

Лабораторная работа 4.

Разработка механизма учета движения материалов.

Учет прихода и расхода материалов проводится в специальных бухгалтерских регистрах, которые называются Регистрами Накопления. Эти объекты являются **регистраторами** движения для документов.

Регистр накопления

Регистры накопления "отвечают" за учет движений ресурсов (финансов, товаров, материалов и т.д.). Они позволяют автоматизировать такие направления как складской учет, взаиморасчеты, планирование. В регистрах накопления хранится информация о поступлении и расходе тех или иных ресурсов, а функциональность этих объектов предоставляет возможность получения остатков на определенный момент времени, расчета итогов, кэширования итогов и т.д.

Когда информация, введенная в документ, записана в базу данных, мы уже можем с ней работать. Например, в течение недели в организацию поступали материалы, каждый раз это оформлялось с помощью соответствующего документа. Как можно узнать, сколько и каких материалов имеется в организации? Если информацию о поступивших материалах хранят лишь документы, то для того, чтобы узнать некие итоговые суммы, придется сначала собирать данные о материалах, просматривая все документы (или документы за какой-то период), потом эти данные обрабатывать. Просматривать документы для того, чтобы получить итоговую информацию о материалах, это слишком медленный процесс. Гораздо лучше было бы при проведении документа выписывать важнейшие данные этого документа в какую-нибудь таблицу, работать с которой быстрее и удобнее, чем со множеством документов. Такие таблицы в 1С:Предприятие называются **регистрами**.

Выше мы упоминали, что в 1С:Предприятие существуют различные регистры. В частности, это регистры накопления, регистры сведений, регистры расчета и регистры бухгалтерии. В данный момент нас интересуют **регистры накопления**. Именно регистры накопления отвечают за движение материальных и финансовых потоков.



Основное назначение регистра накопления – накопление числовой информации в разрезе нескольких измерений. Это объект подчиненный – он не предназначен для интерактивного редактирования и ввода информации. Виды числовой информации, хранимые в регистре накопления, называются **ресурсами**. Числовые данные связаны с конкретными значениями объектов конфигурации, которые здесь выступают как измерения.

Изменение состояния регистра накопления происходит, как правило, при проведении документа и заключается в том, что в регистр добавляется некоторое число записей, каждая запись содержит значение измерений, значение приращений ресурсов, ссылку на документ, который вызвал эти изменения. Документ в этом случае играет роль *регистратора*. Направление приращения – это приход или расход.

Создадим регистр Накопления **ОстаткиМатериалов**. Окно редактирования объекта регистра накопления устроено точно так же, как другие подобные окна. Зададим имя, синоним.

Регистр накопления ОстаткиМатериалов

Основные

Подсистемы

Функциональные опции

Данные

Регистраторы

Формы

Команды

Макеты

Права

Обмен данными

Прочее

Имя: ОстаткиМатериалов

Синоним: Остатки материалов

Комментарий:

Вид регистра: Остатки

Представление списка:

Расширенное представление списка:

Пояснение:

Действия <Назад Далее> Заккрыть Справка

Сделаем его видимым во всех подсистемах, кроме Расчета зарплаты.

Регистр накопления ОстаткиМатериалов

Основные

Подсистемы

Функциональные опции

Данные

Регистраторы

Формы

Команды

Макеты

Права

Обмен данными

Прочее

☒ Бухгалтерия

☐ РасчетЗарплаты

☒ УчетМатериалов

☒ ОказаниеУслуг

☒ Предприятие

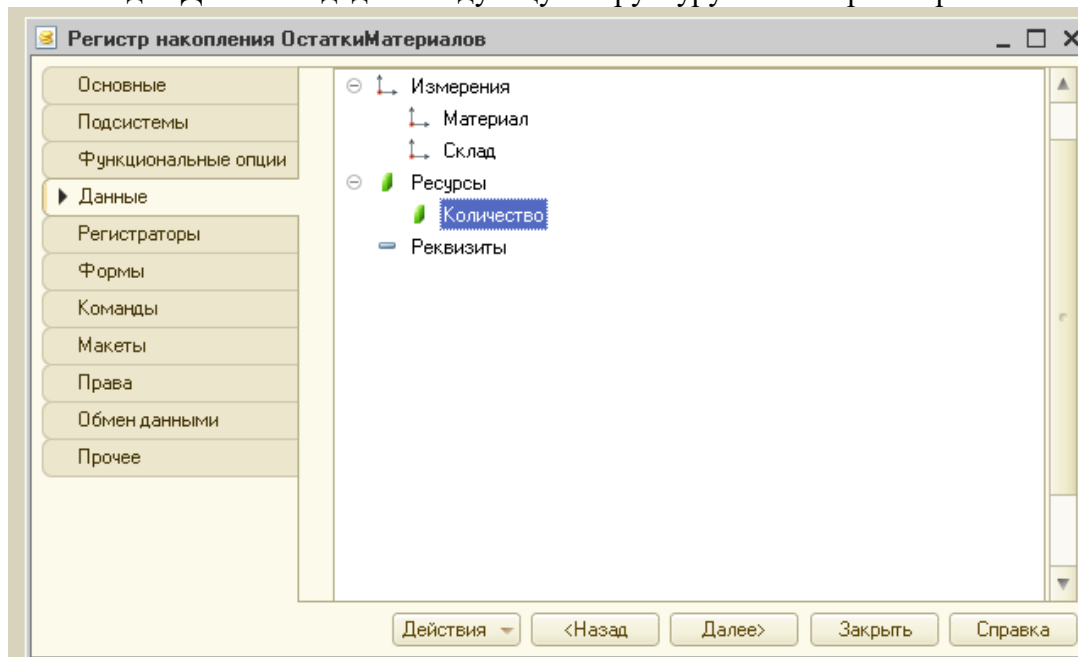
Действия <Назад Далее> Заккрыть Справка

В отличие от справочников для обеспечения видимости регистра накопления требуется явным образом указать это в командном интерфейсе подсистем. Для это перейдем в «Все подсистемы» и там укажем явным образом, что регистр видим в нужных подсистемах.

Регистр накопления должен хранить данные о всех изменениях, которые касаются материалов. Регистр накопления может иметь несколько измерений, в зависимости от того, по каким параметрам мы хотим вести учет, и может иметь несколько реквизитов. Мы хотим учитывать поступление и расход материалов. При этом нам важно знать на какой склад поступили материалы, или с какого склада они убыли.

