Министерство труда, занятости и трудовых ресурсов Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С.Галущака»

|  |  |  |
| --- | --- | --- |
| УТВЕРЖДАЮ  Председатель цикловой комиссии специальности 09.02.03  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Протокол № \_\_ от  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_г | УТВЕРЖДАЮ  Председатель цикловой комиссии специальности 09.02.03  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Протокол № \_\_ от  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г | УТВЕРЖДАЮ  Председатель цикловой комиссии специальности 09.02.03  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Протокол № \_\_ от  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_г |

Методические указания к практическим работам 1-15

ПМ 02 Разработка и администрирование баз данных

Учебная дисциплина: МДК. .02.03 Программирование 1С

Специальность: 09.02.03 Программирование в компьютерных системах

Разработал:

И.Л. Борисов

2016

**Содержание**

[1. Цели 3](#_Toc125663828)

[4.Форма отчета 7](#_Toc125663829)

[5.Критерии оценок 8](#_Toc125663830)

[6.Постановка задачи 9](#_Toc125663831)

[7.Выполнение задание в 1С 12](#_Toc125663832)

[8.Хранилище значения 52](#_Toc125663833)

# Цели

В ходе выполнения работы студенты осваивают общие и профессиональные компетенции:

ОК 1 понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес

ОК 2 Организовывать собственную деятельность, выбирать типовые методы и способы выполнения поставленных задач, оценивать их эффективность и качество

ОК 3 Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность

ОК 8 Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

1.1 Результаты обучения

Усвоенные знания:

* основные положения теории баз данных, хранилищ данных, баз знаний
* основные принципы построения концептуальной, логической и физической; модели данных;
* современные инструментальные средства разработки схемы базы данных;
* методы описания схем баз данных в современных СУБД;
* структуры данных СУБД, общий подход к организации представлений, таблиц, индексов и кластеров;
* методы организации целостности данных;
* способы контроля доступа к данным и управления привилегиями;
* основные методы и средства защиты данных в базах данных;
* модели и структуры информационных систем;
* основные типы сетевых топологий, приемы работы в компьютерных сетях;
* информационные ресурсы компьютерных сетей;
* технологии передачи и обмена данными в компьютерных сетях;
* основы разработки приложений баз данных.

**2.Дидактическое обеспечение**

Источники:

* лекционный материал.

**3.Порядок выполнения**

3.1 Ознакомиться с содержанием задания и сроками сдачи

3.2 Ознакомиться с методическими рекомендациями по выполнению работы

3.3 При необходимости получить консультацию преподавателя

# 4.Форма отчета

4.1 На практическом занятии студенту предлагается ответить на поставленные вопросы при защите выполненной работе.

# 5.Критерии оценок

5.1 «Отлично»– студент правильно отвечает на поставленные вопросы в установленные сроки;

5.2 «Хорошо» – студент отвечает на поставленные вопросы, но допускает 2-3 неточности или работа сдана с задержкой в 1 неделю;

5.3 «Удовлетворительно» – студент отвечает на поставленные вопросы, но допускает 4-5 неточностей, или работа сдана с задержкой в 2 недели;

5.4 «Неудовлетворительно» - студент не готов отвечать на поставленные вопросы

# 6.Постановка задачи

Организация ООО «КанцОптТорг» занимается продажей канцелярских товаров.

Необходимо автоматизировать основные торговые процессы данного предприятия. Организация состоит из одного юридического лица и одного склада. Создать иерархические справочники:

* номенклатура;
* контрагенты;
* сотрудники.

Для контрагента необходимо определить тип отношений:

* клиент;
* поставщик;
* прочее (например, банк или налоговая инспекция).

При этом контрагент может иметь одновременно несколько типов отношений (например, является и поставщиком, и покупателем). Для каждого контрагента необходимо указывать ответственного менеджера.

Создать документ «Поступление товаров», который отражает закупку товаров у поставщика. В документе должен быть реквизит «Ответственный», который содержит сотрудника, отвечающего за этот документ. Создать документ «Реализация товаров», отражающий продажу товаров. Документ также должен содержать реквизит «Ответственный». Обеспечить ввод реализации на основании поступления товаров. Создайте журнал документов «Товарные документы». Сформируйте печатную форму «Реализации товаров». На начальной странице системы отобразите данные журнала документов «Товарные документы».

Создать регистр накопления «Остатки товаров». Регистр должен содержать информацию об остатках номенклатуры. Создать регистр накопления для учета данных о количестве и сумме закупок в разрезе номенклатуры и поставщиков. Создать независимый регистр для учета цен номенклатуры.

Необходимо выполнить проверку, что в документе «Реализация товаров» выбран контрагент, у которого указан флаг «клиент». В формах документов обеспечить автоматический расчет суммы в табличной части, как произведение цены и количества. Также учесть ситуацию изменения суммы – в этом случае нужно пересчитать цену.

Обеспечить контроль остатков для оперативно проводимых документов «Реализация товаров». Также решить задачу расчета себестоимости при продаже товаров. Обеспечить учет суммы выручки, количества проданного товара и списанной себестоимости в оборотном регистре накопления «Продажи».

Создать план счетов «Бух. учет», определить в нем счета, необходимые для отражения операций:

* поступление товаров;
* отгрузка товаров;
* оплата клиента и оплата поставщику.

Создать регистр бухгалтерии для отражения проводок по бухгалтерскому учету. Создать универсальный документ «Бухгалтерская операция». Обеспечить закрытие месяца, необходимо сформировать проводки вида:

Дт Выручка − Кт Прибыль на сумму остатка выручки

Дт Прибыль − Кт Себестоимость на сумму остатка себестоимости

Выполнить закрытие месяца.

Создать план видов расчета «Начисления», в нем будут содержаться виды расчетов, имеющие протяженность во времени. Создать предопределенные элементы – Оклад по тарифу и Невыход сотрудника. Создать документ «Начисление оклада» и «Невыход сотрудника», определить структуру этих документов. Создать регистр расчета «Начисления». Он будет хранить данные о начисленной заработной плате в разрезе сотрудников. Создать регистр сведений, описывающий график работы предприятия. Указать связь регистра расчета с графиком.

Обеспечить расчет оклада сотрудника по дневном тарифу. Отработанные дни месяца определяются методом отклонений. Создать новый вид расчета «Премия суммой» - данный вид расчета не зависит ни от каких данных. Обеспечить начисление премий сотрудникам, указанным в табличной части, вновь созданного документа. Отразить процессы выплаты заработной платы. Для этого потребуется создание регистра накопления.

Необходимо реализовать следующие отчеты:

* создать отчет «Закупки товаров». Пользователь должен иметь возможность задавать период формирования отчета;
* необходимо создать отчет «Остатки товаров по данным документа». Отчет должен выводить остатки по товарам на указанную пользователем дату;
* создать отчет «Валовая прибыль». В нем должна быть представлена информация в виде таблицы. Также в этом отчете построить гистограмму, где в качестве точек выбрана номенклатура, а в качестве серий – контрагентыя;
* Создать отчет «Остатки по счетам БУ». Отчет должен выводить остаток по выбранному счету на указанную пользователем дату. Если счет не задан – должны получаться остатки по всем счетам;
* Создать отчет «Задолженность по зарплате», показывающий на определенную дату, долги перед сотрудниками;
* Создать отчет «Отработанное время» с использованием диаграммы Ганта.;
* Построить отчет, показывающий ABC-классификацию номенклатуры по параметру «Валовая прибыль». Введите такой набор данных о продаже, чтобы в каждый класс входило, как минимум, 2 номенклатуры.

# 7.Выполнение задание в 1С

7.1 Структура базы данных

* 6 подсистем (Продажи, Закупки, Регистры, БухгалтерскийУчет, Зарплата, БизнесПроцесс);
* 2 констант (НаимОрг, НаимСклада);
* 3 справочника (Номенклатура, Контрагенты, Сотрудники);
* 9 документов (ПоступлениеТовара, РеализацияТовара, БухгалтерскаяОперация, ЗакрытиеМесяца, ЗаполненияГрафикаПредприятия, НачислениеОклада, НевыходСотрудника, НачислениеПремии, ВыплатаЗаработнойПлаты);
* 1 перечисления (Вакансии);
* 7 отчетов (ЗакупкиТоваров, ОстаткиТоваров, ВалваяПрибыль, ОстаткиПоСчитамБУ, ЗадолжностьПередСотрудниками, ОтработанноеВремя, КлассификацияНоменклатуры);
* 1 Плана счетов (БухУчет);
* 1 Плана видов расчета (Начисления);
* 2 Регистров сведений (ЦеныНоменклатуры, ГрафикПредприятия);
* 4 Регистра накопления (Остатки, Закупки, Продажи, ЗадолжностьПередСотрудниками);
* 1 Регистра бухгалтерии (РегистрБухУчет);
* 1 Регистра расчета (ЗаработнаПлата).

7.2 Составление элементов конфигурации, определение подсистем, констант, создание и заполнение справочников.

7.2.1 Создание подсистем. В создаваемой конфигурации необходимо будет реализовать шесть «направления»: продажи, закупки, бухгалтерский учет, зарплата, бизнес процессы и служебное направление регистры. Для организации отборов по «направлениям» при разработке конфигурации можно использовать подсистемы. Создаем подсистемы: “Продажи”, “Закупки”, “БухгалтерскийУчет”, “Зарплата”, “БизнесПроцесс”, “Регистры”. Подсистемы выступают как средство дополнительной классификации объектов конфигурации (с возможностью отбора). На структуру объектов ветви «Подсистемы» не накладывается никаких ограничений по числу и вложенности.

Для добавления подсистемы необходимо внутри ветви «Общие» щелкнуть правой клавишей мыши и в открывшемся контекстном меню выбрать «Добавить». Заполняем поле “Имя” к каждой из подсистем и создаем картинку для лучшего восприятия системы. Для добавления картинки нужно заранее подготовить изображение (размер 48Х48), затем в поле “Картинка” нажать на троеточие, по кнопке “Добавить” заполняем поле “Имя” и по кнопке “Выбрать из файла” выбираем подходящее изображение. Затем сохраняем его, нажимаем “ОК” Рисунок 1.

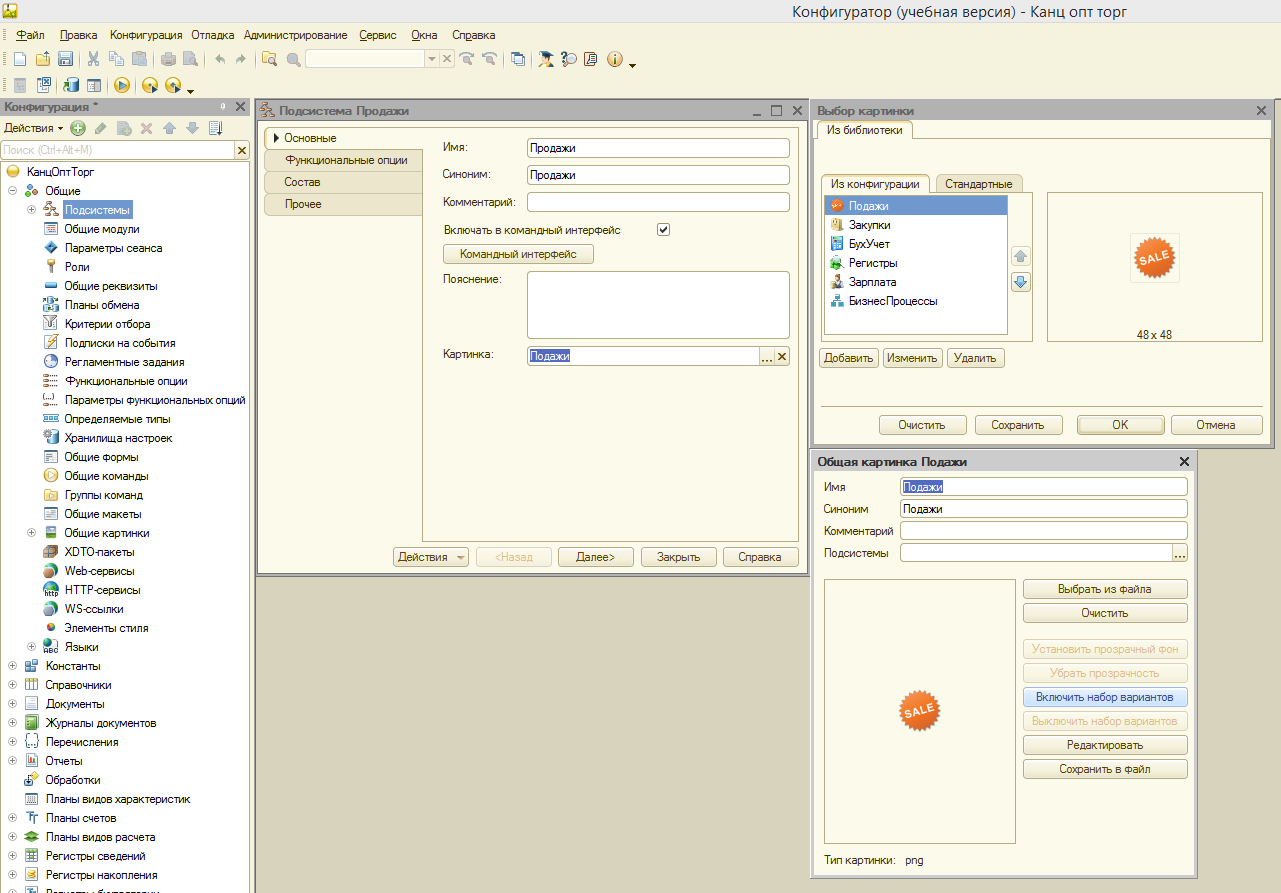


Рисунок 1 – Этапы создания подсистем (на примере подсистемы “Продажи”).

Далее проделываем тоже самое и с остальными подсистемами. После запуска получаем такой внешний вид Рисунок 2:

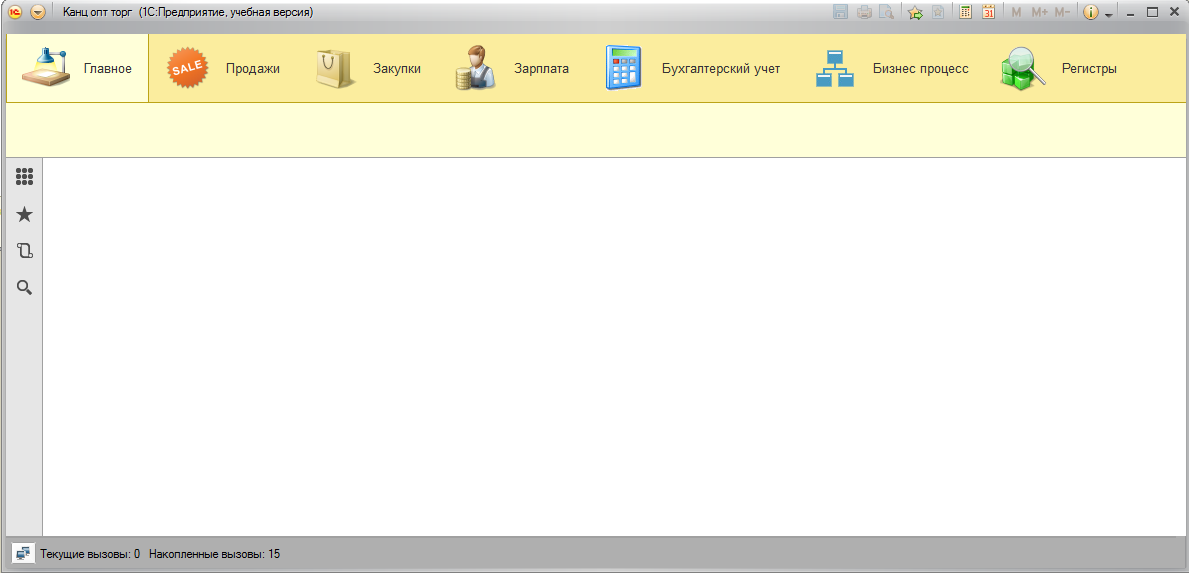


Рисунок 2 – Внешний вид базы в пользовательском режиме.

7.2.2 Создание справочников. Справочник – список объектов однотипной информации. Справочники предназначены, для того чтобы сгруппировать все списки и однозначно идентифицировать объект. Создаем иерархические справочники:

* номенклатура;
* контрагенты;
* сотрудники.

Для создания справочника необходимо на ветви “Справочники” щелкнуть правой клавишей мыши и в открывшемся контекстном меню выбрать «Добавить». Заполняем поле “Имя” к каждому справочнику, на закладке “Подсистемы” ставим галочку на против подсистемы в которую нужно вписать справочник. На закладке “Иерархия” ставим галочку напротив поля “Иерархический справочник”.

Для справочника “Контрагенты” необходимо определить реквизиты. Для этого на закладке “Данные” щелкнуть правой клавишей мыши по реквизиту и в открывшемся контекстном меню выбрать «Добавить».

Для контрагента необходимо определить тип отношений:

* клиент;
* поставщик;
* прочее (например, банк или налоговая инспекция).

При этом контрагент может иметь одновременно несколько типов отношений (например, является и поставщиком, и покупателем). Для этого в справочнике можно определить несколько реквизитов с типом «Булево». Для каждого контрагента необходимо указывать ответственного менеджера. Для этого создаем реквизиты “Клиент”, “Поставщик”, “Прочее” c типом «Булево», “Сотрудник” с типом ссылка на справочник “Сотрудники” Рисунок 3 и заполним данными справочники.

