

1С:Предприятие 8

«1С:Предприятие 8» включает в себя платформу и прикладные решения (конфигурации).

Встроенный язык

Встроенный язык позволяет настроить прикладное решение под свои нужды или разработать свою собственную конфигурацию «с нуля».

Средства встроенного языка 1С:Предприятия 8 дают возможность управлять практически всеми аспектами поведения системы, работать с прикладными объектами, например, справочниками и документами, формировать печатные формы отчетов и выполнять другие самые разнообразные действия.

Прикладное решение использует механизмы 1С:Предприятия 8 и работает только под управлением платформы, оно не может быть использовано самостоятельно, как отдельное приложение.

Формат описания элементов языка

В синтаксических диаграммах используются следующие символы:

Символ Значение

[] В квадратных скобках заключаются необязательные синтаксические элементы.

() Круглые скобки заключают в себе список параметров.

| Вертикальной линией разделяются синтаксические элементы, среди которых нужно выбрать только один.

Контекст выполнения программного модуля

Глобальный контекст образуется:

- значениями свойств и методов глобального контекста;
- функциями встроенного языка и языковыми конструкциями;
- переменными, процедурами и функциями глобального программного модуля и общих модулей, объявленными с ключевым словом **Экспорт**.

Глобальный контекст виден всем программным модулям и определяет общую языковую среду конфигурации.

Локальный контекст

Локальный контекст модуля образуется тем конкретным местом конфигурации задачи, для которого использован программный модуль. Локальный контекст виден только конкретному программному модулю и определяет для модуля набор непосредственно доступных модулю объектов, их свойств и методов.

Виды программных модулей

Общие модули

Глобальный модуль

Модули прикладных объектов

Модули форм

Структура программного модуля

Структуру программного модуля можно подразделить на следующие разделы:

- раздел определения переменных;
- раздел процедур и функций;
- раздел основной программы.

В конкретном программном модуле любой из разделов может отсутствовать.

Элементы встроенного языка

Комментарии

A=B; // Это - комментарий // Это тоже комментарий

В общем случае формат оператора языка следующий:

-метка: Оператор[(параметры)] [ДобключевоеСлово];

В качестве меток используются специальные идентификаторы, начинающиеся с символа '~' (тильда) и состоящие из последовательности букв, цифр и символов '_' (подчеркивание). Чтобы пометить оператор, надо поместить перед ним метку и следующий за ней символ ":".

Элементы встроенного языка

Имена переменных, процедур и функций

Именем переменной, объявленной процедуры или функции, может быть любая последовательность букв, цифр и знаков подчеркивания "_", начинающаяся с буквы или знака подчеркивания "_". Вновь создаваемые имена не должны совпадать с зарезервированными словами языка или именами свойств, непосредственно доступных в текущем контексте. Распознавание имен переменных, процедур и функций ведется без учета регистра букв. Регистр букв (строчные или заглавные) при написании имен переменных, свойств, методов, процедур, функций, а также функций встроенного языка не имеет значения.

Зарезервированные слова

Если; Тогда; ИначеЕсли; Иначе; КонецЕсли; Для;
Каждого; Из; По; Пока; Цикл; КонецЦикла Процедура;
Функция; КонецПроцедуры; КонецФункции; Перемен;
Перейти; Возврат; Продолжить; Прервать; И; Или; Не;
Попытка; Исключение;
ВызватьИсключение; КонецПопытки;
Новый; Выполнить

Специальные символы, используемые в ИСХОДНОМ ТЕКСТЕ

// Двумя знаками «косая черта» начинается комментарий.

| Вертикальная черта в начале строки используется только в строковых константах и означает, что данная строка является продолжением предыдущей (перенос строки).

~ Знаком тильда начинается метка оператора. : Двоеточием заканчивается метка оператора.

;Точка с запятой является символом разделения операторов.

()В круглые скобки заключается список параметров методов, процедур, функций и конструкторов. Также они используются в выражениях встроенного языка.

[]С помощью оператора «квадратные скобки» производится обращение к свойствам объекта по строковому представлению имени свойства.

,Запятая разделяет параметры в списке параметров методов, процедур, функций и конструкторов.

"" В двойные кавычки заключаются строковые литералы. В одинарные кавычки заключаются литералы даты.

Специальные символы, используемые в ИСХОДНОМ ТЕКСТЕ

- .Десятичная точка в числовых литералах. Разделитель, используемый для обращения к свойствам и методам объектов встроенного языка.
- +Символ «плюс» обозначает операцию сложения и конкатенации.
- Символ «минус» обозначает операцию вычитания.
- *Символ «звездочка» обозначает операцию умножения.
- / Символ «косая черта» обозначает операцию деления.
- > Правая угловая скобка обозначает логическую операцию «больше».
- >= Логическая операция «больше или равно».
- < Левая угловая скобка обозначает логическую операцию «меньше».
- <= Логическая операция «меньше или равно».
- = Знак равенства обозначает присвоение или логическую операцию «равно».
- <> Две угловые скобки обозначают логическую операцию «не равно».

Объявление переменных

- *//явное объявление переменной модуля или процедуры или функции*
- **Перем** Итог;
- *//экспортируемая переменная модуля*
- **Перем** Количество Экспорт;
- *//неявное объявление переменных (при первом присвоении значения)*
- ном = 0;
- **Приведение типов**
 - • Число(),
 - • Строка(),
 - • Дата(),
 - • Булево().

Типы данных

Во встроенном языке 1С:Предприятия 8.1 поддерживаются следующие основные типы данных:

- число
- строка
- дата (включает в себя время)
- булево (имеет два значения: Истина и Ложь)
- значение Неопределено
- значение Null (для неуказанных значений в таблицах базы данных)
- Тип (значения специального типа «Тип» нужны для представления и сравнения типов данных, например, «Если ТипЗнч(ВыбДок) = Тип("ДокументСсылка.Чек") Тогда...»).
- другие типы, перечисленные в документации

Конструкции встроеного языка

Конструкция «Если ... Тогда»

- //1-й пример
- Если Состав.Количество() $=$ 0 Тогда
- Предупреждение("Не указано ни одной строки!");
- КонецЕсли;
-
- //2-й пример
- Если Сотр.Категория = Перечисления Категории.Совместитель Тогда
- //обрабатываем как совместителя
- Иначе
- //обычный сотрудник
- КонецЕсли;
-
- //3-й пример
- Если выбор = 1 Тогда
- Сообщить("Выбран 1-й пункт");
- ИначеЕсли выбор = 2 или выбор = 3 Тогда
- Сообщить("Выбран 2-й или 3-й пункт");
- Иначе
- Сообщить("Обработка этого пункта не предусмотрена");
- КонецЕсли;

Конструкции встроенного языка

Конструкция «?(..., ..., ...)»

