**Лабораторная работа №6**

**Тема:** Документы, регистры накопления, СКД

### Цель работы: Исследовать механизмы создания документов и регистров накопления, ознакомиться методикой построения отчетов с использованием системы компоновки данных (СКД)

### Документы

Любая учетная система получает исходные данные из документов. В классическом бухгалтерском учете основа всего – первичные документы, автоматизированные системы учета – не *исключение*.

Для описания документов в *дереве конфигурации* имеется отдельная *ветвь* – **Документы**. В одной из предыдущих занятий мы создали один документ – **ПоступлениеМатериалов**. Сейчас мы займемся работой с ним. Для начала определимся с целью использования этого документа. Мы планируем с его помощью отражать в системе поступление материалов. Исходя из этих целей, нам понадобятся следующие реквизиты документа ([рис. 6.1](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=1#image.6.1).), которые мы зададим на вкладке **Данные** окна редактирования объекта:

**Имя**: *Контрагент*, Тип: СправочникСсылка.*Контрагенты*

**Имя**: **ОтветственныйСотрудник**: Тип: СправочникСсылка.Сотрудники

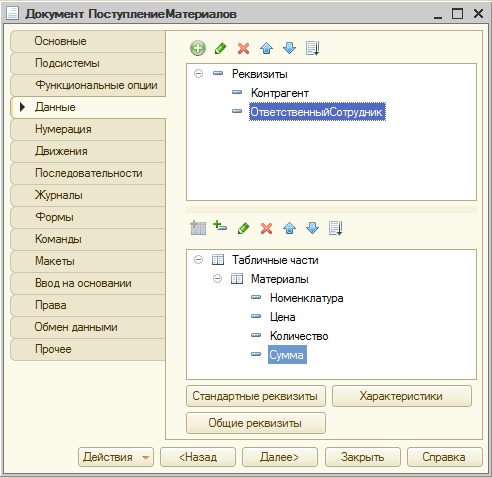
Добавим в состав табличных частей нашего документа новую табличную часть с именем **Материалы** и следующими реквизитами:

**Имя**: Номенклатура, Тип: СправочникСсылка.Номенклатура

**Имя**: Цена, Тип: Число, *длина* 10, *точность* 2

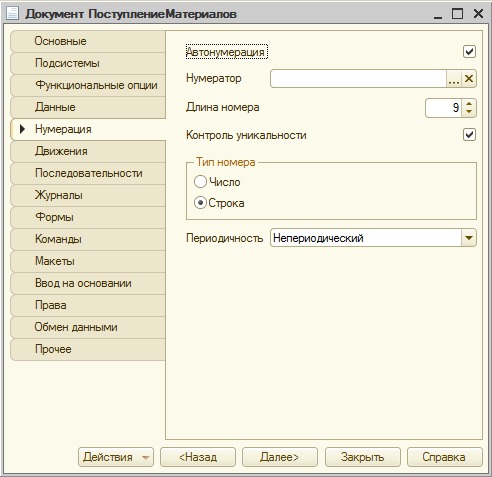
**Имя**: Количество, Тип: Число, *длина* 10, *точность* 3

**Имя**: Сумма, Тип: Число, *длина* 10, *точность* 2



**Рис. 6.1.**Настройка состава реквизитов документа

На закладке **Нумерация**, [рис. 6.2](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=1#image.6.2)., можно задать параметры нумерации документов.



**Рис. 6.2.**Настройка параметров нумерации документа

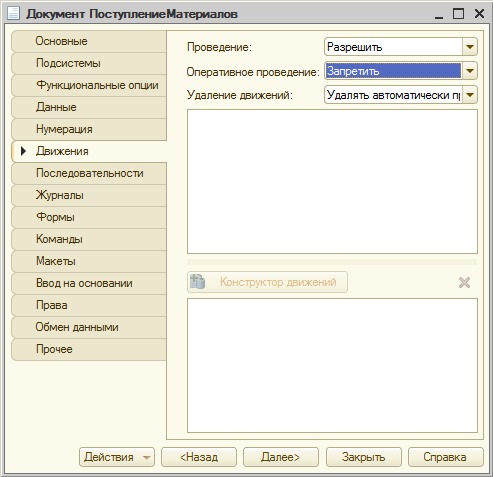
В данном случае документы будут нумероваться автоматически с контролем уникальности номеров.

Нескольким различным видам документов можно назначить сквозную нумерацию с одинаковыми настройками благодаря использования нумератора.

Если в качестве типа номера использована строка – это позволит, при возникновении необходимости, добавлять к номеру различные символьные префиксы.

Настройка периодичности в данном случае установлена в *значение* **Непериодический**, то есть, независимо от срока работы с информационной базой, *нумерация* документов будет продолжаться, а при установке некоторой периодичности (в пределах дня, месяца, квартала, года) – уникальность номеров будет соблюдаться в течение указанного периода.

Закладка **Движения**, [рис. 6.3](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=1#image.6.3)., позволяет управлять проведением документа.



**Рис. 6.3.**Настройка параметров проведения документа

Если документ планируется проводить, то есть – он будет формировать движения *по* регистрам, проведение следует разрешить. **Оперативное проведение** позволяет разработчику настроить различное реагирование кода, отвечающего за *проведение документа*, при *проведении документа*, скажем, более поздней или ранней датой, чем текущая дата.

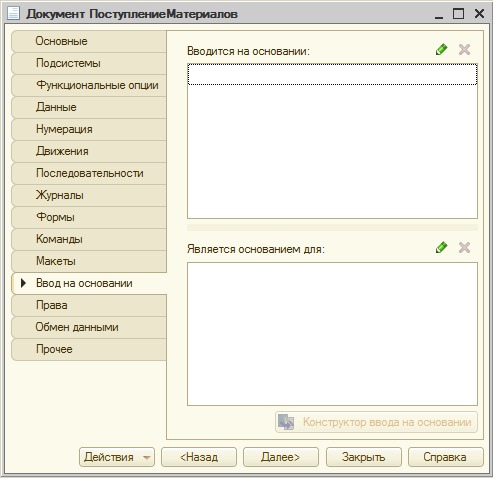
Если мы не планируем реализацию подобной функциональности, мы можем отключить оперативное проведение.

