# Занятие 5. Документы. Конструктор формы

### §5.1. Документы

Документ – объект, отражающий факт хозяйственной деятельности на предприятии. К документам относятся документы поступления и продажи товара, приказы о начислении заработной платы, премировании сотрудников, формирование цен на товары и услуги и т.д. Ключевыми особенностями документа являются: наличие даты (точное положение на оси времени, к которому привязывается документ) и номера документа, а также возможность его проведения (специальное действие, которое меняет учет, к примеру, материальное или финансовое состояние на предприятии).

Документы, как и справочники, являются типообразующими объектами. Ссылка на документ формируется на основе даты создания и типа документа. К основным реквизитам, характеризующим документы одного вида, относят дату и номер. Перечень всех документов одного вида располагается в порядке времени их создания. Т.к. в одну секунду может быть создано множество документов одного вида (вплоть до нескольких тысяч), то для правильной хронологической последовательности, кроме даты, каждый документ дополнительно характеризуется моментом времени, который формируется на основании даты и ссылки на документ.

Создадим документ «**Приходная накладная**», который будет отображать факт поступления товаров на предприятие. Добавим его в подсистему «**Отдел закупок**».

На вкладке «Данные» откроем список «Стандартных реквизитов», увидим, что «**Дата**» и «**Номер**» уже есть, поэтому нет необходимости их добавлять. Добавим следующие реквизиты шапки:

* + **Контрагент**: тип СправочникСсылка.Контрагенты;
  + **Договор**: тип СправочникСсылка.Договора;
  + **Сотрудник**: тип СправочникСсылка.Сотрудники;
  + **Сумма документа**: тип Число, длина 15, точность 2;

Добавим табличную часть «**Товары**» (в которой будем отображать поступление товаров), создадим реквизиты табличной части:

* + **Номенклатура**: тип СправочникСсылка.Номенклатура;
  + **Цена**: тип Число, длина 15, точность 2
  + **Количество**: тип Число, длина 15, точность 2
  + **Сумма**: тип Число, длина 15, точность 2

Откроем вкладку «Нумерация» окна редактирования документа. Нумерация отвечает за то, как будет формироваться номер документа. Как правило, рекомендуется оставлять автоматическую нумерацию с контролем уникальности. При этом, для документов можно указать периодичность. Непериодические документы будут иметь сквозную нумерацию в течение всего времени существования системы. Если установить некий период, то в течение этого периода нумерация будет уникальная, но по окончании, когда начнется новый период, нумерация начнется сначала. В организациях очень часто таким периодом является год. Настройка «Нумератор» нужна для того, чтобы организовать в системе сквозную нумерацию для документов различного вида, у каждого из которых указать один и тот же нумератор (создается отдельно). Оставим настройки по умолчанию.

Запустим систему в режиме отладки и попробуем ввести первую приходную. И практически сразу мы увидим следующие недостатки:

1. в качестве Контрагента можно выбрать как Поставщика, так и Покупателя;
2. отображается список всех договоров, а не только тех, которые заключены с выбранным Контрагентом;
3. можно выбрать приход, как Товаров, так и Услуг;
4. сумма не считается автоматически как цена, умноженная на количество;

Попробуем исправить все вышеперечисленные недостатки. При выборе Контрагента необходимо осуществлять выбор лишь тех, кто находятся в группе «**Поставщики**». Т.к. группа «**Поставщики**» предопределенная, то можно на нее ссылаться при использовании различных механизмов и написании программного кода. Для решения поставленной задачи необходимо настроить соответствующее отбор-условие и наложить его на реквизит «**Контрагент**». Для этого открываем свойства реквизита, переходим на вкладку «Представление», открываем настройку свойства «Параметры выбора» (рис. 27). Настройки, указанные в «Параметрах выбора», будут накладывать условия отбора на реквизиты элементов справочника «Контрагенты» (т.е. того справочника, на элементы которого мы ссылаемся в данном реквизите) в открывающейся форме выбора. Таким образом, мы ограничим список возможных элементов справочника, представляемых в форме выбора.

Т.к. справочник Контрагенты имеет всего 2 уровня иерархии, на первом из которых указана группа («**Покупатели**» или «**Поставщики**»), а на втором – конкретные Контрагенты, то данные группы являются «родителями» для всех контрагентов внутри них. Таким образом, мы должны наложить отбор на поле «**Родитель**» при выборе элементов из справочника «**Контрагенты**».

В открывшемся окне настроек параметров выбора в панели инструментов нажимаем кнопку для добавления отбора и в раскрывшемся списке (рис. 27a) среди всех реквизитов справочника «**Контрагенты**» выбираем реквизит «**Родитель**».