- Коментарий = ? (Количество=0, "Пусто", Строка (Количество)+" шт . ");

Конструкции встроенного языка

Конструкция «Для ... По ... Цикл»

- Для M = 1 По 12 Цикл
- Сообщить(M);
- КонецЦикла;

Конструкции встроенного языка

Конструкция «Для Каждого ... Из ... Цикл»

ПустаяСсылка = Справочники.Номенклатура.

ПустаяСсылка();

Для Каждого СтрокаТЗ Из тзСостав Цикл

//тело цикла

Если СтрокаТЗ.Товар = ПустаяСсылка Тогда

Продолжить; *//возьмем следующую строку*

Иначе

Если СтрокаТЗ.Товар = ВыбТовар Тогда

Прервать; *//досрочный выход из цикла*

КонецЕсли;

КонецЕсли;

КонецЦикла;

Конструкции встроеного языка

Конструкция «Пока ... Цикл»

- » $N = 1$;
- » Выборка = Справочники.Номенклатура.Выбрать();
- » Пока Выборка.Следующий() Цикл
- » Если Выборка.ПометкаУдаления Тогда
- » Продолжить; *//возьмем следующий элемент*
- » КонецЕсли;
- » $N = N + 1$;
- » Если $N > 10$ Тогда
- » Прервать; *//досрочный выход из цикла*
- » КонецЕсли;
- » Сообщить("Товар: " + Выборка.Наименование);
- » КонецЦикла;

Конструкции встроенного языка

Безусловные переходы

- » $N = 0$;
- » Выборка = Справочники.Номенклатура.Выбрать();
- » Пока Выборка.Следующий() Цикл
- » $N = N + 1$;
- » Если $N \leq 10$ Тогда
- » *//пропускаем первые 10 элементов справочника*
- » Перейти ~след;
- » КонецЕсли;
- » Сообщить(Выборка.Наименование);
- » ~след:
- » *//возьмем следующий элемент*
- » КонецЦикла;

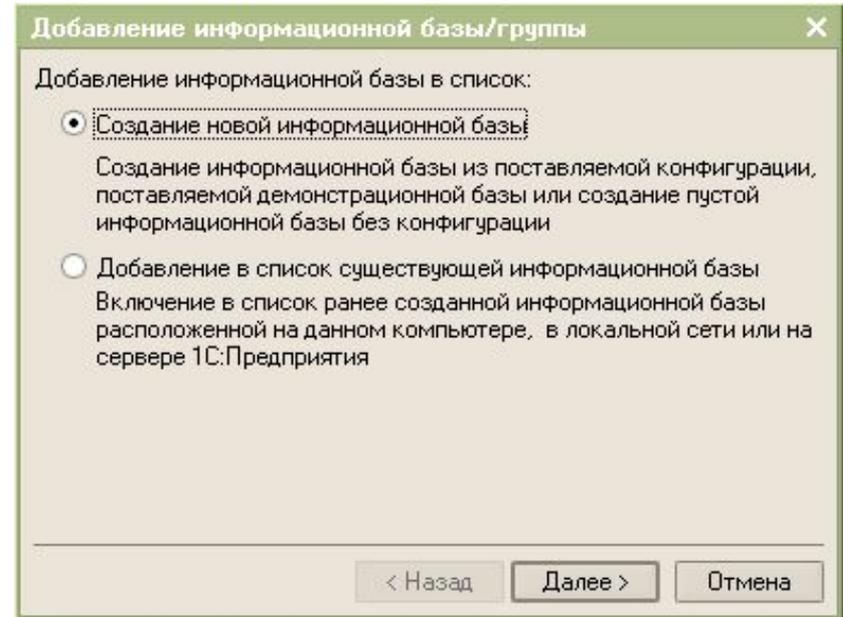
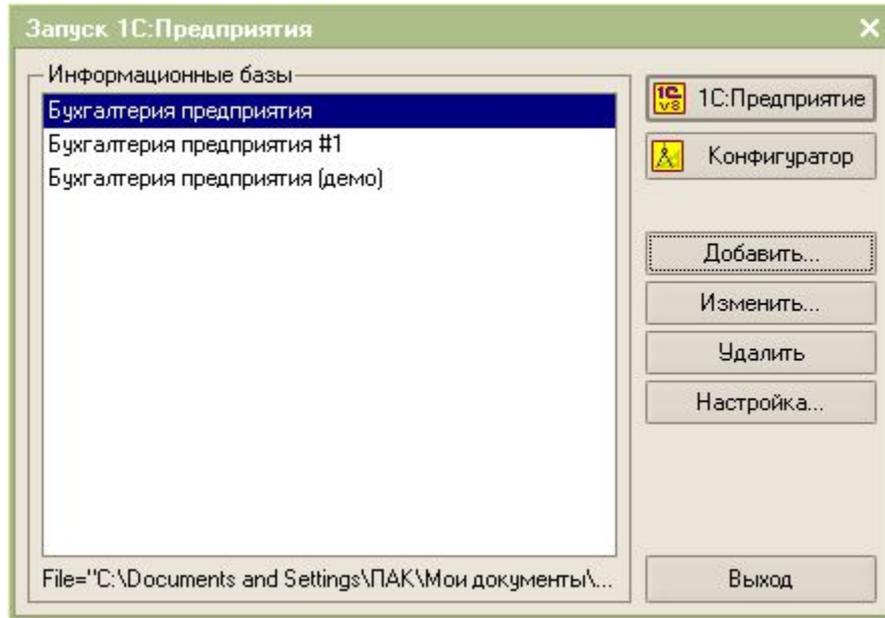
Процедуры и функции

- » Процедура Печать(ТабДок)
 - » Макет = ПолучитьМакет("Основной");
 - » ТабДок.Вывести(Макет.ПолучитьОбласть("Бланк"));
 - » **КонецПроцедуры**
-
- Функция ЕстьПравоИспользования(ИмяОтчета)
 - *//возвращаемое значение функции*
 - Возврат ПравоДоступа("Использование", Отчеты[ИмяОтчета]);
 - **КонецФункции**