**Автоматическое удаление** движений документа предусматривает, при отмене проведения, автоматически удалять движения, которые документ сформировал *по* регистрам. В данный момент в нашей конфигурации пока нет регистров, *по* которым будет проводиться документ. "Формирование движений *по* регистру", более простым языком, означает то, что документ при проведении делает записи в регистре. Регистры можно сравнить с таблицами, содержащими ключевые данные о документах, которые *по* ним проводятся. Работу с различными регистрами мы рассмотрим ниже.

Вкладка **Журналы** позволяет настраивать включение документа в так называемые **журналы документов**. Журнал позволяет организовать совместную работу с документами различных типов, которые включены в этот журнал. Такие документы отображаются в едином списке.

Если *проведение документа* запрещено – то *пользователь* сможет лишь сохранить документ в базе данных. Других воздействий на информационную базу такой документ не произведет. Например, такое поведение может быть характерно для документов, вроде выписанных счетов, которые сами *по* себе воздействия на учет не производят, но их важно хранить в системе для того, чтобы "помнить" о том, какие счета выписаны, важно иметь возможность формировать их *печатные формы*.

Но то, что счет выписан, еще не гарантирует то, что счет будет оплачен, то, что товары, указанные в выписанном счете будут действительно отгружены покупателю. Если продолжить пример с выписанным счетом и перейти на вкладку **Ввод на основании**, [рис. 6.4](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=1#image.6.4)., то окажется, что эта вкладка позволяет настроить ввод одного документа на основании другого.



**Рис. 6.4.**Настройка параметров ввода на основании

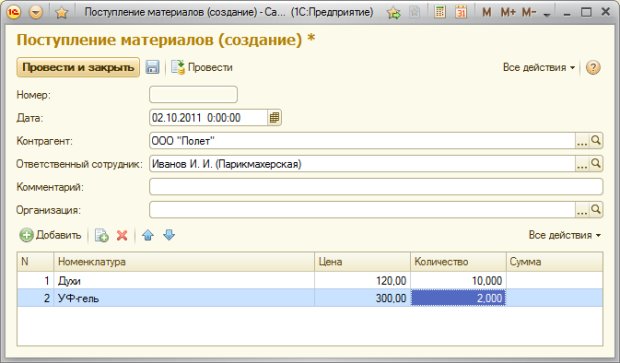
Если, например, мы имеем дело с документом наподобие "**Отгрузка материалов**", окажется, что такой документ вполне логично будет вводить на основании документа "**Счет**" - после оплаты этого счета и фактической отгрузки материалов. Документ отгрузки, в отличие от счета, фиксирует уже свершившийся *факт хозяйственной жизни*, который должен оказать воздействие на состояние информационной базы. Такой документ должен проводиться – то есть – делать записи в соответствующие регистры.

На данном этапе мы можем запустить систему, попытаться поработать с документом, используя автоматически сгенерированную форму, и посмотреть, все ли в данной форме нас устраивает.

Прежде чем продолжать работу с документом **ПоступлениеМатериалов**, приведите данные справочника **Номенклатура** в вашей *информационной базе* к виду, показанному в [таблице 6.1](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=2#table.6.1).

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 6.1. Данные справочника Номенклатура | | | |
| **Наименование** | **Единица измерения** | **Услуга** | **Группа** |
| Парикмахерские услуги |  | Да | Да |
| Завивка | Час | Да |  |
| Стрижка | Час | Да |  |
| Парфюмерия |  | Нет | Да |
| Духи | Штука | Нет |  |
| Одеколон | Штука | Нет |  |
| Прочие материалы |  | Нет | Да |
| УФ-гель | Упаковка | Нет |  |
| Спецодежда |  | Нет | Да |
| Одежда для парикмахера | Штука | Нет |  |
| Уход за волосами |  | Нет | Да |
| Бальзам для волос | Штука | Нет |  |
| Лак для волос | Упаковка | Нет |  |

На [рис. 6.5](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=2#image.6.5). вы можете видеть форму документа после ввода в нее некоторых данных.



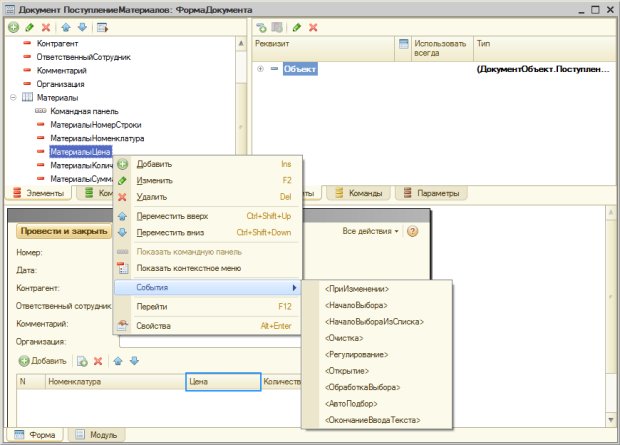
**Рис. 6.5.**Заполнение документа Поступление товаров

Подобный документ, очевидно, заполняется в *информационной базе* *по* данным некоего бумажного документа (*приходной накладной*, например), поступившей от поставщика вместе с поступившими товарами. Цена и количество каждой товарной позиции вводится из документа вручную, с этим здесь ничего поделать нельзя. Но вот ввод суммы *по* каждой из строк табличной части вполне поддается автоматизированному расчету на основе данных о цене и количестве. Еще один важный момент, на который можно обратить внимание – в нашем документе не отображается итоговый показатель *по* табличной части. Как правило, в подобных случаях пользователю важно увидеть общую сумму документа. Общая сумма позволит быстро сверить данные, введенные в *электронный документ*, с его бумажным аналогом. Если общая сумма совпадает – то, почти наверняка, все строки табличной части введены верно.