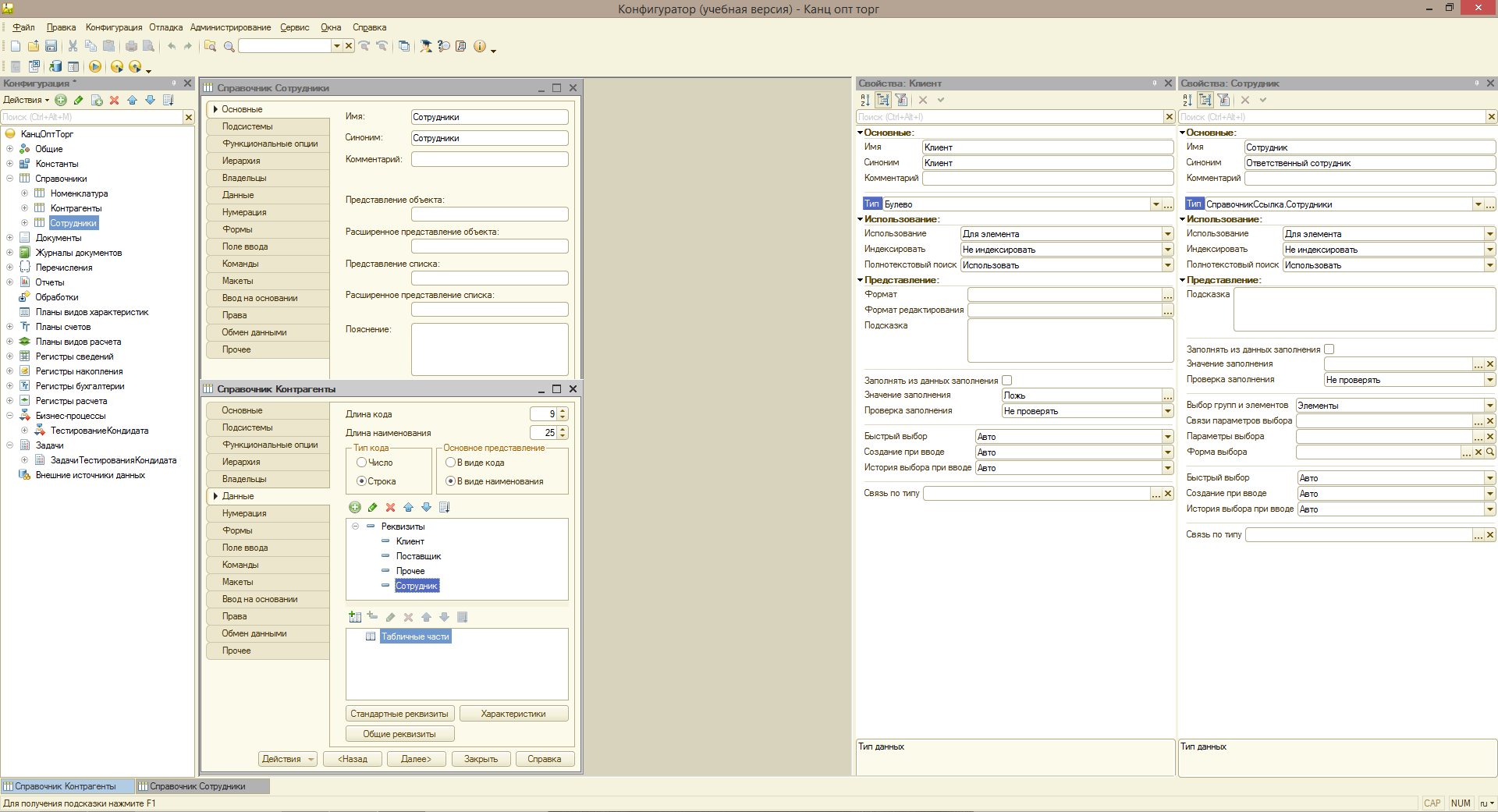


Рисунок 3 – Окна редактирования справочников.

7.2.3 Создание констант. Константы – позволяют хранить в информационной базе данные, которые не изменяются во времени, или изменяются очень редко. Каждая константа позволяет хранить одно значение. Создаем две константы “НаимОрг”, “НаимСклада”.

7.3 Создание и заполнение документов “ПоступлениеТовара” и “РеализацияТовара”. Документы позволяют хранить в прикладном решении информацию о совершенных хозяйственных операциях или о событиях, произошедших в "жизни" предприятия вообще.

7.3.1 Документ “ПоступлениеТовара” отражает закупку товаров у поставщика. Документ содержит реквизиты “Контрагент”, “Ответственный”, который содержит сотрудника, отвечающего за этот документ и “СуммаДокумента”, который содержит сумму всех купленных товаров у поставщика. Также документ содержит табличную часть “Товары”, который содержит информацию о номенклатуре, количестве, цены и суммы номенклатуры отраженных в документы в виде реквизитов табличной части.

7.3.2 Документ “РеализацияТовара”, отражающий продажу товаров. Его можно создать путем копирования документа “ПоступлениеТовара”, для этого необходимо на ветви “Справочники” щелкнуть правой клавишей мыши на документ “ПоступлениеТовара” далее нажать на клавиатуре клавиши CTRL + C, затем нажать клавиши CTRL + V. Переименовать документ в “РеализацияТовара” Рисунок 4.

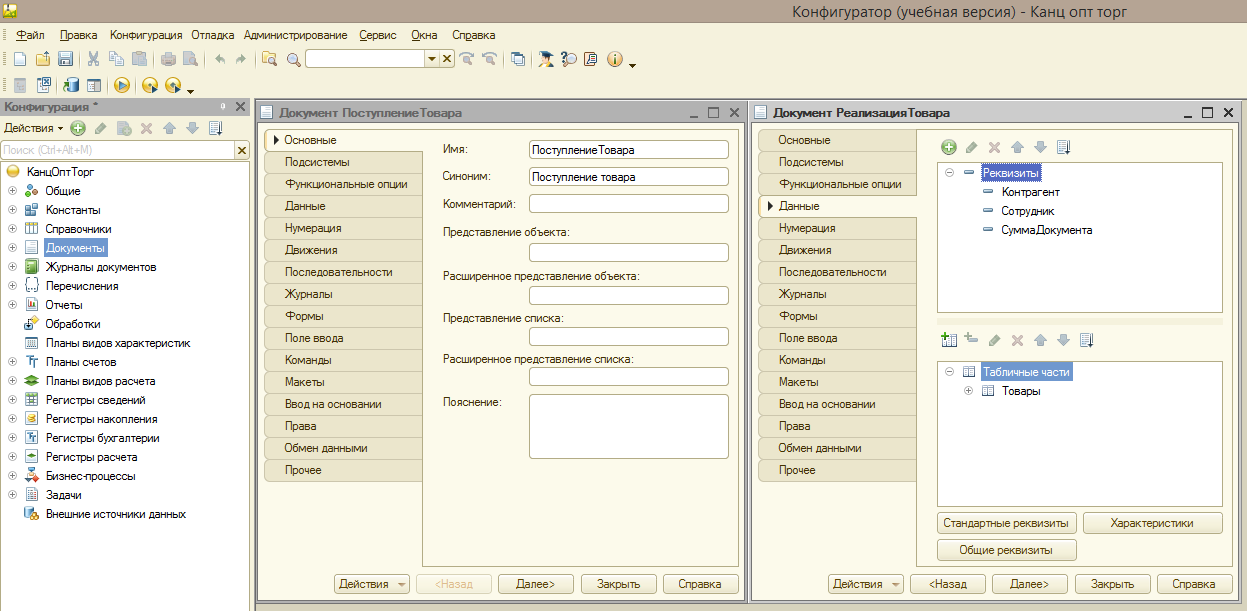


Рисунок 4 – Структура документов “ПоступлениеТовара” и “РеализацияТовара”.

Нужно обеспечьте ввод документа “РеализацияТовара” на основании документа “ПоступлениеТовара”. Для этого в окне редактирования документа “РеализацияТовара” на закладке “Ввод на основании” в поле “Вводится на основании” нажать на зеленый карандаш и проставить галочку на документе “ПоступлениеТовара” жмем “ОК”. Далее наживаем на кнопку “Конструктор ввода на основании” поможет составить процедуру ввода на основании, для этого жмем на кнопку “Заполнить выражение” жмем “ОК” Рисунок 5 и заполним данными документы.

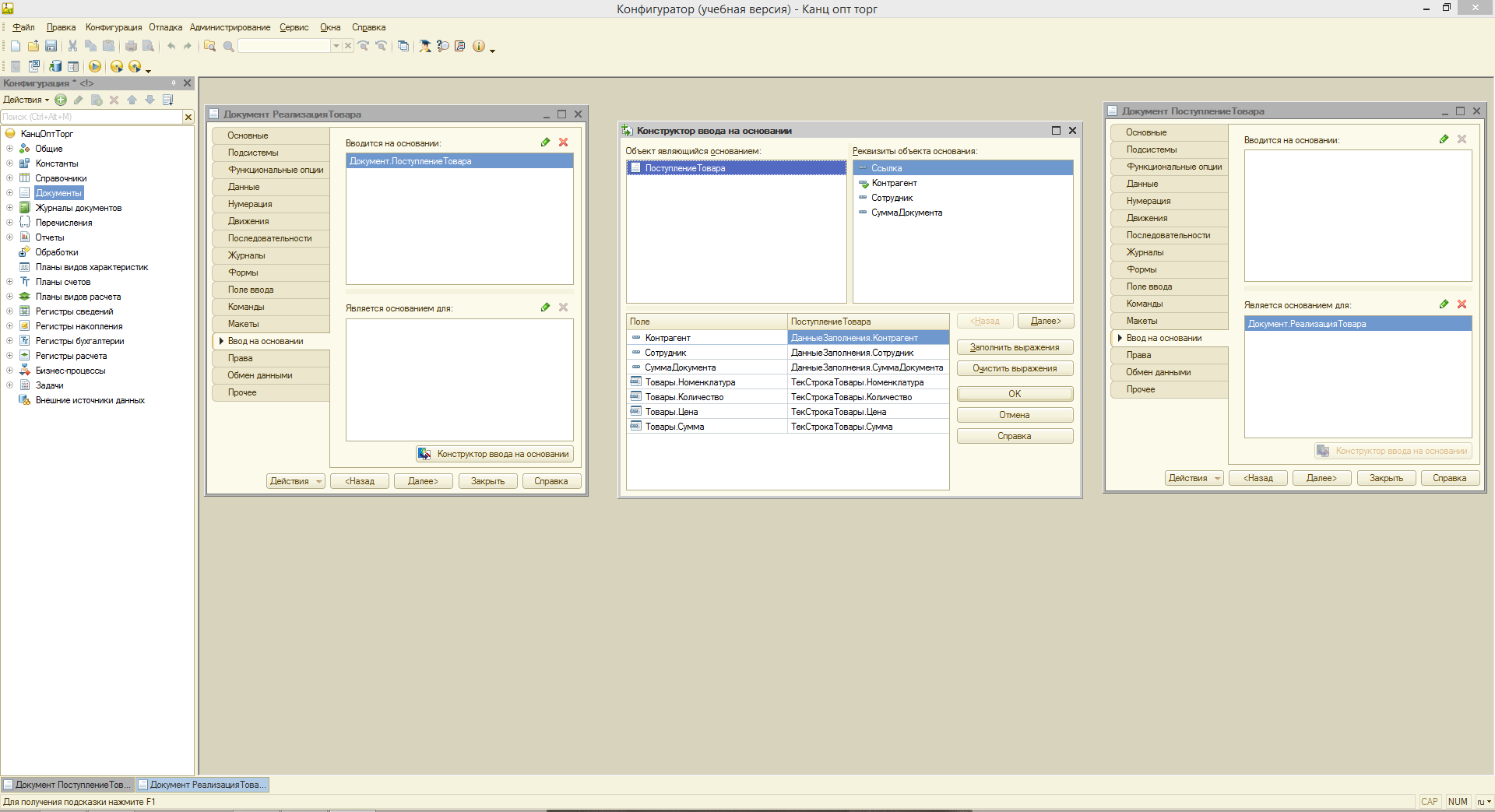


Рисунок 5 – Ввод документа “РеализацияТовара” на основании документа “ПоступлениеТовара”.

Создание обработчиков событий.

Для обработки автоматической обработки ввода товара создадим форму документа и для полей цена, количество, сумма *–* обработчики событий: ПриИзмененииКоличества, ПриИзмененииЦены и ПриИзмененииСуммы что показдано на Рисунок 6:

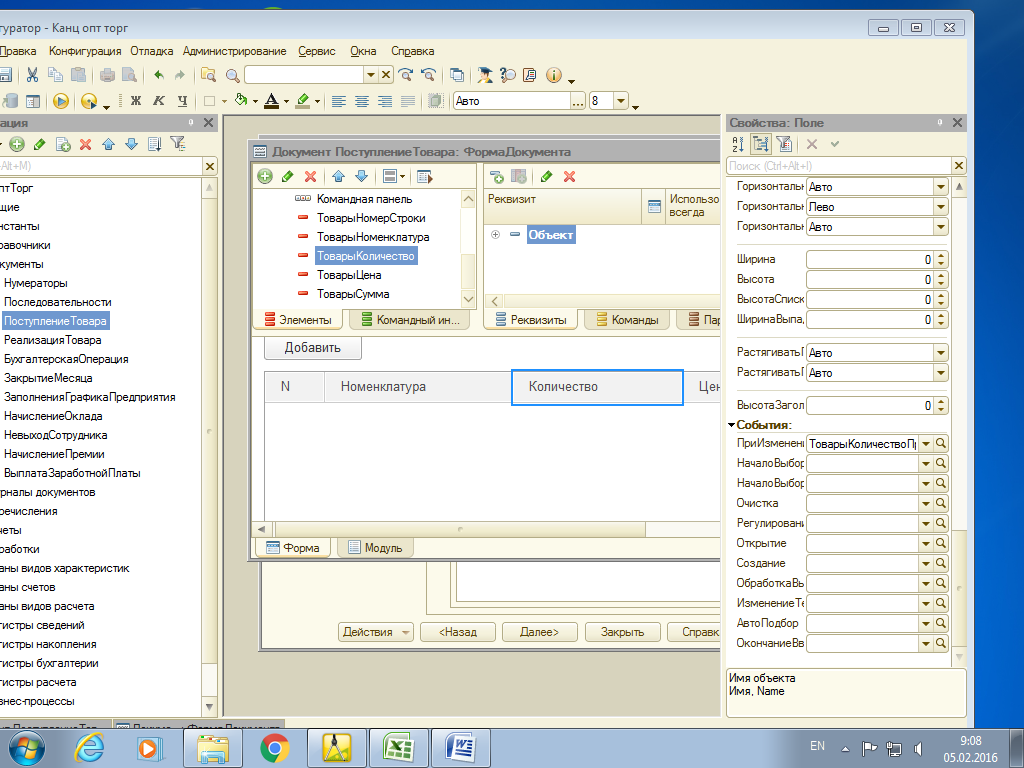
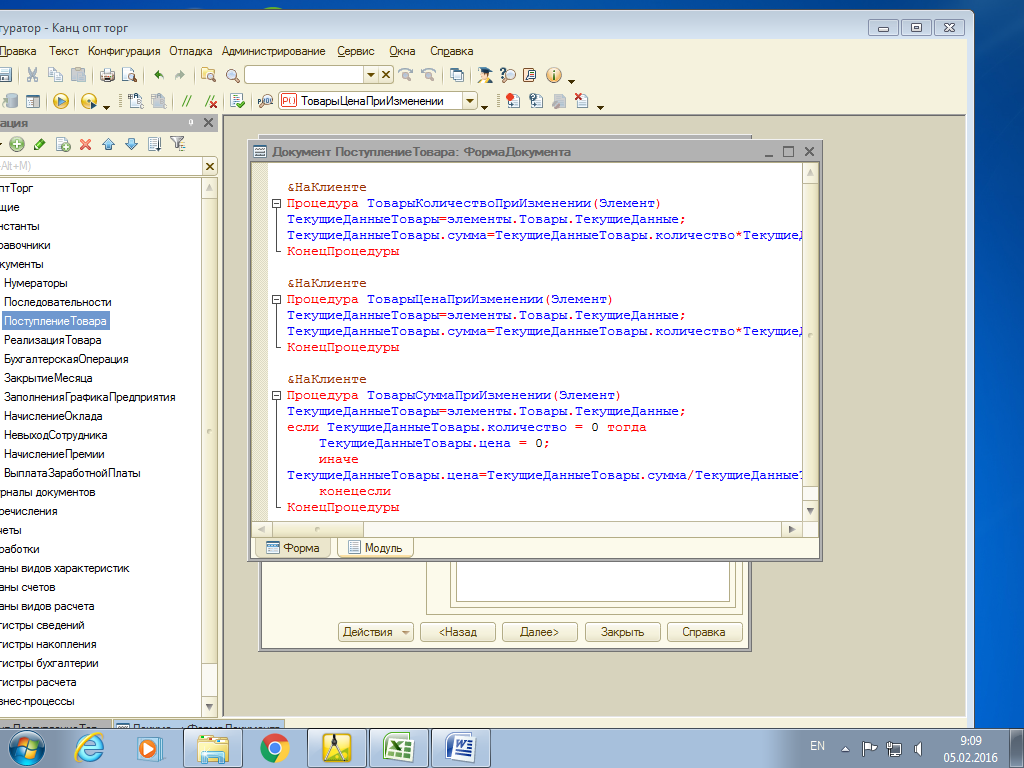


Рисунок 6 *–* Создадим текс обработчиков в модуле формы



Код обработчиков:

&НаКлиенте

Процедура ТоварыКоличествоПриИзменении(Элемент)

ТекущиеДанныеТовары=элементы.Товары.ТекущиеДанные;

ТекущиеДанныеТовары.сумма=ТекущиеДанныеТовары.количество\*ТекущиеДанныеТовары.цена;

КонецПроцедуры

&НаКлиенте

Процедура ТоварыЦенаПриИзменении(Элемент)

ТекущиеДанныеТовары=элементы.Товары.ТекущиеДанные;

ТекущиеДанныеТовары.сумма=ТекущиеДанныеТовары.количество\*ТекущиеДанныеТовары.цена;КонецПроцедуры

&НаКлиенте

Процедура ТоварыСуммаПриИзменении(Элемент)

ТекущиеДанныеТовары=элементы.Товары.ТекущиеДанные;

если ТекущиеДанныеТовары.количество = 0 тогда

ТекущиеДанныеТовары.цена = 0;

иначе

ТекущиеДанныеТовары.цена=ТекущиеДанныеТовары.сумма/ТекущиеДанныеТовары.количество;

конецесли

КонецПроцедуры

7.3.3 Журнал документов «Товарные документы», в котором должны быть графы «Контрагент» и «Ответственный». Создаем Журнал документов «Товарные документы» на закладке “Данные”, в поле “Регистрируемые документы” указываем что в данный журнал входят два документа “ПоступлениеТовара” и “РеализацияТовара”. Добавляем графы «Контрагент» ссылка на реквизиты документов «Контрагент» и граф «Ответственный», ссылка на реквизиты документов «Ответственный» что показано на Рисунок 7.