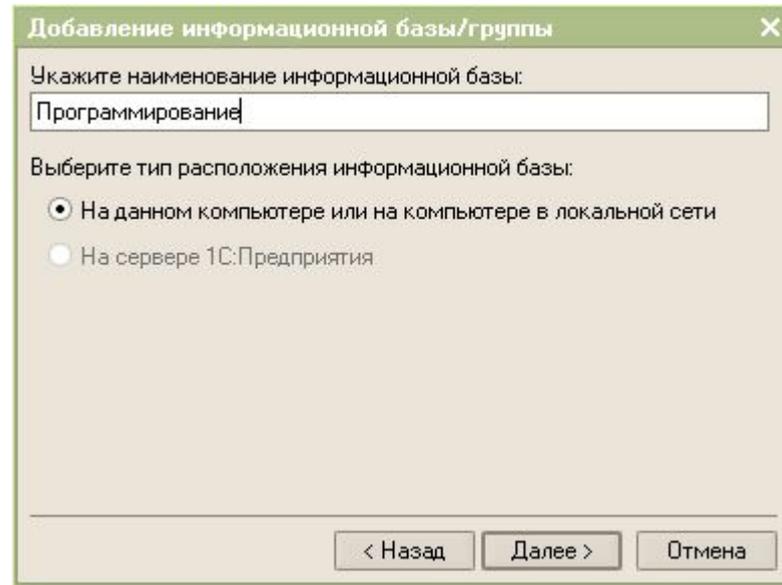
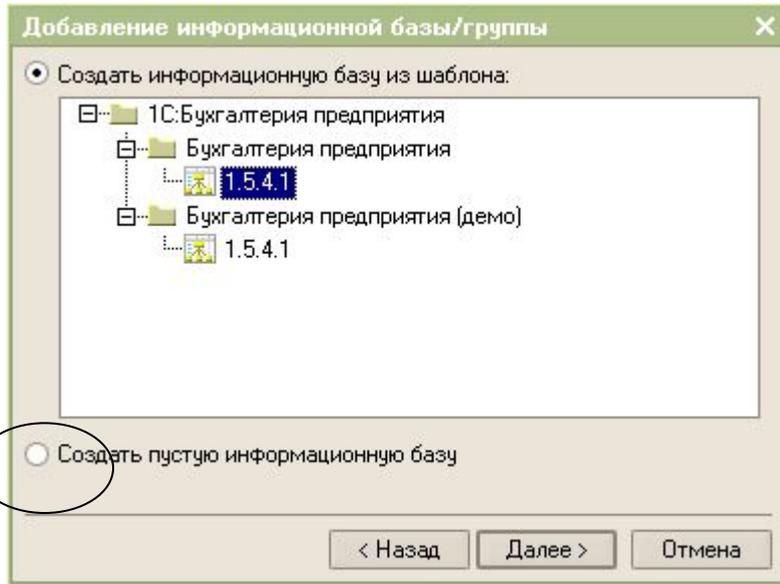
Редактор программных модулей

- » Ctrl+F7 можно произвести проверку модуля на наличие синтаксических ошибок
- » Ctrl-F1 синтаксис-помощник
- » Ctrl-Пробел – контекстная подсказка