Реализуем эту функциональность.

Для того, чтобы автоматически заполнить *поле* сумма *по* каждой из строк табличной части, редактируемой пользователем, очевидно, что рассчитывать сумму имеет смысл либо после заполнения поля **Цена**, либо – после заполнения поля **Количество**, перехватив какие-либо события, имеющие *отношение* к редактируемой табличной части.

В нашем случае это должны быть события, генерируемые при изменении полей **Цена** или **Количество** при вводе данных в определенной строке. Для того, чтобы назначить обработчики подобных событий для определенных элементов табличной части, можно поступить так же, как мы поступали, назначая обработчики событий для любых других элементов формы ([рис. 6.6](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=2#image.6.6).). Для начала, конечно же, нам нужно будет создать собственную форму документа, делается это на закладке **Формы** окна редактирования объекта. С параметрами, предложенными конструктором форм *по* умолчанию, можно согласиться.

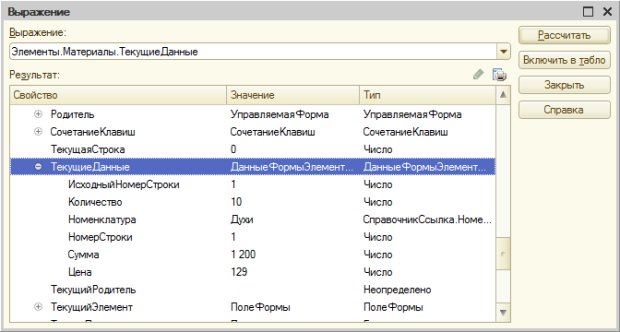
[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_06.jpg)

**Рис. 6.6.**Назначение обработчика полю табличной части

Назначим обработчики событий **ПриИзменении** для полей **МатериалыЦена** и **МатериалыКоличество**.

Теперь нам нужно реализовать следующее: при работе в определенной строке таблицы, при вводе в нее данных, получить эту строку, и, при изменении цены или количества номенклатуры рассчитать сумму.

У табличных полей есть свойство **ТекущиеДанные**, которое, как раз, позволяет обращаться к текущей редактируемой строке. Данные редактируются на клиенте, поэтому мы вполне можем обойтись здесь без вызова серверных процедур, выполнив все необходимые действия на клиенте. Если вы хотите побольше узнать о том, что можно сделать с табличным полем из кода, как и в других случаях, помочь вам в этом могут инструменты отладки. Вот как, например, выглядит свойство **ТекущиеДанные** при срабатывании точки останова в коде модуля нашей формы при отладке кода, который будет представлен ниже, [рис. 6.7](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=2#image.6.7).



**Рис. 6.7.**Просмотр свойства ТекущиеДанные в окне Выражение при отладке кода

Итак, наша задача может быть решена следующим образом:

&НаКлиенте

Процедура МатериалыЦенаПриИзменении(Элемент)

РассчитатьСумму();

КонецПроцедуры

&НаКлиенте

Процедура МатериалыКоличествоПриИзменении(Элемент)

РассчитатьСумму();

КонецПроцедуры

&НаКлиенте

Процедура РассчитатьСумму()

ТекущаяСтрока=Элементы.Материалы.ТекущиеДанные;

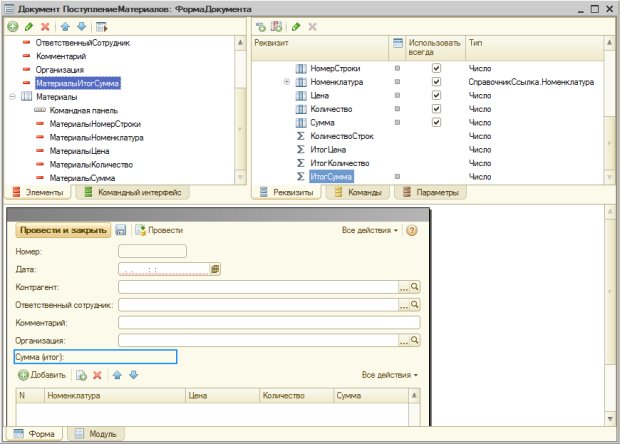
ТекущаяСтрока.Сумма=ТекущаяСтрока.Количество\*ТекущаяСтрока.Цена;

КонецПроцедуры

Из пары обработчиков событий **ПриИзменении** вызывается клиентская процедура **РассчитатьСумму()**. Здесь мы получаем данные текущей строки через свойство **ТекущиеДанные** и вычисляем *поле* **Сумма**, перемножая данные в полях **Количество** и **Цена**.

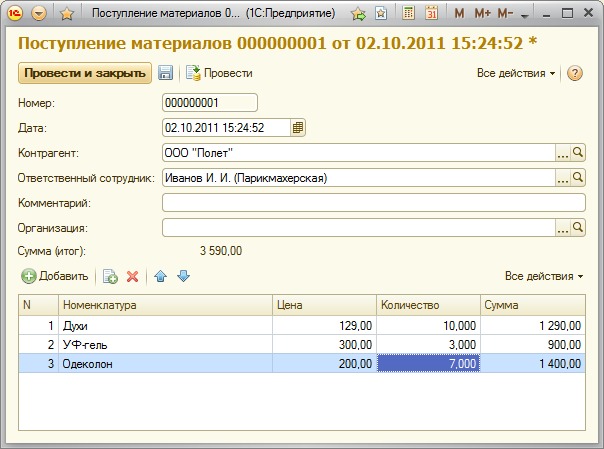
При необходимости, мы можем редактировать *поле* **Сумма** независимо от значений полей **Цена** и **Количество**.