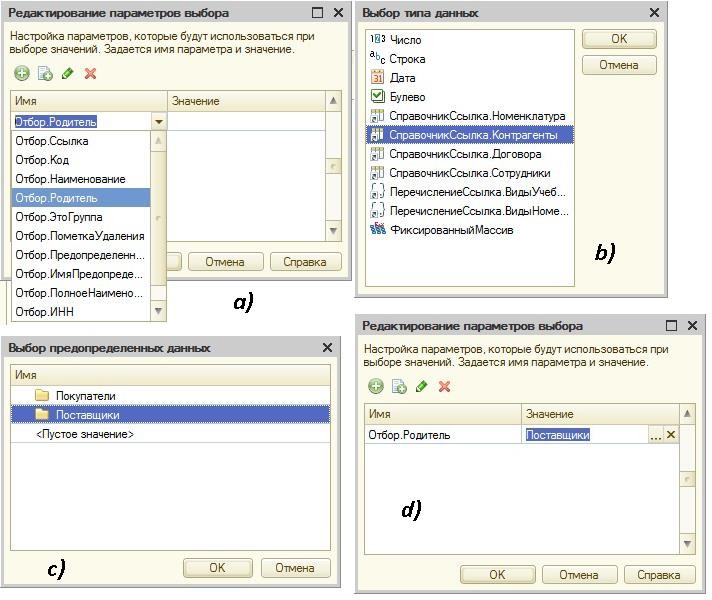


Рис. 27. Настройка «Параметров выбора»

В столбце «Значение» при помощи необходимо выбрать тип значения, которое будет накладываться как условие отбора (рис. 27b). Т.к. значение содержится в самом справочнике «Контрагенты», то указываем СправочникСсылка.Контрагенты. Затем для выбора конкретного значения нажимаем , и из предопределенных данных (рис. 27c) выбираем группу «Поставщики». После выбора требуемых данных отбор выглядит так, как показано на рис. 27d. При необходимости можно наложить и другие условия отбора.



Для ограничения формы выбора «**Договора**» элементами справочника, связанными с выбранным Контрагентом, необходимо реквизиты «**Контрагент**» и «**Договор**» связать между собой. Для связи реквизитов между собой в пределах одного объекта используется настройка «Связи параметров выбора», расположенная на вкладке «Представление» (рис. 28). Выполним такую настройку для реквизита «**Договор**», т.к. на его форму выбора должны быть наложены условия отбора. В окне «Связей параметров выбора» слева указываются другие реквизиты текущего объекта (рис. 28a), от значений которых будет зависеть форма выбора настраиваемого реквизита. В нашем случае это реквизит

«**Контрагент**»: выбираем и переносим его в правую часть окна.

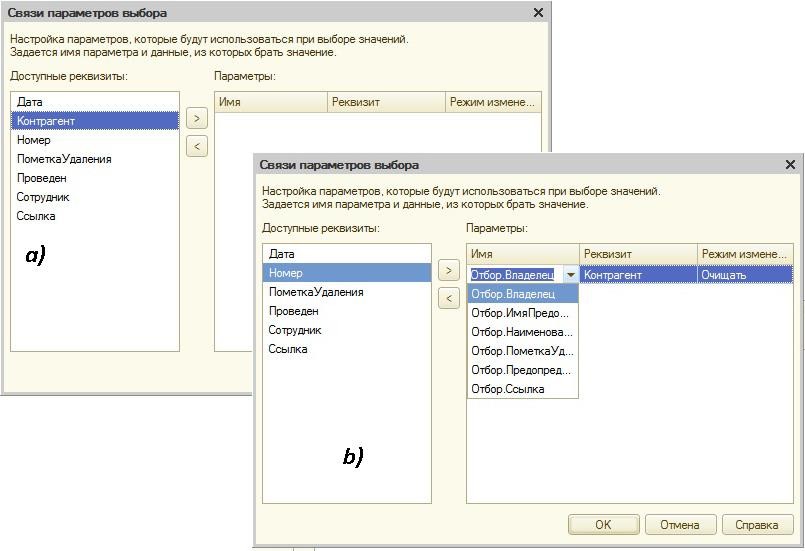


Рис. 28. Настройка «Связи параметров выбора»

Для данного реквизита по умолчанию указывается отбор по владельцу (рис. 28b). Это говорит о том, что в форме выбора настраиваемого реквизита («**Договор**») будут отображены лишь те элементы, для которых поле «**Владелец**» заполнено значением, указанным в выбранном реквизите («**Контрагент**»). В нашем случае это действительно так, и мы оставляем данную настройку отбора. Однако среди различных вариантов может быть указан отбор и по другим полям настраиваемого реквизита (рис. 28b). Дополнительным параметром при настройке отбора является «Режим изменения» связанного реквизита (рис. 28b). Режим «Очищать» говорит о том, что при изменении связанного реквизита («**Контрагент**») автоматически будет очищаться настраиваемый реквизит («**Договор**»). В большинстве случаев так и требуется, тем не менее, данную настройку при необходимости можно и изменить.

