Центросоюз Российской Федерации

Российский университет кооперации

Волгоградский кооперативный институт (филиал)

Кафедра информационных систем в экономике

Курсовая работа

по дисциплине: Автоматизированные информационные системы бухучета

На тему: Разработка объектов прикладного решения «Мастер-класс на «1С: Предприятие 8.2» (Вариант 2)

Выполнил:

Группа 4ПИЭ61д

Проверила:

Иванеева Л.В.

Волгоград 2010

Содержание

Введение 3

1.Описание системы «1С-Предприятие» 4

1.1 Этапы ведения бухучета 4

1.2 Перечислить механизмы платформы «Прикладные механизмы» 5

1.3 Константы в системе 1С:Предприятие 8

1.4 Режим «Конфигуратор» 10

2. Разработать объекты прикладного решения «Мастер-класс на «1С: Предприятие 8.2»: Регистр, Документы, Отчеты 12

2.1 Регистр 12

2.2 Документы 18

2.3 Отчеты 36

Список использованной литературы 63

Введение

Программный продукт "1С:Бухгалтерия 8" включает технологическую платформу "1С:Предприятие 8" и конфигурацию (прикладное решение) "Бухгалтерия предприятия".

Конфигурация "Бухгалтерия предприятия" предназначена для автоматизации бухгалтерского и налогового учета, включая подготовку обязательной (регламентированной) отчетности в организации. Бухгалтерский и налоговый учет ведется в соответствии с действующим законодательством Российской Федерации.

"1С:Бухгалтерия 8" поддерживает решение всех задач бухгалтерской службы предприятия, если бухгалтерская служба полностью отвечает за учет на предприятии, включая, например, выписку первичных документов, учет продаж и т. д. Данное прикладное решение также можно использовать только для ведения бухгалтерского и налогового учета.

В состав конфигурации включен план счетов бухгалтерского учета, соответствующий Приказу Минфина РФ "Об утверждении плана счетов бухгалтерского учета финансово-хозяйственной деятельности организаций и инструкции по его применению" от 31 октября 2000 г. № 94н (в редакции Приказа Минфина РФ от 07.05.2003 № 38н). Состав счетов, организация аналитического, валютного, количественного учета на счетах соответствуют требованиям законодательства по ведению бухгалтерского учета и отражению данных в отчетности. При необходимости пользователи могут самостоятельно создавать дополнительные субсчета и разрезы аналитического учета.

1.Описание системы «1С-Предприятие»

1.1 Этапы ведения бухучета

Первоначальные настройка программы, вводится информация об организации, учетной политике, общая настройка, индивидуальная настройка. Для хранения, каких либо данных, первоначальные настройки заполняются объекты метаданных констант;

Создание справочников при первоначальной загрузки системы, некоторые справочники наполняются нормативной справочной информацией заранее созданной в определении конфигурации. Справочники бывают одноуровневые – информация о которых используется другими справочниками, и многоуровневые – содержит дерево групп подгрупп и таблиц содержимое выбрано элементами этого древа.

Ввод начальных остатков по счетам. Каждая проводка вводится в корреспонденцию счета 00, остатки по счетам на которых ведется аналитический учет вводится с указанием субконта.

Ведения бухгалтерского учета в текущем периоде. Работа на этом этапе выполняется со всеми объектами конфигурации. Анализ состояния счетов бухгалтер может произвести за любой период для самоконтроля или для руководителя предприятия о текущем состоянии бухгалтерского учета. Это выполняется с помощью стандартных отчетов конфигурации.

Формирование итоговой информации за период. Оформляется по завершении отчетного периода, на этом этапе проводится бухгалтерские итоги и формируются регламентируемые отчеты для руководителя и внешних контрагентов.

С 1 по 3 этапы выполняются подготовительно для начала работы, а 4 и 5 выполняются в течении всего жизненного цикла эксплуатации программы.

1.2 Перечислить механизмы платформы «Прикладные механизмы»

Понятие прикладных объектов.

Одним из важных принципов, заложенных в основу системы 1С:Предприятие 8.0, является стандартизация технологии разработки прикладных решений. Такой подход позволяет разработчику сосредоточить свои усилия на создании бизнес-логики приложения и не заниматься технологическими подробностями, такими как организация взаимодействия с базой данных, обработка транзакционных блокировок, нюансы программирования экранных форм и пр. Технологическая платформа 1С:Предприятия 8.0 берет на себя эти и многие другие технические вопросы, позволяя разработчику сконцентрироваться на решении прикладной задачи.

Это возможно благодаря тому, что прикладные решения в 1С:Предприятии разрабатываются в терминах проблеммно-ориентированных объектов, набор которых определен на уровне технологической платформы.

Справочники.

Справочники служат для описания таких сущностей как товары, контрагенты, валюты, склады и пр. Все эти сущности имеют общие свойства: внутренняя идентификация объекта в системе, необходимость поддержки иерархии и группировки элементов, необходимость поддержки вложенных таблиц и т.д.

Константы.

Предназначены для хранения постоянной или условно-постоянной информации. Например, константа может хранить название организации, фамилию, имя и отчество руководителя или главного бухгалтера, и т.д.

Перечисления.

Предназначены для описания перечня значений, которые может принимать какая-либо переменная. Например, перечисление может хранить значения, описывающие статус покупателя: "оптовый" и "розничный".

Документы.

Служат для описания таких сущностей как счета, накладные, заказы и пр. Эти сущности фиксируют различные события, происходящие в жизни организации, они привязаны ко времени, содержат вложенные таблицы, должны отражаться в учетных механизмах и т.д.

Оперативный учет.

Система поддерживает проведение документов пользователями в режиме "реального времени", что особенно актуально при многопользовательской работе.

Хранение сведений.

Предназначен для хранения многомерных сведений о значениях различных величин. Такими значениями могут быть, например, курсы валют или цены на товары конкурентов по состоянию на определенную дату. Эта информация может быть как статической, так и изменяющейся с течением времени - в этом случае для нее предусмотрено хранение истории изменений.

Описание характеристик.

Предназначен для хранения информации о характеристиках различных объектов. Позволяет пользователю создавать всевозможные характеристики, описывать тип этих характеристик и задавать их значения. Может использоваться, например, для предоставления пользователю возможности описывать товары произвольным количеством произвольных характеристик (цвет, размер, запах и т.д.). Позволяет создавать и хранить название характеристики и тип данных, который должны принимать значения этой характеристики.

Учет движения средств.

"Отвечает" за учет движений ресурсов (финансов, товаров, материалов и т.д.). Позволяет 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.

Бухгалтерский учет.

Предназначены для построения модели, реализующей систему двойной записи бухгалтерского учета. Реализуют многоуровневые планы счетов с фиксированной или переменной разрядностью кодов, многоуровневый и многомерный аналитический учет, учет по нескольким планам счетов, учет по нескольким организациям, опциональное ведение количественного, суммового и валютного учета по отдельным разрезам аналитики и т.д.

Сложные периодические расчеты.

Служат для реализации моделей расчета заработной платы. Позволяют описывать различные виды расчета (например, оклад, персональная доплата, алименты, штраф и т.д.), задавать правила, по которым одни виды расчета могут влиять на результаты других видов расчета, и хранить промежуточные данные и конечные результаты расчетов. С помощью этих объектов может быть организован расчет основных начислений, расчет налога на доходы физических лиц, распределение результатов расчета заработной платы для целей отражения в бухгалтерском учете и т.д.

Бизнес-процессы.

Позволяют создавать формализованные описания типичных последовательностей работ, выполняемых в организации, и на их основе формировать списки задач, которые необходимо выполнить тому или иному сотруднику организации в данный момент. Например, процесс продажи товара может быть представлен как последовательность выписки счета, его утверждения, получения наличной оплаты и отгрузки товара со склада. За выполнение каждого из этапов могут отвечать различные сотрудники, таким образом, в любой момент времени можно определить, в каком состоянии находится процесс продажи товара и кто из сотрудников в данный момент должен выполнить какие-либо действия.

Общие механизмы.

Механизм ввода на основании, реализуемый системой 1С:Предприятие, позволяет создавать новые документы, справочники и т.д., используя информацию, которая уже хранится в информационной базе.

Критерии отбора помогают отбирать различную информацию, хранящуюся в информационной базе, например, все документы, относящиеся к указанному поставщику.

1.3 Константы в системе 1С:Предприятие

В системе 1С:Предприятие константы предназначены для хранения постоянной или условно-постоянной информации. Такая информация либо совсем не изменяется в процессе деятельности предприятия, либо изменяется достаточно редко. Наиболее простой пример подобной информации — название организации, которое, как правило, не меняется.

Основная причина использования констант заключается в том, что в них один раз заносится какая-либо информация, которая затем может многократно использоваться при формировании документов, в расчетах, при построении отчетных форм. Информация, внесенная в константу время от времени может редактироваться, но сути константы это не меняет: единожды введенное в систему, значение константы используется многократно.

Рассмотрим такой пример. Наиболее часто на документах предприятия встречается подпись директора и главного бухгалтера. Естественно, что подписи эти должностные лица должны ставить самостоятельно. Но помимо самой подписи, в документах требуется ее расшифровка — фамилия подписавшего документ. Конечно, можно ввести фамилии директора и главного бухгалтера непосредственно в формы документов. Но, если какая-либо из фамилий изменится, то придется вновь редактировать все бланки документов и исправлять фамилии. Поэтому гораздо удобнее создать в Конфигураторе 2 константы — для хранения фамилий директора и главного бухгалтера, ввести эти фамилии в константы один раз, а в многочисленных бланках документов использовать идентификаторы констант для получения их значений. При смене директора или главного бухгалтера достаточно внести изменения только в константы — ввести и них новые фамилии — и все изменения автоматически будут отражены в тех местах, где эти константы используются.

Конфигуратор системы 1С:Предприятие позволяет создавать практически неограниченное количество констант для хранения любой нужной информации.

Следует обговорить, что в Конфигураторе создастся только описание константы как места для хранения какой-либо информации. Но в данном контексте для простоты будет использоваться термин «константа».

Управление списком констант.

