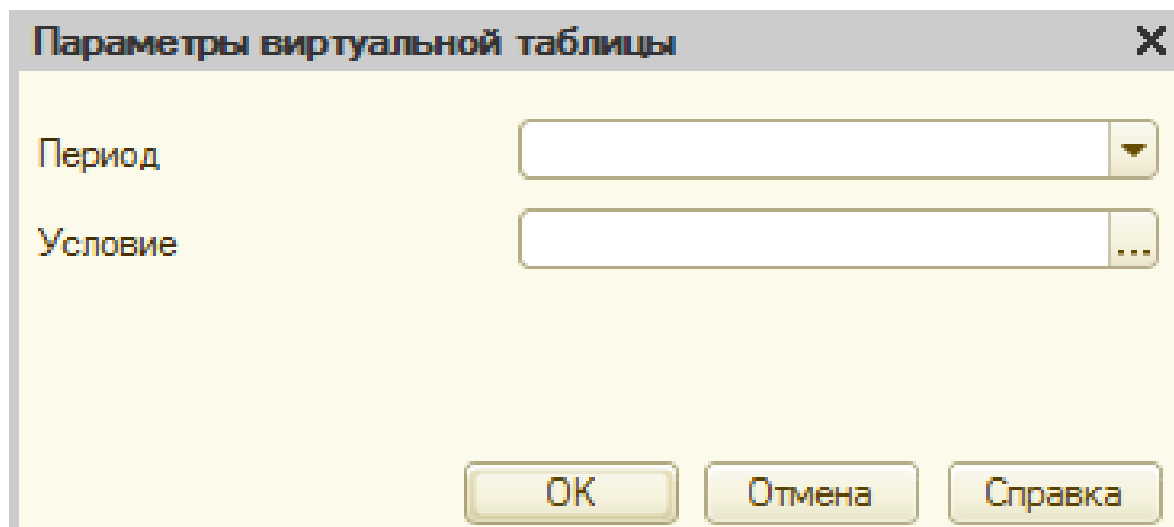
Выполняя повторный запуск конструктора с обработкой результата, попробуйте выполнить разные варианты упорядочивания, разные типы итогов. Обращайте внимание, как на изменение текста запроса, так и на изменение кода, формируемого конструктором в модуле формы.

Работа с виртуальными таблицами

Работа с виртуальными таблицами имеет некоторую особенность. Заключается она в том, что в общем случае обязательно использование параметров виртуальных таблиц. Правильное использование данных параметров в некоторых случаях напрямую сказывается на оптимальности и быстродействии выполнения запроса.



Для задания параметров необходимо нажать на кнопку «Параметры виртуальной таблицы» и в открывшемся диалоговом окне указать их значения (имена параметров запроса).

The image shows a dialog box titled "Параметры виртуальной таблицы" (Parameters of virtual table) with a close button (X) in the top right corner. Inside the dialog, there are two labels: "Период" (Period) and "Условие" (Condition). Next to "Период" is a text input field with a dropdown arrow on the right. Next to "Условие" is a text input field with an ellipsis (...) button on the right. At the bottom of the dialog, there are three buttons: "ОК" (OK), "Отмена" (Cancel), and "Справка" (Help).

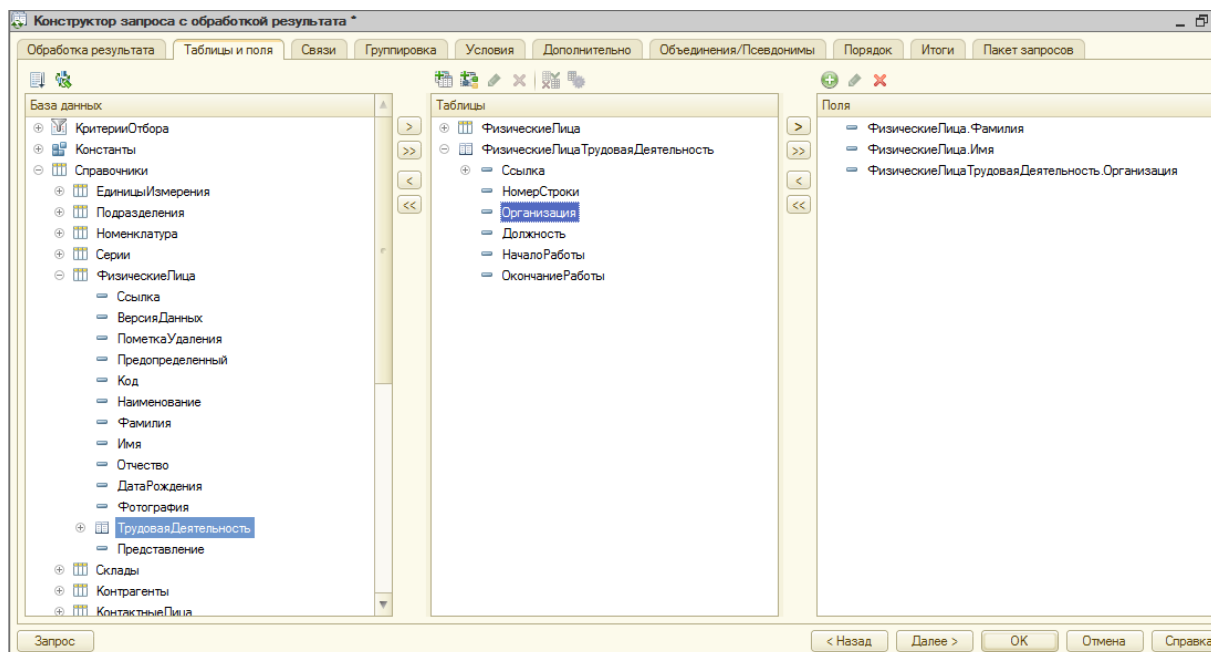
Состав параметров напрямую зависит от выбранной виртуальной таблицы. При определении условия можно повторно вложено вызвать конструктор запроса для определения подзапроса.