Вторая задача из тех, которые мы поставили себе выше, заключается в выводе на форму итоговых сведений *по* табличному полю. Ее можно реализовать различными способами, но лучше всего воспользоваться стандартными итоговыми показателями табличного поля, которые можно найти в составе табличного поля на закладке **Реквизиты** редактора форм, [рис. 6.8](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=3#image.6.8).

[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_08.jpg)

**Рис. 6.8.**Вывод итогового показателя для поля Сумма на форму

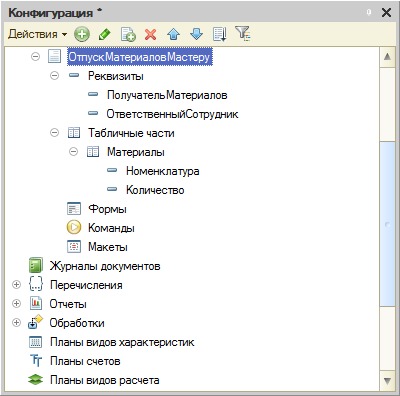
Этот *реквизит* – **ИтогСумма** – нужно перетащить на вкладку **Элементы**. Он будет отображаться на форме, изменяясь при изменениях суммы в строках табличной части, [рис. 6.9](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=3#image.6.9).



**Рис. 6.9.**Форма после модификации

Документ **ПоступлениеМатериалов** мы разработали, настроили его форму. Документ, хотя его проведение и разрешено, пока, фактически, не проводится – мы не реализовали *механизмы* проведения, у нас нет регистров, *по* которым он будет проводиться.

Данную функциональность мы реализуем ниже, а сейчас займемся еще одним документом, который, являясь, *по* составу реквизитов и *по* особенностям устройства формы, очень похожим на документ **ПоступлениеМатериалов**, выполняет противоположную ему функцию – а именно – отвечает за списание материалов. В нашей системе материалы выбывают при передаче их в производство. Мы вполне можем создать новый документ копированием предыдущего и изменением некоторых его реквизитов. Так и поступим. Скопируем документ и приведем состав его реквизитов к показанному на [рис. 6.10](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=3#image.6.10).

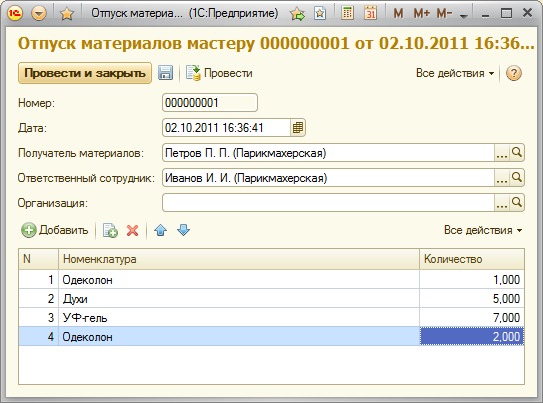


**Рис. 6.10.**Создание документа ОтпускМатериаловМастеру

Включим данный документ в состав подсистемы **ОперативныйУчетМатериалов**, вместо реквизита **Контрагент** у него будет *реквизит* **ПолучательМатериалов** с типом **СправочникСсылка.Сотрудники**.

В табличной части документа мы используем лишь два реквизита – это **Номенклатура** и **Количество**. Показатели стоимости списываемой номенклатуры мы будем рассчитывать автоматически. При работе с этим документом нас вполне устроит форма, генерируемая автоматически.

Форма нашего нового документа будет выглядеть так, как показано на [рис. 6.11](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=3#image.6.11).



**Рис. 6.11.**Документ ОтпускМатериаловМастеру в работе

Документы созданы, но сейчас они, во-первых, заполняются практически полностью вручную (за исключением поля **Сумма** в табличной части документа **ПоступлениеМатериалов**), а, во-вторых, нам сейчас довольно сложно будет понять, каков *остаток*материалов, числящихся за определенным сотрудником.

Единственный способ, которым можно сделать это сейчас – программно или "вручную" просмотреть все документы поступления материалов, выводя некоторые итоговые показатели. При заполнении документа отпуска материалов мы вынуждены заранее самостоятельно проверять остатки материалов *по* существующим документам, самостоятельно решать, какова цена этих материалов, контролировать остатки. Это не очень удобно, не производительно, и решением этих задач мы сейчас займемся.

### Регистры накопления

Представим себе организацию, в которой все сведения о приходе материалов хранятся лишь в виде приходных документов. Для того, чтобы узнать количество и *стоимость* имеющихся в организации материалов, нам понадобится обращаться к документам, просматривать каждый из них, выписывая нужные данные, после чего суммировать их, получая нужные данные. Такой подход неудобен – он слишком медленный как для нашего воображаемого "ручного" случая, так и для автоматизированного учета.

Логичнее было бы, в *дополнение* к документам, вести специальные таблицы, в которые, при приеме материалов и при их списании, вносить краткие сведения об этом. Если, скажем, ежедневно, подводить итоги *по* этим таблицам и выводить остатки материалов, то, для того, чтобы сказать, сколько и каких материалов имеется в организации на определенную дату, достаточно обратиться к соответствующей графе таблицы. В системе 1С:Предприятие такими таблицами являются *регистры накопления*. Как следует из названия, они предназначены для отражения, накопления, неких показателей. И отражение в регистрах прихода и расхода материалов – один из типичных примеров их использования.

В нашей организации ведется учет материалов в привязке к ответственным лицам, которые их получают и с которых эти материалы списывают при отпуске в производство. Нам нужно хранить информацию о количестве материалов и об их стоимости, а так же, при списании в производство, иметь сведения о том, какому именно мастеру эти материалы переданы. Эти соображения позволяют нам спроектировать структуру *регистра накопления*, который мы сейчас будем создавать.

При планировании состава *регистра накопления* нужно понять, какие именно данные мы собираемся в нем хранить, после чего "разложить" эти данные *по* измерениям, ресурсам и реквизитам регистра.