Вся работа с объектами метаданных типа «Константа» ведется в окне «Конфигурация — Метаданные». Для констант отведена отдельная ветвь дерева метаданных, которая начинается у ключевого слова «Константы». Приемы создания, редактирования свойств и удаления констант совпадают с общими приемами управления объектами метаданных, изложенными в главе «Метаданные» на стр. 16.

Свойства константы.

В этом параграфе будут описаны специфические свойства константы, в дополнение к общим свойствам объектов метаданных, о которых говорилось в п. «Свойства объектов метаданных» на стр. 21.

Свойства константы редактируются при помощи палитры свойств «Свойства Константы». Необходимо обратить внимание, что константа является типизированным объектом метаданных (см. и. «Типизированные и типообразующие объекты метаданных», стр. 16).

Закладка «Дополнительные».

Периодический. Включение опции «Периодический» позволяет указать для константы хранение истории изменений значения константы. Такая константа в системе 1С:Предприятие называется периодической и обладает следующим свойством: в отличие от «обычных» (не периодических) констант, система 1С:Предприятие хранит значения такой константы но датам их изменения. При обращении к значению такой константы на некоторую дату выдается ее значение на эту дату, либо, если таковое отсутствует, — на ближайшую предыдущую дату, на которую устанавливалось значение константы.

К периодической константе можно обращаться при помощи встроенного языка системы 1С:Предприятие — получать и записывать значение константы на указываемую дату. Однако следует иметь в виду, что при получении значения константы на какую-то дату извлекается ее значение на указанную или (если значение константы на указанную дату отсутствует) на ближайшую предыдущую дату, а запись значения периодической константы выполняется всегда на указанную дату.

При использовании конфигурации для периодической константы можно вызвать окно «История», отражающее изменение значения константы по датам. Окно представляет собой таблицу, состоящую из двух граф: дата изменения значения константы и значение на эту дату. Окно «История» позволяет редактировать историю изменений значения константы напрямую: вводить новые строки, редактировать или удалять существующие.

1.4Режим «Конфигуратор»

Система «1С:Предприятие 8» имеет два основных режима запуска – "Конфигуратор" и "1С:Предприятие".

Первый из этих режимов предназначен для создания ИБ (информационной базы), внесения изменений в ее Конфигурацию, а также для выполнения административных функций.

В режиме "1С:Предприятие" пользователь запускает Конфигурацию на выполнение, как бы "проигрывая" файл ИБ. При этом программная часть системы использует структуры, созданные на этапе конфигурирования, предоставляя пользователю возможность заполнить их конкретными значениями.

Если в режиме "Конфигуратор" с помощью встроенного языка определены соответствующие алгоритмы обработки, то в режиме "1С:Предприятие" пользователь будет вызывать их работу, давая системе соответствующие команды.

2. Разработать объекты прикладного решения «Мастер-класс на «1С:Предприятие 8.2»: Регистр, Документы, Отчеты

2.1 Регистр

Теперь перейдем к добавлению «кубика», который на первый взгляд может показаться не совсем очевидным. Добавим регистр накопления Финансовые операции.

Сразу возникает вопрос: что такое регистр? Непонятно! Зачем он вообще нужен?

Попробуем объяснить просто. По условиям задачи наша система должна не просто хранить список наших знакомых и учитывать разнообразные события, происходящие в нашей жизни. Она должна уметь вести базовый финансовый учет - как минимум регистрировать денежные потоки и предоставлять минимальную отчетность, что же с нашими финансами происходит.

Регистрация различных событий, связанных с поступлением и затратой денег, будет производиться такими объектами конфигурации, как Документы. Чуть позже мы добавим парочку документов в нашу конфигурацию.

Кроме, собственно, регистрации фактов поступления и расходования денежных средств, мы наверняка захотим получать какие-то отчеты, наглядно отображающие происходящее с нашими финансами. Такие отчеты вполне можно было бы построить по тем данным, которые содержатся во вводимых пользователем документах поступления и расходования денег. Но представьте, что через месяц (или год), например, вы решите дополнить свой финансовый учет новыми документами или функционально расширить уже существующие.

Внести изменения в конфигурацию, как вы уже поняли, задача несложная. Как в этом случае быть с отчетами? Окажется, что наши отчеты, которые мы создавали раньше, будут работать неправильно. Потому что они учитывают только часть поступлений и расходов, ту часть, которая оформляется старыми документами. Значит, нам придется переделывать и все отчеты. А их в реальной конфигурации может быть несколько десятков...

Чтобы избежать этой работы и большого количества ошибок, которые могут возникнуть в результате такой переделки отчетов, в системе 1C: Предприятие применяется следующий подход.

Для хранения данных о движениях каких-либо сущностей (например, финансов) используются специальные объекты конфигурации - регистры. Документы, которые отражают события финансовой деятельности, заносят свои данные в эти регистры. А отчеты строятся по данным этих регистров.

В результате мы получаем независимую структуру. С одной стороны, однажды создав регистры нужной нам структуры, мы можем сразу же создать нужные отчеты, которые будут красиво и понятно показывать пользователям данные, содержащиеся в этих регистрах. С другой стороны, при появлении новых видов документов в системе нам нужно будет всего лишь правильно написать алгоритм, по которому их данные помещаются в эти регистры. При этом все созданные нами отчеты не потребуют переделки и будут показывать правильные данные. Итак, для нашей задачи достаточно создать один регистр накопления -Финансовые операции. В нем мы будем хранить данные о том, как часто к нам поступают деньги и как часто мы их расходуем.

Регистры накопления находятся в ветке дерева объектов конфигурации, которая так и называется: Регистры накопления. Добавим новый регистр накопления.

Назовем регистр Финансовые Операции и отнесем его к подсистеме Финансы. После чего можно перейти к закладке Данные.

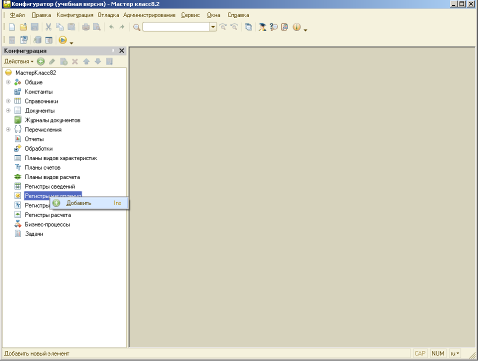


Рисунок 1. Создание регистра накопления

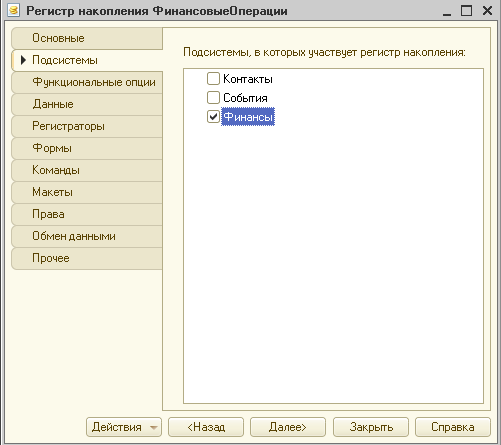


Рисунок 2. Данные регистр накопления

Ресурсы - это те данные, которые мы хотим получать из регистра. В нашем случае мы хотели бы знать сумму наших средств. Значит, нам нужен всего один числовой ресурс - Сумма.

Измерения - это то, в каких разрезах мы хотели бы получать данные из регистра. Вряд ли нам будет интересно каждый раз наблюдать только лишь остаток денег «вообще». Например, для анализа наших финансовых потоков, нас может заинтересовать, какие события или знакомые «приносят» нам больше всего денег или наоборот - «потребляют» © Поэтому, понадобится два измерения - Событие и Друг.

Реквизиты - это дополнительная информация, сопровождающая каждую запись в регистре. В нашем случае, мы не будем использовать реквизиты, но ничто не мешает вам добавить их в регистр в будущем и помещать в них любую вспомогательную информацию.

Итак, добавим перечисленные измерения и ресурс. Для добавления будем пользоваться контекстным меню, которое вызывается на соответствующих ветках данных регистра.

Добавим измерение.

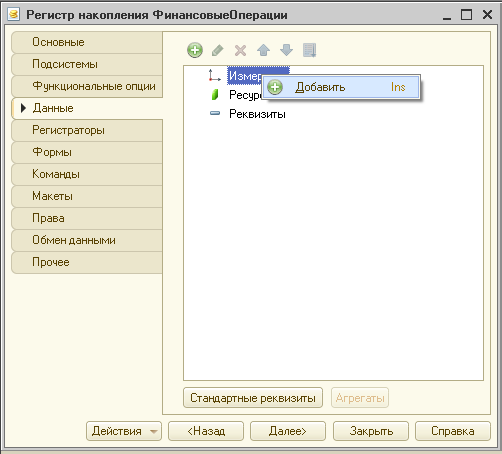


Рисунок 3. Добавление измерения

В палитре свойств укажем, что называться оно будет Событие, и выберем тип – Справочник Ссылка События.

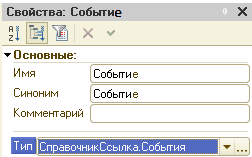


Рисунок 4. Свойства измерения Событие

Добавим второе измерение - Друг. Тип – Справочник Ссылка Друзья.

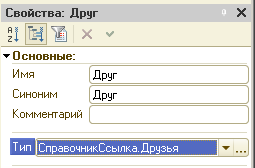


Рисунок 5. Свойства измерения Друг

Теперь добавим ресурс - Сумма. Тип и длину оставим такими, какие предложены по умолчанию - нас это устраивает. Скорректируем только точность - с 0 на 2.

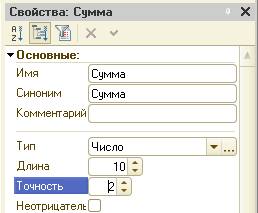


Рисунок 6. Свойства ресурса Сумма

В результате регистр накопления ФинансовыеОперации будет выглядеть следующим образом.