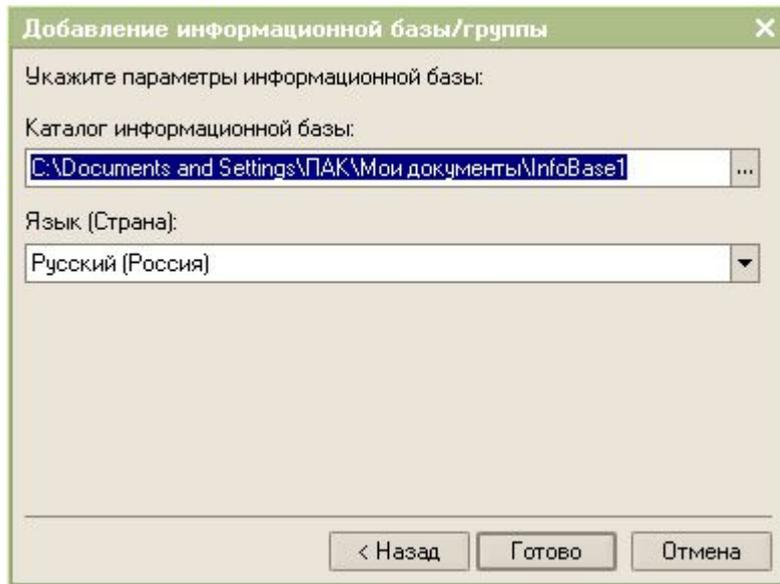
Создание новой ИБ



Создание новой ИБ



Создание новой ИБ



Добавление информационной базы/группы

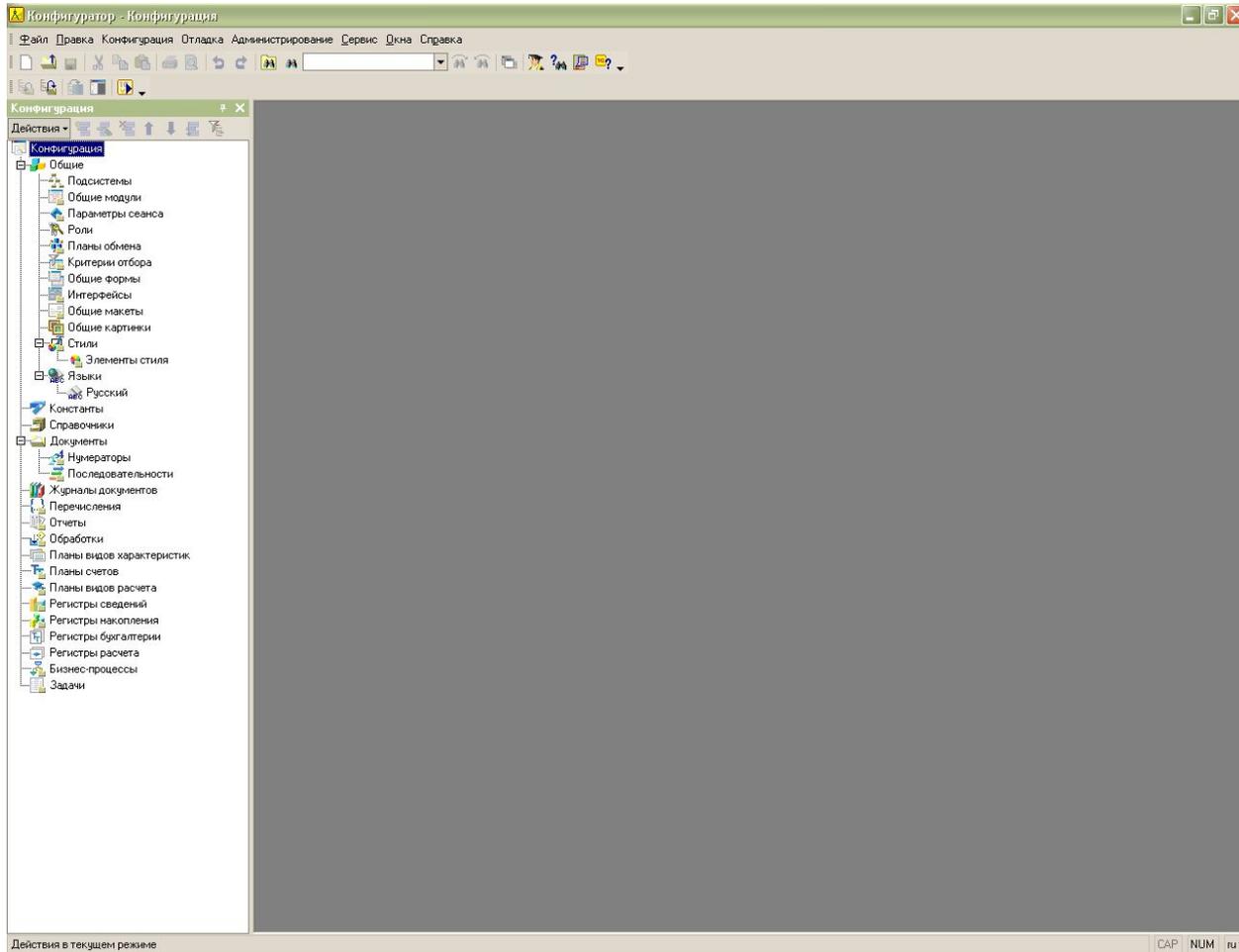
Укажите параметры информационной базы:

Каталог информационной базы:
C:\Documents and Settings\ПАК\Мои документы\InfoBase1 ...

Язык (Страна):
Русский (Россия) ▼

< Назад Готово Отмена

Создание новой ИБ



Демонстрационный пример

- **Справочники** предназначены для хранения условно-постоянной информации. Они используются в других объектах, например, документах, регистрах и отчетах. В нашем примере будут следующие справочники:
 - Номенклатура. Содержит список товаров.
 - Контрагенты. Содержит список клиентов и поставщиков, а также всех внешних организаций.
 - Склады. Содержит список складов (мест хранения) организации.
 - **Документы** используются для регистрации событий, происходящих в жизни предприятия. Документы сохраняются в информационной базе, имеют дату и время, а также номер. В нашей конфигурации будут созданы следующие документы:
 - Приходная Накладная. Осуществляет оприходование товара на склад, добавляет запись о приходе товара в регистр накопления Остатки.
 - Расходная Накладная. Регистрирует реализацию товара, уменьшает количество в регистре Остатки и добавляет запись в регистр Продажи.
 - В **регистрах** накопления хранится информация об остатках или оборотах, связанных с объектами предметной области. Нам понадобятся следующие регистры накопления:
 - Остатки. Хранит записи о движении товаров на складе, предоставляет информацию об остатке товаров на каждом складе.
 - Продажи. Хранит записи о продажах, предоставляет информацию об объеме продаж каждого товара в разрезе контрагентов.
 - **Отчеты** предназначены для отображения на экране и печати итоговых и детальных данных. Отчеты не хранят никаких данных в информационной базе, они лишь извлекают информацию из других объектов, обрабатывают ее и выводят результаты в выходную форму. В нашей простой конфигурации мы создадим всего два отчета:
 - Остатки. Показывает остатки товаров на выбранном складе. Данные будут браться из регистра накопления Остатки.
 - Анализ Продаж. Показывает объем продаж заданной группы товаров. Для получения данных будет использован регистр накопления Продажи.