Для настройки выбора только Товаров, а не Услуг, в качестве значений реквизита «**Номенклатура**» табличной части «**Товары**», следует воспользоваться настройкой «Параметров выбора» (рис. 27). Настраиваемым реквизитом при этом будет, как нетрудно догадаться, «**Вид номенклатуры**». Тип значения, соответственно, будет ПеречислениеСсылка.ВидыНоменклатуры, откуда в качестве значения выбирается «**Товар**».

### §5.2. Конструктор управляемой формы. Модуль формы

Таким образом, первые три проблемы из обозначенного списка были решены с использованием стандартных механизмов платформы. Однако настройку автоматического расчета суммы таким же образом не сделать. Система просто не может знать алгоритм, который бы брал данные из двух, трех и т.д. ячеек табличной части документа, что-то с ними делал и помещал бы в третью ячейку: таких алгоритмов может быть сколь угодно много и всех их предусмотреть просто невозможно. Поэтому необходимо систему

«научить» нужному алгоритму с использованием встроенного языка 1С.

Мы заполняем табличную часть в режиме исполнения: выбираем номенклатуру, вводим цену, количество. Т.е. все действия, которые выполняются – выполняются интерактивно. Соответственно, обработку необходимо применить к интерактивным действиям пользователя, т.е. соответствующий программный код должен располагаться в модуле формы. Поэтому в окне редактирования объекта переходим на вкладку «Формы» (рис. 29).

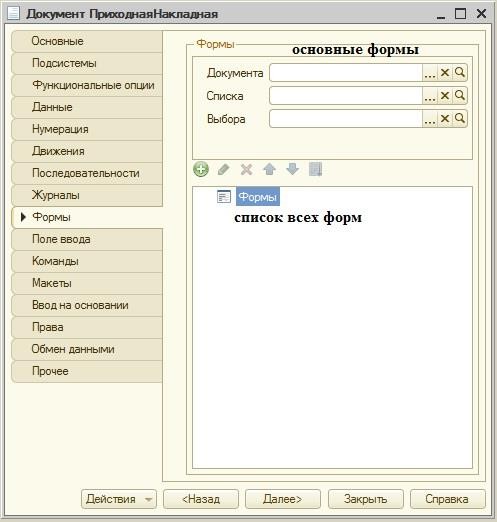


Рис. 34. Настройка форм объекта

В верхней части окна «Формы» располагается перечень возможных типов форм (элемента, списка и т.д.) для объекта данного вида. Для каждого типа назначается форма, являющаяся основной (рис. 29). Программист может создать сколько угодно большое число форм одного типа, отличающихся функциональностью. Список всех форм располагается ниже панели инструментов (рис. 29). Для объекта, из всех форм определенного типа, только одна является основной. Основная форма – форма, загружаемая по умолчанию, при интерактивном обращении к ней в режиме исполнения. Все остальные формы могут загружаться программно в результате работы различных алгоритмов. Для создания основной формы необходимо выбрать требуемый тип формы и нажать (рис. 29), либо выбрать нужную форму из списка всех форм при помощи .



При создании/редактировании экземпляра документа открывается форма элемента (форма документа). Сейчас система ее создает автоматически, нам же данный механизм необходимо изменить. Нажимаем для создания основной формы документа, в результате запускается специальный мастер. Оставляем все параметры по умолчанию. По окончании работы мастера запускается конструктор формы документа (рис. 30).