Значит, измерениями для нашего регистра будут склад и материал. В качестве ресурса будем учитывать только количество поступившего или убывшего материала.

На закладке Данные зададим следующую структуру нашего регистра:

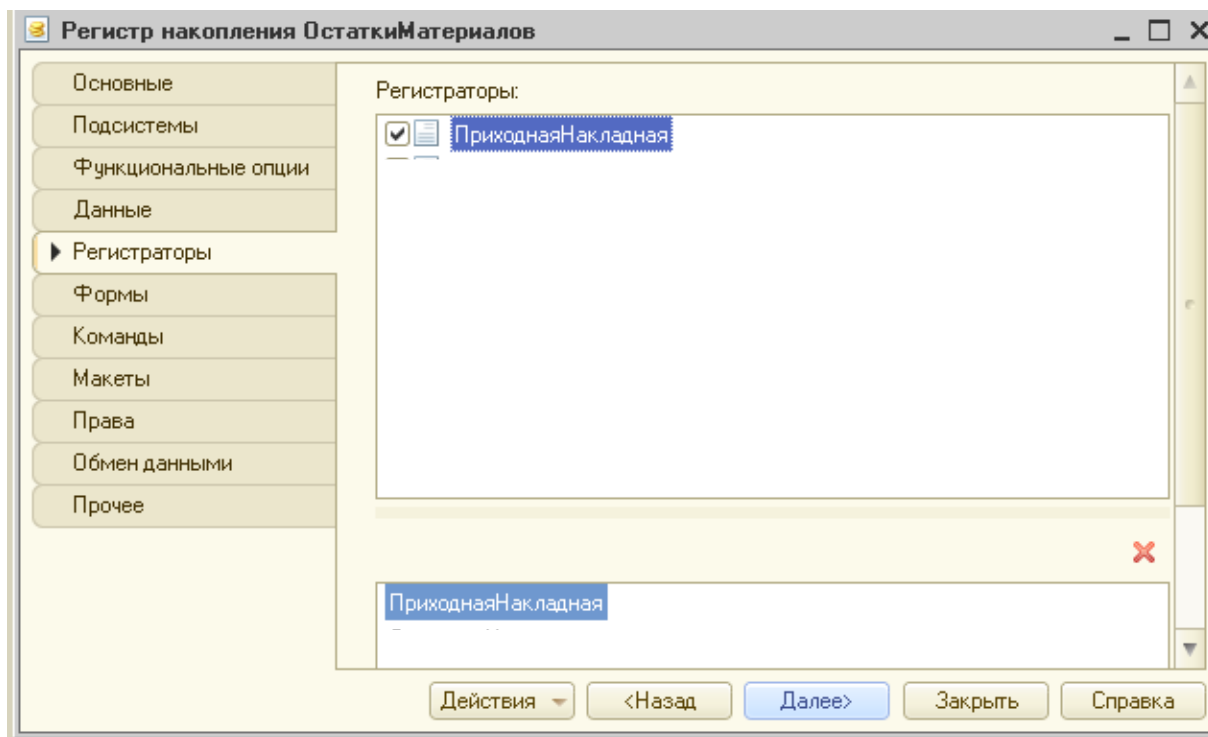


Материалы – ссылка на справочник Номенклатура

Склад – ссылка на Справочник Склады.

В ресурсы добавим **Количество** – число, целое, не отрицательное

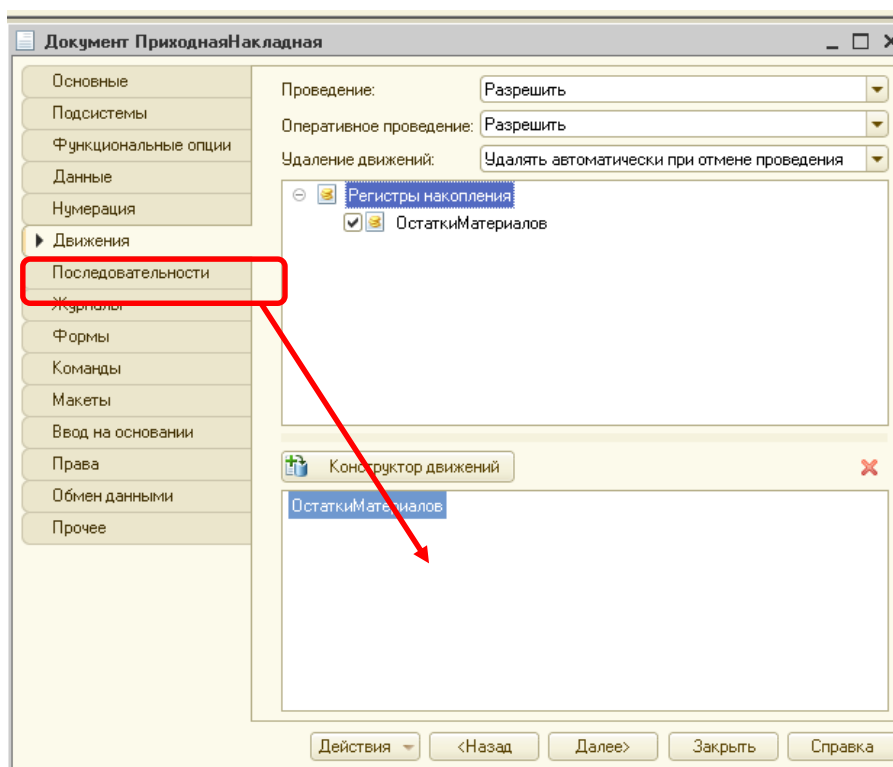
В качестве регистратора отметим созданный нами ранее документ: **Приходная накладная**



В ранее разработанных документах теперь надо определить алгоритм учета, который соответствует каждому документу.

Поэтому откроем в режиме редактирования документ Приходная накладная и перейдем на закладку Движение.