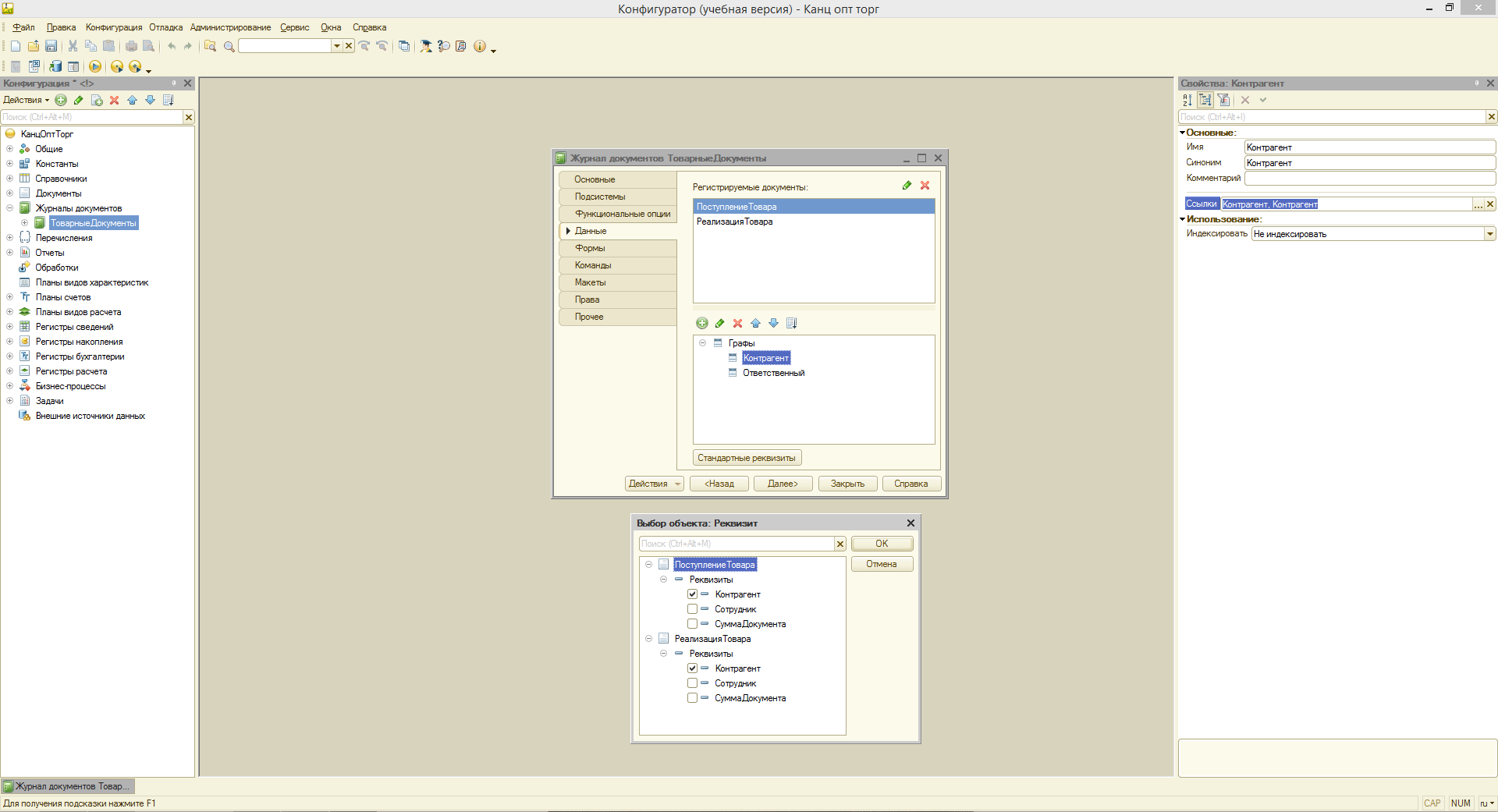


Рисунок 6 – Журнал документов «Товарные документы».

7.3.4 Сформировать печатную форму документа «Реализации товаров». Для этого в окне редактирования документа «Реализации товаров» на закладке “Макеты” нажать на кнопку “Конструкторы” и выбрать “Конструктор печати”. Далее создаем имя нажимаем “Далее”, указываем какие реквизиты будут печататься, нажимаем “Далее”, указываем какие реквизиты табличной части будут печататься, нажимаем “ОК” Рисунок 7.

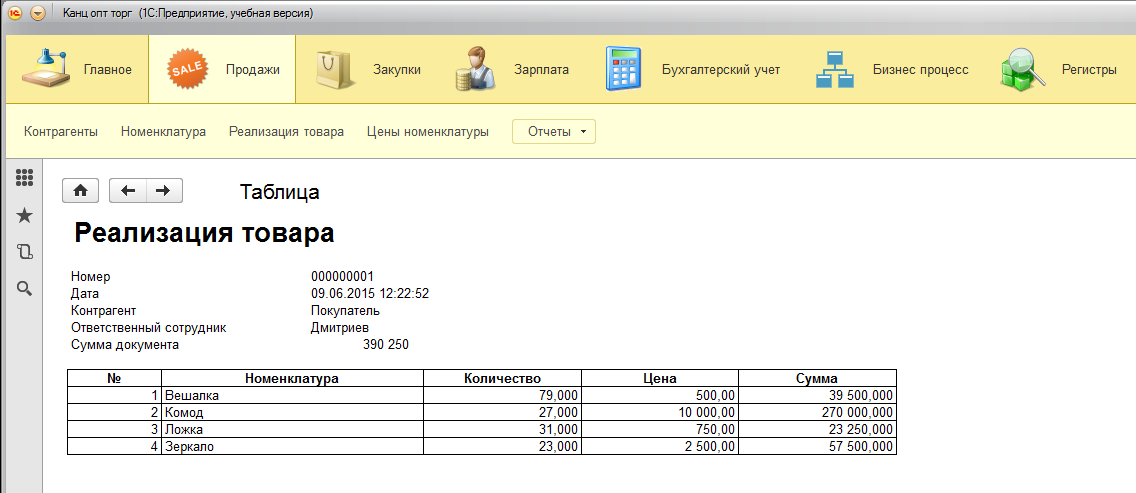


Рисунок 7 – Печатная форма документа “РеализацияТовара”.

7.4 Создание регистра накопления «Остатки товаров». Регистр накопления образует многомерную систему измерений и позволяет "накапливать" числовые данные в разрезе нескольких измерений. Регистр должен содержать информацию об остатках номенклатуры.

7.4.1 Создаем регистр накопления внутри ветви “Регистры накопления”, называем его “Остатки”, вид регистра “Остатки”. Определяем структуру регистра, для этого в закладке “Данные”, создаем одно измерение “Номенклатура” указываем тип данных “справочник номенклатура” и ресурсы “Количество”, “СуммаСебестоимости” и “СебестоимостьЕдиницы” Добавляем один реквизит «выручка» тип данных – число. Как показано на Рисунок 8.

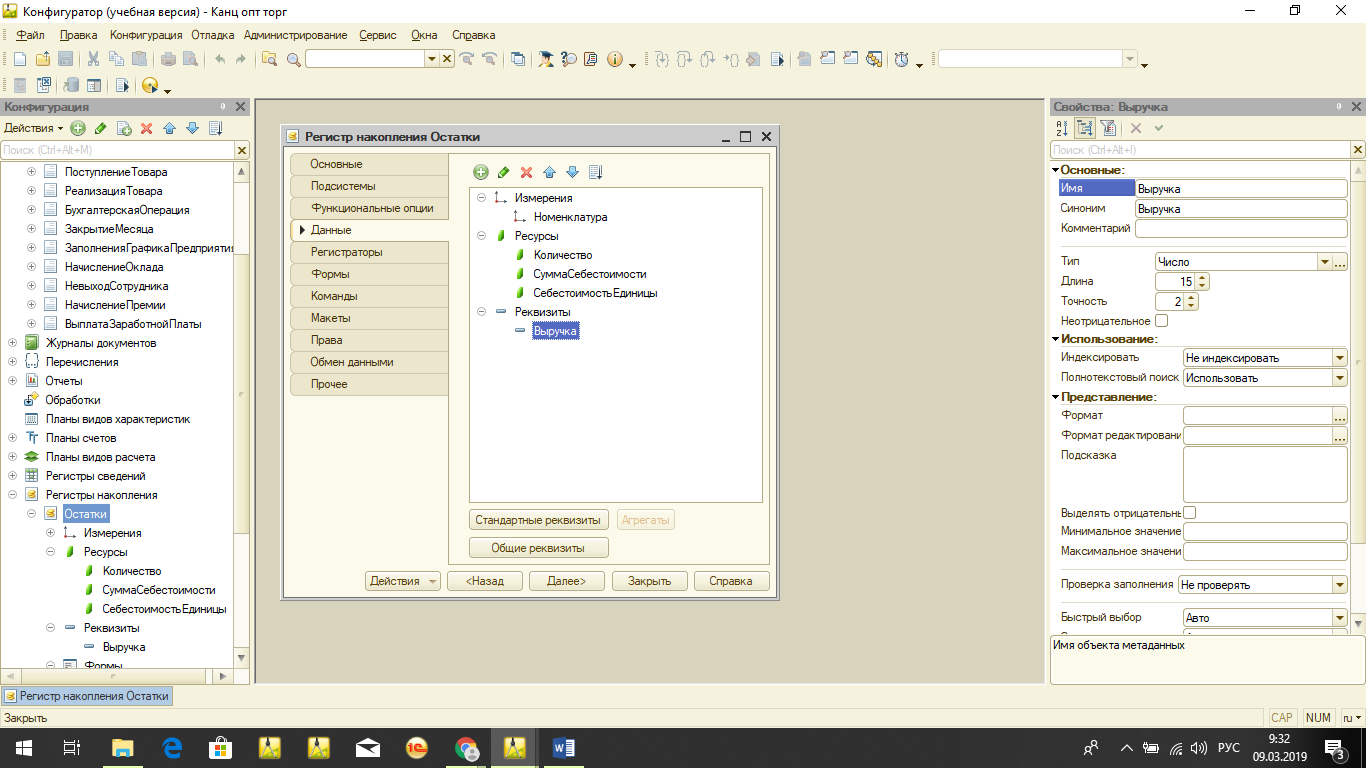


Рисунок 8 – Структура регистра накопления “Остатки”.

7.4.2 Отмечаем что регистраторами для этого регистра выступают и документ “ПоступлениеТовара”, и документ “РеализацияТовара”. Для этого в окне редактирования регистра “Остатки” в закладке “Регистраторы” ставим галочку на против этих документов. Теперь нужно обеспечить проведение документов «Реализация товаров» и «Поступление товаров» по этому регистру. Для того чтобы разработать программный код проведения необходимо воспользоваться конструктором движений, располагается он на закладке “Движения”, в окне редактирования документов. Создаем движения регистра накопления нажимаем кнопку “Конструктор движения”. Указываем табличную часть “Товары”, для документа тип движения “ПоступлениеТовара” будет “Приход”, для “РеализацияТовара” – “Расход”, нажимаем кнопку “Заполнить выражения”, для незаполненных полей вписываем текст вручную как указано на Рисунок 9, нажимаем “ОК”.

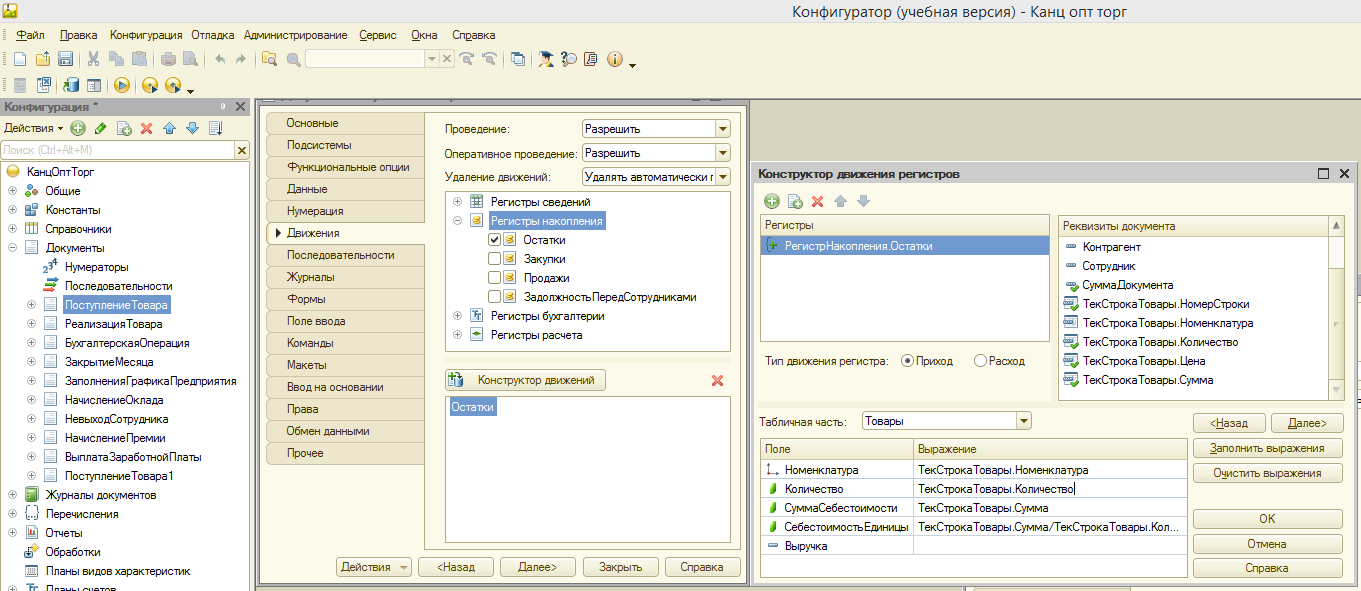


Рисунок 9 – Создание движения регистра накопления “Остатки” (на примере документа “ПоступлениеТовара”).

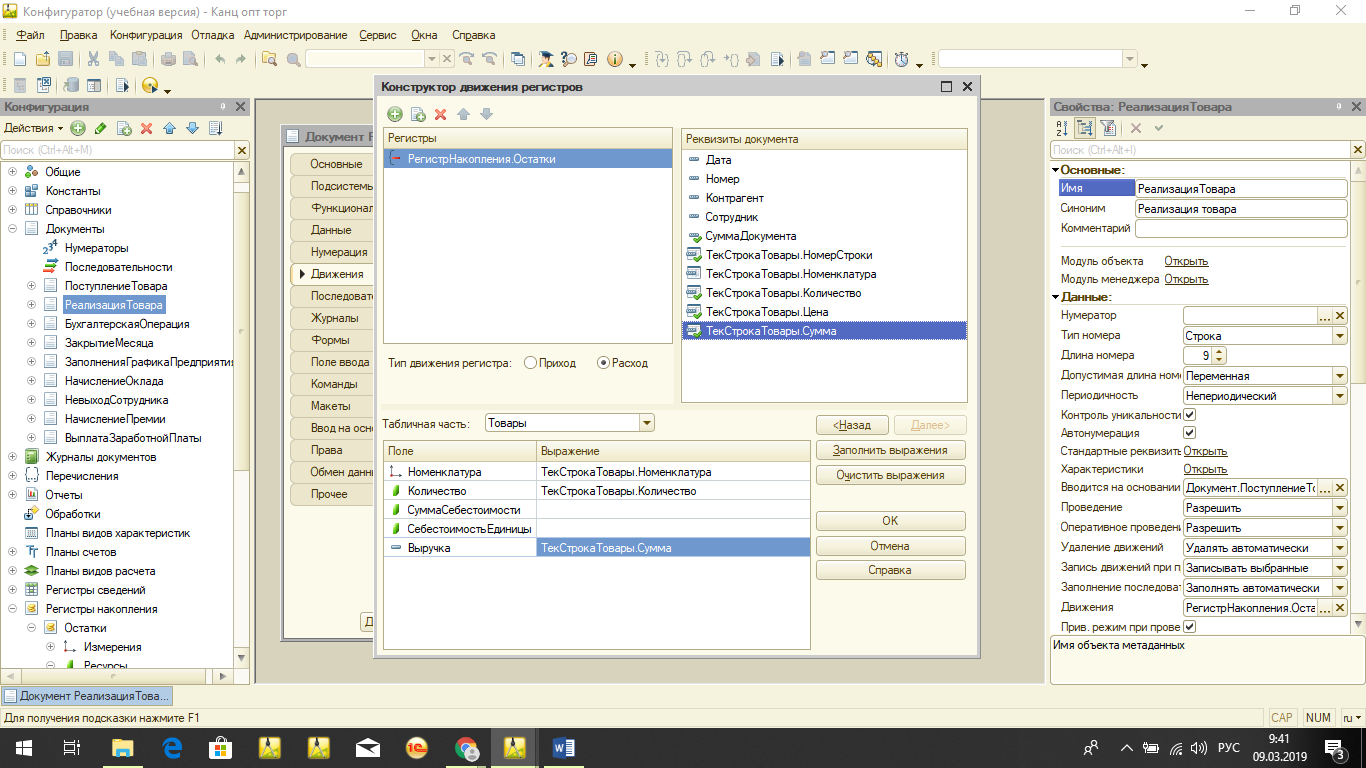


Рисунок 9-1 – Создание движения регистра накопления “Остатки” (на примере документа “РеализацияТовара”).

7.4.3 Добавим команду отображения движений на формы документов. Заходим во вкладку “Командный интерфейс”, разворачиваем группу перейти и указываем, что для “Остатков” будет видимость. Тоже самое делаем и для другого документа Рисунок 10.