Создание справочников

Справочник Номенклатура

Имя: Номенклатура

Синоним: Номенклатура

Комментарий: Номенклатура

Действия <Назад Далее> Закрывать Справка

Справочник Номенклатура

Иерархический справочник

Вид иерархии: Иерархия групп и элементов

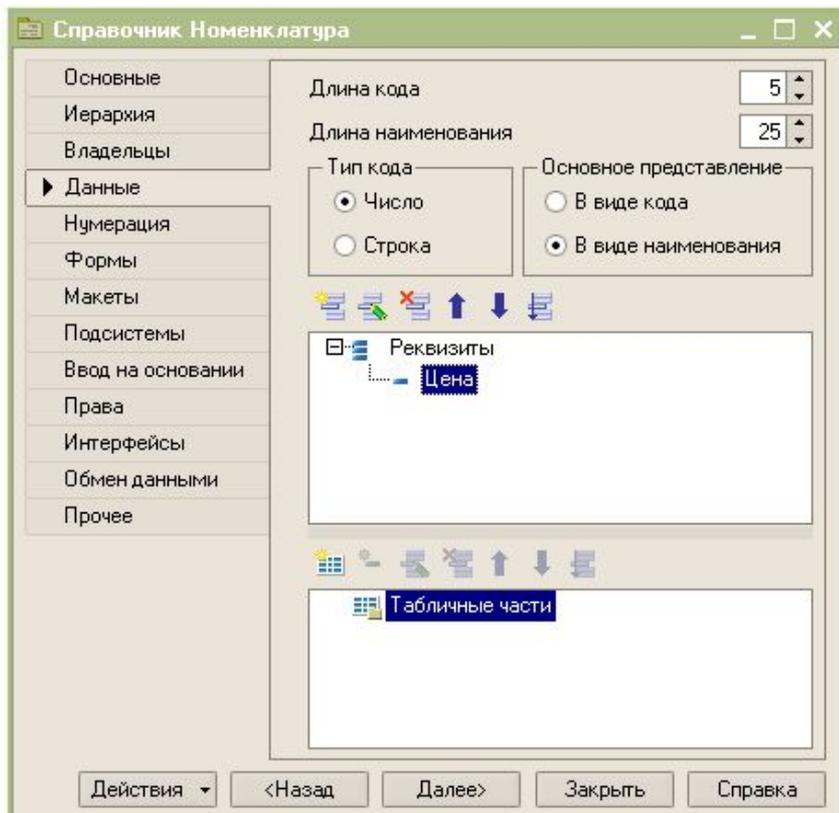
Размещать группы сверху

Ограничение количества уровней иерархии

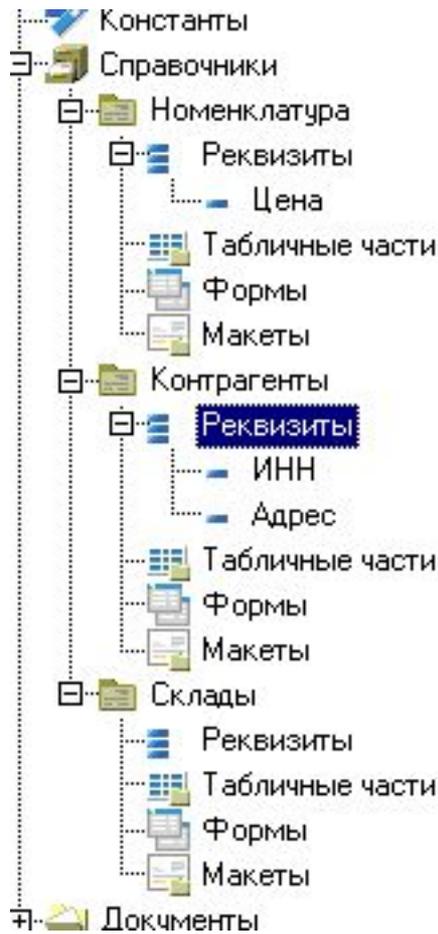
Количество уровней иерархии 2

Действия <Назад Далее> Закрывать Справка

Создание справочников