Отметим наш созданный регистр Накопления Остатки Материалов



Теперь нажмем Конструктор движений. В нем надо обязательно выбрать табличную часть, а заполнить поля требуемыми выражениями. Заполнять надо устанавливая курсор

на соответствующее поле в реквизитах документа и нажимая кнопку «Заполнить выражение»

Конструктор движения регистров

Регистры: РегистрНакопления.ОстаткиМатериалов

Тип движения регистра: ☒ Приход ☐ Расход

Табличная часть: **Материалы**

Реквизиты документа:

- Дата
- Номер
- Склад
- ТекСтрокаМатериалы.НомерСтроки
- ☒ ТекСтрокаМатериалы.Материал
- ТекСтрокаМатериалы.Количество
- ТекСтрокаМатериалы.Цена
- ТекСтрокаМатериалы.Сумма

Поле	Выражение
Материал	Дата
Склад	
Количество	

Кнопки: <Назад, Далее>, **Заполнить выражения**, Очистить выражения, ОК, Отмена, Справка

Конструктор движения регистров

Регистры: РегистрНакопления.ОстаткиМатериалов

Тип движения регистра: ☒ Приход ☐ Расход

Табличная часть: **Материалы**

Реквизиты документа:

- Дата
- Номер
- Склад
- ТекСтрокаМатериалы.НомерСтроки
- ☒ ТекСтрокаМатериалы.Материал
- ТекСтрокаМатериалы.Количество
- ТекСтрокаМатериалы.Цена
- ТекСтрокаМатериалы.Сумма

Поле	Выражение
Материал	ТекСтрокаМатериалы.Материал
Склад	Склад
Количество	ТекСтрокаМатериалы.Количество

Кнопки: <Назад, Далее>, Заполнить выражения, Очистить выражения, ОК, Отмена, Справка

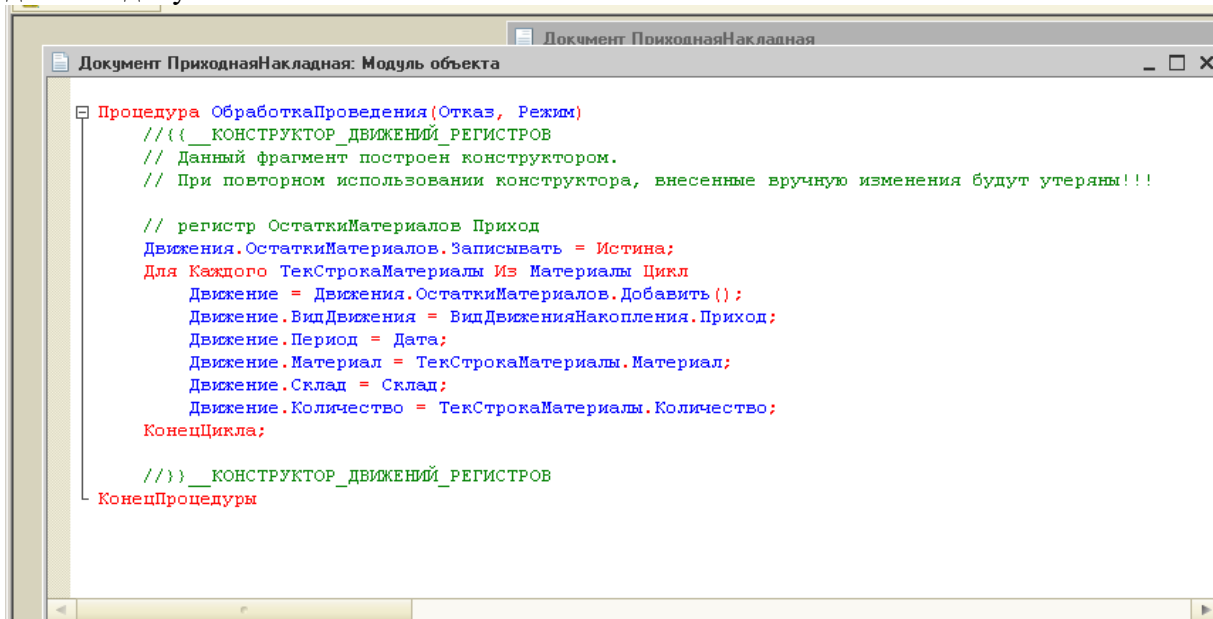
Обратите внимание, что по умолчанию у нас выбран тип движения Приход. Для данного документа он нам и нужен.

Проверим корректность заполнения полей:

- перенос значений можно выполнить двойным щелчком по соответствующему полю в реквизитах документа.

После ввода данных нажмем ОК.

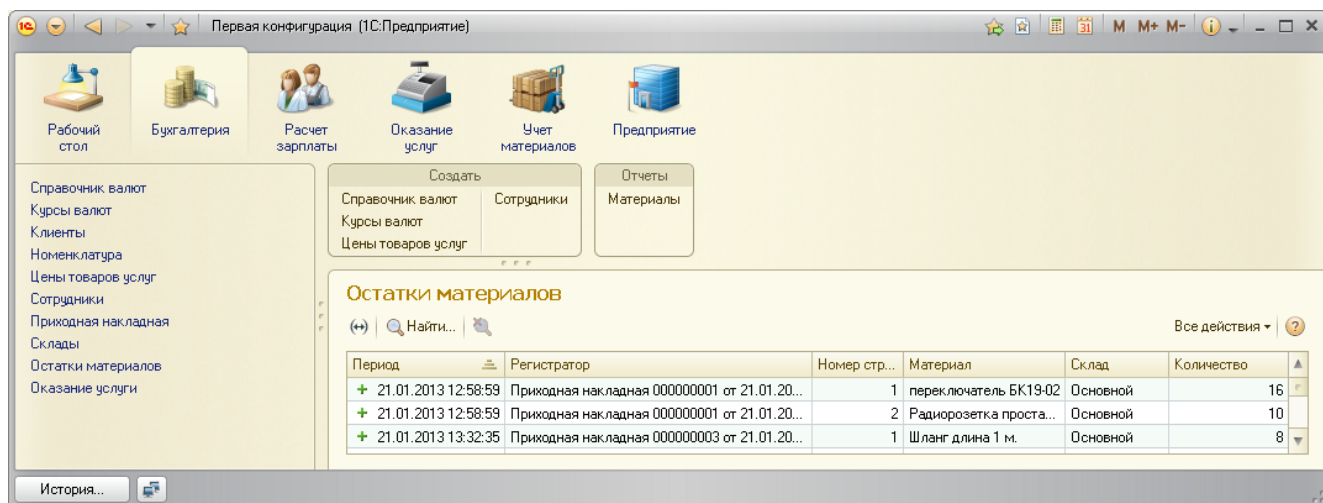
Нам будет выведена автоматически сгенерированная процедура учета движения для данного документа.



Метод Добавить – означает, что добавляется новая запись, так как есть параметр Для Каждого – это означает, что добавляется новая запись для каждой строки табличной части документа.