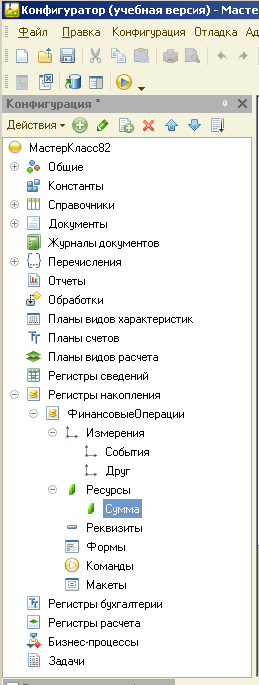


Рисунок 7. Регистр Финансовые Операции

2.2 Документы

Теперь перейдем к добавлению документов, регистрирующих поступление и расход денежных средств. Начнем с документа Поступление Денег. Документы находятся в ветке дерева объектов конфигурации, которая так и называется: Документы. Добавим новый документ.

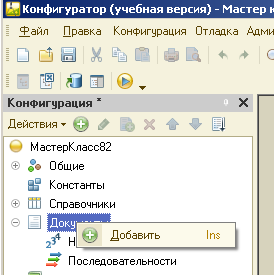


Рисунок 8. Добавление нового документа

Назовем его Поступление Денег, привяжем к подсистеме Финансы и перейдем на закладку Данные.

У каждого документа стандартно есть два реквизита - номер и дата создания. Однако, кроме них, всегда хочется иметь еще какую-то информацию, характеризующую конкретный документ. Например, в нашем случае хотелось бы понимать, от кого или в результате какого события к нам поступили денежные средства. Чтобы не создавать по документу на каждый факт поступления денег, позволим регистрировать одним документом несколько однотипных событий. К примеру, удобно за один день зафиксировать все поступления в одном документе - чтобы потом упростить поиск и анализ этой информации.

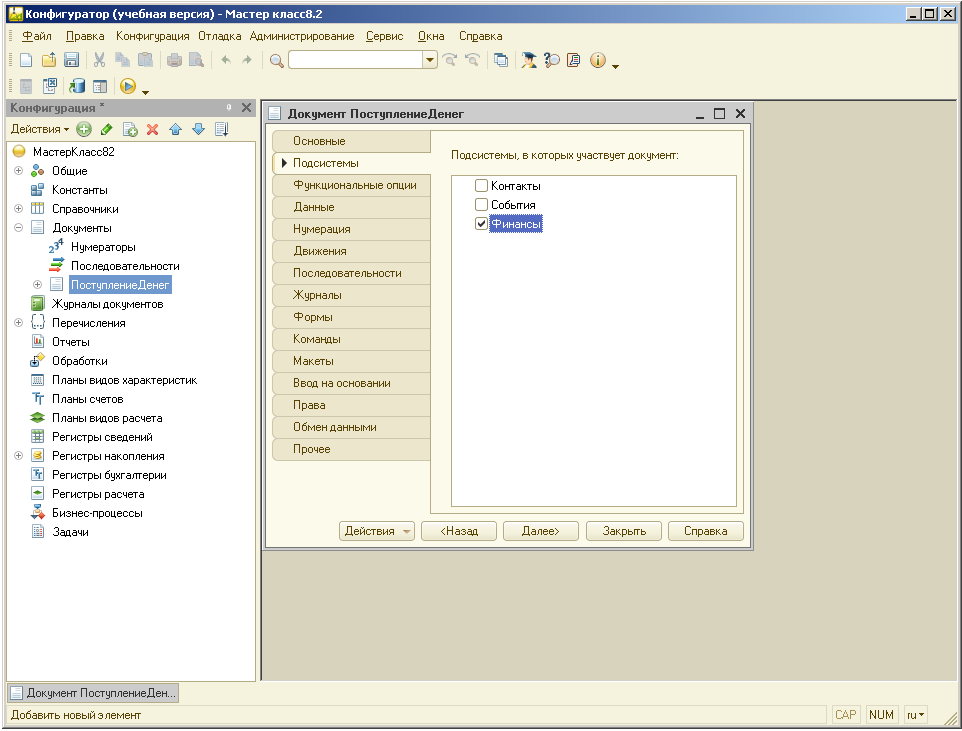


Рисунок 9. Создание документа ПоступлениеДенег

Как вы уже догадались, поможет нам в этом табличная часть. Создаем табличную часть Поступления.

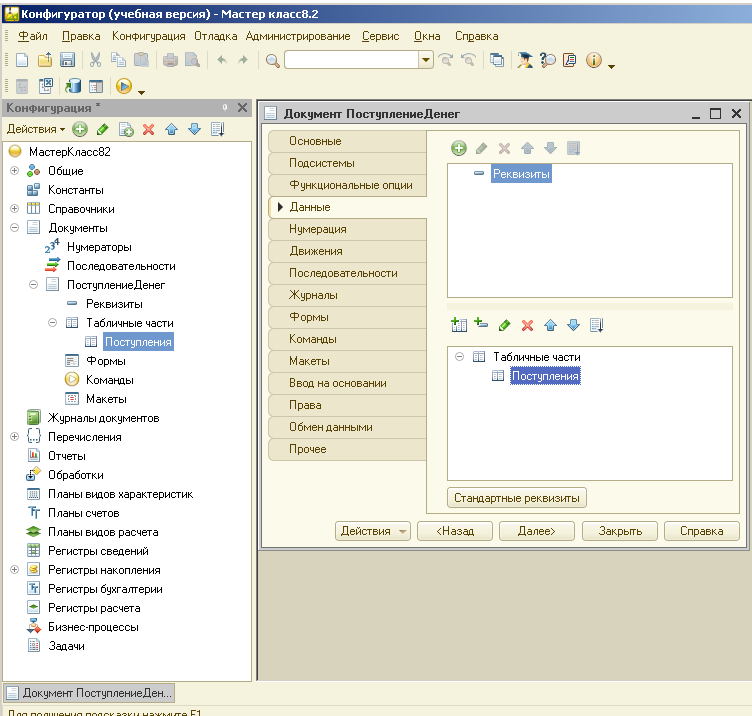


Рисунок 10. Создание табличной части Поступления

Для того чтобы иметь возможность отслеживать потоки по событиям и знакомым, заведем два реквизита только что созданной табличной части -Событие и Друг. Для фиксации, собственно, самой суммы операции - реквизит Сумма:

Событие, тип - СправочникСсылка.События

Друг, тип - СправочникСсылка.Друзья

Сумма, тип - Число, длина - 10, точность - 2, неотрицательное

В результате этих действий данные документа будут иметь следующий вид:

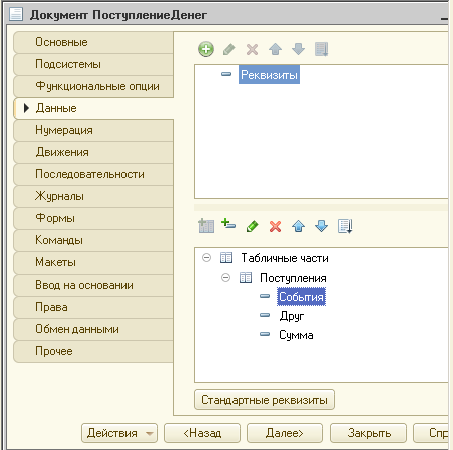


Рисунок 11. Заполнение табличной части реквизитами

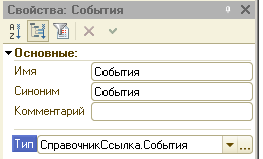


Рисунок 12. Реквизит Событие

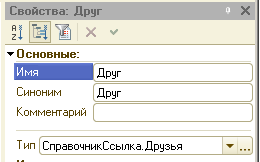


Рисунок 13. Реквизит Друг

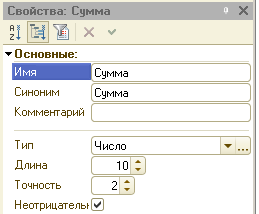


Рисунок 14. Реквизит Сумма

Заключительный этап работы с документом - перейдем на закладку Движения.

Как мы уже объясняли ранее, документы записывают какие-то данные в регистры, а отчеты показывают данные из регистров в том виде, который удобен для пользователя.

Движения документа - это как раз те записи, которые документ делает в регистрах. В нашем случае - в регистре Финансовые операции.

Поэтому раскроем ветку Регистры накопления, отметим флажком, что наш документ Поступление денег будет делать движения по регистру Поступпвние Денег и нажмем кнопку Конструктор движений, чтобы система помогла нам сформировать алгоритм записи данных в этот регистр.

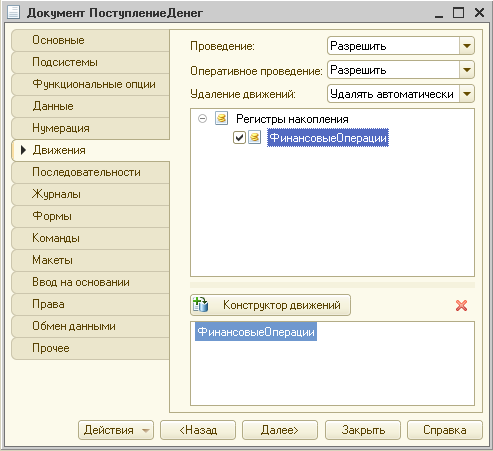


Рисунок 15. Конструирование движений документа по регистру

Откроется конструктор движений документа. Здесь мы оставим тип движений как есть («Приход», т.к. наш документ будет регистрировать поступление денег), выберем табличную часть - Поступления и нажмем кнопку Заполнить выражения.

В таблице, расположенной в нижней части окна, мы увидим, что система автоматически установила соответствие между реквизитами регистра накопления и реквизитами документа. Так, в измерение регистра Событие будут записаны данные реквизита Событие документа, в измерение Друг - данные реквизита Друг табличной части документа, а в ресурс Сумма - данные реквизита Сумма табличной части документа.

После заполнения и проверки таблицы соответствий можно нажать кнопку ОК.

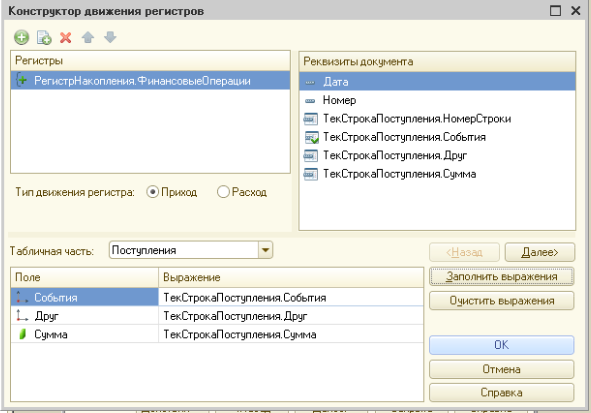


Рисунок 16. Конструктор движений

Система сформирует процедуру записи данных документа в регистр и покажет ее на экране (рис. 17).