Итак, нам нужно хранить следующие данные:

* Номенклатурная позиция
* Ответственный сотрудник, на котором числится данная позиция
* Количество номенклатуры
* Стоимость номенклатуры
* Данные о мастере, которому переданы материалы для использования.

**Измерения регистра**, или разрезы, в которых хранятся данные, позволяют нам ответить на вопросы о том, какие именно данные хранятся в регистре. В нашем случае нам нужно знать две основных характеристики – это, за каким **ответственным лицом**закреплена та или иная **номенклатурная позиция**. Очевидно, измерениями из нашего списка данных будут номенклатурная позиция и ответственный сотрудник.

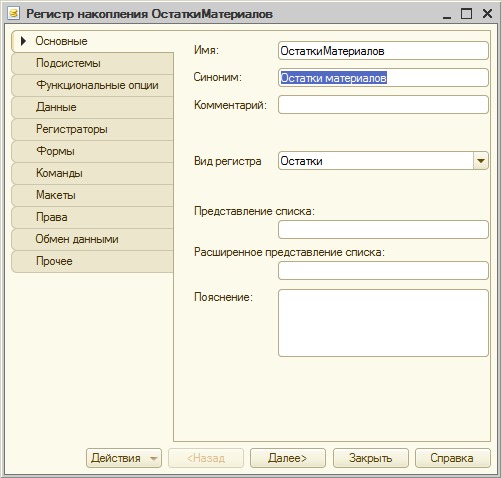
**Ресурсы регистра** – это всегда числовые значения, характеризующие хранимые данные. Числовые значения – это количество и сумма, и именно они будут ресурсами нашего регистра.

**Реквизиты регистра** играют вспомогательную роль, и, в нашем случае, логично будет в *реквизите* регистра хранить сведения о мастере, получившем материалы для работы – на тот случай, если нам понадобится узнать – кто именно эти материалы использовал.

Еще один важный вопрос, который нужно решить, проектируя *регистр*, заключается в том, будет ли этот *регистр* **регистром остатков**или **регистром оборотов**. Нас интересуют и сведения об остатках материалов, и сведения об оборотах, поэтому при настройке регистра следует указать вид регистра – **Остатки**. *Регистр* с видом **Остатки** позволяет нам работать и с остатками и с оборотами

Предложенная здесь структура *регистра накопления* – это лишь один из вариантов того, как можно организовать хранение описываемых данных. Подобную схему учета можно реализовать, скажем, с помощью пары регистров, один из которых используется исключительно для целей хранения суммовых остатков материалов – то есть, те данные, которые нужны для финансовых отчетов, другой – для хранения данных *по* центрам ответственности. В любом случае, каждая конкретная схема учета может потребовать и собственной структуры регистров, и наш пример – лишь демонстрация одного из возможных вариантов.

Обсудив теоретическую часть вопроса, перейдем к практике. Создадим новый *регистр накопления*, назовем его **ОстаткиМатериалов**, *параметр* **Вид регистра** оставим в значении **Остатки**, [рис. 6.12](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=4#image.6.12).



**Рис. 6.12.**Регистр накопления ОстаткиМатериалов

Включим *регистр накопления* в состав подсистемы **ОперативныйУчетМатериалов**.

На вкладке **Данные** создадим следующие измерения, ресурсы и реквизиты:

**Измерения**:

**Имя**: Номенклатура, Тип: СправочникСсылка.Номенклатура, Запрет незаполненных значений – установлено.

**Имя**: ОтветственныйСотрудник, Тип: СправочникСсылка.Сотрудники, Запрет незаполненных значений – установлено.

**Ресурсы**

**Имя**: Количество, Тип: число, *длина* 10, *точность* 3

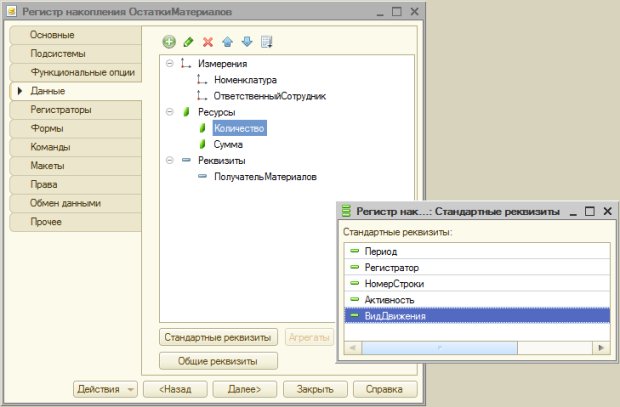
**Имя**: Сумма, Тип: число, *длина* 10, *точность* 2

**Реквизиты**:

**Имя**: ПолучательМатериалов, Тип: СправочникСсылка.Сотрудники

Обратите внимание на имена этих реквизитов, на их типы, а так же – на стандартные реквизиты регистра ([рис. 6.13](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=4#image.6.13).) – эти данные пригодятся нам при работе над процедурой *проведения документа*.

Исключим из состава реквизитов регистра общий *реквизит* **Организация**. Сейчас в нем нет необходимости. Для организации хранения данных в регистре в разрезе различных организаций нам понадобилось бы новое измерение – Организация, благодаря наличию которого мы смогли бы работать с материалами различных организаций.

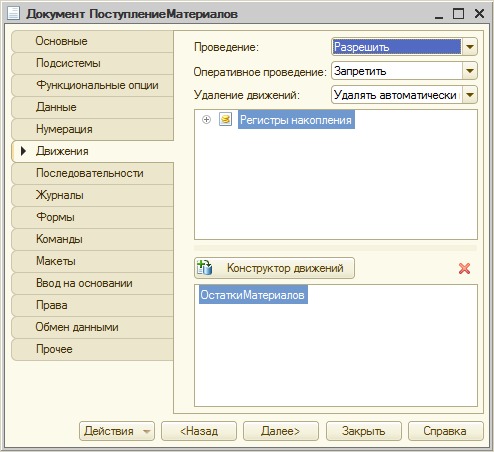
[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_13.jpg)

**Рис. 6.13.**Регистр накопления ОстаткиМатериалов, состав данных

Перейдем на вкладку **Регистраторы** окна редактирования объекта и выберем в качестве документов-регистраторов документы – **ПоступлениеМатериалов** и **ОтпускМатериаловМастеру**.