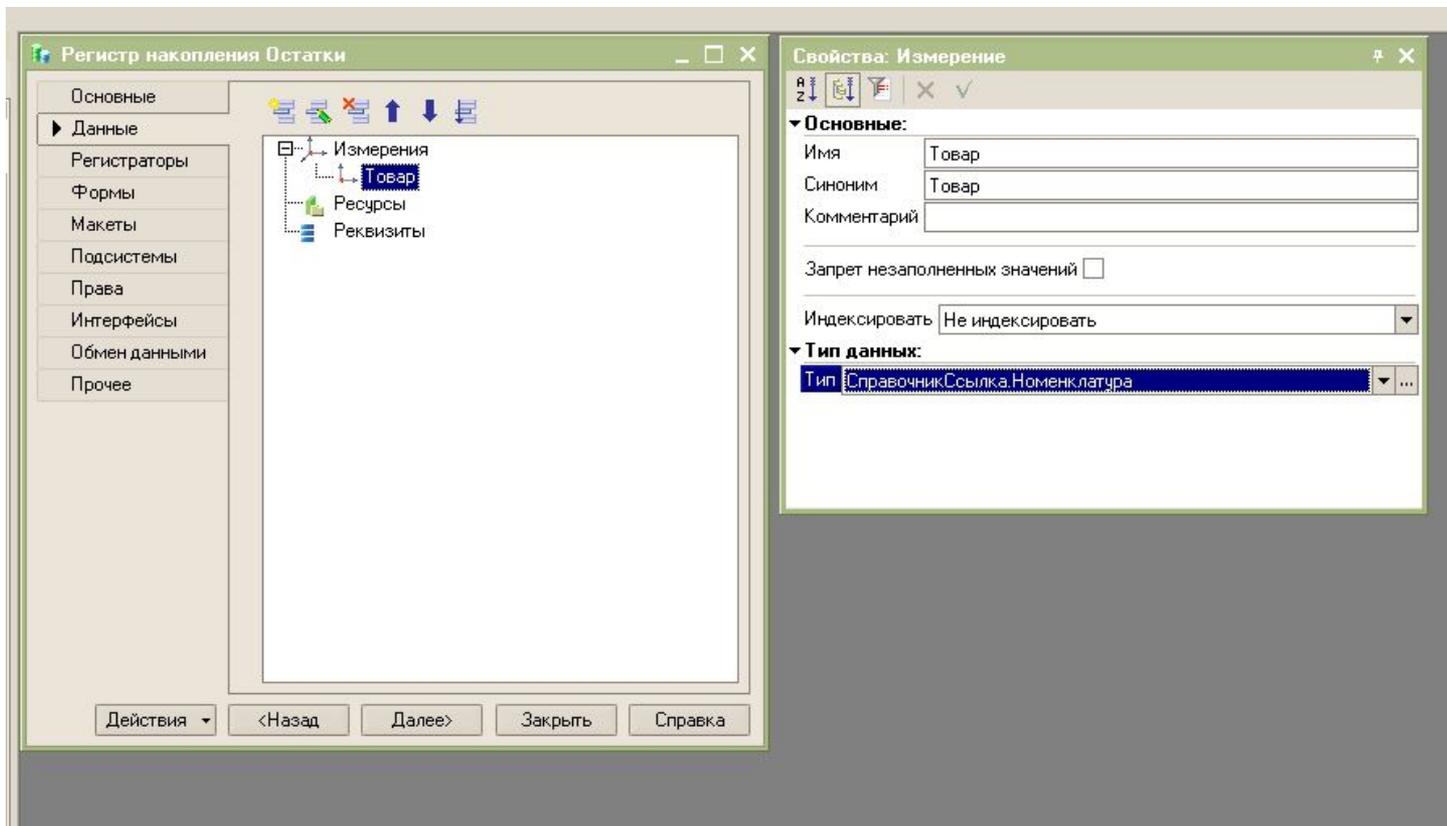
Создание справочников



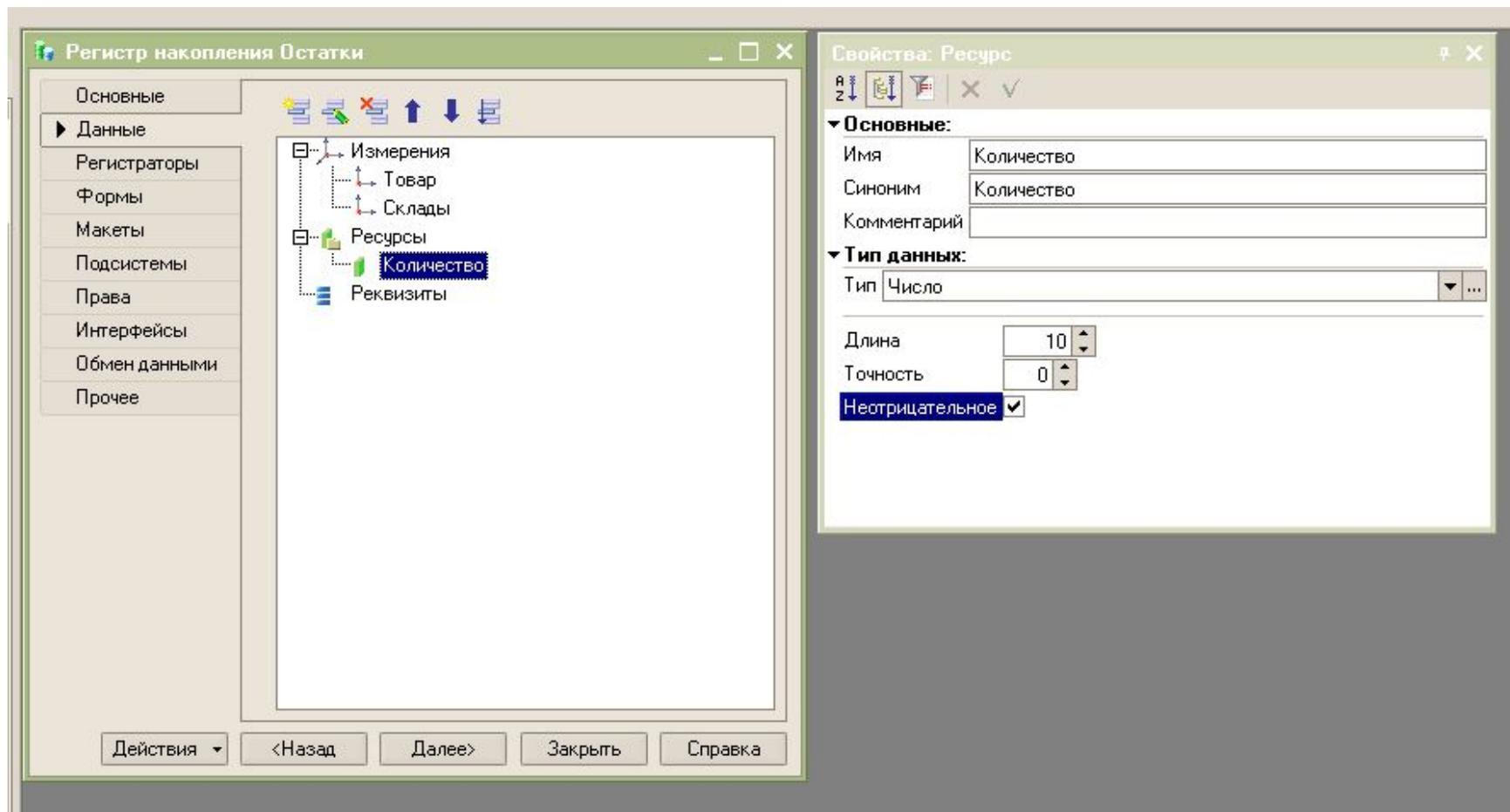
В справочнике Контрагенты, вдобавок к стандартным реквизитам Код и Наименование, мы заведем реквизиты ИНН и Адрес. У справочника Склады реквизитов создавать не нужно, нам будет достаточно встроенных реквизитов Код и Наименование.

Создание регистров

Создадим регистр Остатки, который будет хранить остатки товаров на каждом складе. Для решения этой задачи он должен иметь два измерения: Товар и Склад, и один ресурс Количество.