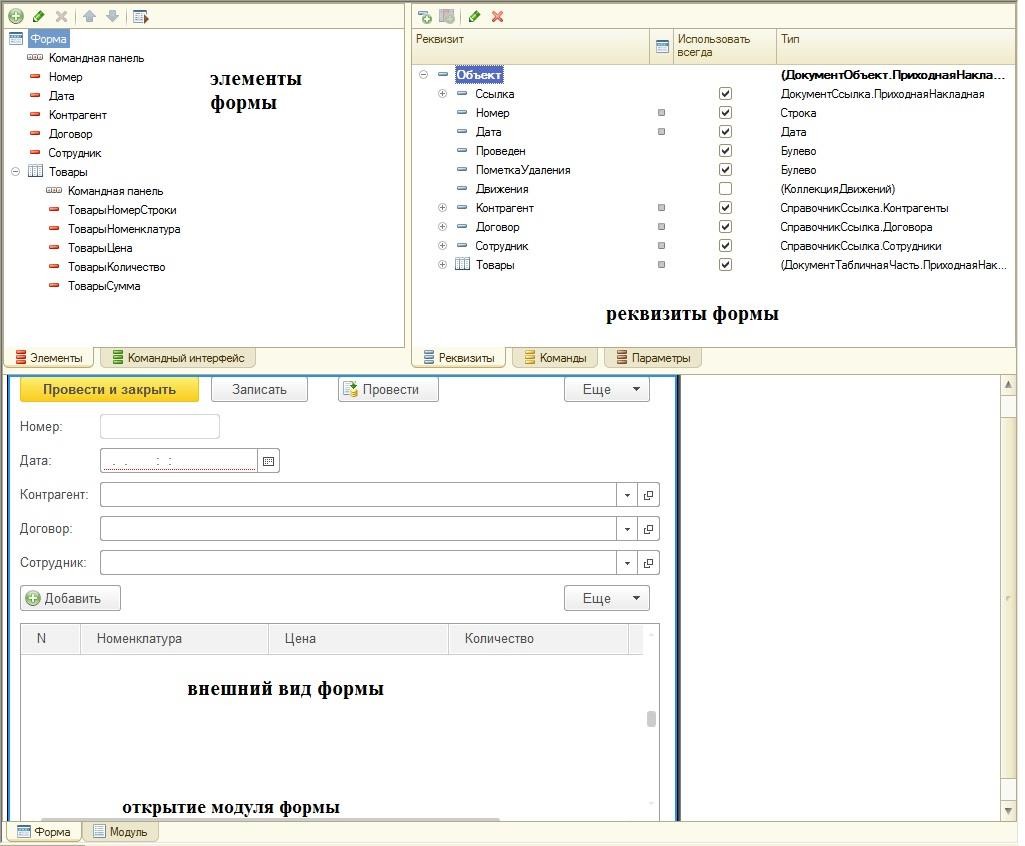


Рис. 30. Конструктор формы

В верхней правой части конструктора формы (рис. 30) располагаются элементы, отвечающие за данные формы. Сюда относятся, в первую очередь, реквизиты формы, команды формы, которые могут быть расположены в виде кнопок или гиперссылок. Это те данные, которые, которые отвечают за функциональность формы. Также здесь можно добавить параметры формы, которые либо будут передаваться в другие открываемые формы, либо использоваться в различных программных алгоритмах.

В левой верхней части конструктора (рис. 30) располагается перечень интерактивных элементов – т.е. тех элементов, которые будут расположены на форме, и которые будут доступны для редактирования, использования конечным пользователем. Здесь производится настройка внешнего интерфейса: как элементы будут располагаться, как группироваться и т.д. Интерактивные элементы, выполняющие какие-либо действия, а не являющиеся просто декорациями, связаны с реквизитами и командами формы (через специальное свойство «ПутьКДанным»). Также здесь настраивается командный интерфейс: панель навигации для перехода к определенным разделам приложения и командная панель.

Внизу (рис. 30) отображается внешний вид – как создаваемая форма будет выглядеть в режиме исполнения. Также имеется закладка для перехода к модулю формы – это то место, где располагается программный код, описание обработчиков событий. В модуле формы описываются как некоторые служебные (вспомогательные) процедуры и функции, так и обработчики, связанные с элементами формы. Каждый элемент формы (и сама форма в целом) имеет ряд событий, которые могут возникать в результате работы системы. Для просмотра списка всех событий необходимо открыть свойства конкретного элемента в дереве элементов (см. рис. 30), и затем перейти к вкладке «События» (рис. 31).

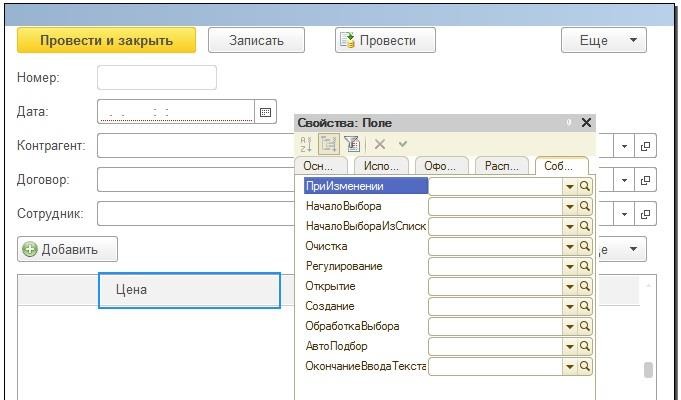


Рис. 31. События элементов формы

Каждое событие интерактивного элемента формы может быть связано лишь с одной процедурой – обработчиком события. Имя процедуры может быть произвольным. Сами процедуры-обработчики описываются в модуле формы. Для того, чтобы связать событие с процедурой, которая станет обработчиком данного события, используется выпадающий список , в котором из списка всех процедур указывается та, которую следует назначить в качестве обработчика. Для автоматического создания процедуры-обработчика определенного события и связи события и его обработчика, можно использовать кнопку около требуемого события элемента. Следует обратить внимание, что часть событий выполняется на стороне клиента, а часть – на стороне сервере. Поэтому в модуле формы любую процедуру, функцию (не только обработчики событий) необходимо предварять директивой, указывающей, где происходит выполнение программного кода.



Для реализации автоматического расчета Суммы, необходимо для элемента «**Цена**» описать обработчик события «**При изменении**». Система задает вопрос, какие процедуры необходимо создать:

* обработчик на клиенте;
* обработчик на клиенте и безконтекстную процедуру на сервере (для случая, когда необходимо получать какие-то данные с сервера);
* обработчик на клиенте и контекстную серверную процедуры (аналогично предыдущему случаю, но передается весь контекст формы);

Для нас будет достаточно обработчика на клиенте т.к. никакие данные с сервера (из базы данных нам не потребуются). В результате создается шаблон процедуры- обработчика.