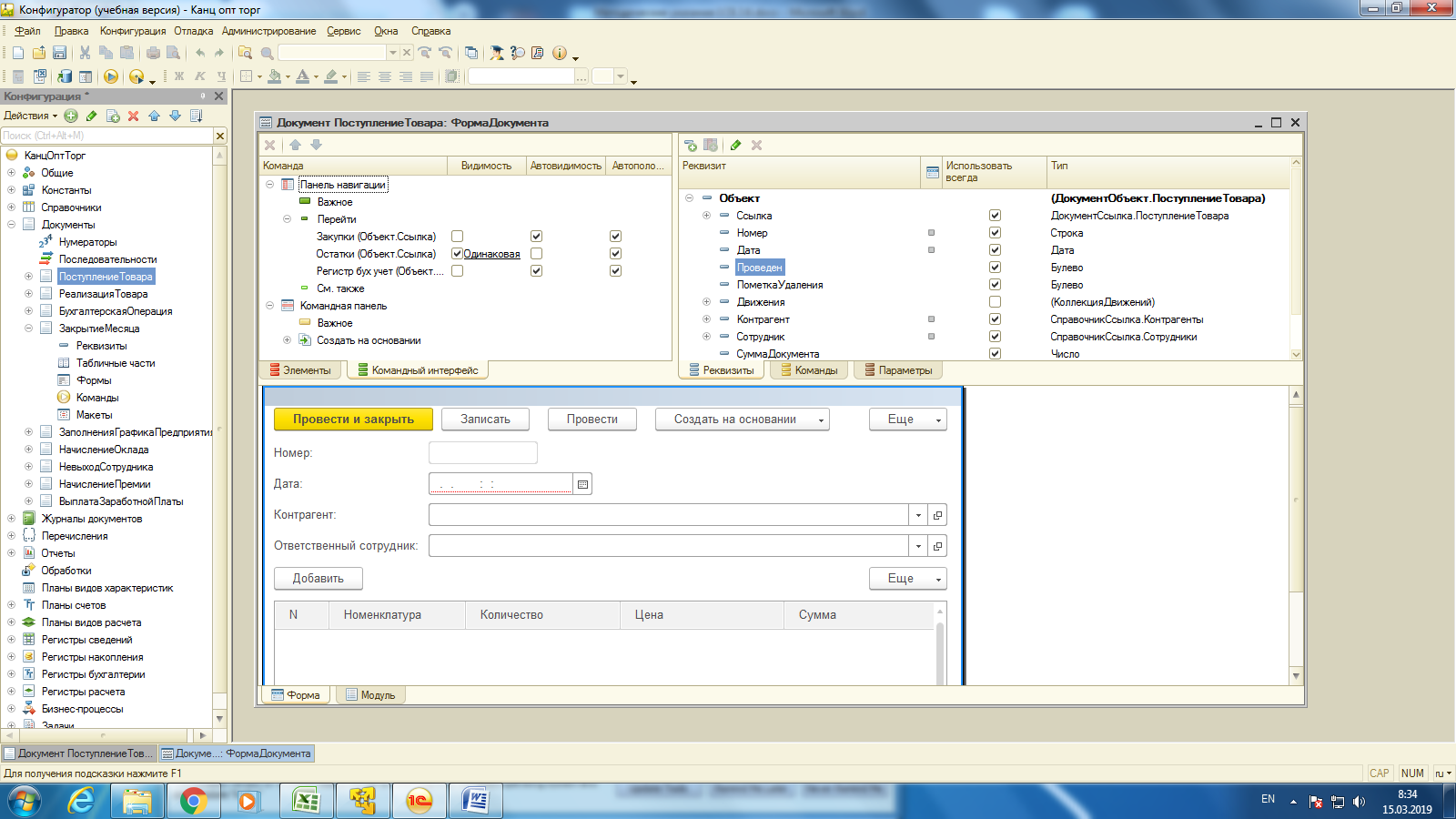


Рисунок 10 – Форма документа “РеализацияТовара”.

7.4.4 Выполним проведение существующих документов в пользовательском режиме. Для этого в пользовательском режиме выделяем существующие документы “ПоступлениеТовара” и “РеализацияТовара”, нажимаем кнопку “Еще” – “Провести” Рисунок 11.

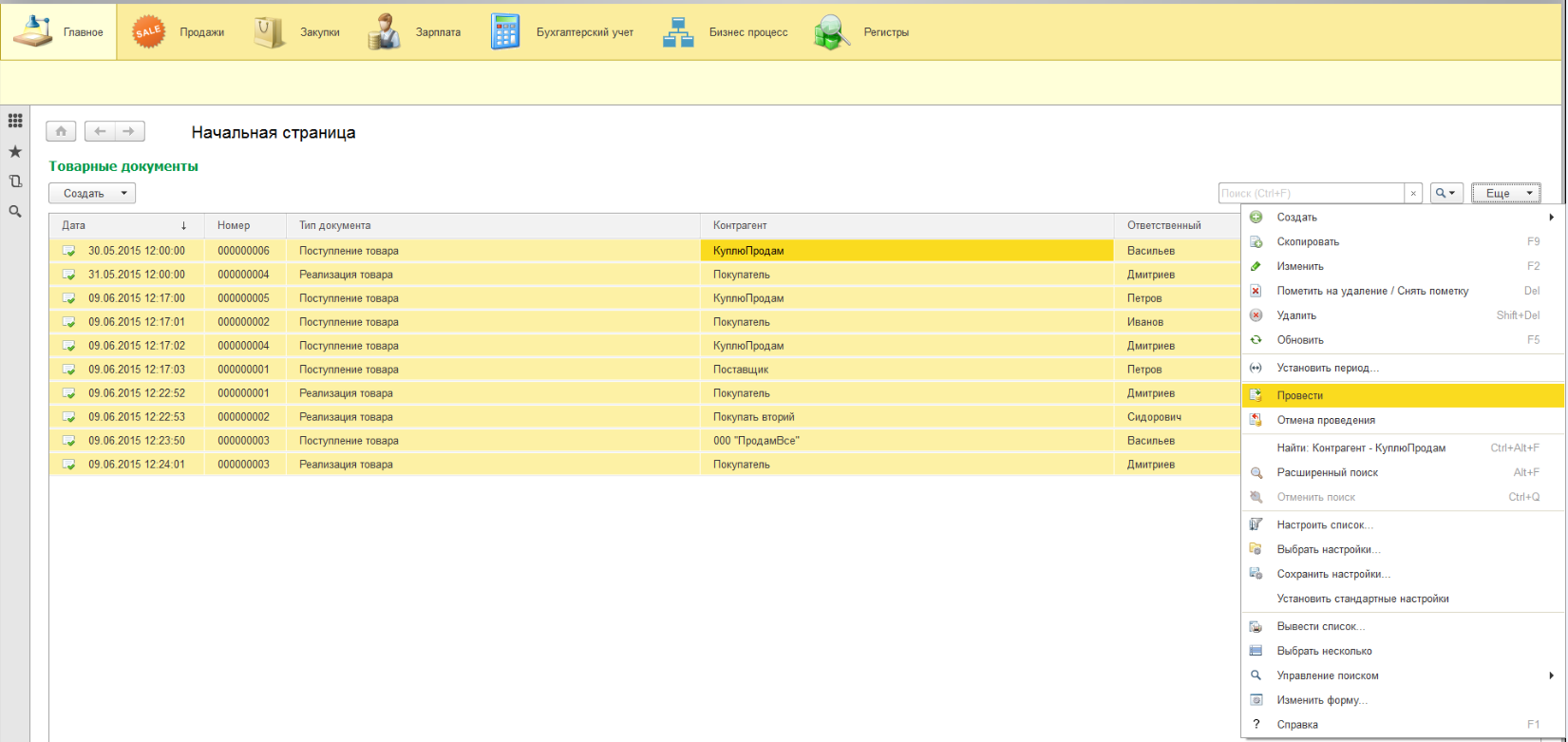


Рисунок **11** – **Выполнение проведения документов.**

7.5 Создание регистр накопления “Закупки” для учета данных о количестве и сумме закупок в разрезе номенклатуры и поставщиков, и независимый регистр “ЦеныНоменклатуры” для учета цен номенклатуры.

7.5.1 Создаем регистр накопления “Закупки” указываем вид регистра “Обороты”. Определяем структуру регистра, для этого создаем два измерения “Номенклатура” и “Контрагент” указываем тип данных “справочник номенклатура” и “справочник контрагенты” соответственно. Создаем ресурсы “Количество” и “Сумма” Рисунок 8. Отмечаем что регистратором для этого регистра выступает документ “ПоступлениеТовара”. Воспользуемся конструктором движения этого документа и создадим движение по регистру “Закупки”. Перепроводим существующий документ “ПоступлениеТовара” в пользовательском режиме. После проведения документа регистр заполнится данными Рисунок 12.

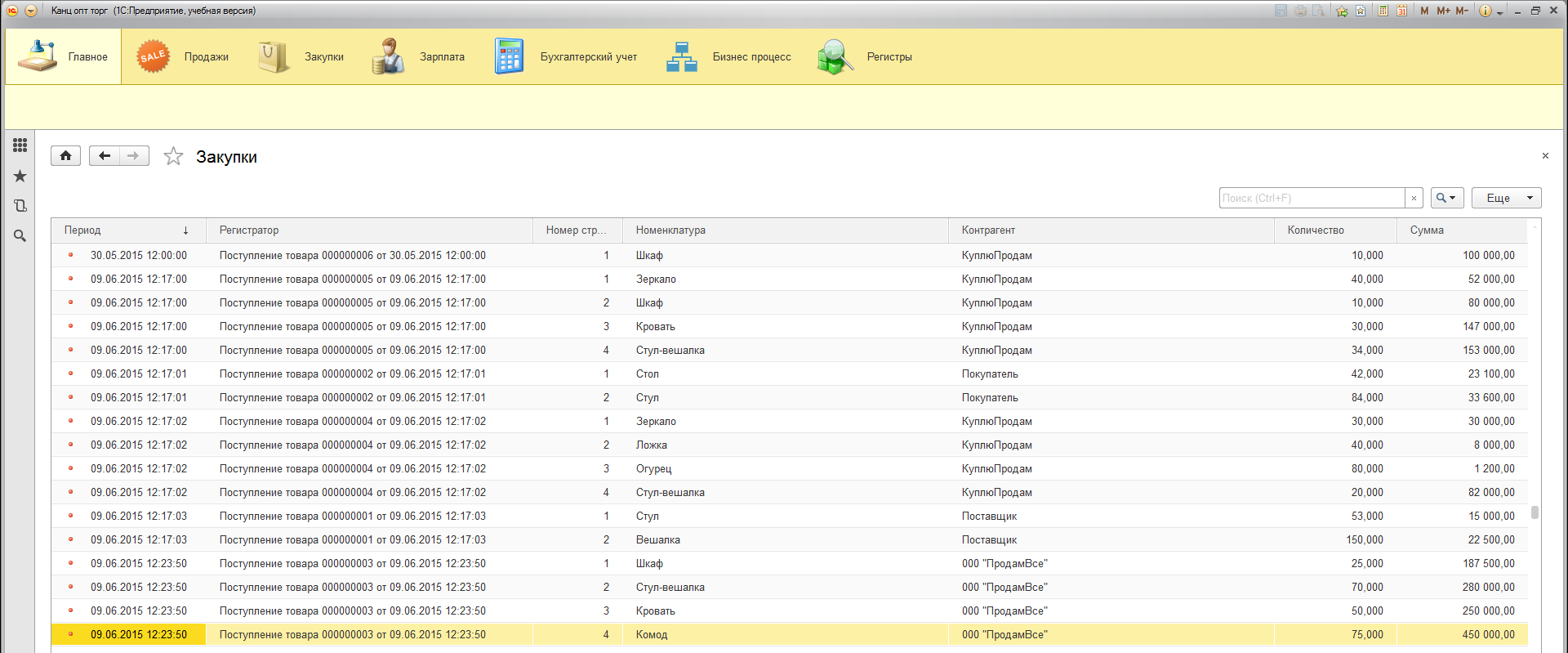


Рисунок **12** – **Форма регистра накопления “Закупки” в пользовательском режиме.**

7.5.2 Создаем регистр сведений “ЦеныНоменклатуры”. Указываем что изменятся данные регистра будут не чаще чем раз вдень, для этого в поле “Периодичность” выбираем значения “В пределах дня”. Заполняем данными регистр, создаем одно измерение “Номенклатура” указываем тип данных “справочник номенклатура” и один ресурс “Цена”. Заполним его данными Рисунок 13.

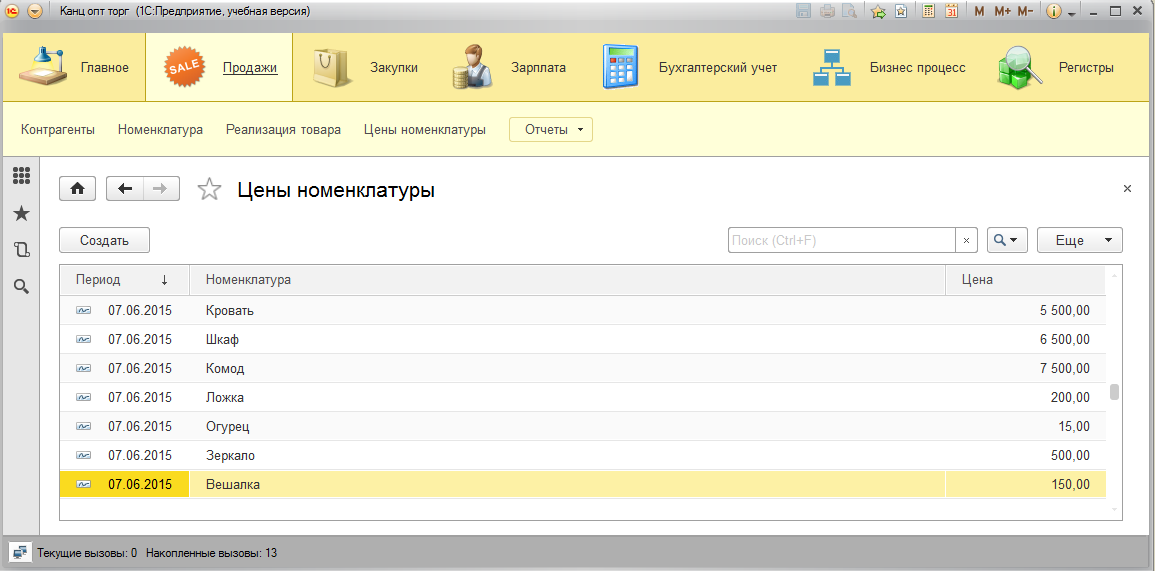


Рисунок 13 – Форма регистра сведений “ЦеныНоменклатуры” в пользовательском режиме.

7.6 Модернизации документов “ПоступлениеТовара” и “РеализацияТовара”.

7.6.1 Выполним заполнение реквизит “Сумма документа” для этого заходим в модуль объекта документа “РеализацияТовара” нажав правой кнопкой мыши по названия документа в ветви “Документы”. Создаем событие модуля объекта “ПередЗаписью” Рисунок 14.

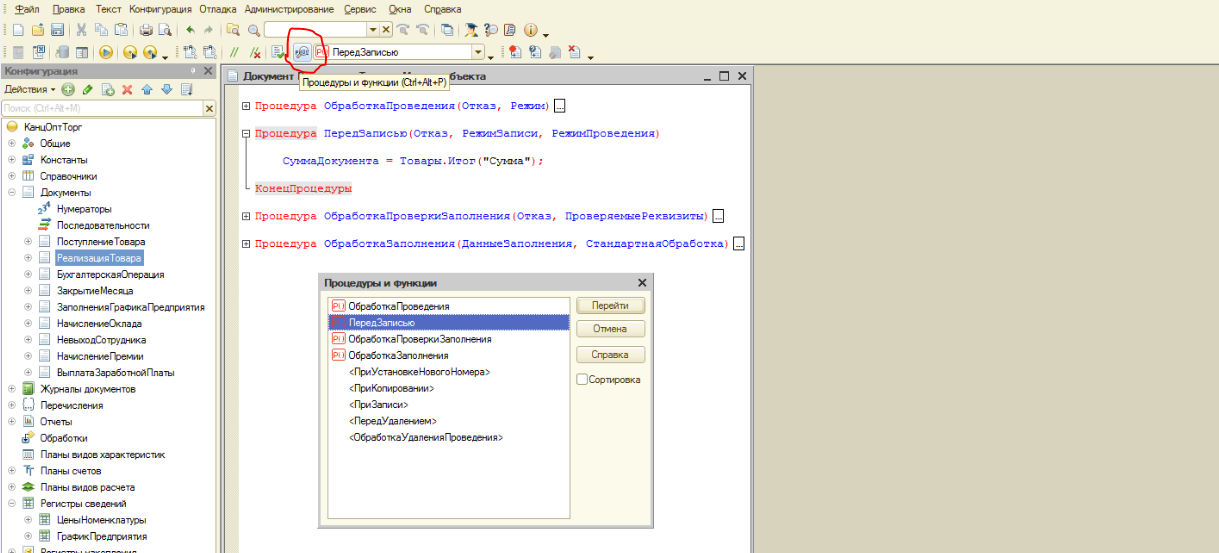


Рисунок 14 – Модуль объекта “РеализацияТовара”.

В этой процедуре необходимо обеспечить заполнение реквизита “Сумма документа”, для этого нужно вписать код процедуры для заполнения. Код процедуры выглядит следующим образом:

Процедура ПередЗаписью (Отказ, РежимЗаписи, РежимПроведения)

СуммаДокумента = Товары.Итог("Сумма");

КонецПроцедуры

Аналогично выполняем и с документом “ПоступлениеТовара”.

7.6.2 Выполнить проверку, что в документе “РеализацияТовара” выбран контрагент, у которого указан флаг “клиент”. Для этого в модуле объекта Документа “РеализацияТовара” создать Событие “ОбработкаПроверкиЗаполнения”. В этой процедуре необходимо обеспечить проверку, выбран ли флажок реквизита “Контрагент”, для этого используется условие выбора. Код процедуры выглядит следующим образом:

Процедура ОбработкаПроверкиЗаполнения (Отказ, ПроверяемыеРеквизиты)

Если НЕ Контрагент.Клиент тогда

Отказ = истина;

Сообщить ("В документе должен быть клиент");

Конецесли

КонецПроцедуры

Аналогично выполняем и с документом “ПоступлениеТовара” (Вместо клиента проверяем Поставщика).