Вид движения Приход – означает, что запись в регистре Накопления будет отмечена знаком + (плюс.)

Проверим работу нашего регистра накопления, перейдем в режим исполнения и проведем ранее введенные документы повторно.



Мы видим, что появились несколько строк в регистре накопления, в нашем примере две первые строки относятся к одному документу.

Ранее у нас был разработан документ Оказание услуги. Ранее необходимо было добавить в документ вторую табличную часть Материалы. Если это не было выполнено ранее, то сделать это сейчас.

Добавить в документ ОказаниеУслуги вторую табличную часть, назвав ее Материалы.
Добавить в нее реквизиты

- Материал – ссылка на справочник Номенклатура
- Количество – число, целое неотрицательно;
- Цена – число целое, точность 2 цифры после запятой, неотрицательное;
- Сумма- число целое, точность 2 цифры после запятой, неотрицательное;

Обеспечить механизм расчета суммы по ранее реализованному алгоритму.

Теперь добавим учет использования материалов при оказании услуг в регистре накопления ОстаткиМатериалов. В отличие от алгоритма обработки Приходной накладной укажем, что в данном случае тип движения – Расход.

На закладке Движение документа ОказаниеУслуг переходим в режим конструктора движений, устанавливаем требуемые параметры и проверяем сгенерированный код процедуры.

Конструктор движения регистров

Регистры

- РегистрНакопления.ОстаткиМатериалов

Реквизиты документа

- Дата
- Номер
- Склад
- Клиент
- Мастер
- ТекСтрокаМатериалы.НомерСтроки
- ТекСтрокаМатериалы.Материал
- ТекСтрокаМатериалы.Количество
- ТекСтрокаМатериалы.Цена
- ТекСтрокаМатериалы.Сумма

Тип движения регистра: ☐ Приход ☒ Расход

Табличная часть: Материалы

Поле	Выражение
Материал	ТекСтрокаМатериалы.Материал
Склад	Склад
Количество	ТекСтрокаМатериалы.Количество

<Назад Далее>

Заполнить выражения

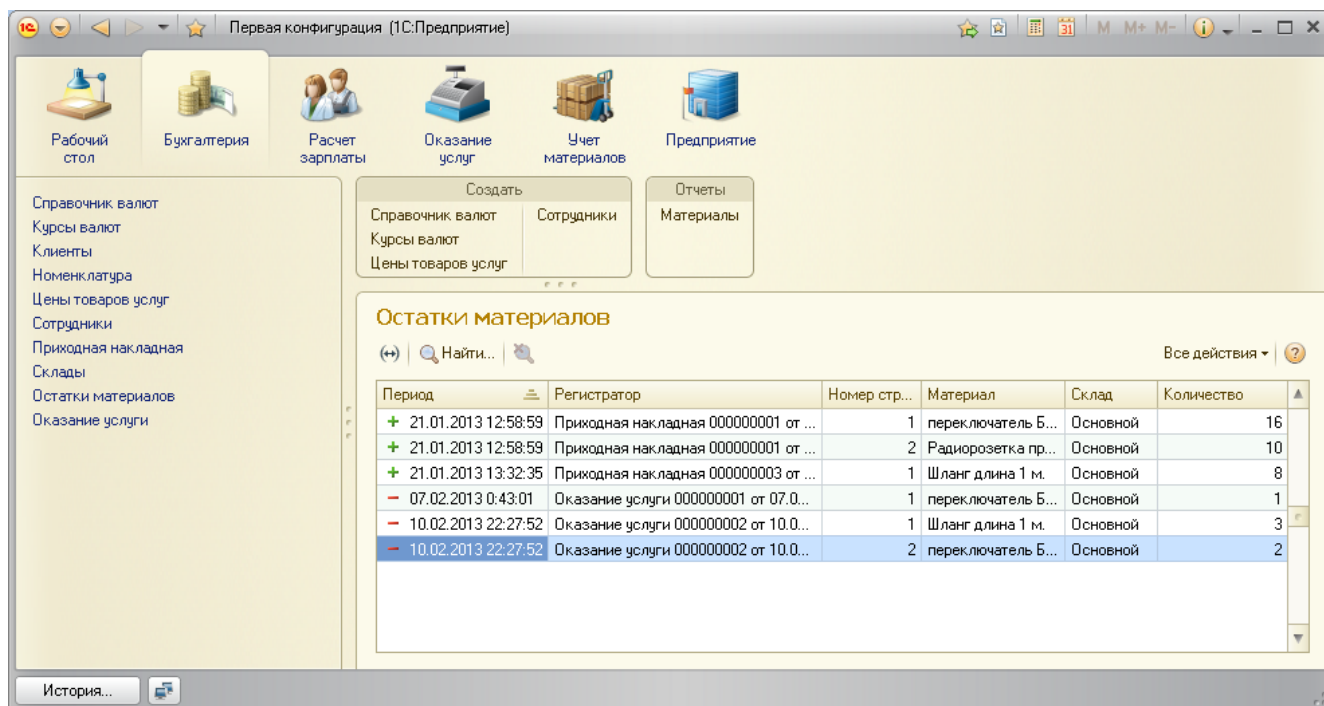
Очистить выражения

ОК

Отмена

Справка

Все остальное без изменений. Перейдите в режим исполнения и создайте несколько новых документов Учет услуг. После этого откройте регистр Остатки Материалов и отметьте, что теперь в нем появились записи со знаком – (минус)