Теперь необходимо определить, как обратиться к данным, которые сейчас находятся в табличной части документа? Для этого воспользуемся отладкой: введем в процедуре заглушку ***A=1;*** и установим в этой строке точку останова. Запустим систему в режиме отладки и попытаемся создать один документ «**Приходная накладная**».

Добавим строку табличной части и введем в поле «**Цена**», допустим, «**5**». Система перебрасывает нас в модуль объекта в точку останова. Используем инструмент

«Вычислить выражение». Мы можем попытаться обратиться к значениям реквизитов или элементов формы.

Все реквизиты хранятся в основном реквизите – «**Объект**». В поле «Вычислить» вводим «Объект», в результате получаем значения реквизитов (рис. 32).

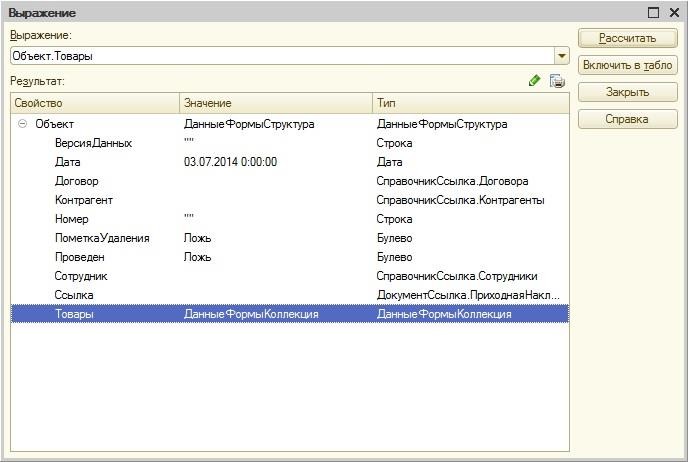


Рис. 32. Содержимое реквизитов документа

Однако тем самым мы не получили желаемого: мы не видим строку табличной части с ценой. Известно, что табличная часть называется «Товары», поэтому находим ее среди всех полученных реквизитов (рис. 32). Данный реквизит имеет тип ДанныеФормыКоллекция. Коллекцию значений мы можем посмотреть в отдельном окне при помощи кнопки , в результате чего открывается табличная часть «Товары» (рис. 33).

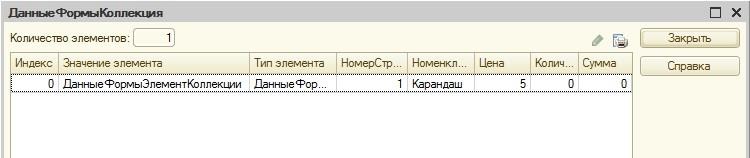


Рис. 33. Просмотр табличной части в отдельном окне

Здесь мы видим наш товар с заданной ценой, но опять – нет ничего, чтобы указывало на то, как к этому полю обратиться. Пробуем просмотреть соответствующую запись также в отдельном окне (рис. 34).

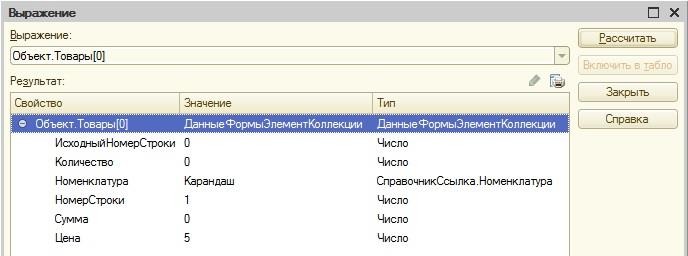


Рис. 34. Просмотр выделенной строки табличной части в отдельном окне

В результате срабатывает инструмент «Вычислить выражение» для конкретной строки табличной части: **Объект.Товары[0]**. Данный вариант невозможен к использованию, т.к. строк может быть великое множество, и нам будет неизвестно, в какой конкретно строке мы находимся в данный момент. Таким образом, поиск решения среди реквизитов объекта не дал желаемого результата.

Для обращения к элементам формы следует обратиться к коллекции «Элементы». В результате мы получим окно, содержащее гораздо больше элементов, доступных для просмотра (рис. 35).