7.6.3 В формах документов обеспечить автоматический расчет суммы в табличной части, как произведение цены и количества. В форме документа “РеализацияТовара” создаем событие “ПриИзменинии”, для этого выделяем реквизит “Количество”, щелкаем правой клавишей мыши и в открывшемся контекстном меню выбрать “События” - “ПриИзменинии”. Код процедуры выглядит следующим образом:

&НаКлиенте

Процедура ТоварыКоличествоПриИзменении(Элемент)

ТекущиеДанныеТовары = Элементы.Товары.ТекущиеДанные;

ТекущиеДанныеТовары.сумма =ТекущиеДанныеТовары.количество \* ТекущиеДанныеТовары.цена;

КонецПроцедуры

Аналогично поступаем и с реквизитом “Цена”:

&НаКлиенте

Процедура ТоварыЦенаПриИзменении(Элемент)

ТекущиеДанныеТовары = Элементы.Товары.ТекущиеДанные;

ТекущиеДанныеТовары.сумма =ТекущиеДанныеТовары.количество \* ТекущиеДанныеТовары.цена;

КонецПроцедуры

Также учесть ситуацию изменения суммы – в этом случае нужно пересчитать цену. Для этого в форме документ создаем событие “ПриИзменинии” для реквизита “Сумма”. Код процедуры выглядит следующим образом:

&НаКлиенте

Процедура ТоварыСуммаПриИзменении(Элемент)

ТекущиеДанныеТовары = элементы.Товары.ТекущиеДанные;

Если ТекущиеДанныеТовары.количество = 0 тогда

ТекущиеДанныеТовары.цена = 0;

Иначе

ТекущиеДанныеТовары.цена = ТекущиеДанныеТовары.сумма / ТекущиеДанныеТовары.количество;

Конецесли

КонецПроцедуры

7.7 Создание отчета “ЗакупкиТоваров”. Отчет должен выводить информацию о количестве и сумме закупки в разрезе номенклатуры и ответственного за закупку. Создаем отчет “ЗакупкиТоваров” и открываем схему компоновки данных. Создаем новый набор данных “Запрос” и открываем конструктор запроса. Выбираем табличную часть документа “ПоступлениеТовара”, затем выбираем реквизиты из табличной части “Номенклатура”, “Количество”, “Сумма” и “Сотрудник” Рисунок 15.

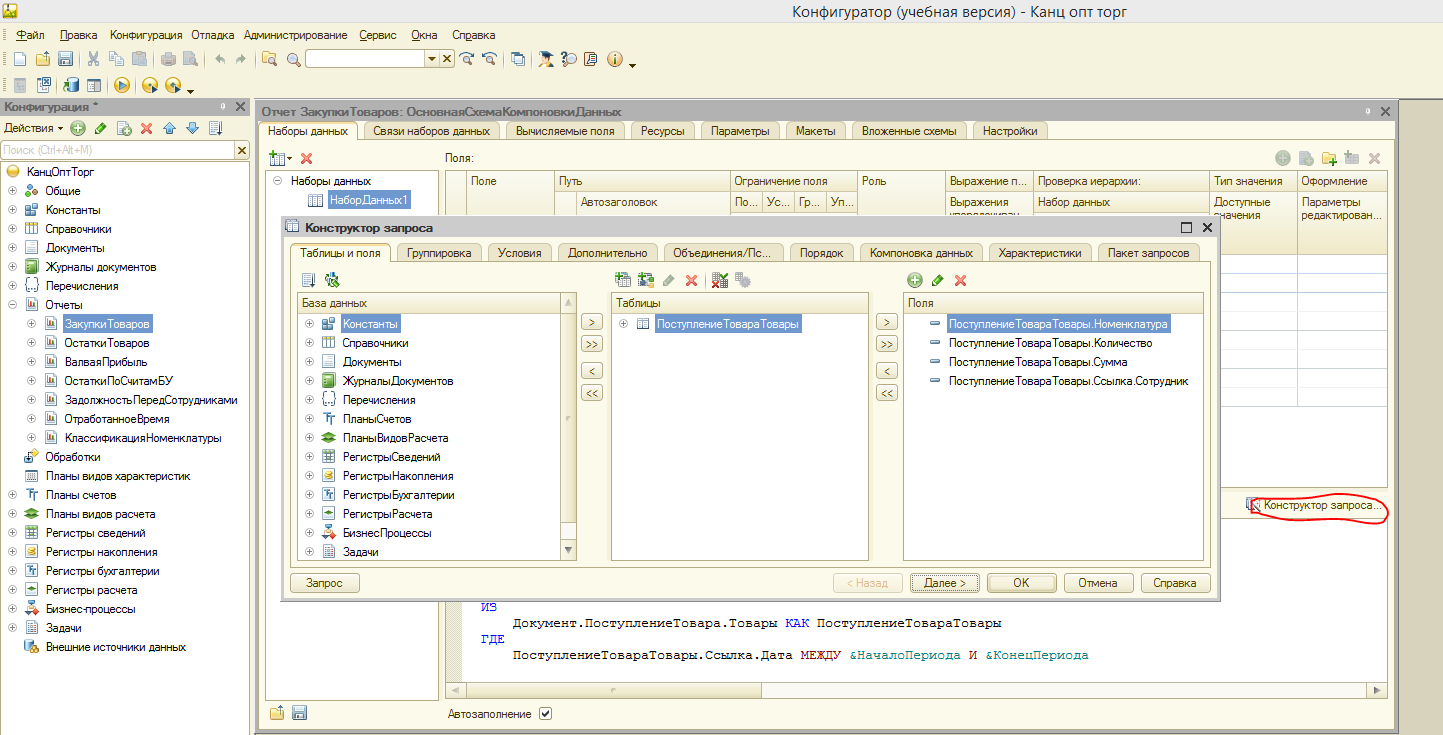


Рисунок 15 – Схема компоновки данных отчета “ЗакупкиТоваров”.

В закладке “Условия” прописываем условие, при котором пользователь может иметь возможность задавать период формирования отчета как показано на Рисунок 16, жмем “ОК”.

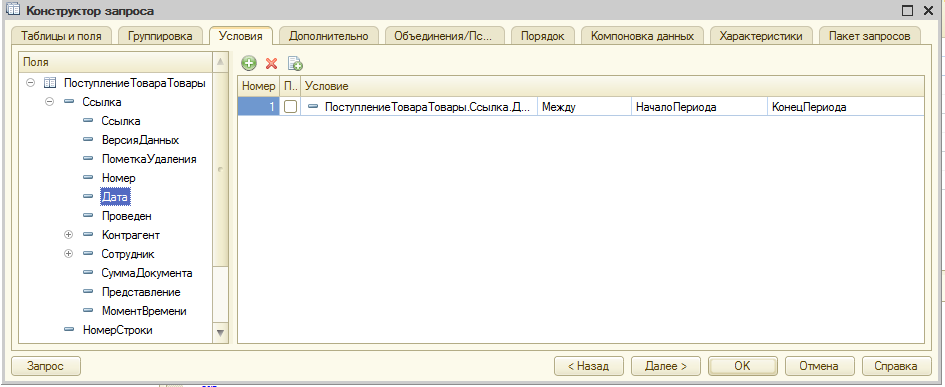


Рисунок 16 – Наложение условия на отчет “ЗакупкиТоваров”.

Определяем ресурсы отчета на закладке “Ресурсы”, жмем на кнопку с двумя стрелочками ( >> ). Заходим на закладку “Параметры”, указываем что дата выводится в формате “даты” без времени. Переходим на закладку “Настройки”, создаем с помощью конструктора новую настройку. Выбираем тип отчета “Список”, далее выводим все поля, группируем по реквизитам “Номенклатура” и “Сотрудник”. Включим параметры в пользовательские настройки, чтобы они были сразу же доступны в форме отчета Рисунок 17.

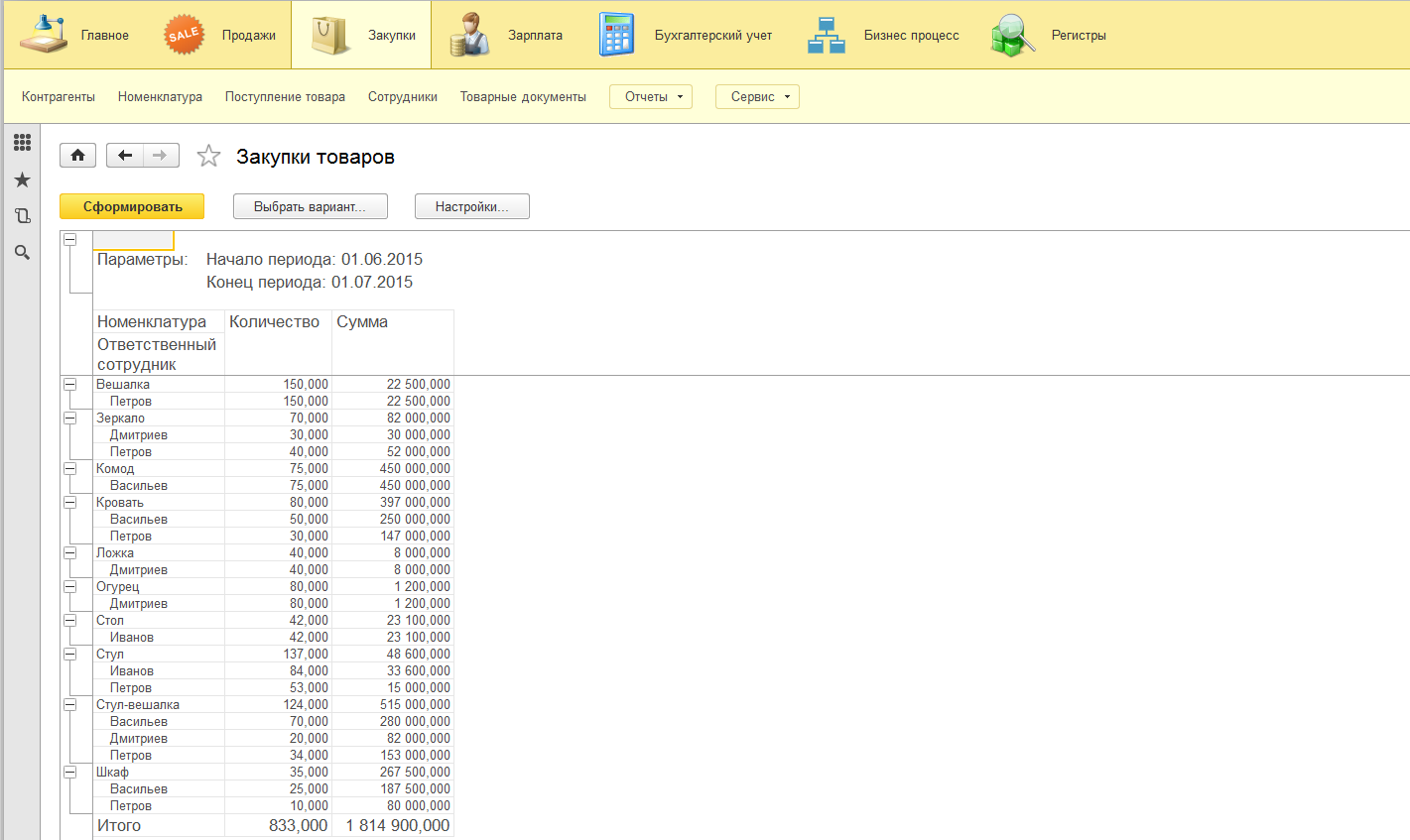
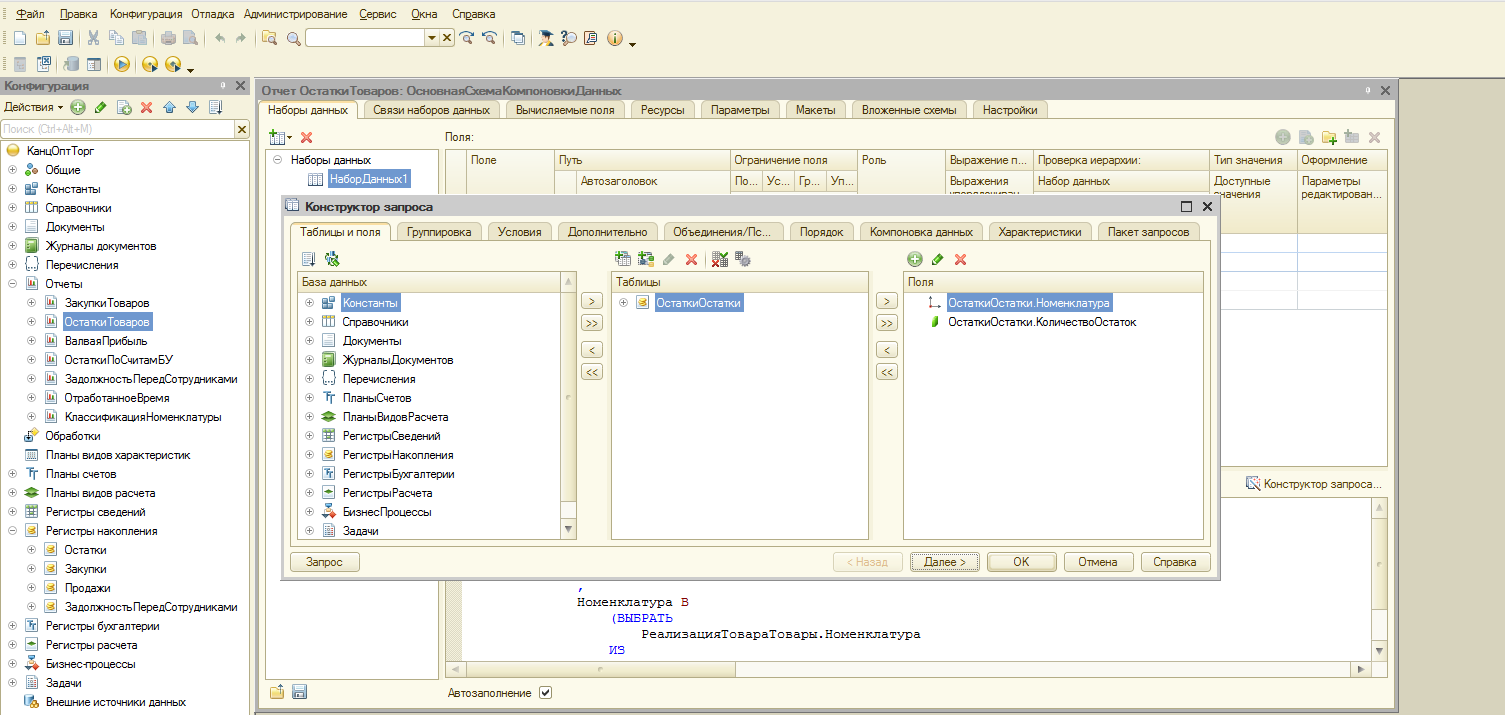


Рисунок 17 – Форма отчета “ЗакупкиТоваров” **в пользовательском режиме.**

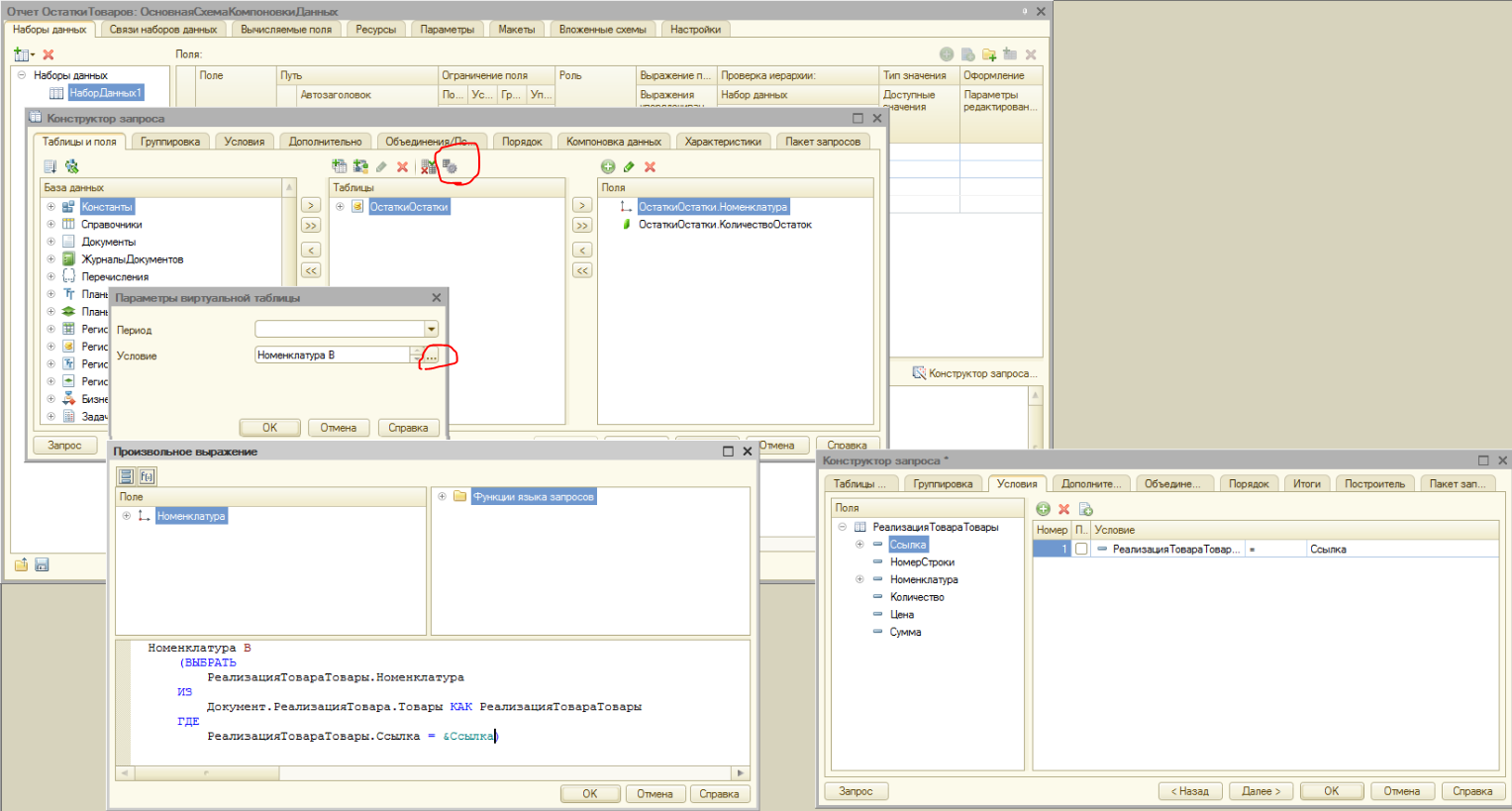
7.8 Создание отчета “Остатки товаров по данным документа”. Отчет должен выводить остатки по товарам на указанную пользователем дату. При этом в форме отчета пользователь должен указать документ “Реализация товаров”, именно по товарам из табличной части этого документа необходимо выводить данные об остатках. Создаем отчет “ОстаткиТоваров”. Создаем новый набор данных “Запрос” и открываем конструктор запроса. Выбираем виртуальную таблицу реквизита накопления “ОстаткиОстатки”, затем выбираем измерение “Номенклатура”, и регистр “Количество” Рисунок 18.