Построение запросов по нескольким таблицам

В реальной практике очень редко бывает ситуация, когда данные, которые необходимо получить определены в одной таблице. Чаще всего они находятся в различных таблицах, но должны быть получены вместе (при этом должна соблюдаться связность данных).

На закладке «Таблицы и поля» можно одновременно указывать несколько таблиц-источников данных. Для их увязки можно использовать раздел «Условия», либо можно использовать закладку «Связи». Данная закладка появляется в конструкторе запроса в случае, если в качестве источников данных выбрана более чем одна таблица.

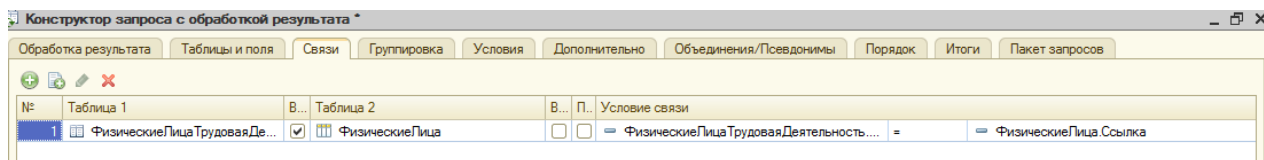
Если ни какой связи явно определенными источниками не определять, то в результат запроса войдет полная комбинация записей из определенных таблиц.



Закладка «Связи» отвечает за раздел «Соединение» языка запроса.

Виды соединений:

1. Внутреннее (ни один из флагов «Все» не отмечен).
2. Левое Внешнее Соединение (отмечен один из флагов «Все»).
3. Правое Внешнее Соединение (отмечен один из флагов «Все»).
4. Полное Внешнее Соединение (отмечены оба флага «Все»).



Связь данных в таблицах устанавливается с помощью так называемого «Условия связи».

Рассмотрим примеры. Условие соединения:
Таблица1.Номер1=Таблица2.Номер2.

Таблица №1	
Номер	Номер1
Лак для волос	1
Краска для волос	2
Маска для лица	3

Таблица №2	
ЕдИзм	Номер2
Шт.	1
Гр.	3
упаковка	4
банка	1

Внутреннее соединение: в результат выполнения запроса войдут только данные записей из обеих таблиц, для которых выполняется условие соединения, т.е.

Лак для волос	Шт.
Лак для волос	Банка
Маска для волос	Гр.

Левое внешнее соединение: в результате выполнения запроса войдут данные из записей, для которых выполняется условие соединения «не вошедшие» из Таблицы №1. Т.е. в результате запроса войдут все данные из Таблицы №1, и для тех записей результата запроса, для которых выполнялось условие соединения в полях, куда помещаются данные из Таблицы №2, будут стоять значения, для которых условие не выполняется, будет стоять Null.

Лак для волос	Шт.
Лак для волос	Банка
Маска для волос	Гр.
Краска для волос	Null

Правое внешнее соединение обратно левому.

Лак для волос	Шт.
Лак для волос	Банка
Маска для волос	Гр.
Null	упаковка

Полное внешнее соединение. В результате запроса войдут как записи, для которых выполнялось условие соединения, так и записи, полученные из «не вошедших» данных из обеих таблиц.

Лак для волос	Шт.
Лак для волос	Банка
Маска для волос	Гр.
Краска для волос	Null
Null	упаковка

Задание.