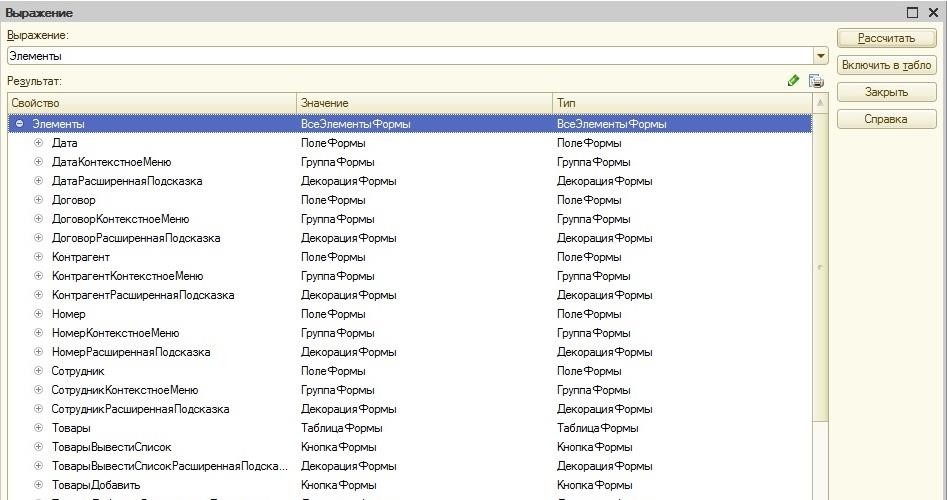


Рис. 35. Список интерактивных элементов, доступных для просмотра при отладке

Это на самом деле ожидаемо т.к. элементов пользовательского интерфейса достаточно много, настроек и параметров и характеристик – тоже. Главная проблема – как среди всего этого множества отыскать то, что требуется именно нам.

В первую очередь поиск необходимо начинать по основному наименованию элемента. Табличная часть называется «**Товары**», поэтому в первую очередь и ищем – **Элементы.Товары**. Раскроем найденную группу «**Товары**». Табличная часть имеет тип – ДанныеФормыКоллекция (см. рис. 32), а конкретная строка табличной части – ДанныеФормыЭлементКоллекции. Среди перечня всех элементов найдем элемент соответствующего типа. Это элемент под названием **ТекущиеДанные**. Раскрыв данный элемент, мы увидим данные в строке, в которой мы в данный момент находимся. Более того, выделив поле «**Цена**», в поле «Выражение» инструмента «Вычислить выражение» мы увидим полный путь к этому полю: **Элементы.Товары.ТекущиеДанные.Цена**, что нам и требовалось получить. Аналогичным образом, осуществляется доступ к содержимому полей «**Количество**» и «**Сумма**».

Закрываем отладку приложения, возвращаемся к редактированию модуля формы документа. В созданной процедуре-обработчике убираем точку останова и текст- заглушку. Пишем следующий программный код:

|  |  |
| --- | --- |
| СтрокаРасчета = Элементы.Товары.ТекущиеДанные; СтрокаРасчета.Сумма = СтрокаРасчета.Цена\*СтрокаРасчета.Количество; | Л.7 |

Первая строчка содержит в себе строку-ссылку на текущие данные (введена для большей читабельности кода). Вторая строка производит требуемые действия с элементами, входящими в указанную строку-ссылку.

Аналогичным образом создадим обработчик события «ПриИзменении» для поля

«Количество» и добавим в него точно такие же строчки программного кода. Запустим систему в режиме отладки и попробуем добавить приходную накладную теперь. В конце нажимаем кнопку «Записать» (без проведения) после чего закрываем текущую приходную.

Создадим документ «**Расходная накладная**» копированием. В системе «1С: Предприятие» очень сильно развит инструмент создания объектов (прикладных и подчиненных) «на основании» других. Для того чтобы просто создать точную копию объекта, необходимо в дереве конфигурации выделить требуемый объект и нажать кнопку

«Добавить копированием» , либо использовать функциональную клавишу F9. После этого необходимо изменить имя прикладного объекта, и выполнить требуемую настройку реквизитов и поведения объекта. Реквизиты шапки, табличные части и их реквизиты объекта также можно добавлять копированием.



Также имеется возможность создания на основе прикладного объекта одного вида, прикладного объекта другой сущности (не стоит путать с инструментом «Ввод на основании», далее в пособии). К примеру, из объекта типа «Справочник» создать объект типа «Документ». Для этого необходимо в дереве конфигурации выделить исходный объект и перетащить его на класс объектов, экземпляр которого необходимо создать. При этом скопируются только те настройки, реквизиты, табличные части (и их реквизиты) которые могут существовать у объектов различных классов и имеют одинаковый тип данных.

В нашем примере мы не будем создавать копированием объекты различных сущностей. Выделим документ «**Приходная накладная**» и при помощи «Добавить копированием» создадим новый документ «**Расходная накладная**». Включим документ в подсистему «**Отдел продаж**». Для реквизита «**Контрагент**» изменим настройку

«Параметры выбора»: теперь мы должны выбирать контрагентов из списка

«**Покупателей**». Для реквизита «**Номенклатура**» табличной части «**Товары**» сбросим настройку «Параметры выбора» – мы продаем не только товары, но и оказываем услуги, и все это делаем в рамках одного документа «Расходная накладная».

После того, как мы внесли все необходимые изменения, можно запустить систему в режиме отладки и посмотреть на результат. Если вдруг оказалось, что при создании документа копированием, система сохранила условия отбора, накладываемые на формы выбора, как у исходного документа (к примеру, можно выбрать только контрагентов из списка поставщиков) – тогда необходимо удалить форму документа «**Расходная накладная**» и просто заново ее создать с описанием требуемого функционала.