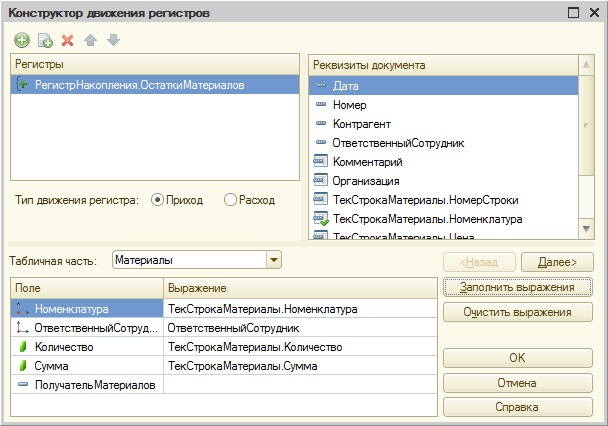
На данном этапе настройка *регистра накопления* окончена, перейдем к настройкам документов. Начнем с документа **ПоступлениеМатериалов**.

Откроем окно редактирования объекта для этого документа, перейдем на вкладку **Движения** ([рис. 6.14](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=4#image.6.14).) и нажмем на кнопку **Конструктор движений**



**Рис. 6.14.**Документ ПоступлениеМатериалов, вкладка Движения

В конструкторе, выберем тип движения регистра – **Приход**, в *поле* **Табличная часть** укажем табличную часть документа **Материалы**, нажмем на кнопку **Заполнить выражения**. Автоматический механизм установления соответствия между данными документа и регистра не всегда работает правильно (в том случае, если не может однозначно определить соответствия, или тогда, когда соответствие, определенное им *по* его логике, отличается от желаемого), поэтому проверим правильность установленных соответствий. В итоге окно **Конструктора** **движения регистра** должно выглядеть так, как показано на [рис. 6.15](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=4#image.6.15).



**Рис. 6.15.**Конструктор движений

После нажатия на кнопку ОК, в модуле объекта документа будет сформирована такая процедура обработки проведения (так она выглядит после удаления комментариев о том, что код построен конструктором движений):

Процедура ОбработкаПроведения(Отказ, Режим)

// регистр ОстаткиМатериалов Приход

Движения.ОстаткиМатериалов.Записывать = Истина;

Для Каждого ТекСтрокаМатериалы Из Материалы Цикл

Движение = Движения.ОстаткиМатериалов.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

Движение.Период = Дата;

Движение.Номенклатура = ТекСтрокаМатериалы.Номенклатура;

Движение.ОтветственныйСотрудник = ОтветственныйСотрудник;

Движение.Количество = ТекСтрокаМатериалы.Количество;

Движение.Сумма = ТекСтрокаМатериалы.Сумма;

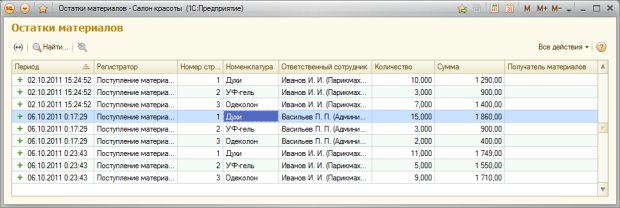
КонецЦикла;

КонецПроцедуры

Эта процедура объявлена в модуле объекта, она исполняется на *сервере. Установка* свойства **Движения.ОстаткиМатериалов.Записывать** в *значение* **Истина** говорит системе о том, что она должна записать в *регистр*движения, сформированные в процедуре. Движения заполняются в цикле обхода табличной части документа. Они будут физически записаны в *регистр* после того, как будет сформирован полный набор записей. Процедура проведения вполне может использовать и другие *механизмы* обращения к данным. Например, нужные данные могут быть получены с помощью запроса.

Так как мы проводим документ, отвечающий за приход товаров, *параметр* **Движение.ВидДвижения** устанавливается в *значение***ВидДвиженияНакопления.Приход**. Период устанавливается равным дате документа (мы напрямую работаем с реквизитами документа). Остальные данные заполняются, опять же, либо из реквизитов документа, либо из реквизитов текущей строки табличной части.

Запустим конфигурацию в режиме 1С:Предприятие, проверим работу механизма на практике. Для этого перепроведем существующие документы **ПоступлениеМатериалов**, можем ввести и новые документы этого вида. При проведении или перепроведении данные из документов попадают в *регистр накопления* **ОстаткиМатериалов**, [рис. 6.16](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=4#image.6.16).

[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_16.jpg)

**Рис. 6.16.**Данные в регистре накопления

Обратите внимание на то, что *регистры накопления*, как объекты, которые не предназначены изначально для "ручной" работы пользователя, не выводятся в командном интерфейсе даже при указании подсистем, в которые они входят. Нам, при разработке, понадобится просматривать регистры.

Для того, чтобы открыть окно регистра можно либо воспользоваться командой **Главное меню > Все функции** и в появившемся окне **Все функции** найти нужный *регистр*, либо открывать его с помощью команды в интерфейсе, предварительно самостоятельно добавив эту команду в нужный раздел интерфейса.

При записи данных о приходе материалов нам, в нашем случае, нет нужды в каких-либо дополнительных проверках вводимых данных, поэтому нас вполне устроит стандартная процедура проведения.

Следующим документом, проведение которого мы хотим организовать, является документ, списывающий материалы – **ОтпускМатериаловМастеру**. Им мы займемся в следующей лекции, а сейчас построим отчет **ОстаткиМатериалов**.