Рисунок 18 – Схема компоновки данных отчета “ОстаткиТоваров”.



Обращаемся к параметрам виртуальной таблицы, именно здесь нужно наложить условие номенклатура должна входить текущий документ. Для этого заходим в настройки виртуальной таблицы, нажимаем на кнопку троеточие в поле “Условие”, записываем условие, для этого пишем “Номенклатура В ( )” выделяем область внутри скобочек, щелкаем правой кнопкой мыши и в открывшемся контекстном меню выбрать «Конструктор запроса». Открывается новый конструктор вложенного запроса, в нем мы обращаемся к табличной части документа “Реализация товаров”, выбираем из этой табличной части “Номенклатура” и ограничиваем условием, что нас интересует не все реализации товаров, а только товары из табличной части конкретного документа, а этот документ однозначно характеризуется ссылкой. Накладываем условие на ссылку Рисунок 19.

Рисунок 19 – Схема компоновки данных отчета “ОстаткиТоваров” (Вложенный запрос).



Определяем ресурсы отчета на закладке “Ресурсы”, жмем на кнопку с двумя стрелочками ( >> ).. Заходим на закладку “Параметры”, указываем что дата выводится в формате “даты” без времени, убираем галочку с поля “Ограничение доступности” параметра “ДокументРеализация”. Переходим на закладку “Настройки”, создаем с помощью конструктора новую настройку. Выбираем тип отчета “Список”, далее выводим все поля. Включим параметры в пользовательские настройки, чтобы они были сразу же доступны в форме отчета Рисунок 20.

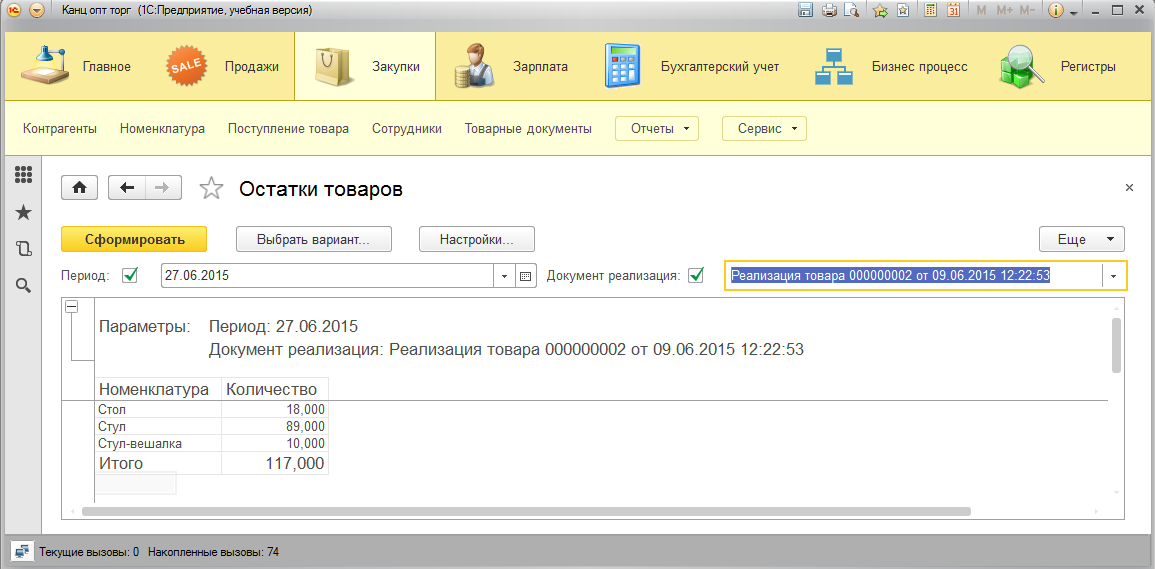


Рисунок 20 – Форма отчета “ОстаткиТоваров” **в пользовательском режиме.**

7.9 Обеспечение контроля остатков для оперативно проводимых документов “Реализация Товаров”. Также решение задачи расчета себестоимости при продаже товаров.

7.9.1 Заходим в модуль объекта документа “РеализацияТоваров” и в процедуру “ОтработкаПроведения”. В этой процедуре необходимо обеспечить контроля остатков, для этого нужно вписать код процедуры для заполнения. Код процедуры выглядит следующим образом:

Процедура ОбработкаПроведения (Отказ, Режим)

// регистр Остатки Расход

Движения.Остатки.Записывать = Истина;

Для Каждого ТекСтрокаТовары Из Товары Цикл

Движение = Движения.Остатки.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Период = Дата;

Движение.Номенклатура = ТекСтрокаТовары.Номенклатура;

Движение.Количество = ТекСтрокаТовары.Количество;

Движение.Выручка = ТекСтрокаТовары.Сумма;

КонецЦикла;

Движения.Записать();

Если Режим = РежимПроведенияДокумента.Оперативный тогда

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ОстаткиОстатки.Номенклатура,

| -ОстаткиОстатки.КоличествоОстаток КАК Количество

|ИЗ

| РегистрНакопления.Остатки.Остатки(

| ,

| Номенклатура В

| (ВЫБРАТЬ

| РеализацияТовараТовары.Номенклатура

| ИЗ

| Документ.РеализацияТовара.Товары КАК РеализацияТовараТовары

| ГДЕ

| РеализацияТовараТовары.Ссылка = &Ссылка)) КАК ОстаткиОстатки

|ГДЕ

| ОстаткиОстатки.КоличествоОстаток < 0";

Запрос.УстановитьПараметр("Ссылка", Ссылка);

РезультатЗапроса = Запрос.Выполнить();

если НЕ РезультатЗапроса.Пустой() тогда

Отказ = ИСТИНА;

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Сообщить ("В документе "+Ссылка+" образовались отрецательные остатки");

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

сообщить ("По номенклатуре "+ВыборкаДетальныеЗаписи.Номенклатура+" остаток " +ВыборкаДетальныеЗаписи.Количество);

КонецЦикла;

конецЕсли;

конецЕсли;

Если Отказ Тогда

Возврат;

КонецЕсли;

7.9.2 Решение задачи расчета себестоимости при продаже товаров. Код добавляется в процедуру обработки проведения документа “РеализацияТоваров” и выглядит следующим образом:

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ОстаткиОстатки.Номенклатура,

| ОстаткиОстатки.КоличествоОстаток КАК Количество,

| ОстаткиОстатки.СуммаСебестоимостиОстаток КАК Сумма

|ИЗ

| РегистрНакопления.Остатки.Остатки(

| &МоментВремени,

| Номенклатура В

| (ВЫБРАТЬ

| РеализацияТовараТовары.Номенклатура

| ИЗ

| Документ.РеализацияТовара.Товары КАК РеализацияТовараТовары

| ГДЕ

| РеализацияТовараТовары.Ссылка = &Ссылка)) КАК ОстаткиОстатки";

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Запрос.УстановитьПараметр("Ссылка", Ссылка);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Движения.Остатки.Записывать = Истина;

Движения.Продажи.Записывать = Истина;

СписаннаяСебестоимость = 0;

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

Для каждого Движение Из Движения.Остатки Цикл

Если Движение.Номенклатура = ВыборкаДетальныеЗаписи.Номенклатура Тогда

Движение.СуммаСебестоимости = Движение.Количество \* ВыборкаДетальныеЗаписи.Сумма/ВыборкаДетальныеЗаписи.Количество;

Движение.СебестоимостьЕдиницы = ВыборкаДетальныеЗаписи.Сумма/ВыборкаДетальныеЗаписи.Количество;

ДвижениеПродажи = Движения.Продажи.Добавить();

ДвижениеПродажи.Период = Дата;

ДвижениеПродажи.Номенклатура = ВыборкаДетальныеЗаписи.Номенклатура;

ДвижениеПродажи.Контрагент = Контрагент;

ДвижениеПродажи.Сумма =Движение.Выручка;

ДвижениеПродажи.Себестоимость = Движение.СуммаСебестоимости;

ДвижениеПродажи.Количество = Движение.Количество;

СписаннаяСебестоимость = СписаннаяСебестоимость + Движение.СуммаСебестоимости;

КонецЕсли;

КонецЦикла;

КонецЦикла;

КонецПроцедуры

7.10 Обеспечение учета суммы выручки, количества проданного товара и списанной себестоимости в оборотном регистре накопления “Продажи”.

Создание отчета «Валовая прибыль». В нем представлена информация в виде таблицы. В строках – клиенты, в колонках – номенклатура, на пересечении – сумма валовой прибыли.

Также в этом отчете нужно построить гистограмму, где в качестве точек выбрана номенклатура, а в качестве серий – контрагенты.

7.10.1 Создаем регистр накопления называем его “Продажи”, вид регистра “Обороты”. Определяем структуру регистра, для этого в закладке “Данные”, создаем два измерения “Номенклатура”, тип данных “справочник номенклатура” и “Контрагент”, тип данных “справочник контрагент” и ресурсы “Количество”, “Сумма” и “Себестоимость”. Отмечаем что регистратором для этого регистра выступает документ “РеализацияТовара”. Заходим в модуль объекта документа “РеализацияТоваров” и в процедуру “ОтработкаПроведения”. В этой процедуре необходимо обеспечить движение регистра накопления, для этого нужно вписать код процедуры для заполнения (см. код написанный в пункте 3.9.2).

7.10.2 Создаем отчет «ВаловаяПрибыль». Создаем новый набор данных “Запрос” и открываем конструктор запроса. Выбираем виртуальную таблицу реквизита накопления “ПродажиОбороты”, затем выбираем измерения “Номенклатура” и “Контрагент”, и ресурсы “Сумма” и “Себестоимость”, непосредственно в запросе создаем вычисляемое поле, для этого двойным кликом мыши нажимаем на регистр “Сумма” откроется конструктор произвольных выражений. В нем создать операцию вычитания суммы оборота на эго себестоимость в итоге получается валовая прибыль, далее удаляем поле “Себестоимость” Рисунок 21.

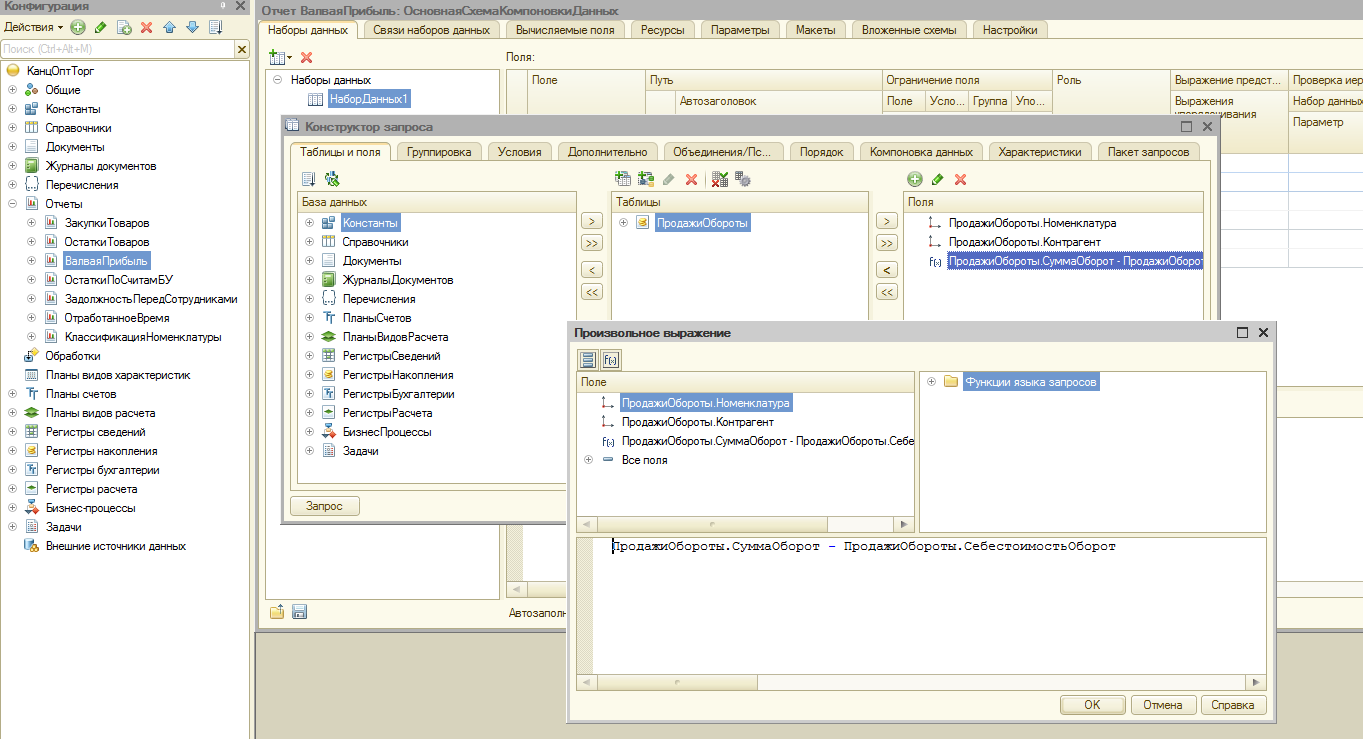


Рисунок 21 – Схема компоновки данных отчета “ВаловаяПрибыль”.

Определяем ресурсы отчета. Заходим на закладку “Параметры”, указываем что дата выводится в формате “даты” без времени. Переходим на закладку “Настройки”, создаем с помощью конструктора новую настройку. Выбираем тип отчета “Таблица”, далее выводим все поля, определяем поля по которым будет проводится группировка строк и колонок отчета это будет “Контрагент” и “Номенклатура” соответственно. Также создает еще одну настройку типа “Диаграмма”, группировка точек которой будет проводится относительно “Контрагента”, для серий указываем “Номенклатура” Включим параметры в пользовательские настройки, чтобы они были сразу же доступны в форме отчета Рисунок 22.

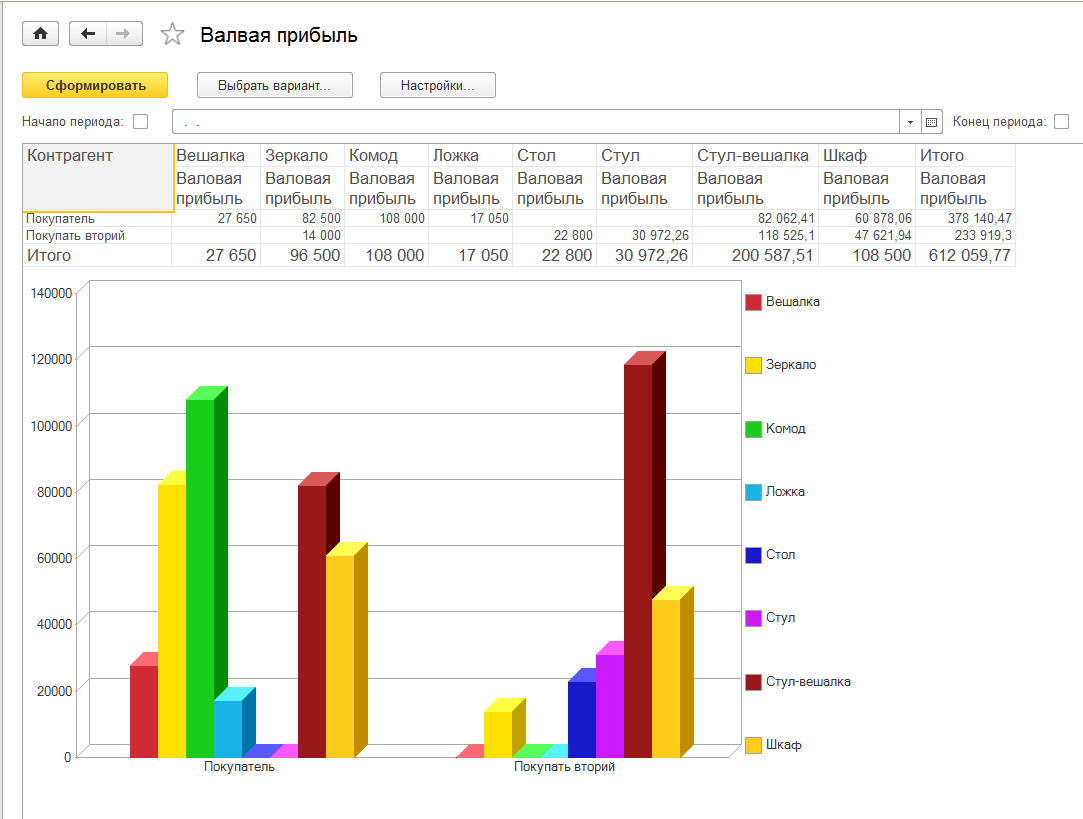


Рисунок 22 – Форма отчета “ВаловаяПрибыль” **в пользовательском режиме.**

7.11 Создание плана счетов “Бух. учет”, определение в нем счетов. Создание регистр бухгалтерии для отражения проводок по бухгалтерскому учету.