Обратим внимание на следующее: в модуле формы документа «**Приходная накладная**» для события ПриИзменении полей «**Цена**» и «**Количество**» был прописан одинаковый программный код. Создав копированием документ «**Расходная накладная**», данный код также оказался дважды написан в модуле документа. Таким образом, одни и те же действия несколько раз были описаны в различных частях конфигурации. Это не совсем рационально. Правильнее этот программный код описать в одном единственном месте (в общем модуле), а в остальных частях конфигурации делать вызов соответствующей процедуры/функции.

Создадим общий модуль «**Работа с документами**», который будет выполняться исключительно на клиенте (установите соответствующую галочку в настройках модуля). Добавим в модуль следующую процедуру:

|  |  |
| --- | --- |
| Процедура ПересчитатьСумму(СтрокаРасчета) Экспорт СтрокаРасчета.Сумма = СтрокаРасчета.Цена\*СтрокаРасчета.Количество;  КонецПроцедуры | Л.8 |

СтрокаРасчета – параметр процедуры, передаваемый ей в точке вызова. В модуле объекта для документов «**Приходная накладная**», «**Расходная накладная**» для имеющихся обработчиков событий изменим программный код на следующий:

|  |  |
| --- | --- |
| СтрокаРасчета = Элементы.Товары.ТекущиеДанные; РаботаСДокументами.ПересчитатьСумму(СтрокаРасчета); | Л.9 |

В первой строке кода мы получаем текущие данные, для которых нужен обработчик. Во второй строке мы вызываем процедуру из общего неглобального модуля, и передаем эти данные через параметр процедуры.

***Подсказка***: При написании программного кода обратите внимание на то, что система автоматически выдает список конструкций, которые могут быть. Также при описании прикладных объектов, при нажатии на точку система показывает список возможных реквизитов объекта, из которых можно выбрать. Вы всегда можете ввести первые символы конструкции (или прикладного объекта, реквизита и т.д.), которую хотите использовать, после чего нажать комбинацию клавиш «Ctrl+Пробел». В результате система выполнить синтаксический поиск по первым введенным символам и выведет список возможных методов, событий, объектов, реквизитов и т.д., которые можно использовать; вам будет достаточно в появившемся списке выбрать подходящий вариант. Это позволяет, во-первых, разрабатывать код более быстро и эффективно, а, во-вторых, снижает вероятность возникновения ошибок из-за опечаток при наборе. Однако это будет действовать не всегда: например, вы присвоили переменной имя объекта. Не факт, что система всегда сможет определить реквизиты данной переменной, как и объекта; у исходного объекта она обязана это делать, но если было переприсвоение, то это не совсем обязательно. В данном случае вам необходимо выполнять контроль корректности ввода данных самостоятельно.

### §5.3. Журнал документов. Команды формы.

Иногда бывают ситуации, когда пользователю необходимо ряд документов различного вида представить в едином списке в хронологическом порядке. В нашем случае – это перечень всех документов «**Приходная накладная**» и «**Расходная накладная**». Для таких целей используется объект «Журнал документов». Создадим новый журнал «**Приходно-расходные операции**», включим в подсистему «**Общий отдел**».

На закладке «Данные» указываем перечень всех документов, которые необходимо отобразить в журнале: укажем оба наших документа (рис. 36). По умолчанию, если больше ничего не настраивать, то при загрузке журнала система динамически сформирует список, который будет в хронологическом порядке содержать дату, номер и тип документа. Если необходимо вывести какую-то дополнительную информацию, то для этого следует воспользоваться добавлением граф на вкладке «Данные» (рис. 36).

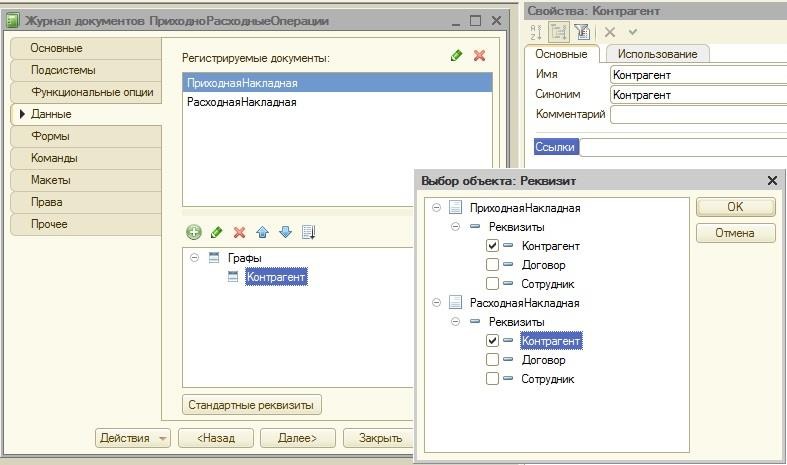


Рис. 36. Журнал документов

При добавлении очередной графы указывается ее имя и синоним, а в свойстве

«Ссылки» – реквизиты, которые будут выбираться из документов каждого вида, входящих в журнал (рис. 36). Допустимо указание только реквизитов шапки, реквизиты табличных частей недоступны для выбора. Дополнительно следует обратить внимание, что тип графы журнала – составной. Это означает, что из различных документов в данной графе можно указать реквизиты различных типов, если это необходимо. В нашем случае создадим графы «**Контрагент**» и «**Сотрудник**», и укажем для каждого из документов соответствующие реквизиты. Запустим систему и посмотрим результат.