### Отчет ОстаткиМатериалов, введение в СКД

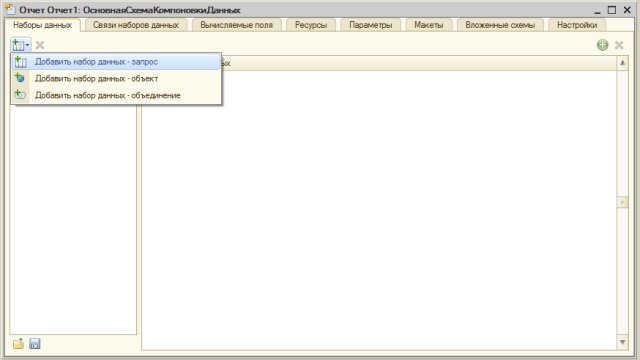
Ранее мы рассматривали построение простейшего отчета самостоятельно формируя *макет* отчета, код вывода данных. Такой подход имеет право на жизнь, им нужно владеть, *по* крайней мере, на тот случай, если вам придется редактировать отчет в какой-либо конфигурации, где применяется такой подход. Однако, наиболее правильным способом построения отчетов в системе 1С:Предприятие 8.2. является использование *системы компоновки данных*, или, сокращенно, СКД. Начиная с этой лабораторной работы мы будем строить отчеты именно с использованием СКД.

Итак, прежде чем строить отчет, поймем, чего мы ждем от этого отчета.

Нам хотелось бы получать сведения о количественных и суммовых остатках номенклатуры *по* ответственным лицам на заданную дату. Эти данные мы можем найти в регистре накопления **ОстаткиМатериалов**.

Добавим в *дереве конфигурации* новый отчет, назовем его **ОстаткиМатериалов**. Включим его в состав подсистемы **ОперативныйУчетМатериалов**.

На закладке **Основные** нажмем на кнопку с увеличительным стеклом в *поле* **Основная схема компоновки данных**. Появится окно конструктора макета, где мы можем задать имя (нас устроит имя *по* умолчанию – **ОсновнаяСхемаКомпоновкиДанных**), тип макета ограничен единственным – **Схема компоновки данных**. Нажмем в этом окне **Готово** и попадем в окно **конструктора СКД**. Здесь нам, в первую *очередь*, нужно добавить новый **источник данных**, в нашем случае это будет **Запрос**, [рис. 6.17](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.17).

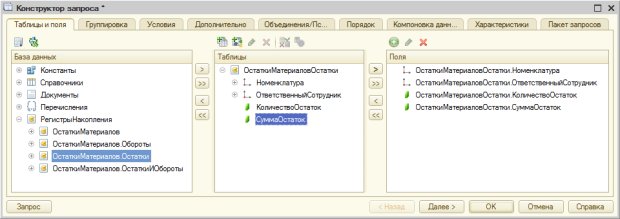
[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_17.jpg)

**Рис. 6.17.**Добавление нового набора данных - запроса

Когда набор данных, названный **НаборДанных1**, будет добавлен, мы можем нажать на кнопку **КонструкторЗапроса**, находящуюся над полем **Запрос** в нижней части окна. Это приведет к открытию окна конструктора запроса.

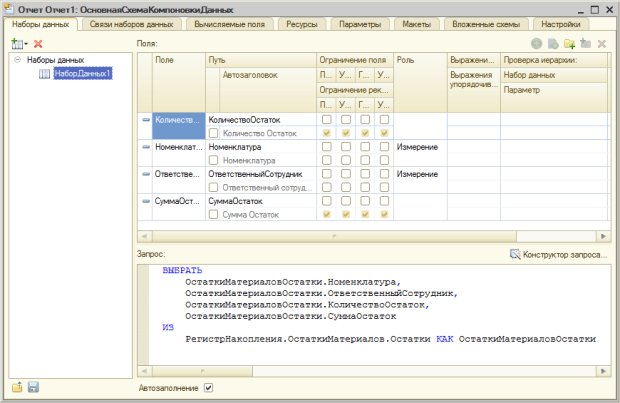
Из виртуальной таблицы *регистра накопления* **ОстаткиМатериалов** выберем следующие поля, [рис. 6.18](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.18).

* Номенклатура
* ОтвественныйСотрудник
* КоличествоОстаток
* СуммаОстаток

[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_18.jpg)

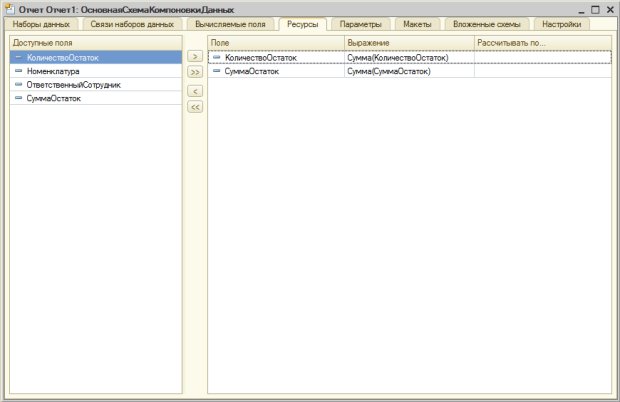
**Рис. 6.18.**Создание запроса

На этом работа с конструктором запроса завершена – остальные настройки мы будем делать в конструкторе СКД. Благодаря установленному *по* умолчанию флагу **Автозаполнение**, на вкладке **Наборы данных** после создания запроса мы можем видеть заполненный *список* полей, [рис. 6.19](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.19). – с этими полями мы сможем работать при создании отчета.

[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_19.jpg)

**Рис. 6.19.**Автозаполнение списка полей на закладке Наборы данных

Переместимся в окне редактора СКД на вкладку **Ресурсы**, из списка **Доступные поля** перенесем в *список*, находящийся в правой части окна, поля, *по* которым можно вычислять итоги. В нашем случае это поля **КоличествоОстаток** и **СуммаОстаток**. *По* умолчанию этим полям в *поле* **Выражение** будет назначена агрегатная *функция* **Сумма**, нас устроит такое положение дел, [рис. 6.20](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.20).