Как можно заметить, процедура довольно проста. В цикле обходится табличная часть документа и для каждой строки, содержащейся в этой табличной части, в регистр добавляется новая запись, содержащая данные этой строки документа.

Мы могли бы написать этот алгоритм сами, но для облегчения работы мы воспользовались конструктором движений. Конструктор сформировал этот код автоматически, о чем говорят комментарии в начале и в конце процедуры.

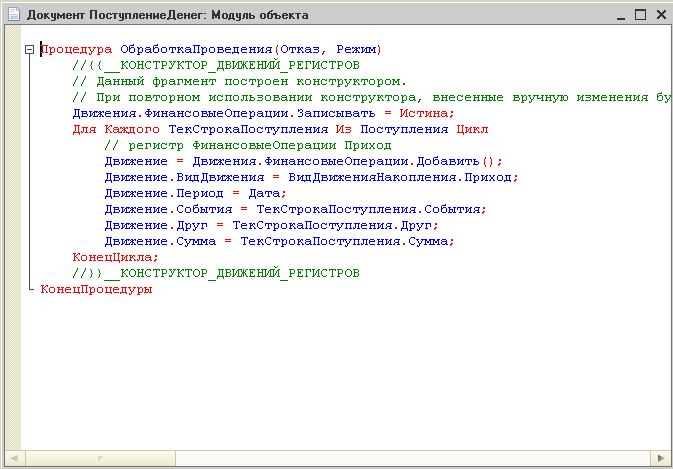


Рисунок 17. Процедура записи данных документа в регистр

Теперь запустим систему в режиме 1С:Предприятие и с помощью специальной обработки загрузим демонстрационные данные в документ Поступление денег из файла «06-ПоступлениеДенег.xml».

Перейдем на закладку Финансы и откроем список документов Поступление денег.

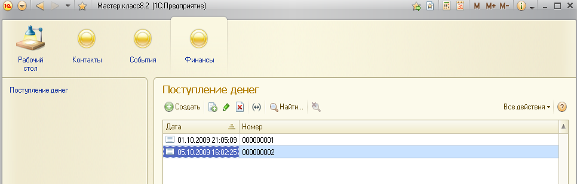


Рисунок 18. Документы Поступление денег

Откроем первый документ. Пока мы просто загрузили данные в документ и эти данные никак не отразились в нашем учете. Для того чтобы добавленные нами данные повлияли на состояние учета, нужно документ записать и провести. В момент проведения как раз и будут добавлены записи в регистр накопления Финансовые операции.

Чтобы записать и провести документ, нажмем кнопку Провести и закрыть.

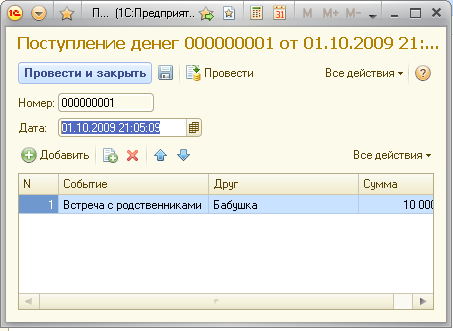


Рисунок 19. Проведение документа Поступление денег

Успешная запись и проведение документа будут отражены системным сообщением платформы «1С:Предприятие 8.2» (в правом нижнем углу экрана).

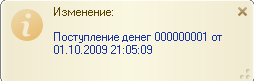


Рисунок 20. Успешная запись и проведение документа Поступление денег

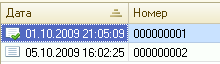


Рисунок 21. Проведенный документ

Сам же факт успешного проведения документа в системе отражается в виде отдельного значка в списке документов.

Проверить факт проведения, т.е. помещения нужной информации в записи регистра Финансовые операции можно следующим способом: зайти в главное меню ► Все функции... ► раскрыть узел Регистры накопления » Финансовые операции

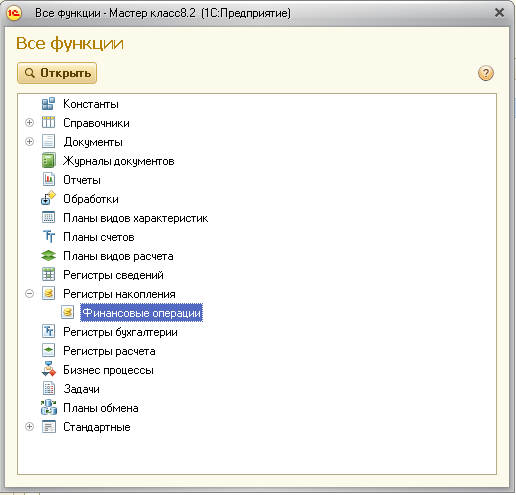


Рисунок 21. Все функции

Откроется форма записей регистра накопления Финансовые операции. На данный момент времени в регистре содержится лишь одна запись - проведенного первого документа (кликнув по этой записи, можно всегда открыть документ, ее породивший). Обратите внимание на значок «+», который символизирует добавление информации к значениям ресурсов.

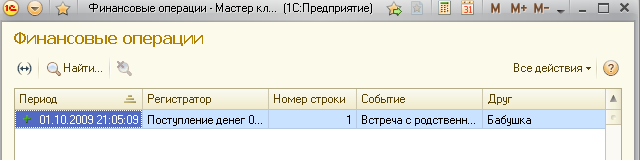


Рисунок 22. Записи регистра накопления

Переключившись на основное окно программы, проведем второй документ. После успешного проведения вернемся обратно в форму списка регистра Финансовые операции. Видно, что там появилась вторая запись, соответствующая второму документу.

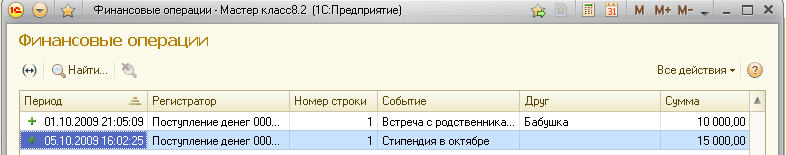


Рисунок 23. Записи регистра накопления

С поступлением денег мы закончили. Но надо фиксировать еще и расходы. Возвращаемся в конфигуратор. По логике вещей, нужно создать документ РасходДенег. Можно быстро создать его вручную по образу и подобию уже существующего документа ПоступлениеДенег - ведь по своему составу они будут практически идентичны. А можно воспользоваться сервисом платформы «1С:Предприятие 8.2» - созданием новых объектов конфигурации на основании уже существующих путем их копирования.

Для этого нужно установить курсор на существующий документ Поступление Денег в списке объектов метаданных и нажать кнопку Добавить копированием (F9).

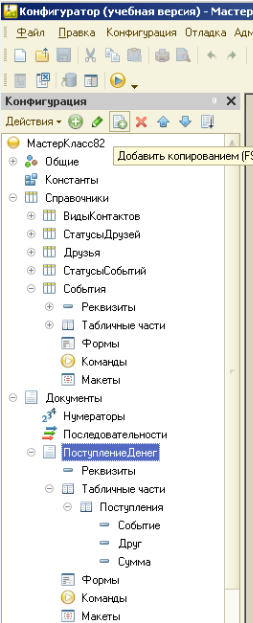


Рисунок 24. Добавление копированием

После нажатия на эту кнопку появится ПоступлениеДенег1 - точная копия Поступление Денег.

Нам останется только переименовать этот документ и немного подправить его свойства. Этим и займемся.

Прежде всего откроем окно редактирования его свойств и изменим имя документа – Расход Денег. Помня про подсистемы, привяжем новый документ к подсистеме Финансы.

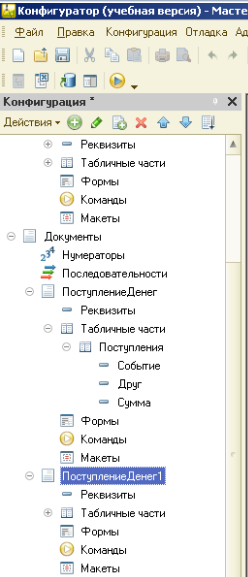


Рисунок 25. Копия документа Поступление денег

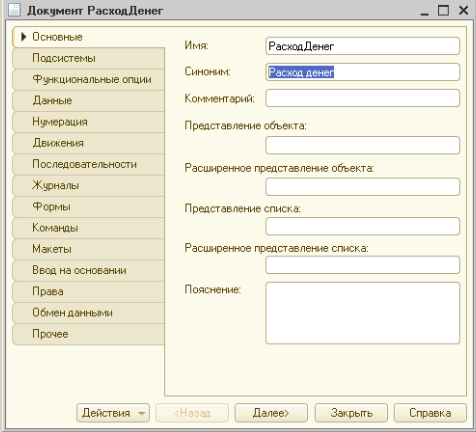


Рисунок 26. Документ Расход денег

На закладке Данные изменим при помощи палитры свойств название табличной части - Затраты. Остальные реквизиты не трогаем - они нас полностью устраивают.

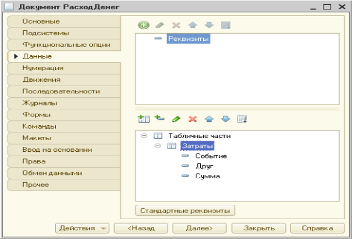


Рисунок 27. Переименование табличной части

Переходим на закладку Движения. Здесь нам нужно будет исправить движения документа. Сейчас они скопированы из документа ПоступлениеДенег. Тот документ учитывал поступления денег. Здесь же нам нужно учитывать расходы. Запустим конструктор движений. Система предупредит нас, что существующая процедура проведения будет замещена. Согласимся.