1. Получите данные о контактных лицах, их телефонах, полном наименовании контрагентов.

2. Получите список пяти наиболее дорогих (по ценам) товаров.
3. Получите данные о том, какой контрагент, на какую сумму поставил фирме товара. В результате запроса должны присутствовать итоги по группам справочника «Контрагенты».
4. Получите список 5 самых продаваемых (по количеству) товаров.

Работа с временными таблицами

Использование временных таблиц помогает повысить скорость выполнения запросов, сделать процесс построения сложных запросов более простым и организовать исполнение таких запросов «поэтапно».

Возможность использования временных таблиц определяется наличием:

1. Объекта «МенеджерВременныхТаблиц».
2. Свойства «МенеджерВременныхТаблиц» объекта «Запрос».
3. Расширением языка запросов по работе с временными таблицами.

В тексте запроса для работы с временными таблицами можно использовать следующие ключевые слова:

1. Поместить.
2. Индексировать По.
3. Уничтожить.

В качестве иллюстрации использования рассмотрим следующий код.

```
//Создание менеджера временной таблицы
```

```
МенеджерВТ = Новый МенеджерВременныхТаблиц;
```

```
Запрос = Новый Запрос;
```

```
Запрос.МенеджерВременныхТаблиц = МенеджерВТ;
```

```
Запрос.Текст = "ВЫБРАТЬ
```

```
| ПродажаТоваров.Номенклатура,
```

```
| СУММА(ПродажаТоваров.Количество) КАК Колчество
```

```
| СУММА(ПродажаТоваров.Сумма) КАК Сумма
```

```
|ПОМЕСТИТЬ ТЧР
```

```
|ИЗ
```

```
| Документ.ПродажаТоваров.Товары КАК ПродажаТоваров
```

```
|ГДЕ
```

```
| ПродажаТоваров.Ссылка = &ЗначениеСсылки
```

```
|
```

```
|СГРУППИРОВАТЬ ПО
```

```
| ПродажаТоваров.Номенклатура";
```

```
Запрос.УстановитьПараметр («ЗначениеСсылки», ПродажаТоваров);
```

```
//Создание временной таблицы
```

```
Результат = Запрос.Выполнить();
```

Использование временной таблицы:

Запрос = Новый Запрос;

//в свойство «МенеджерВременныхТаблиц» записываем менеджер,

//в контексте которого была создана временная таблица

Запрос.МенеджерВременныхТаблиц = МенеджерВТ;

Запрос.Текст = “ВЫБРАТЬ

| ТЧР.Номенклатура Как Товар

| Остатки.КоличествоОстаток, ТЧР.Количество

|ИЗ

|ТЧР

|Левое Соединение

|РегистрыНакопления.ОстаткиНоменклатуры.Остатки(,Номенклатура

В (Выбрать Номенклатура ИЗ ТЧР)) КАК Остатки

|По ТЧРасходной.Номенклатура = Остатки.Номенклатура”;

Результат = Запрос.Выполнить();

В качестве источников данных для временных таблиц могут использоваться:

1. Таблица значений.

2. Табличная часть.

3. Результат запроса.

Для удаления временных таблиц, созданных в контексте экземпляра объекта «МенеджерВременныхТаблиц», нужно использовать метод «Закрыть()».

МенеджерВТ.Закрыть();

Для удаления одной таблицы:

Запрос = Новый Запрос;

Запрос.МенеджерВременныхТаблиц = МенеджерВТ;

Запрос.Текст = «Уничтожить ТЧРеализацииИОстатки»;

Результат = Запрос.Выполнить();

В тексте запроса можно использовать predefined данные системы.

Запрос = Новый Запрос;

Запрос.Текст = “ВЫБРАТЬ

| Номенклатура.Код,

| Номенклатура.Наименование

|ИЗ
| Справочник.Номенклатура КАК Номенклатура
|ГДЕ
|Номенклатура.ОснЕдиницаИзмерения
ЗНАЧЕНИЕ(Справочник.ЕдиницыИзмерения.Штука)”;

Результат = Запрос.Выполнить();
//Обход результата запроса

1. Контрольные вопросы.

- 1) Для чего предназначен объект конфигурации «Регистр накопления».
 - 2) Почему следует использовать регистры, хотя необходимая информация содержится в других объектах.
 - 3) Для чего нужны измерения регистра, ресурсы и реквизиты.
 - 4) Что такое движения регистра и что такое регистратор.
 - 5) Для чего предназначен объект конфигурации «Регистр сведений».
 - 6) Какими особенностями обладает объект конфигурации Регистр сведений.
 - 7) В чем главные отличия регистра сведений от регистра накопления.
- Как получить значения ресурсов наиболее поздних записей регистра средствами встроенного языка.