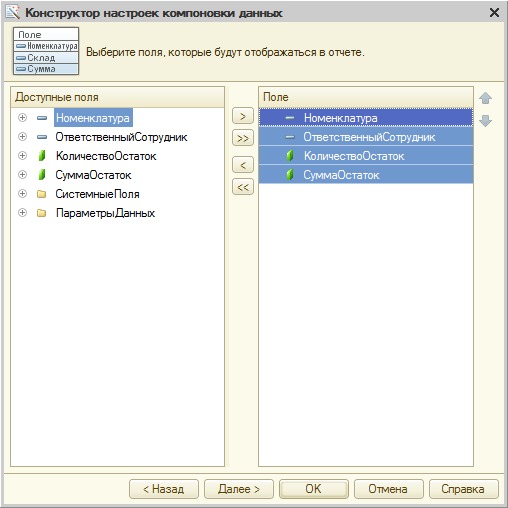
[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_20.jpg)

**Рис. 6.20.**Настройка состава ресурсов отчета

Теперь займемся настройкой внешнего вида отчета.

Перейдем на закладку **Настройки**, вызовем кнопкой с соответствующим названием **Конструктор настроек**и выберем на его первой странице тип отчета – **таблицу**. Нажмем на кнопку **Далее** и в следующем окне выберем поля, которые будут отображаться в отчете в следующем порядке ([рис. 6.21](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.21).):

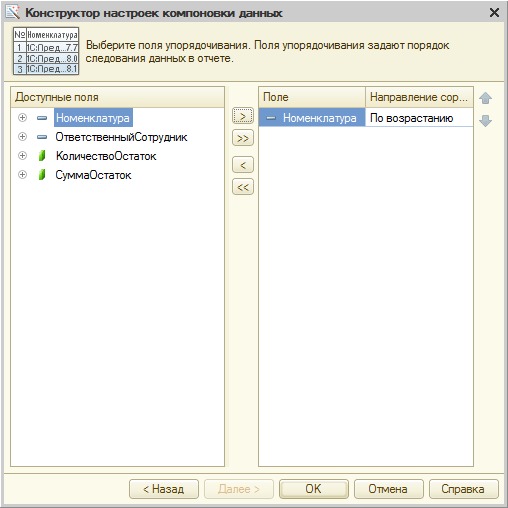
* Номенклатура
* ОтветственыйСотрудник
* КоличествоОстаток
* СуммаОстаток



**Рис. 6.21.**Выбор полей, которые будут отображаться в отчете

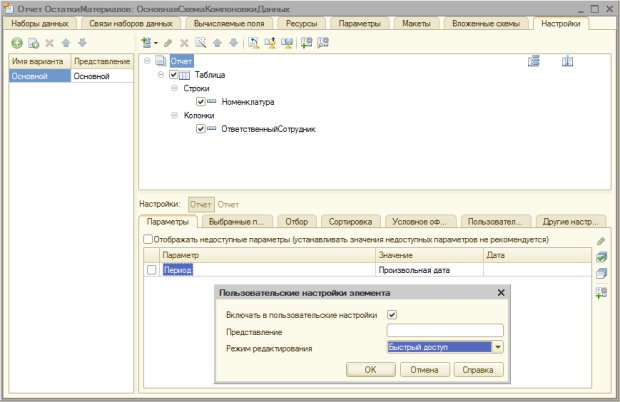
Нажмем кнопку **Далее**, в следующем окне конструктора, служащим для настройки группировки таблиц, в группу **Строки** добавим *поле***Номенклатура**, в *поле* **Колонки** – **ОтветственныйСотрудник**. Тип группировки оставим в состоянии **Без иерархии**.

В следующем окне конструктора, который позволяет задать упорядочение отчета, зададим упорядочивание *по* полю **Номенклатура**, *по* возрастанию, [рис. 6.22](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.22).



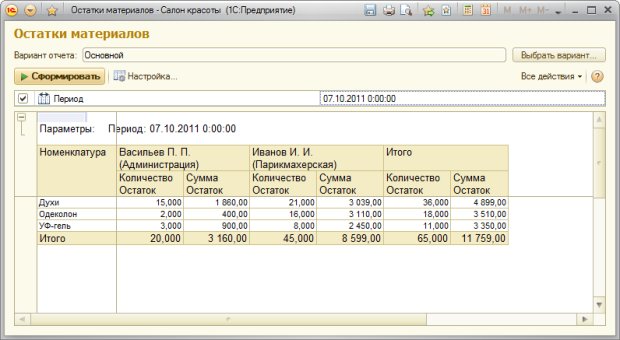
**Рис. 6.22.**Настройка упорядочивания отчета

Нажмем **ОК**, в отчет будет добавлена новая *таблица*. В нижней части формы конструктора СКД, на закладке **Параметры**, выделим *параметр* **Период** и нажмем на кнопку **Свойства элемента пользовательских настроек**. В появившемся окне установим флаг **Включать в пользовательские настройки**, режим редактирования оставим в значении **Быстрый доступ**, [рис. 6.23](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.23). Это позволит нам вывести данный *параметр* в форму отчета, позволит пользователю выбирать нужный период перед построением отчета

[](https://www.intuit.ru/EDI/13_12_15_4/1449958784-19792/tutorial/1046/objects/6/files/06_23.jpg)

**Рис. 6.23.**Настройка вывода параметра

Запустим систему в режиме 1С:Предприятие и построим отчет, [рис. 6.24](https://www.intuit.ru/studies/courses/2318/618/lecture/13474?page=5#image.6.24).



**Рис. 6.24.**Готовый отчет по количественным и суммовым остаткам материалов

Перед построением отчета, если мы хотим задать *параметр* **Период**, установим флаг перед этим параметром и выберем нужную дату.