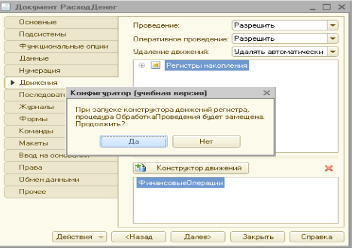


Рисунок 28. Конструктор движений

Тут все выполняем по аналогии с предыдущим документом. Меняем только тип движения документа на Расход и для заполнения выбираем табличную часть Затраты. Снова заполняем движения и получаем готовый модуль проведения при нажатии на кнопку ОК.

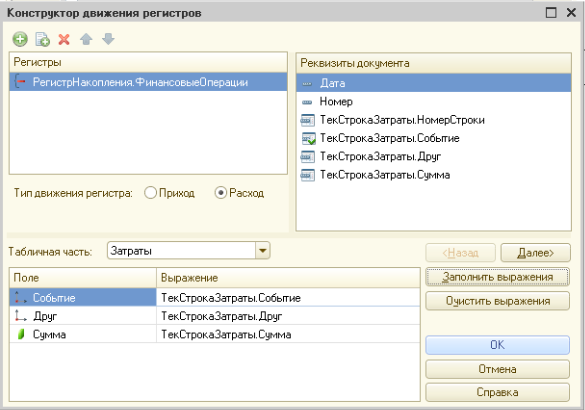


Рисунок 29. Конструктор движений

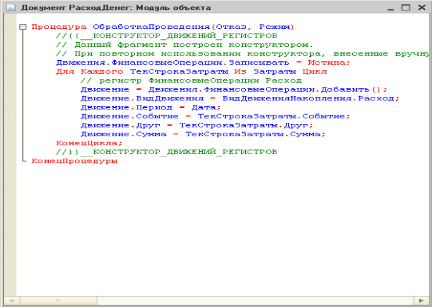


Рисунок 30. Модуль проведения документа Расход денег

Можно заметить, что отличие от документа ПоступлениеДенег - по сути всего в одной строке, которая задает вид движений. В нашем случае в регистр будут добавляться движения расхода.

Запускаем пользовательский режим и загружаем обработкой данные по расходам денег из файла «07-РасходДенег.xml». Переходим на закладку Финансы и открываем список документов Расход денег.

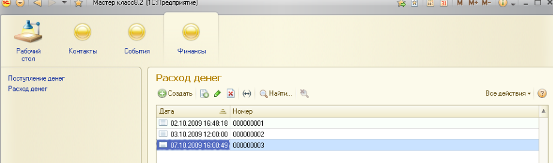


Рисунок 31. Список документов Расход денег

Проведем эти три документа и проконтролируем, что записи о фактах затрат денежных средств отразились в регистре накопления Финансовые операции.

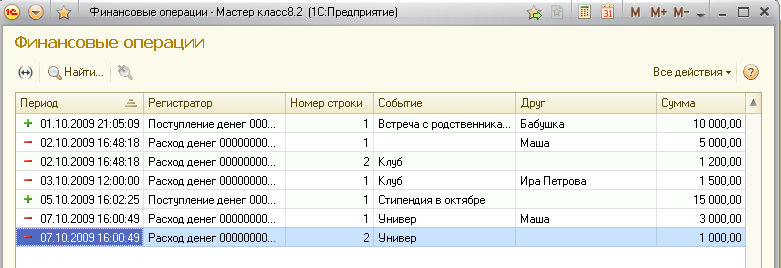


Рисунок 32. Записи регистра накопления

Нетрудно догадаться, что «расходные» записи отражаются в списке регистра значком «-». Кроме этого, видно, что несколько строк одного документа попадают в регистр в виде отдельных записей - обратите внимание на реквизит Номер строки.

На этом учетную (регистрирующую) часть нашей программы можно считать завершенной. Мы уже сейчас можем заполнять список своих друзей знакомых, регистрировать поступление денег и их траты.

Но «просто считать» мало - хотелось бы получать из системы информацию в удобной для восприятия виде. Для этого предусмотрен механизм отчетов.

2.3 Отчеты

Следующим шагом в разработке нашей конфигурации будет создание отчетов, в удобном виде отображающих информацию о нашей деятельности. Отчеты мы создадим тоже «без программирования», используя исключительно средства визуального конструирования.

Отчеты находятся в ветке дерева объектов конфигурации, которая так и называется - Отчеты. Добавим новый отчет. Для начала это будет отчет Затраты.

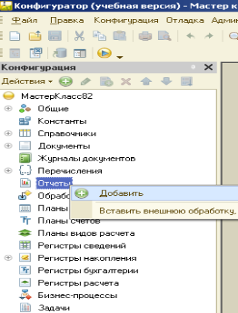


Рисунок 33. Создание нового отчета

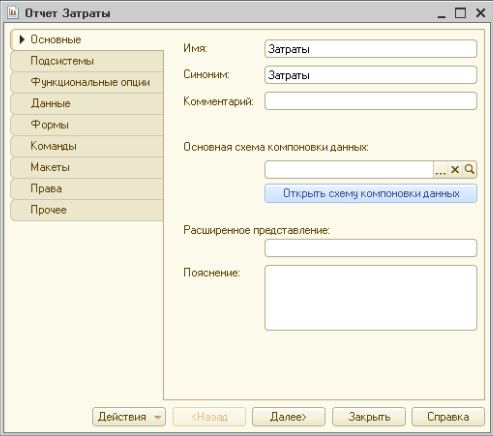


Рисунок 34. Отчет Затраты

Так как это новый отчет и он, естественно, еще не содержит никакой схемы компоновки, система откроет конструктор макета этого отчета, где предложит создать макет, содержащий схему компоновки. Согласимся и нажмем на кнопку Готово.

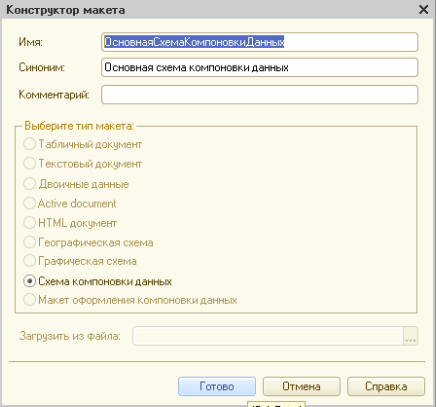


Рисунок 35. Конструктор макета

После этого система откроет конструктор схемы компоновки данных. Пока схема компоновки пустая.

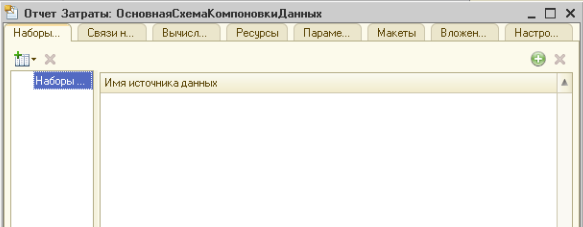


Рисунок 36. Пустая схема компоновки данных

Нам предстоит описать набор данных, из которого отчет будет получать данные, описать структуру отчета.

Добавим набор данных - запрос. То есть данные для этого отчета мы будем получать запросом из базы данных 1С Предприятия.

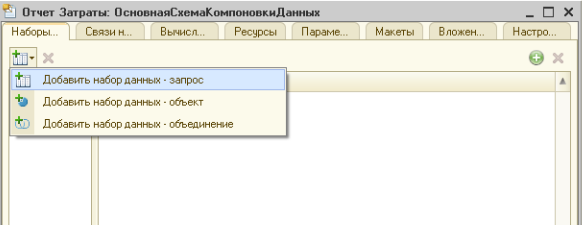


Рисунок 37.Создание нового запроса

Можно ввести текст запроса вручную, а можно использовать конструкта. Мы выберем второй способ и нажмем на кнопку Конструктор

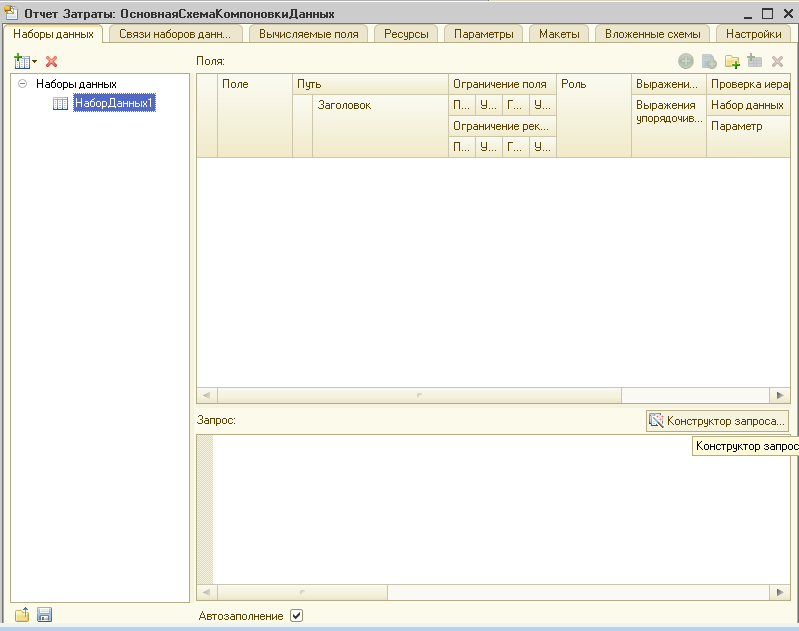


Рисунок 38. Вызов конструктора запроса

В открывшемся окне конструктора запроса слева перечислены таблицы, изкоторых мы можем получать данные. Раскроем ветку Регистры Накопления и двойным щелчком мыши выберем из нее таблицу Финансовые Операции. Остатки И Обороты.

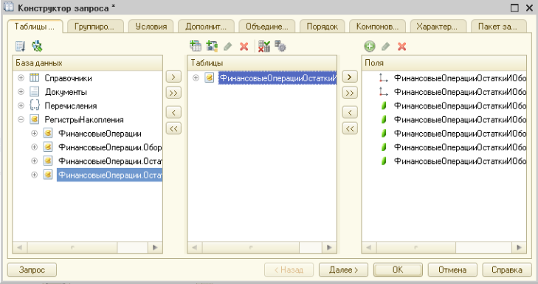


Рисунок 39. Выбор таблицы Финансовые Операции. Остатки И Обороты

В среднем списке выберем эту таблицу и нажмем на кнопку Добавить все поля. Все поля этой таблицы будут выбраны в качестве полей запроса.