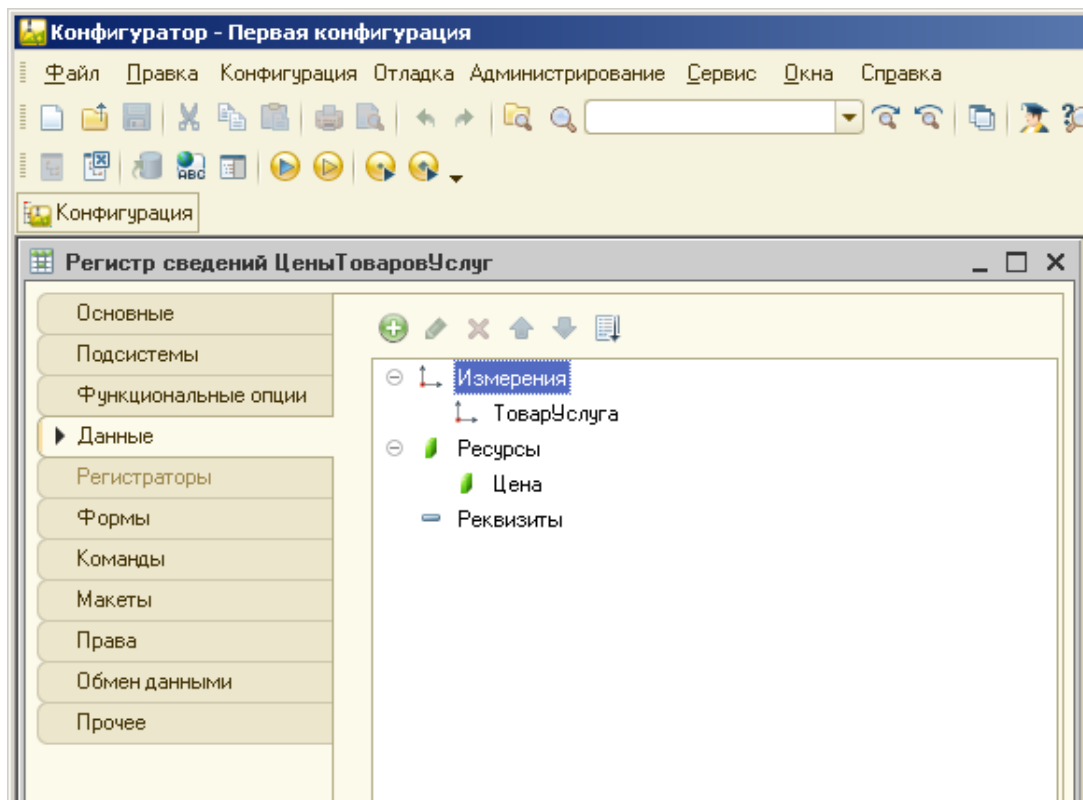


Разработка универсальной функции обращения к регистру сведений.

Для того чтобы учитывать изменение ценовой политики, и чтобы у нас при формировании документа Оказание услуг автоматически подставлялась последняя цена, надо написать специальную функцию.

Создадим периодический регистр Сведений ЦеныТоваровУслуг, который будет хранить цены на товары и услуги, предоставляемые нашей фирмой. Регистр независимый, периодический, период месяц.

Структура этого регистра следующая



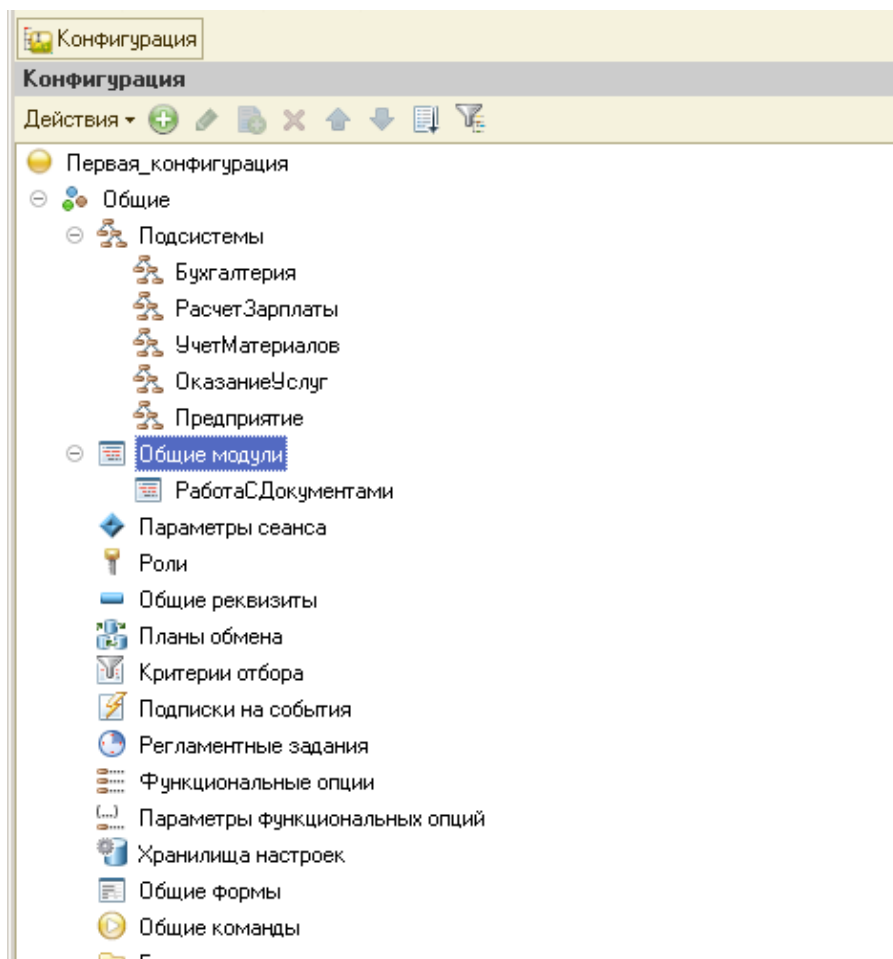
Измерение ТоварУслуга – это ссылка на справочник Номенклатура.

Разработаем функцию получения актуальной цены на материалы. Эту функцию можно будет применить при формировании документов.