7.11.1 Создаем плана счетов “БухУчет”, затем на закладке “Прочее” нажимаем кнопку “Предопределенные”. Создаем счета Рисунок 23:

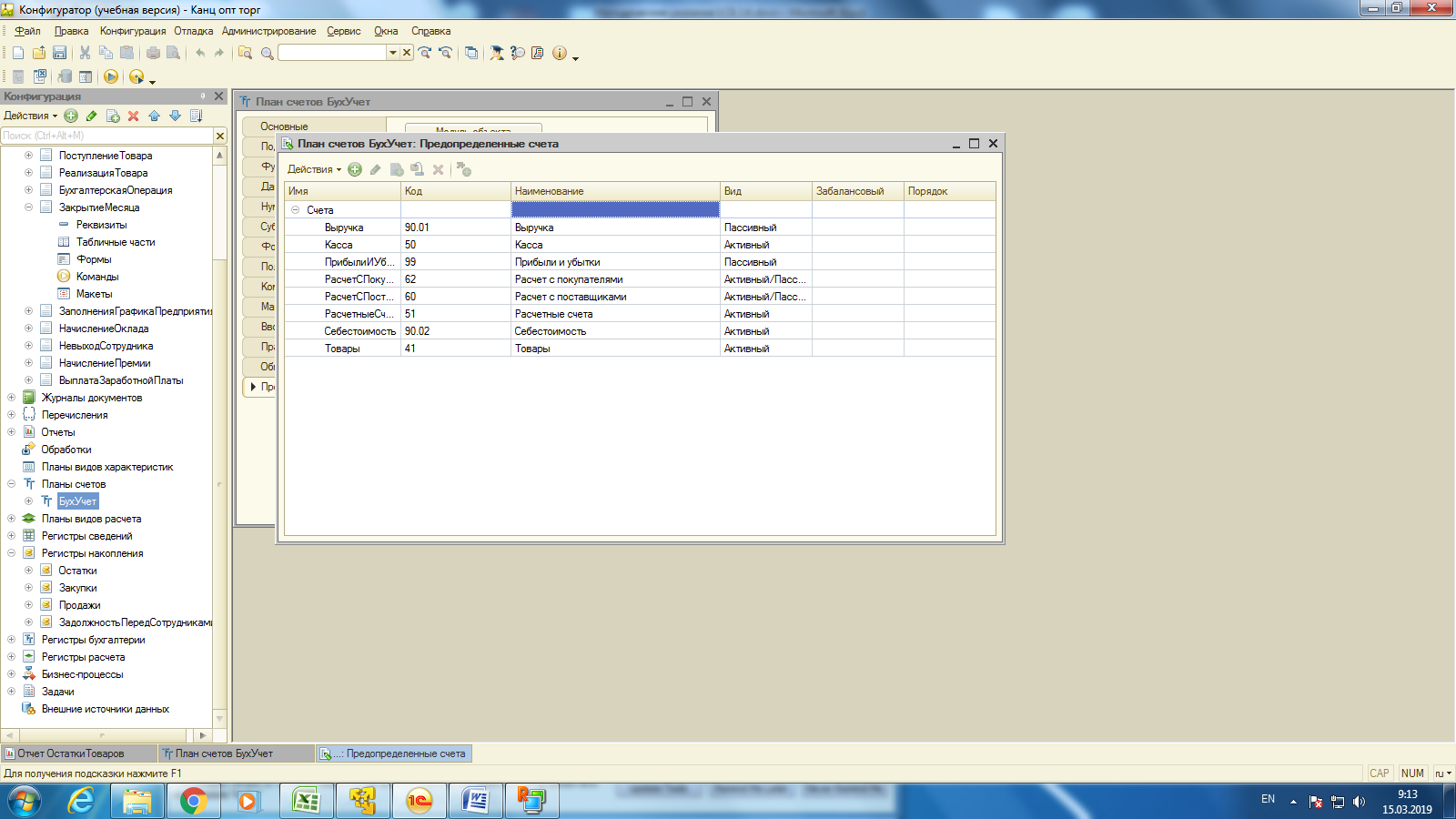


Рисунок 23 – Окно редактирования плата счетов “БухУчет”определение в нем счетов.

7.11.2 Создаем регистр бухгалтерии “РегистрБухУчет”, связываем с планом счетов путем добавления в регистр в поле “План счетов” план счетов “БухУчет” и указываем флажок “Корреспондент”. Создаем ресурс “Сумма”. Отмечаем что регистраторами для этого регистра выступают документы “ПоступлениеТовара” и “РеализацияТоваров”.

7.12 Обеспечение проведение документов “Поступление товаров” и “Реализация товаров” по бухгалтерскому учету. Для этого заходим в модуль объекта документа “РеализацияТовара” в “ОбработкаПроведения”. В этой процедуре в самом конце необходимо обеспечить движение регистра бухгалтерии “РегистрБухУчет”. Код процедуры выглядит следующим образом:

Вписываем код и в модуль объекта документа “Поступление товаров”

// регистр БухУчет

Движения.РегистрБухУчет.Записывать = Истина;

Движение = Движения.РегистрБухУчет.Добавить();

Движение.СчетДт = ПланыСчетов.БухУчет.Товары;

Движение.СчетКт = ПланыСчетов.БухУчет.РасчетСПоставщиками;

Движение.Период = Дата;

Движение.Сумма = СуммаДокумента;

КонецПроцедуры

Аналогично вписываем код и в модуль “ОбработкаПроведения” объекта документа “Реализация товаров”

Движения.РегистрБухУчет.Записывать = Истина;

Проводка = Движения.РегистрБухУчет.Добавить();

Проводка.Период = дата;

Проводка.СчетДт = ПланыСчетов.БухУчет.Себестоимость;

Проводка.СчетКт = ПланыСчетов.БухУчет.Товары;

Проводка.Сумма = СписаннаяСебестоимость;

Проводка = Движения.РегистрБухУчет.Добавить();

Проводка.Период = дата;

Проводка.СчетДт = ПланыСчетов.БухУчет.РасчетСПокупателями;

Проводка.СчетКт = ПланыСчетов.БухУчет.Выручка;

Проводка.Сумма = СуммаДокумента;

КонецПроцедуры

7.13 Создание отчета “Остатки по счетам БУ”. Отчет должен выводить остаток по выбранному счету на указанную пользователем дату. Если счет не задан – должны получаться остатки по всем счетам. Выполнение закрытия месяца.

7.13.1 Создаем отчет “ОстаткиПоСчетамБУ”. Создаем новый набор данных “Запрос” и открываем конструктор запроса. Выбираем виртуальную таблицу реквизита бухгалтерии “РегистрБухУчетОстатки”, затем выбираем измерение “Счет”, и ресурсы “СуммаОстатокДт” и “СуммаОстатокКт”. Определяем ресурсы отчета. Заходим на закладку “Параметры”, указываем что дата выводится в формате “даты” без времени. Переходим на закладку “Настройки”, создаем с помощью конструктора новую настройку. Выбираем тип отчета “Список”, далее выводим все поля. Включим параметры в пользовательские настройки, чтобы они были сразу же доступны в форме отчета, затем на закладке “Отбор” в настройке вариантов, указываем здесь “Счет” по умолчанию отбор снимаем Рисунок 24.

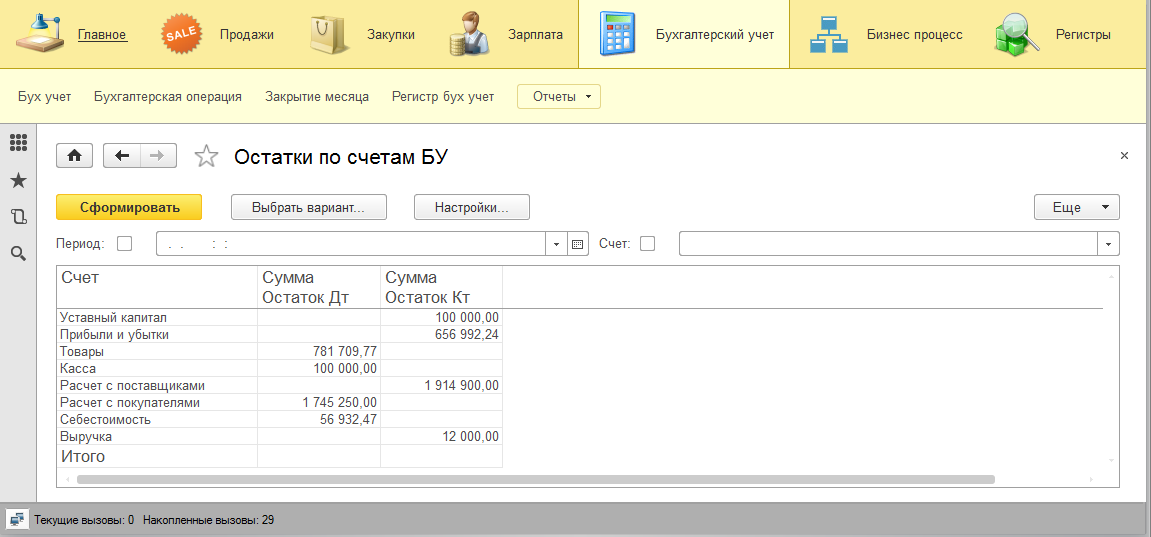


Рисунок 24 – Форма отчета “ОстаткиПоСчетамБУ” **в пользовательском режиме.**

7.13.2 Выполним закрытие месяца. Необходимо сформировать проводки вида:

Дт Выручка Кт Прибыль на сумму остатка выручки Дт Прибыль Кт Себестоимость на сумму остатка себестоимости

Создаем новый документ “ЗакрытиеМесяца”, у этого документа не будет никаких реквизитов. Обеспечение проведение “ЗакрытиеМесяца”, по регистру бухгалтерии “РегистрБухУчет”. Для этого заходим в модуль объекта Документа “ЗакрытиеМесяца” в “ОбработкаПроведения”. В этой процедуре необходимо обеспечить движение регистра бухгалтерии “РегистрБухУчет”.

Код процедуры выглядит следующим образом:

Процедура ОбработкаПроведения (Отказ, РежимПроведения)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| РегистрБухУчетОстатки.Счет,

| РегистрБухУчетОстатки.СуммаОстатокДт,

| РегистрБухУчетОстатки.СуммаОстатокКт

|ИЗ

| РегистрБухгалтерии.РегистрБухУчет.Остатки(&МоментВремени, Счет В

|(&СчетВыручка, &СчетСебестоимость), ) КАК РегистрБухУчетОстатки";

Запрос.УстановитьПараметр("МоментВремени", КонецМесяца(Дата)+1);

Запрос.УстановитьПараметр("СчетВыручка", ПланыСчетов.БухУчет.Выручка);

Запрос.УстановитьПараметр("СчетСебестоимость", ПланыСчетов.БухУчет.Себестоимость);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

СуммаВыручки = 0;

СуммаСебестоимости = 0;

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

Если ВыборкаДетальныеЗаписи.Счет = ПланыСчетов.БухУчет.Выручка Тогда

СуммаВыручки = ВыборкаДетальныеЗаписи.СуммаОстатокКт;

ИначеЕсли ВыборкаДетальныеЗаписи.Счет = ПланыСчетов.БухУчет.Себестоимость Тогда

СуммаСебестоимости = ВыборкаДетальныеЗаписи.СуммаОстатокДт;

КонецЕсли;

КонецЦикла;

Движения.РегистрБухУчет.Записывать = Истина;

Проводка = Движения.РегистрБухУчет.Добавить();

Проводка.Период = Дата;

Проводка.СчетДт = ПланыСчетов.БухУчет.Выручка;

Проводка.СчетКт = ПланыСчетов.БухУчет.ПрибылиИУбытки;

Проводка.Сумма = СуммаВыручки;

Проводка = Движения.РегистрБухУчет.Добавить();

Проводка.Период = Дата;

Проводка.СчетДт = ПланыСчетов.БухУчет.ПрибылиИУбытки;

Проводка.СчетКт = ПланыСчетов.БухУчет.Себестоимость;

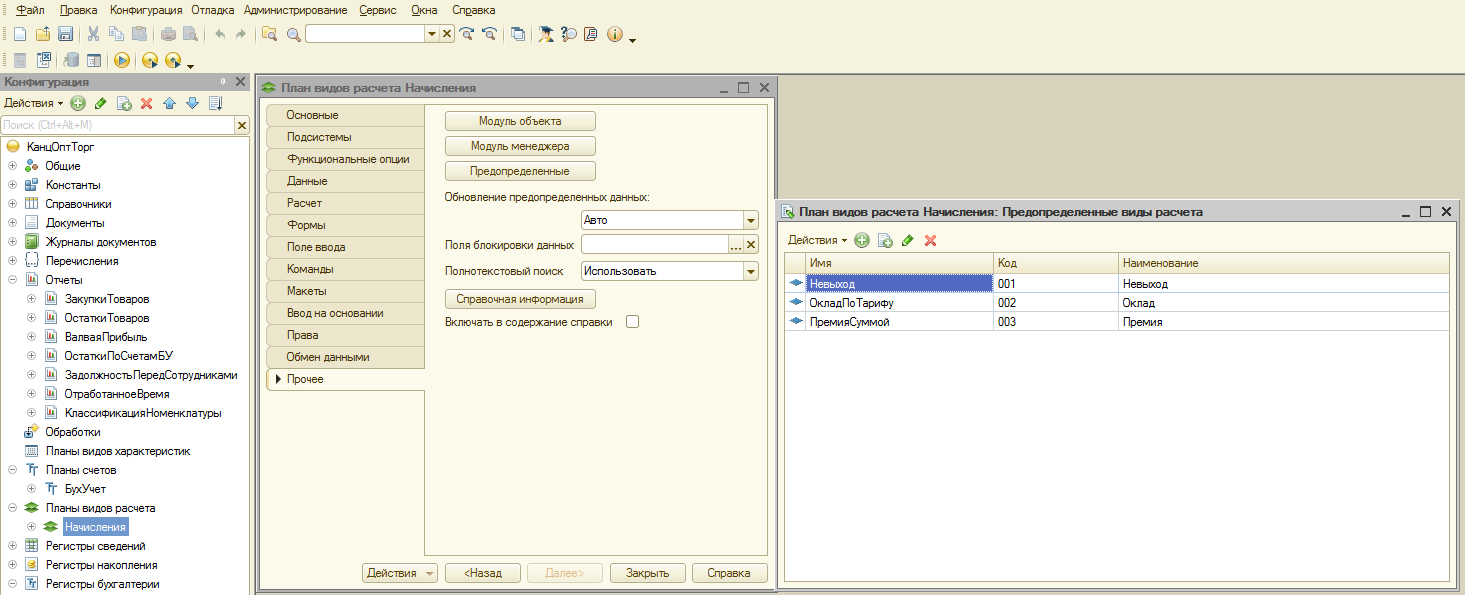
Проводка.Сумма = СуммаСебестоимости;

КонецПроцедуры

7.14 Создание плана видов расчета “Начисления”, в нем будут содержаться виды расчетов, имеющие протяженность во времени. Создание регистра сведений, описывающий график работы предприятия. Создание документа “Начисление оклада” и “Невыход сотрудника”, определение структуры этих документов. Создание регистра расчета “Начисления”. Он будет хранить данные о начисленной заработной плате в разрезе сотрудников.

7.14.1 Создаем плана расчта “Начисления”, затем на закладке “Расчет” ставим галочку напротив поля “Использует период действия” т. е. виды расчета этого плата могут иметь протяженность во времени. На закладке “Прочее” нажимаем кнопку “Предопределенные”. Создаем виды расчета Рисунок 25:

Рисунок 25 – Окно редактирования плата счетов “Начисления” определение в нем счетов.



7.14.2 Создаем регистр сведений “ГрафикПредприятия”. Режим записи – “Подчинение регистру”. Определяем структуру регистра, для этого создаем одно измерение “Дата” и один ресурс “РабочийДень”. Отмечаем что регистратором для этого регистра выступает документ “ЗаполненияГрафикаПредприятия”, который еще нужно создать. Создаем документ “ЗаполненияГрафикаПредприятия”, у этого документа нет реквизитов, в модуле объектов этого документа процедуре “ОбработкаПроведения” необходимо обеспечить заполнение регистра сведений “ГрафикПредприятия” данными о рабочих днях. Код процедуры выглядит следующим образом:

Процедура ОбработкаПроведения (Отказ, РежимПроведения)

Движения.ГрафикПредприятия.Записывать = Истина;

ТекущаяДата = НачалоГода(Дата);

КонецГода = КонецГода(Дата);

Пока ТекущаяДата<= КонецГода Цикл

Движение = Движения.ГрафикПредприятия.Добавить();

Движение.Дата = ТекущаяДата

Если ДеньНедели(ТекущаяДата)<6 тогда

Движение.РабочийДень = 1;

Иначе

Движение.РабочийДень = 0;

КонецЕсли;

ТекущаяДата = ТекущаяДата + 24\*3600;

КонецЦикла;

КонецПроцедуры

После проведения документа регистр заполнится данными.

7.14.3 Создадим регистр расчета “Начисления”. Связываем этот регистр с планом видов расчета “Начисления”, ставим флажок возле поля “Период действия”. В поле “График” указываем тот регистр где хранится график предприятия, отмечаем в “Значение графика” ресурс “РабочийДень” и в поле “Дата графика” значение “Дата”. Создаем измерение “Сотрудник” тип “ссылка на справочник сотрудники”, ресурс “Результат” и реквизит “РасчетныеДанные”.

Создаем документы “Начисление оклада” и “Невыход сотрудника”. Реквизиты документов “Сотрудник” тип “ссылка на справочник сотрудники” и для документа “Невыход сотрудника” реквизиты “ДатаНачала” и “ДатаОкончания” и для документа “Начисление оклада” еще один реквизит “ДневнаяЗарплата”. Впишем в оба документа в модуль объекта следующий код:

В документ “Невыход сотрудника”.