На этом создание нашего запроса завершено - нажмем кнопку ОК. Система создаст текст запроса и автоматически заполнит поля системы компоновки данных.

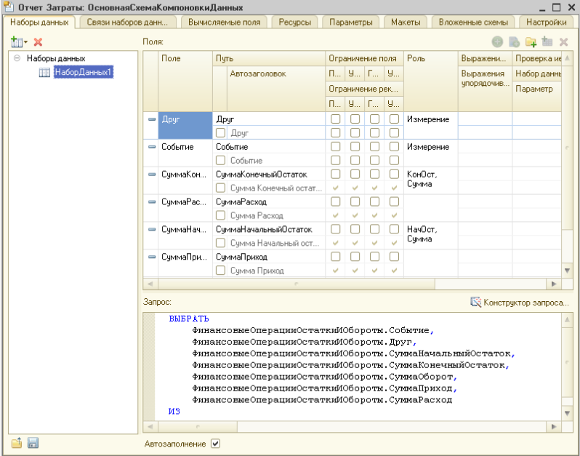


Рисунок 39. Поля схемы компоновки

Перейдем на закладку Ресурсы. Здесь двойным щелчком выберем следующие поля:

СуммаКонечныйОстаток

СуммаНачальныйОстаток

СуммаОборот

СуммаПриход

СуммаРасход

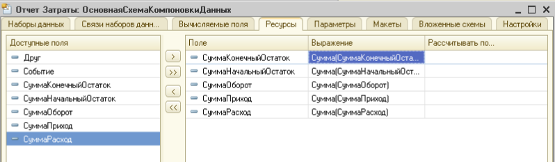


Рисунок 40. Ресурсы схемы компоновки

Перейдем на закладку Настройки. Здесь для того, чтобы создать структуру нашего отчета, мы тоже воспользуемся конструктором. Для этого нажмем на кнопку Конструктор настроек...

Конструктор настроек позволяет быстро создать несколько простых вариантов настройки отчета. Для наших целей хорошо подойдет список, поэтому ничего не меняем и просто нажимаем кнопку Далее >.

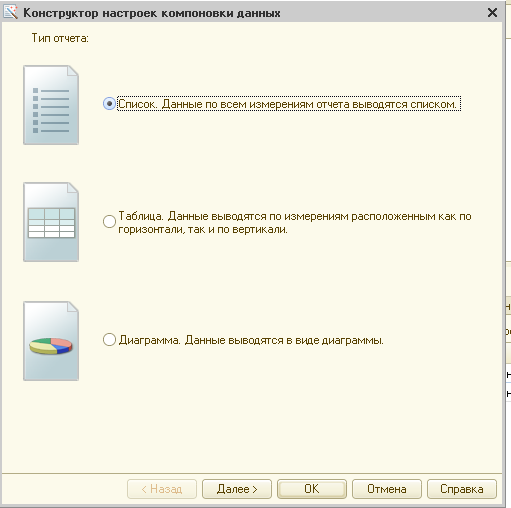


Рисунок 41. Выберем отчет в виде списка

На следующем шаге выберем поля, которые будут отображаться в нашем отчете. Двойным щелчком в списке Доступные поля выберем следую-

поля:

Друг

Событие

СуммаОборот

СуммаПриход

СуммаРасход

Нажмем Далее >.

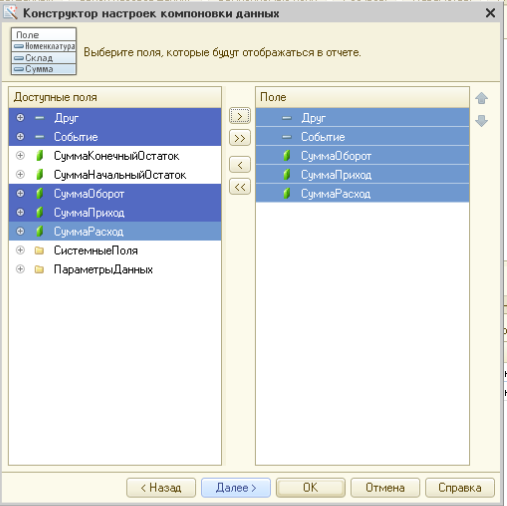


Рисунок 42. Выберем поля для отображения в отчете

Теперь нам нужно выбрать поля, по которым данные в отчете будут сгруппированы. Нам интересно, чтобы можно было видеть движения по каждому знакомому в отдельности и еще - по каждому событию, связанному с этим знакомым. Поэтому двойным щелчком из доступных полей выбираем оба - Друг и Событие. После чего можно нажать ОК, т.к. конструирование вида отчета мы закончили.

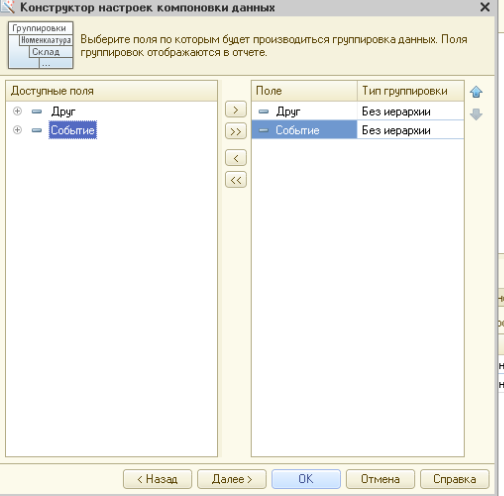


Рисунок 43. Поля группировки в отчете

Система покажет структуру отчета

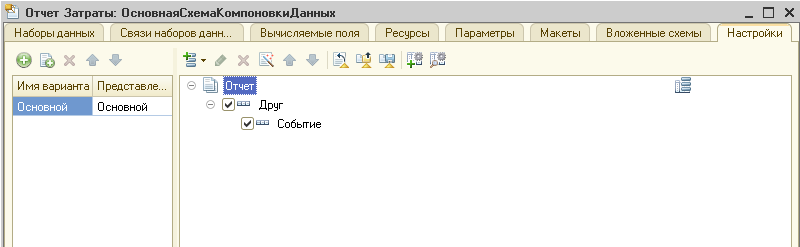


Рисунок 43. Готовая структура отчета

Практически закончили. Дадим возможность пользователю отчета произвольно задавать период выборки данных - это будет полезно, когда в регистре накопится много записей за довольно ощутимый период. Сделать это просто - ниже структуры отчета нам доступны два параметра: Начало периода и Конец периода.

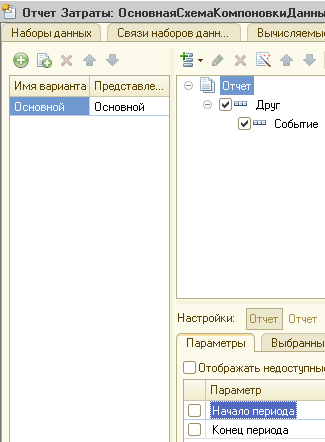


Рисунок 44. Параметры отчета

Достаточно выделить мышью каждый из параметров и выбрать вариант Свойства элемента пользовательских настроек.

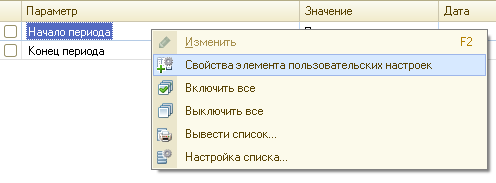


Рисунок 45. Свойства элемента пользовательских настроек

В открывшемся окне достаточно включить флажок Включать в пользовательские настройки и нажать ОК.

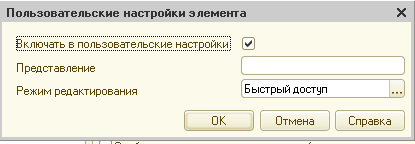


Рисунок 46. Включать в пользовательские настройки

Выполнив эту операцию для обоих параметров (Начало периода и Конец периода), продолжаем настройку нашего отчета. Для этого мы закрываем окно настройки схемы компоновки данных и возвращаемся в окно параметров отчета Затраты. Переключаемся на закладку Подсистемы и привязываем наш отчет к подсистеме Финансы.

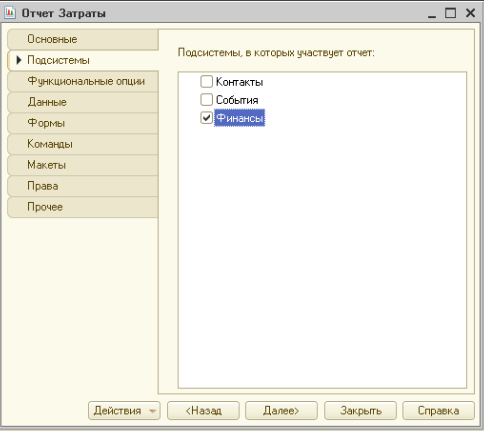


Рисунок 47. Подсистема Финансы

Запустим систему в режиме 1С Предприятие и посмотрим, как выглядит наш отчет. Искать наш отчет следует на закладке Финансы - обратите внимание, как система самостоятельно отделила его в отдельную область на экране.

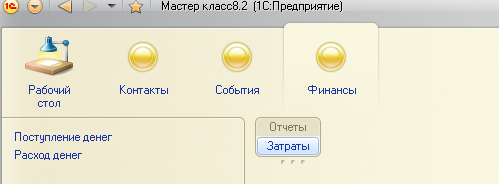


Рисунок 48. Открытие отчета затраты

Открыв окно отчета, сформируем его, нажав на кнопку Сформировать.