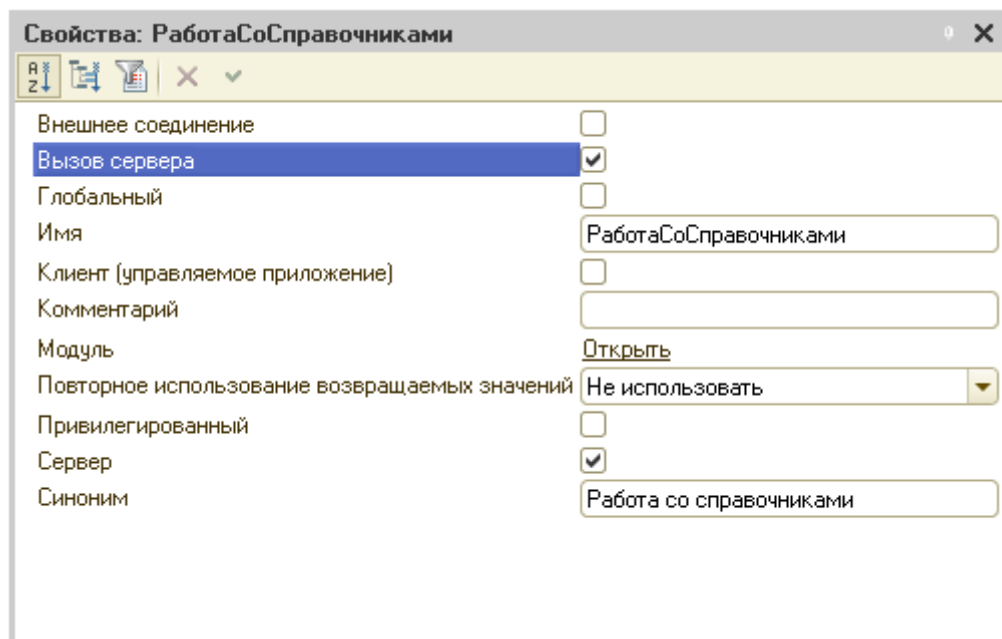
Создадим функцию РозничнаЦена(). Поместим ее в общий модуль, чтобы она была доступна для всех объектов. Функция должна автоматически подставлять последнюю актуальную цену на товар или услугу. Актуальной будет последняя цена на дату, по которой проводится документ. Если у нас есть 2 цены на товар 1, при этом на 01.01.2015 была задана цена 1000, а на 01.02.2015 – цена 1200, то при проведении документа 15.01.2015 в наш документ автоматически должна быть поставлена цена 1000, а вот при создании нового документа 10.02.2015 нам уже подставится цена 1200.

Теперь перейдем к созданию данной функции.

Откроем конфигуратор

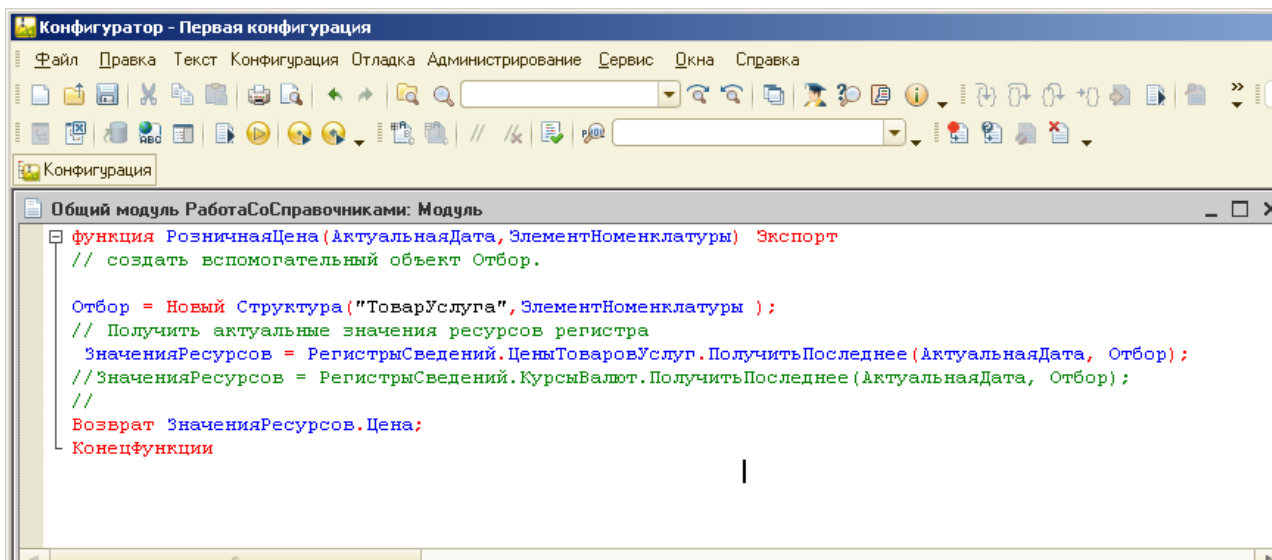


Добавим новый модуль РаботаСоСправочниками



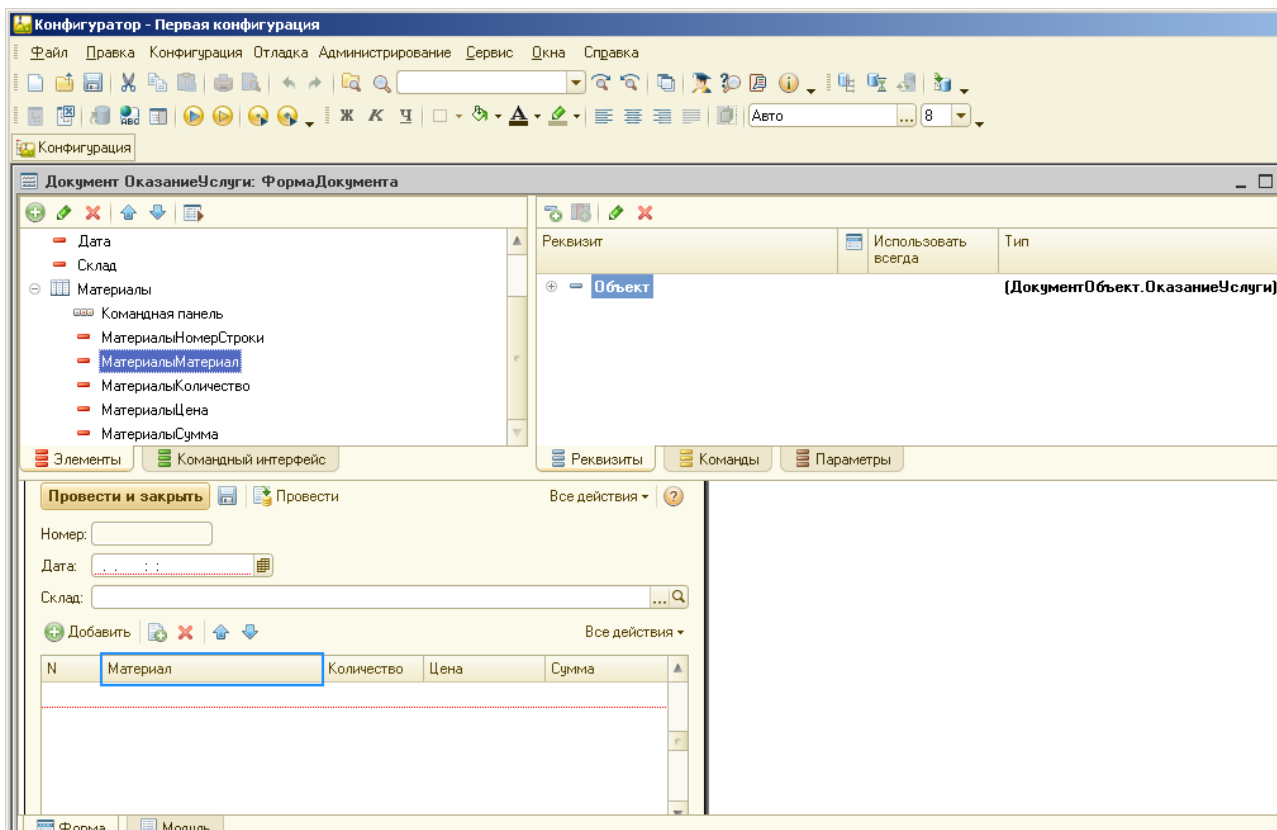
Установим галочку Вызов сервера, чтобы экспортные процедуры и функции и модуля можно вызвать с клиента.