Процедура ОбработкаПроведения (Отказ, Режим)

// регистр ЗаработнаПлата

Движения.ЗаработнаПлата.Записывать = Истина;

Движение = Движения.ЗаработнаПлата.Добавить();

Движение.Сторно = Ложь;

Движение.ВидРасчета = ПланыВидовРасчета.Начисления.Невыход;

Движение.ПериодДействияНачало = ДатаНачала;

Движение.ПериодДействияКонец = КонецДня(ДатаОкончания);

Движение.ПериодРегистрации = Дата;

Движение.Сотрудник = Сотрудник;

КонецПроцедуры

В документ “Начисление оклада”, необходимо обеспечить расчет оклада сотрудника по дневном тарифу.

Процедура ОбработкаПроведения (Отказ, Режим)

// регистр ЗаработнаПлата

Движения.ЗаработнаПлата.Записывать = Истина;

Движение = Движения.ЗаработнаПлата.Добавить();

Движение.Сторно = Ложь;

Движение.ВидРасчета = ПланыВидовРасчета.Начисления.ОкладПоТарифу;

Движение.ПериодДействияНачало = НачалоМесяца(Дата);

Движение.ПериодДействияКонец = КонецМесяца(Дата);

Движение.ПериодРегистрации = Дата;

Движение.Сотрудник = Сотрудник;

Движение.РасчетныеДанные = ДневнаяЗарплата;

Движения.Записать();

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

|ЗаработнаПлатаДанныеГрафика.РасчетныеДанные,

|ЗаработнаПлатаДанныеГрафика.РабочийДеньФактическийПериодДействия КАК Факт

| ИЗ

| РегистрРасчета.ЗаработнаПлата.ДанныеГрафика(Регистратор = &Ссылка) КАК |ЗаработнаПлатаДанныеГрафика";

Запрос.УстановитьПараметр("Ссылка", Ссылка);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

Результат = ВыборкаДетальныеЗаписи.РасчетныеДанные \* ВыборкаДетальныеЗаписи.Факт;

Движение.Результат = Результат;

КонецЕсли;

Движения.ЗаработнаПлата.Записать();

КонецПроцедуры

7.15 Обеспечение начисление премий сотрудникам, указанным в табличной части, вновь созданного документа. Создаем новый документ “НачислениеПремии”, создаем табличную часть документа с реквизитами “Сотрудник” тип “ссылка на справочник сотрудники” и “СуммаПремии”. В модуле объекта документа в “ОбработкаПроведения”, необходимо обеспечить движение регистра расчетов “ЗаработнаПлата”. Код процедуры выглядит следующим образом:

Процедура ОбработкаПроведения (Отказ, Режим)

Движения.ЗаработнаПлата.Записывать = Истина;

Для Каждого ТекСтрокаСотрудники Из Сотрудники Цикл

// регистр ЗаработнаПлата

Движение = Движения.ЗаработнаПлата.Добавить();

Движение.Сторно = Ложь;

Движение.ВидРасчета = ПланыВидовРасчета.Начисления.ПремияСуммой;

Движение.ПериодДействияНачало = НачалоМесяца(Дата);

Движение.ПериодДействияКонец = КонецМесяца(Дата);

Движение.ПериодРегистрации = Дата;

Движение.Сотрудник = ТекСтрокаСотрудники.Сотрудник;

Движение.Результат = ТекСтрокаСотрудники.СуммаПремии;

КонецЦикла;

КонецПроцедуры

7.16 Отражение процессов выплаты заработной платы. Для этого потребуется создание регистра накопления и документа “ВыплатаЗаработнойПлаты”. Создание отчета “Задолженность по зарплате”, показывающий на определенную дату, долги перед сотрудниками. Создание отчета с использованием диаграммы Ганта.

7.16.1 Создаем регистр накопления “ЗадолжностьПередСотрудниками”, вид регистра “Остатки”. Одно измерение “Сотрудник” тип “ссылка на справочник сотрудники” и ресурс “Долг”. Отмечаем что регистраторами для этого регистра выступают документы “НачислениеОклада”, “ВыплатаЗаработнойПлаты” и “НачислениеПремии”. В модуле объекта документов в “ОбработкаПроведения”, необходимо обеспечить движение регистра расчетов “ЗаработнаПлата”. Код процедуры выглядит следующим образом:

В документ “НачислениеОклада”.

Движения.ЗадолжностьПередСотрудниками.Записывать = Истина;

Движение = Движения.ЗадолжностьПередСотрудниками.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

Движение.Период = Дата;

Движение.Сотрудник = Сотрудник;

Движение.Долг = Результат;

КонецПроцедуры

В документ “НачислениеПремии”.

// регистр ЗадолжностьПередСотрудниками Приход

Движения.ЗадолжностьПередСотрудниками.Записывать = Истина;

Для Каждого ТекСтрокаСотрудники Из Сотрудники Цикл

Движение = Движения.ЗадолжностьПередСотрудниками.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

Движение.Период = Дата;

Движение.Сотрудник = ТекСтрокаСотрудники.Сотрудник;

Движение.Долг = ТекСтрокаСотрудники.СуммаПремии;

В документ “ВыплатаЗаработнойПлаты”.

Процедура ОбработкаПроведения (Отказ, Режим)

// регистр ЗадолжностьПередСотрудниками Расход

Движения.ЗадолжностьПередСотрудниками.Записывать = Истина;

Для Каждого ТекСтрокаСотрудники Из Сотрудники Цикл

Движение = Движения.ЗадолжностьПередСотрудниками.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Период = Дата;

Движение.Сотрудник = ТекСтрокаСотрудники.Сотрудник;

Движение.Долг = ТекСтрокаСотрудники.Сумма;

КонецЦикла;

КонецПроцедуры

7.16.2 Создаем отчет “ЗадолжностьПередСотрудниками” Создаем новый набор данных “Запрос” и открываем конструктор запроса. Выбираем виртуальную таблицу реквизита накопления “ЗадолжностьПередСотрудникамиОстатки”, затем выбираем измерение “Сотрудник”, и ресурс “Долг”. Определяем ресурсы отчета. Заходим на закладку “Параметры”, указываем что дата выводится в формате “даты” без времени. Переходим на закладку “Настройки”, создаем с помощью конструктора новую настройку. Выбираем тип отчета “Список”, далее выводим все поля. Включим параметры в пользовательские настройки, чтобы они были сразу же доступны в форме отчета Рисунок 26.

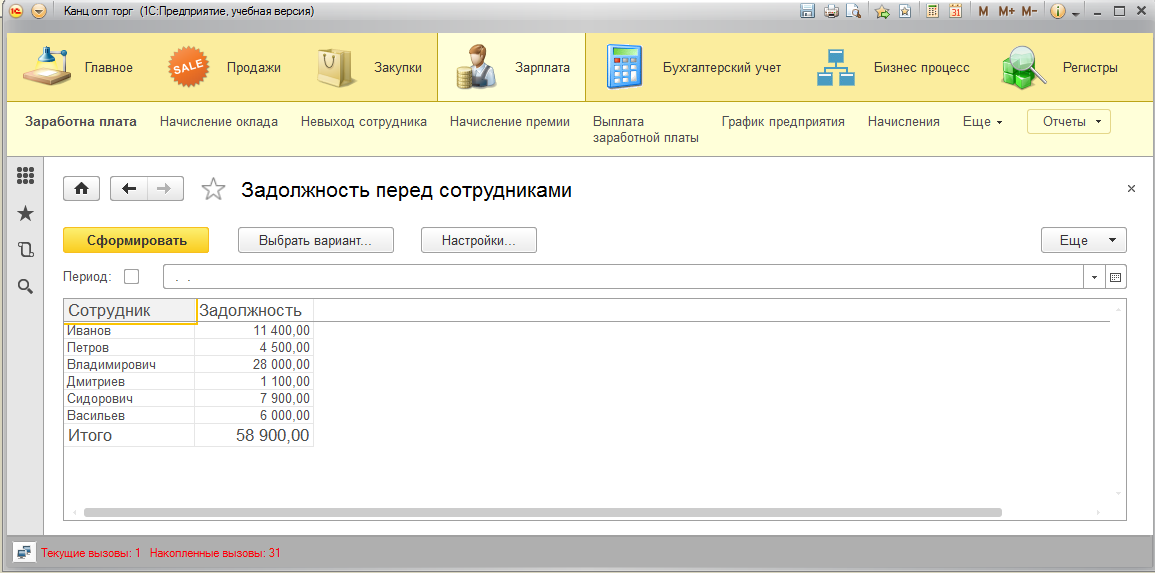


Рисунок 26 – Форма отчета “ЗадолжностьПередСотрудниками” **в пользовательском режиме.**

7.16.4 Создаем отчет “ОтработанноеВремя” с использованием диаграммы Ганта. Создаем форму отчета по умолчанию, создаем на форме реквизит “ДиаграммаГанта” с типом “ДиаграммаГанта” Рисунок 27. Перетаскиваем реквизит в форму отчета и создаем событие “ПриСозданииНаСервере”, со следующем кодом:

&НаСервере

Процедура ПриСозданииНаСервере (Отказ, СтандартнаяОбработка)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

|ЗаработнаПлатаФактическийПериодДействия.Сотрудник,

|ЗаработнаПлатаФактическийПериодДействия.ВидРасчета,

|ЗаработнаПлатаФактическийПериодДействия.ПериодДействияНачало,

|ЗаработнаПлатаФактическийПериодДействия.ПериодДействияКонец

|ИЗ

|РегистрРасчета.ЗаработнаПлата.ФактическийПериодДействия КАК |ЗаработнаПлатаФактическийПериодДействия";

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

Точка = ДиаграммаГанта.УстановитьТочку(ВыборкаДетальныеЗаписи.Сотрудник);

Серия = ДиаграммаГанта.УстановитьСерию(ВыборкаДетальныеЗаписи.ВидРасчета);

Значение = ДиаграммаГанта.ПолучитьЗначение(Точка,Серия);

Интервал = Значение.Добавить();

Интервал.Начало = ВыборкаДетальныеЗаписи.ПериодДействияНачало;

Интервал.Конец = ВыборкаДетальныеЗаписи.ПериодДействияКонец;

КонецЦикла;

КонецПроцедуры

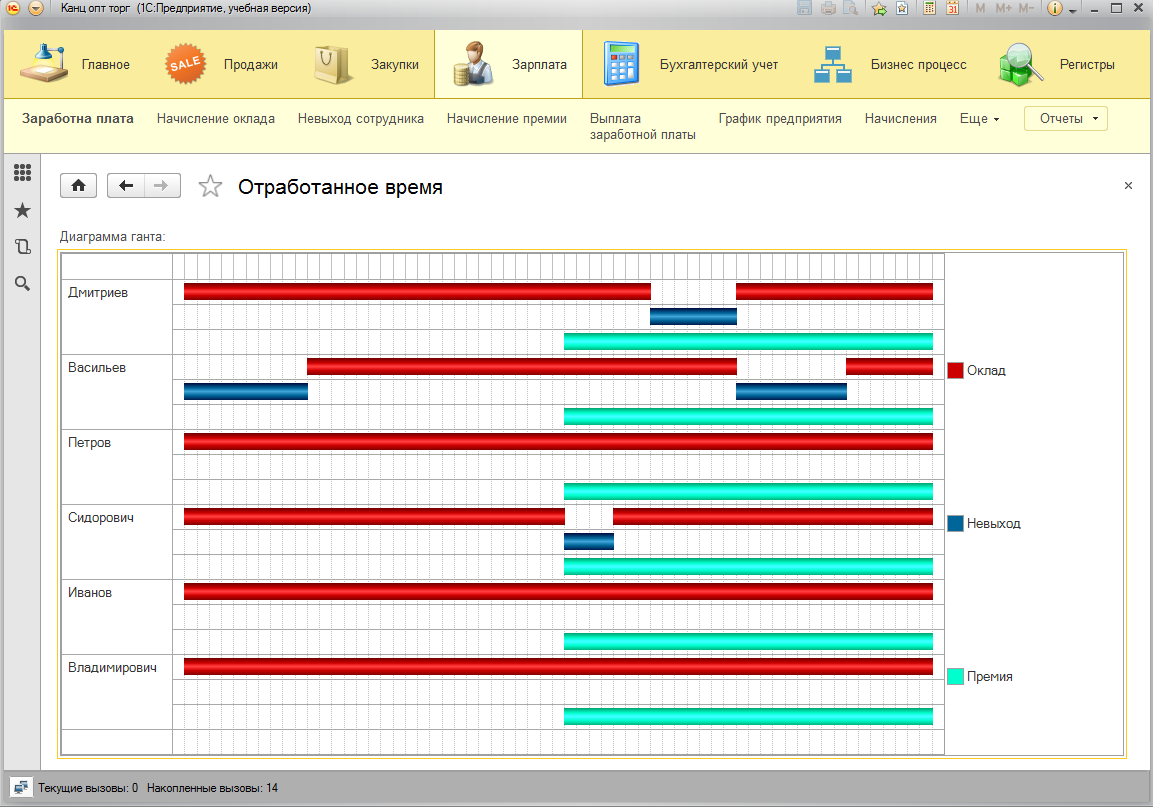


Рисунок 27 – Форма отчета “ОтработанноеВремя” **в пользовательском режиме**

7.17 Построение отчета, показывающего ABC-классификацию номенклатуры по параметру «Валовая прибыль». Создаем отчет “КлассификацияНоменклатуры”. Создаем новый набор данных “Запрос” и открываем конструктор запроса. Выбираем виртуальную таблицу реквизита накопления “ПродажиОбороты”, затем выбираем измерение “Номенклатура”, и ресурсы “Сумма” и “Себестоимость”, непосредственно в запросе создаем вычисляемое поле, для этого двойным кликом мыши нажимаем на регистр “Сумма” откроется конструктор произвольных выражений. В нем создать операцию вычитания суммы оборота на эго себестоимость в итоге получается валовая прибыль копируем это поле еще раз назовем его “Класс”, далее удаляем поле “Себестоимость”. Определяем ресурсы отчета. Заходим на закладку “Параметры”, указываем что дата выводится в формате “даты” без времени, создаем новый параметр “Период” тип параметра “Стандартный период”. Переходим на закладку “Настройки”, создаем с помощью конструктора новую настройку. Выбираем тип отчета “Список”, далее выводим все поля. Включим параметры в пользовательские настройки, чтобы они были сразу же доступны в форме отчета. Далее специальным образом нужно определить ресурс “Класс”, в поле “Выражение” вписываем код (КлассификацияABC("Сумма(Класс)", 3)).

Далее воспользуемся закладной “Макеты” создаем макет поля и указываем поле “Класс”, далее нужно указать область, где будет описываться поле “Класс” Рисунок 28.

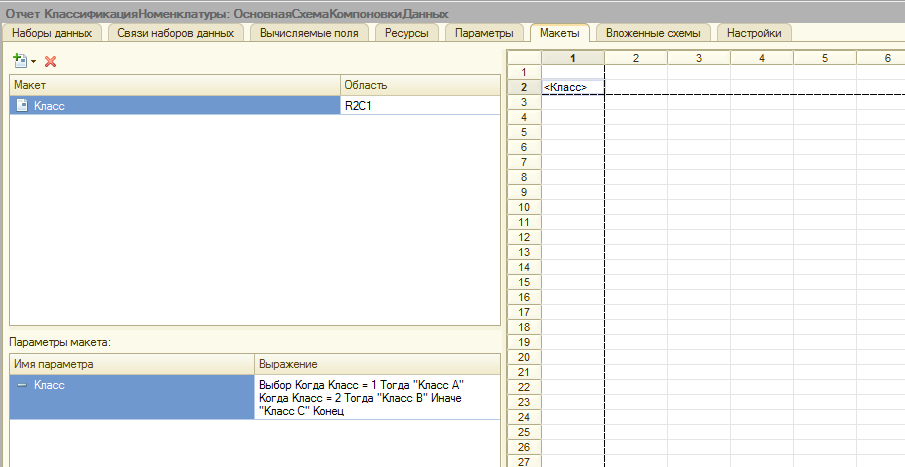


Рисунок 28 – Схема компоновки данных отчета “КлассификацияНоменклатуры”, составление макета.

В результате отчет примет вид как на Рисунок 29.

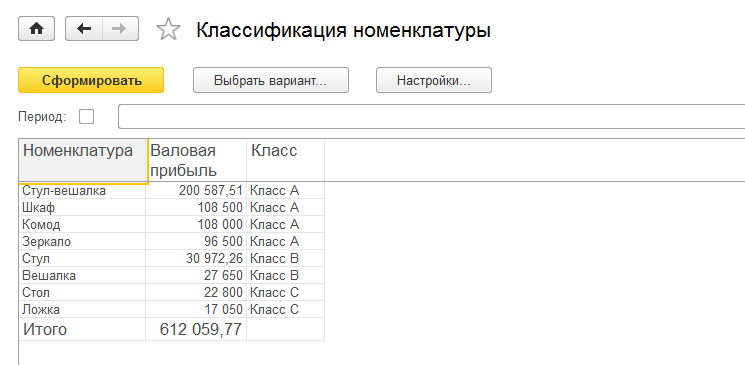


Рисунок 29 – Форма отчета “КлассификацияНоменклатуры” **в пользовательском режиме.**

# 8.Хранилище значения

Объект «ХранилищеЗначения» предназначен для хранения значения «как оно есть». Он может быть записан в поля информационной базы, имеющие соответствующий тип. Это позволяет сохранять в информационной базе любые значения (в том числе «не формализуемые», тип которых не может быть выбран в качестве типа поля, например «Картинка»). Большинство объектов, которые имеют неизменяемое значение, а также универсальные коллекции, могут быть преобразованы в значение «ХранилищеЗначения». К значению, хранящемуся в объекте, нельзя обращаться, его можно только извлечь из хранилища.

Не рекомендуется хранить в реквизитах этого типа ссылки на другие объекты информационной базы.

Для знакомства с объектом «ХранилищеЗначения» создайте у справочника «ФизическиеЛица» реквизит «Фото» (тип «ХранилищеЗначения»).