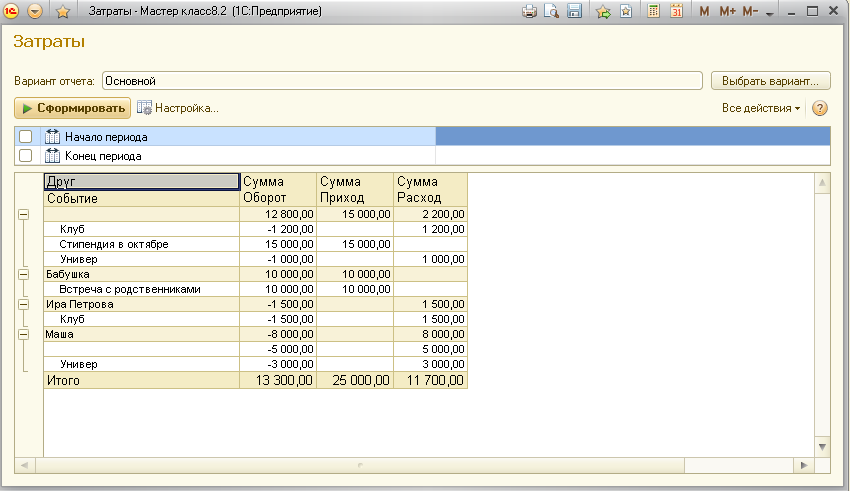


Рисунок 49. Отчет Затраты

Убедившись, что отчет работает и показывает нам динамику поступления и расходования денежных средств в разрезах тех измерений, которые мы задали в свое время в регистре, возвращаемся в конфигуратор. Создадим еще один простенький отчет, способный быстро отобразить нам текущий остаток доступных денежных средств. Называем его СколькоДенег и относим сразу к подсистеме Финансы. Аналогично отчету Затраты создаем схему компоновки данных (на закладке Основные). В качестве источника данных снова выбираем Запрос.

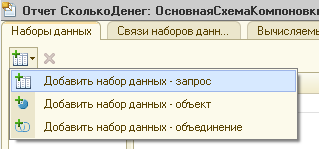


Рисунок 50. Запрос для отчета СколькоДенег

В открывшемся окне снова воспользуемся Конструктором запросов. Но для этого отчета выберем другую таблицу - ФинансовыеОперацииОстатки.

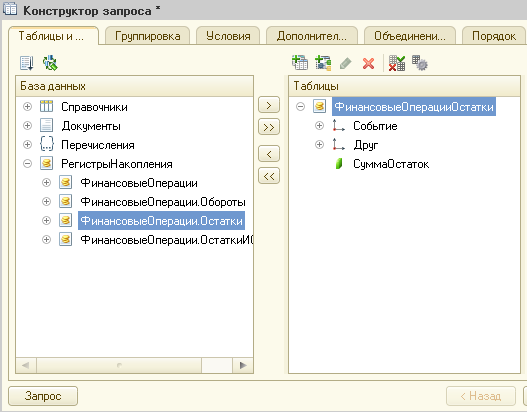


Рисунок 51. Источник данных для отчета СколькоДенег

В средней части раскрываем узел и выбираем только одно поле - СуммаОстаток.

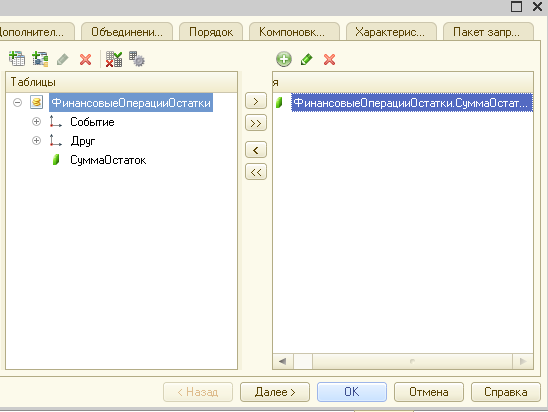


Рисунок 52. Поле СуммаОстаток

Закрываем конструктор запросов кнопкой ОК. После чего попадаем в окно схемы компоновки данных, закладка Наборы данных. Тут нас все устраивает. Переходим на закладку Ресурсы, где единственный наш ресурс СуммаОстаток, и выбираем.

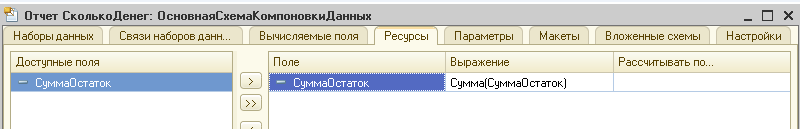


Рисунок 53. Ресурс СуммаОстаток

Переходим на закладку Настройки. Вновь воспользуемся Конструктором настроек. Снова нас устроит предлагаемый по умолчанию тип Список.

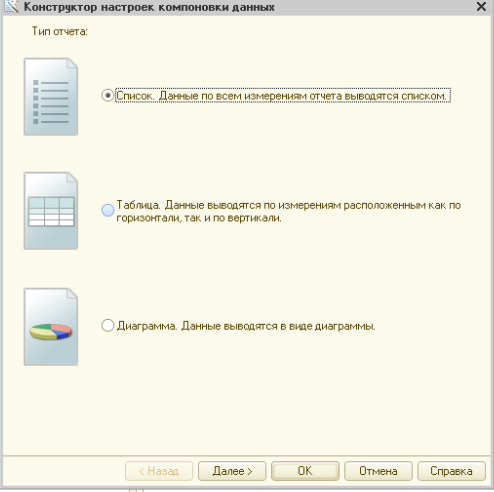


Рисунок 54. Конструктор настроек

На следующей закладке из Параметров данных выберем стандартное поле Период (чтобы иметь возможность видеть дату, на которую будет стро¬иться отчет) и поле СуммаОстаток.



Рисунок 55. Поля отчета

Выбрав поля, можно сразу нажать ОК, т.к. большего нам от этого отчета не потребуется. Далее для параметра Период, как и в отчете Затраты, включаем его в пользовательские настройки.

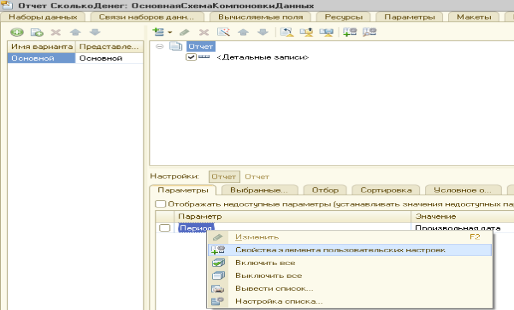


Рисунок 56. Включаем пользовательские настройки

Значение параметра Период устанавливаем в Начало этого дня, чтобы отчет всегда автоматически был готов показать информацию на текущий день.

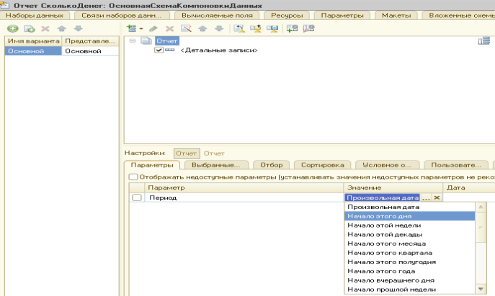


Рисунок 57. Начало этого дня для Периода

В итоге настройки нашего отчета будут выглядеть вот так.

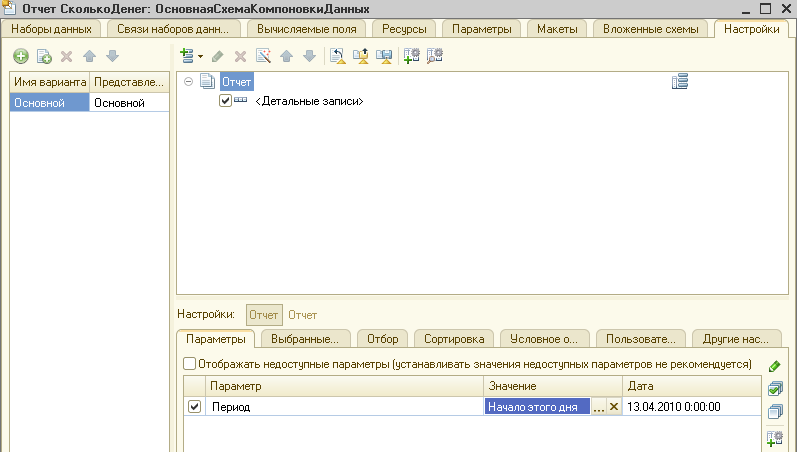


Рисунок 58. Настройки отчета СколькоДенег

Запускаем пользовательский режим и проверяем работу этого отчета.

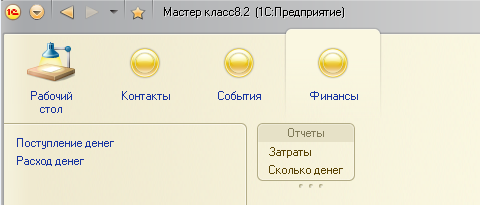


Рисунок 59. Отчет Сколько денег

Информация, сообщаемая отчетом, крайне проста, но полезна - вы всегда сможете посмотреть, сколько финансов доступно на текущий момент времени.

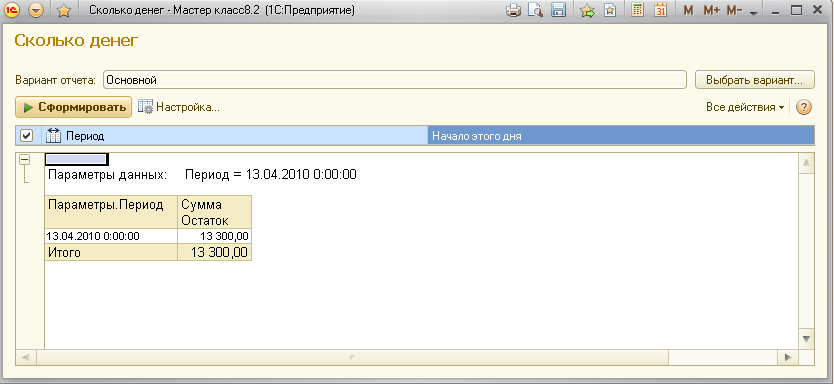


Рисунок 60. Результат отчета Сколько денег

Сделаем еще один отчет, и на этом пример разработки нашей системы CRM можно будет считать оконченным. Возвращаемся в конфигуратор, создаем новый отчет, называем его ГрафикПоДням, относим к подсистеме Финансы и создаем схему компоновки данных. В схеме источником данных вновь выбираем Запрос и запускаем Конструктор запросов.