Поместим в него следующий текст:



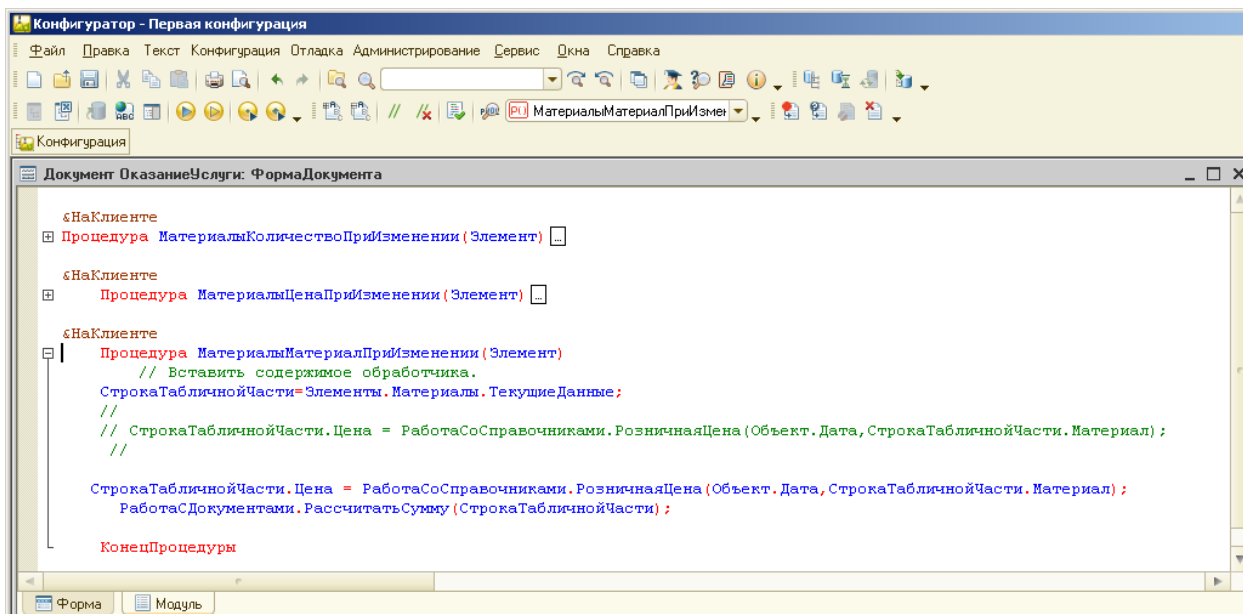
Здесь “ТоварУслуга” – это имя измерения в РегистреСведений, в котором мы храним цены на наши товары и услуги. ЗначениеРесурсов.Цена – это последняя цена на заданное значение измерения.

Откроем форму документа ОказаниеУслуги.



Предполагаем, что актуальная цена должна подставляться сразу как только мы выберем товар.

Поэтому выбираем поле МатериалыМатериал и событие ПриИзменении и создаем новую процедуру на клиенте:



Теперь заполним регистр сведений ЦеныТоваровУслуг и проверим, как работают все наши разработанные функции.

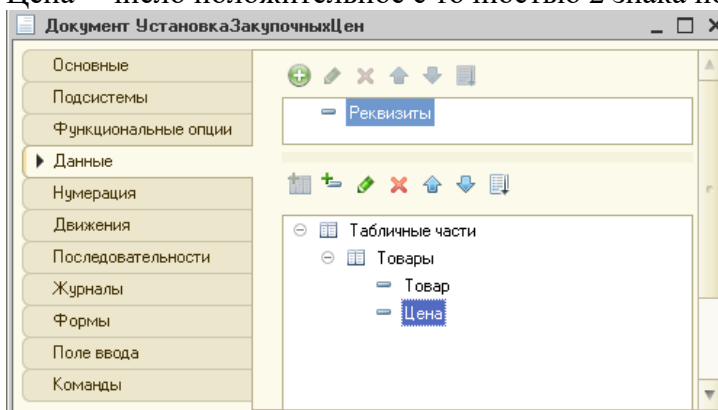
Проверить можно ли добавить новую запись в регистр Сведений и в этом же сеансе работы.

Задание для самостоятельной работы.

Предполагаем, что цены на закупку товаров иные, чем при продаже их в рамках предоставляемых нашей организацией услуг. Кроме того в реальном бизнесе цены устанавливаются специальными документами.

1. Создать 2 документа УстановкаЗакупочныхЦен и УстановкаРозничныхЦен. Документа содержат табличную часть Товары, состоящую из 2-х атрибутов:

- Товар- ссылка на справочник Номенклатура
- Цена – число положительное с точностью 2 знака после запятой.



2. Создать 2 регистра сведений ЗакупочныеЦены и РозничныеЦены. Оба регистра подчинены регистраторам, в качестве которых выступают документы УстановкаЗакупочныхЦен и УстановкаРозничныхЦен. Регистры сведений являются периодическими, но их периодичность задается позицией регистратора.