Далее предположим, что в журнале необходимо отображать также сумму документа. Рассмотрим решение на примере документа «**Приходная накладная**». Для этого создадим еще один реквизит шапки в документе: «**Сумма документа**», тип Число, длина 15, точность 2. По той причине, что форма документа уже была ранее создана, данный реквизит автоматически на форме не появится (проверьте это, запустив систему в режиме исполнения). Для добавления вновь созданного реквизита, необходимо отрыть форму документа, в разделе «Данные формы» найти основной реквизит «**Объект**», в нем

«**Сумму документа**», и перетащить его к «Элементам формы» (расположим ниже табличной части «Товары», но не в ней).

Далее указанный реквизит необходимо заполнить нужным значением. Создадим специальную кнопку – команду формы, которая будет рассчитывать сумму документа и заносить ее в указанный реквизит. В области «Данные формы» откроем вкладку

«Команды», далее «Команды формы». Добавим новую команду «**Рассчитать сумму**» и перетащим ее к элементам формы, разместив в одну строку с реквизитом «**Сумма документа**». По умолчанию, система все элементы формы автоматически выстраивает друг под другом. Для того чтобы изменить такое расположение, элементы необходимо группировать. Для группировки в «Элементы формы» можно добавить следующие группы:



* Обычная группа – для простой группировки элементов;
* Страницы – для создания группы, которая будет содержать несколько закладок, в каждой из которых может быть несколько элементов;
* Страница – отдельная закладка, создается внутри группы типа «Страницы»;
* Командная панель формы – для создания команд на форме на специальной панели;

Так создадим обычную группу и переместим в нее элемент «**Сумма документа**» и кнопку «**Рассчитать сумму**». В панели свойств созданной группы можно настроить положение элементов, отображение заголовков и т.д. Для настройки положения элементов используется свойство «**Группировка**». Аналогичным образом расположим элементы формы «**Дата**» и «**Номер**» в одну строку, дополнительно изменив их заголовки. В результате всех преобразований форма должна выглядеть, как показано на рис. 37.

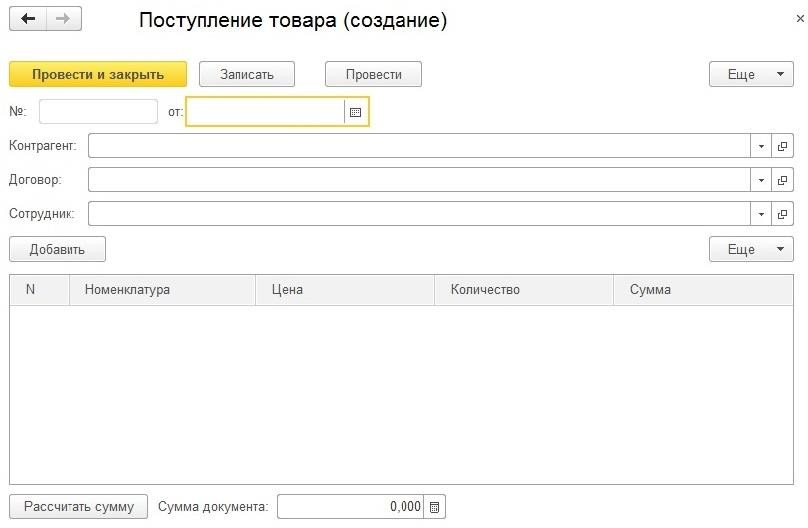


Рис. 37. Измененная форма документа «**Приходная накладная**»

Затем для кнопки необходимо описать механизм заполнения суммы. Для этого выбирается созданная команда формы, и в панели свойств открывается «Действие» для данной команды (на клиенте). В открывшуюся процедуру добавляем следующий код:

|  |  |
| --- | --- |
| Объект.СуммаДокумента = Объект.Товары.Итог("Сумма"); | Л.10 |

В данном программном коде применяется функция ***Итог()***, которая производит суммирование числовых значений в указанном столбце таблицы (столбец «**Сумма**»). Результат записывается в нужный реквизит. Проверьте механизм работы на практике.

### Темы для самостоятельного изучения:

1. Конструктор ввода документов «на основании».
2. Нумераторы документов.
3. Последовательности документов. Восстановление последовательности.
4. Макеты и печатные формы документов.