В качестве источника данных выбираем таблицу ФинансовыеОперации-ОстаткиИОбороты. До этого момента процесс ничем не отличается от того, что мы делали в отчете Затраты.

А теперь мы провернем вот такой трюк. Учитывая то, что наш отчет называется ГрафикПоДням, вы верно предположили, что в отчете мы будем получать не цифровые данные, а графические. Подготовим данные из регистра в нужном для отчета виде. Для этого отметим таблицу ФинансовыеОпера-цииОстаткиИОбороты в среднем окне конструктора запросов и нажмем кнопку Параметры виртуальной таблицы.

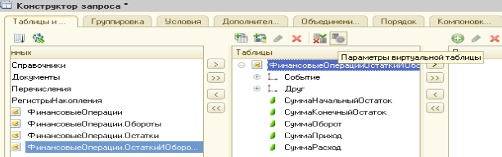


Рисунок 61. Настройка параметров виртуальной таблицы

В открывшемся окне в поле Периодичность выбираем День и закрываем это окно кнопкой ОК.

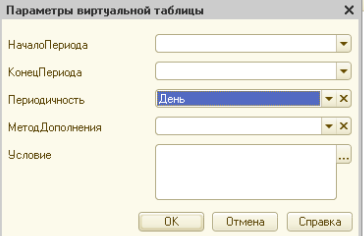


Рисунок 62. Параметры виртуальной таблицы

После этого выбираем два поля из нашей таблицы: Период и СуммаКонечныйОстаток.

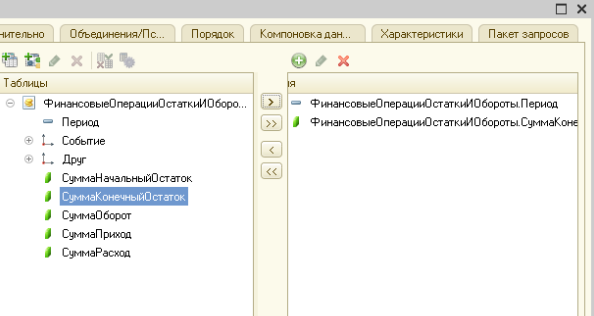


Рисунок 63. Поля Период и СуммаКонечныйОстаток

Закрываем констриктор запросов кнопкой ОК. Переходим на закладку Ресурсы и выбираем там поле СуммаКонечныйОстаток.

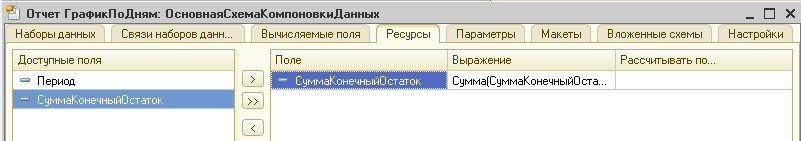


Рисунок 64. Ресурсы для отчета

Переходим на закладку Настройки и вызываем конструктор настроек. В этот раз выбираем вариант Диаграмма. После чего нажимаем Далее>.

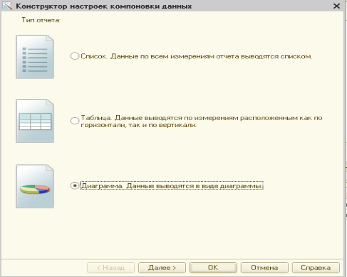


Рисунок 65. Диаграмма

Выбираем поля Период и СуммаКонечныйОстаток и жмем Далее>.

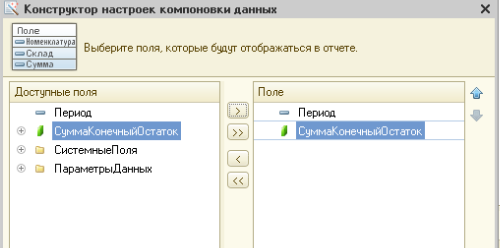


Рисунок 66. Поля диаграммы

В следующем окне настройки диаграммы помещаем поле Период в раздел Точки. Жмем Далее>.

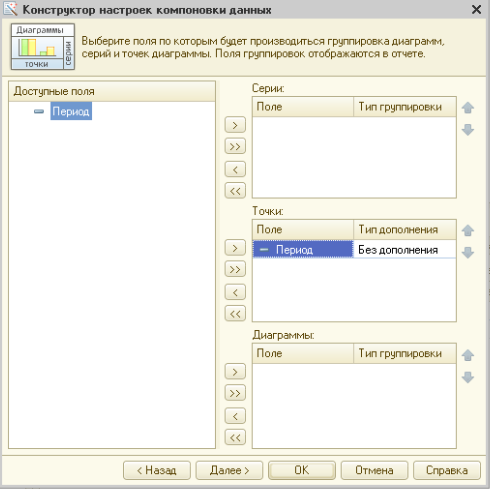


Рисунок 67. Точки диаграммы

В поля упорядочивания тоже добавляем наш Период. Жмем Далее>.

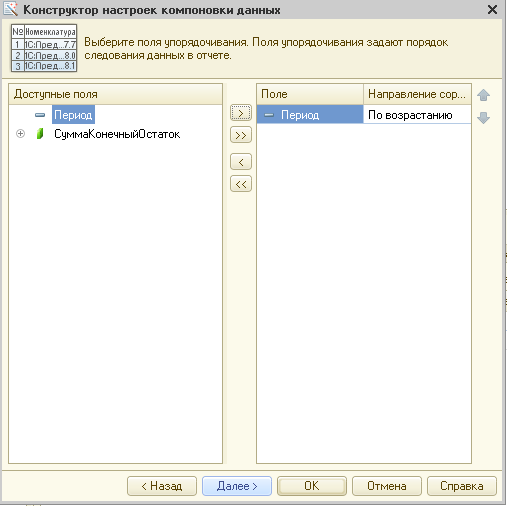


Рисунок 68. Поля упорядочивания диаграммы

Тип диаграммы - График. На этом настройка внешнего вида нашей диаграммы закончена и можно нажать ОК.

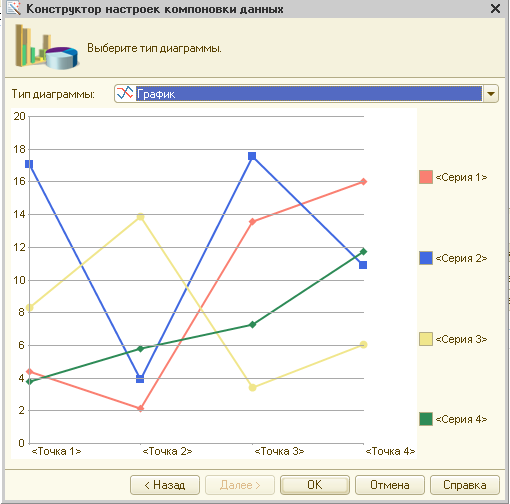


Рисунок 69.Тип диаграммы

Как и в предыдущих отчетах, для параметров Начало периода и Конец периода активизируем включение в пользовательские настройки.

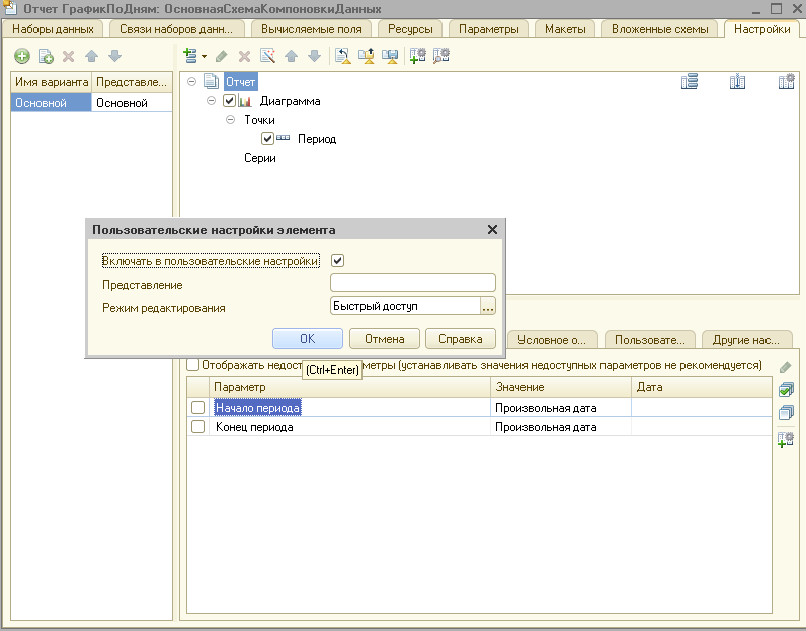


Рисунок 70. Включение в пользовательские настройки

Проверяем, что получилось, в пользовательском режиме. Запускаем и формируем отчет График по дням.

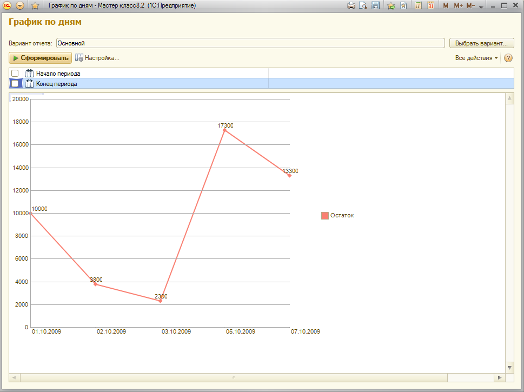


Рисунок 71. График по дням

Наглядно видно, как вели себя наши финансовые показатели в зависимости от поступления или расходования денежных средств.

Список использованной литературы

1)Иванеева Л.В. \ Иванеева Л.В. «Автоматизированные информационные системы бухучета»\Автоматизированные информационные системы бухгалтерского учета. Методические указания к курсовой работе для студентов специальности 080801 «Прикладная информатика (в экономике)» всех форм обучения - Волгоград: Волгоградский кооперативный институт, 2008 – 13 с.

2)Рыбалка В.В. «Пример быстрого разработки приложения на платформе 1С:Предприятие 8.2»

3)HTTP://V8.1C.RU/BUHV8/321.HTM

4)HTTP://RU.WIKIPEDIA.ORG/WIKI/БУХУЧЁТ