Регистр сведений ЗакупочныеЦены

Основные

Имя: ЗакупочныеЦены

Синоним: Закупочные цены

Комментарий:

Периодичность: По позиции регистратора

Режим записи: Подчинение регистратору

Основной отбор по периоду

Представление записи:

Расширенное представление записи:

Представление списка:

Действия <Назад Далее> Закрывать Справка

Регистр сведений ЗакупочныеЦены

Регистраторы:

ПриходнаяНакладная

ОказаниеУслуг

УстановкаЗакупочныхЦен

Действия <Назад Далее> Закрывать Справка

3. Перейти на закладку Движение в документе УстановкаЗакупочныхЦен и обеспечить создание кода движения через Конструктор движений в регистре Закупочные цены.

Документ УстановкаЗакупочныхЦен

Основные

Подсистемы

Функциональные опции

Данные

Нумерация

Движения

Последовательности

Журналы

Формы

Поле ввода

Команды

Макеты

Проведение: Разрешить

Оперативное проведение: Разрешить

Удаление движений: Удалять автоматически при отмене

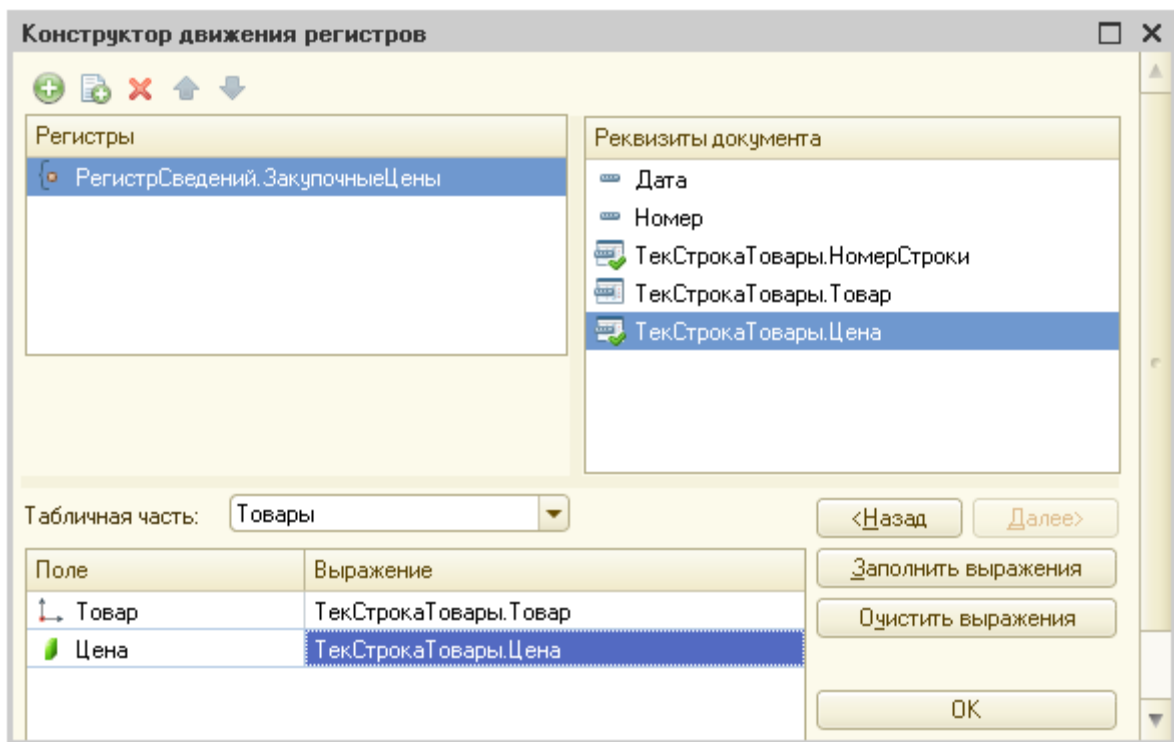
Регистры сведений

ЗакупочныеЦены

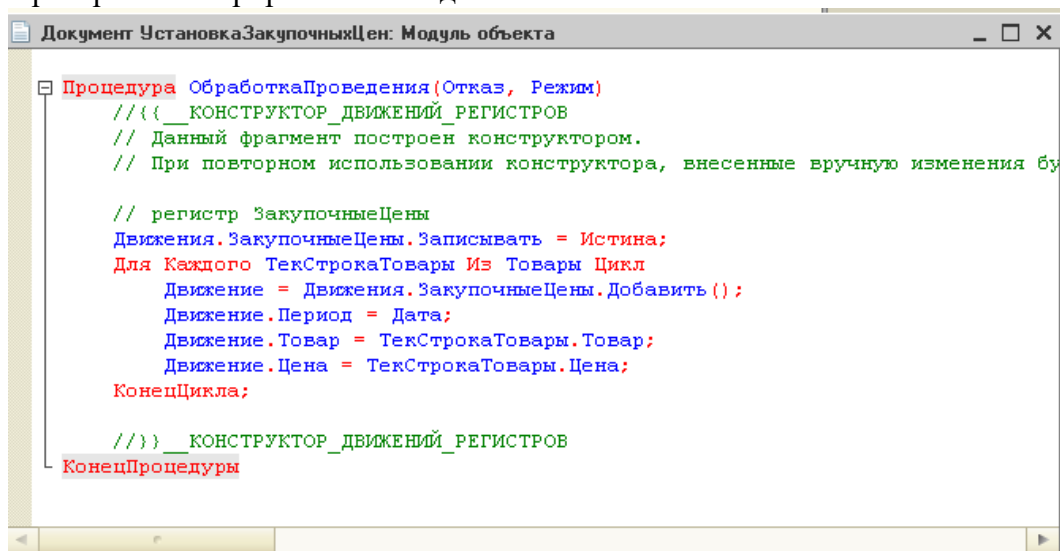
Конструктор движений

ЗакупочныеЦены

Действия <Назад Далее> Закрывать Справка



Проверьте сгенерированный код



4. Разработайте 2 новые функции: ЗакупочнаяЦена и РозничнаяЦена по аналогии с ранее разработанной функцией Розничная цена, указав там ссылку на вновь созданные регистры сведений.
5. Проверьте работоспособность механизмов подстановки разных цен в разные документы.

Подготовьте отчет, представив в отчете тексты разработанных функций, разработанный интерфейс, примеры формирования цен при создании документов. Прокомментируйте подстановку разных цен и подтвердите этот механизм соответствующими скриншотами.

Сохраните отчет, сохраните разработанную конечную конфигурацию

Отправьте отчет и разработанную конфигурацию преподавателю.

Предъявите работающую конфигурацию преподавателю, получите зачет по данной практической работе.