Создание регистров



Создание регистров

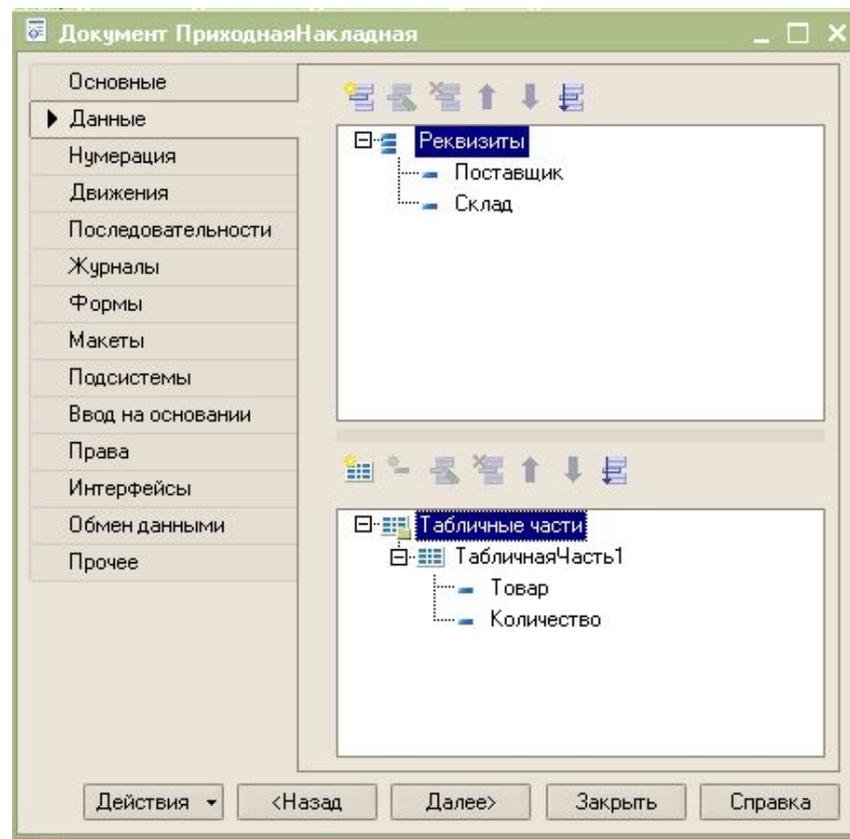
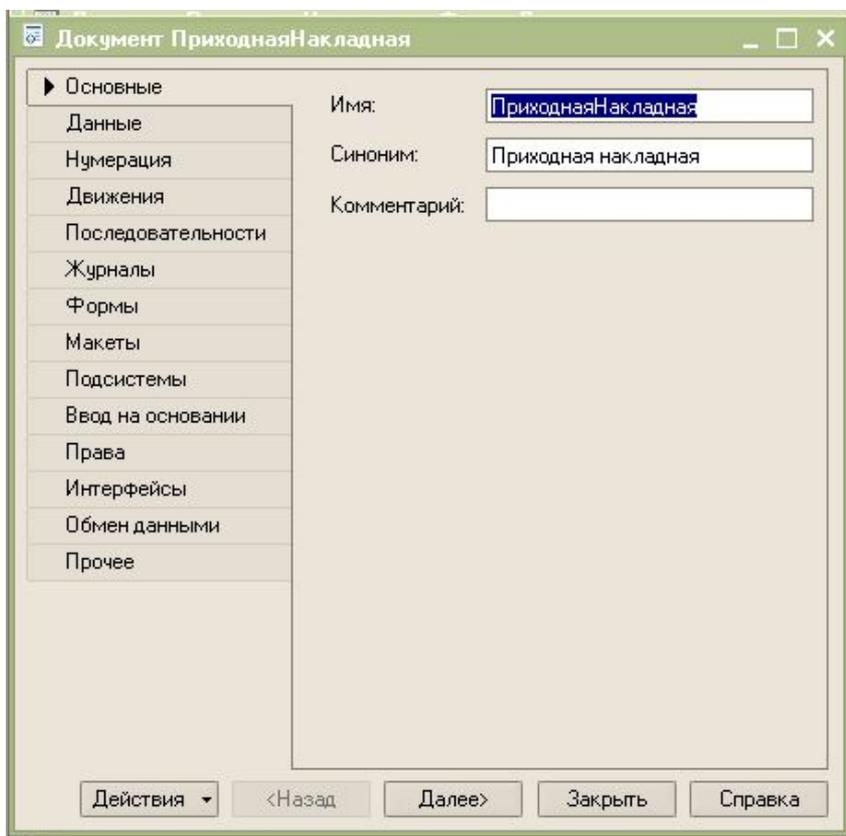
Регистр Продажи создается таким же способом В нем будут два измерения Контрагент и Товар, и один ресурс Сумма. Регистр будет хранить объем продаж за период в разрезе контрагентов и товаров, т е это будет *регистр оборотов*.

The screenshot displays the configuration of a 'Регистр накопления Продажи' (Sales Accumulation Register). The interface is divided into several panes:

- Left Pane (Tree View):** Shows a hierarchical structure of accounting elements. Under 'Регистры накопления' (Accumulation Registers), the 'Продажи' (Sales) folder is expanded, showing 'Измерения' (Dimensions) with 'Контрагент' (Counterparty) and 'Товар' (Goods), 'Ресурсы' (Resources) with 'Сумма' (Sum) selected, and 'Реквизиты' (Attributes), 'Формы' (Forms), and 'Макеты' (Templates).
- Main Window:** Titled 'Регистр накопления Продажи', it shows a tree view of the register's structure. Under 'Измерения' (Dimensions), 'Контрагент' and 'Товар' are listed. Under 'Ресурсы' (Resources), 'Сумма' is selected and highlighted.
- Right Pane (Properties):** A dialog titled 'Свойства: Ресурс' (Resource Properties) is open. It shows the following configuration for the 'Сумма' resource:
 - Имя (Name):** Сумма
 - Синоним (Synonym):** Сумма
 - Комментарий (Comment):** (empty)
 - Тип данных (Data Type):** Число (Number)
 - Длина (Length):** 12
 - Точность (Precision):** 2
 - Неотрицательное (Non-negative):**

Создание документов

У любого документа в системе 1С Предприятие 8.0 есть встроенные реквизиты Дата и Номер. Мы создадим дополнительные реквизиты Поставщик и Склад, а также табличную часть Товары с реквизитами Товар и Количество.



Создание документов

В окне редактирования документа на закладке «Движения» отметьте регистры, по которым он может делать движения. Для приходной накладной нужно отметить регистр *Остатки*, а для расходной накладной — регистры *Остатки* и *Продажи*.

Далее создадим алгоритм проведения приходной накладной, при котором в регистре *Остатки* будет увеличиваться количество товаров. Воспользуемся для этого конструктором движений.

The image shows two windows from a software application. The left window is titled "Документ Приходная Накладная" and has a sidebar with various options. The "Движения" (Movements) tab is active, showing a tree view of "Регистры накопления" (Accumulation registers) with "Остатки" (Residuals) checked. Below this is a "Конструктор движений" (Movement constructor) button and a list containing "Остатки".

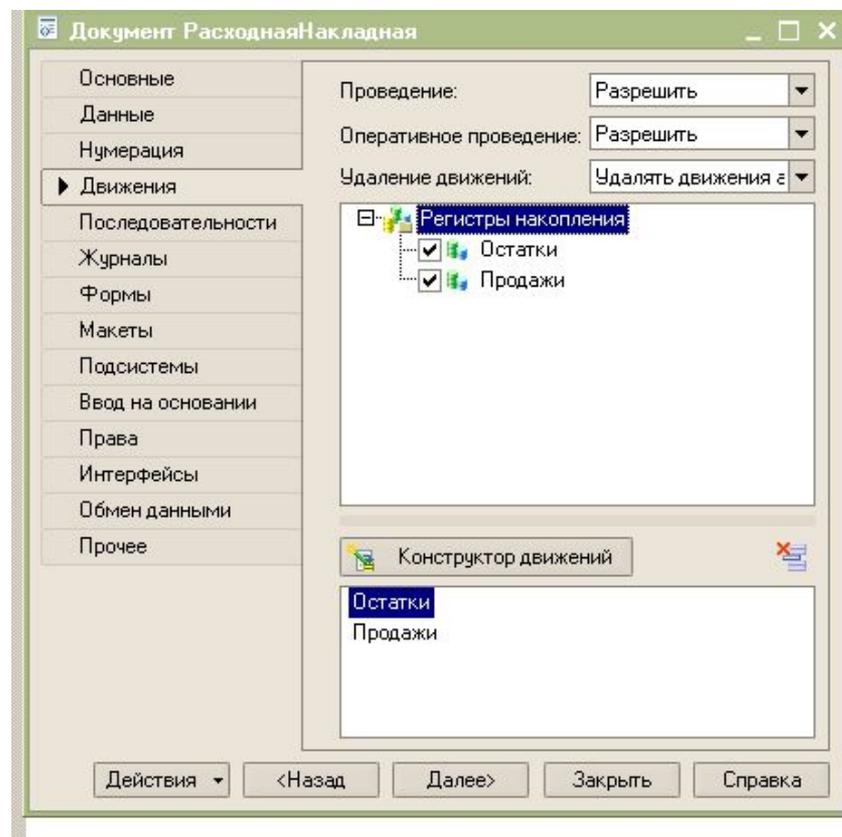
The right window is titled "Конструктор движения регистров" (Register movement constructor). It shows a list of registers with "РегистрНакопления.Остатки" selected. The "Тип движения регистра" (Register movement type) is set to "Приход" (Receipt). The "Табличная часть" (Table part) is set to "ТабличнаяЧасть1". Below this is a table with the following data:

Поле	Выражение
Товар	ТекСтрокаТабличнаяЧасть1.Товар
Склады	Склад
Количество	ТекСтрокаТабличнаяЧасть1.Количество

At the bottom of the right window, there are buttons for "Назад" (Back), "Далее" (Next), "Заполнить выражения" (Fill expressions), "Очистить выражения" (Clear expressions), "OK", "Отмена" (Cancel), and "Справка" (Help).

Создание документов

Документ РасходнаяНакладная создается аналогичным образом, за исключением того, что он делает движения типа «расход» по регистру Остатки и добавляет записи в регистр оборотов Продажи.



Создание документов

Движение документа Расходная Накладная

Конструктор движения регистров

Регистры

- РегистрНакопления.Остатки
- РегистрНакопления.Продажи

Тип движения регистра: Приход Расход

Реквизиты документа

- Дата
- Склад
- Покупатель
- ТекСтрокаТабличнаяЧасть1.Товар
- ТекСтрокаТабличнаяЧасть1.Количество
- ТекСтрокаТабличнаяЧасть1.Цена
- ТекСтрокаТабличнаяЧасть1.Сумма
- ТекСтрокаТабличнаяЧасть1.Всего

Табличная часть: ТабличнаяЧасть1

Поле	Выражение
Товар	ТекСтрокаТабличнаяЧасть1.Товар
Склады	Склад
Количество	ТекСтрокаТабличнаяЧасть1.Количество

<Назад Далее>

Заполнить выражения

Очистить выражения

OK

Отмена

Справка

Создание документов

Движение документа РасходнаяНакладная

Конструктор движения регистров

Регистры

- РегистрНакопления.Остатки
- РегистрНакопления.Продажи**

Реквизиты документа

- Дата**
- Склад
- Покупатель
- ТекСтрокаТабличнаяЧасть1.Товар
- ТекСтрокаТабличнаяЧасть1.Количество
- ТекСтрокаТабличнаяЧасть1.Цена
- ТекСтрокаТабличнаяЧасть1.Сумма
- ТекСтрокаТабличнаяЧасть1.Всего

Табличная часть: ТабличнаяЧасть1

Поле	Выражение
Контрагент	Покупатель
Товар	ТекСтрокаТабличнаяЧасть1.Товар
Сумма	ТекСтрокаТабличнаяЧасть1.Всего

<Назад Далее>

Заполнить выражения

Очистить выражения

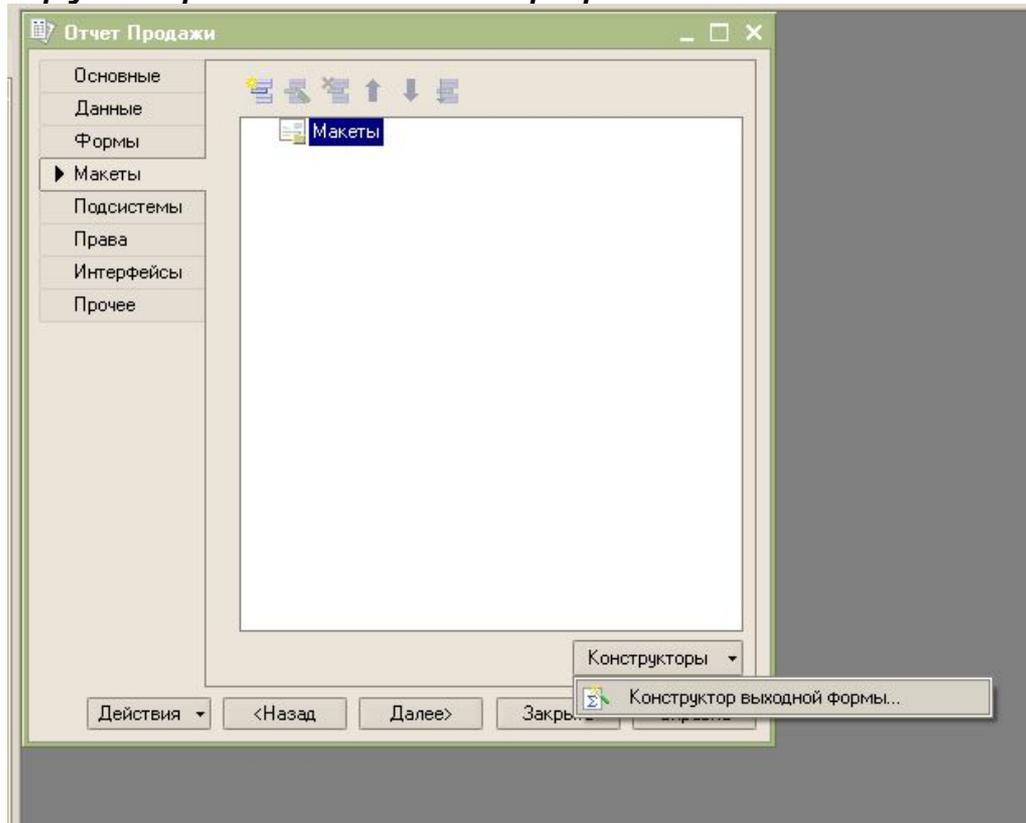
OK

Отмена

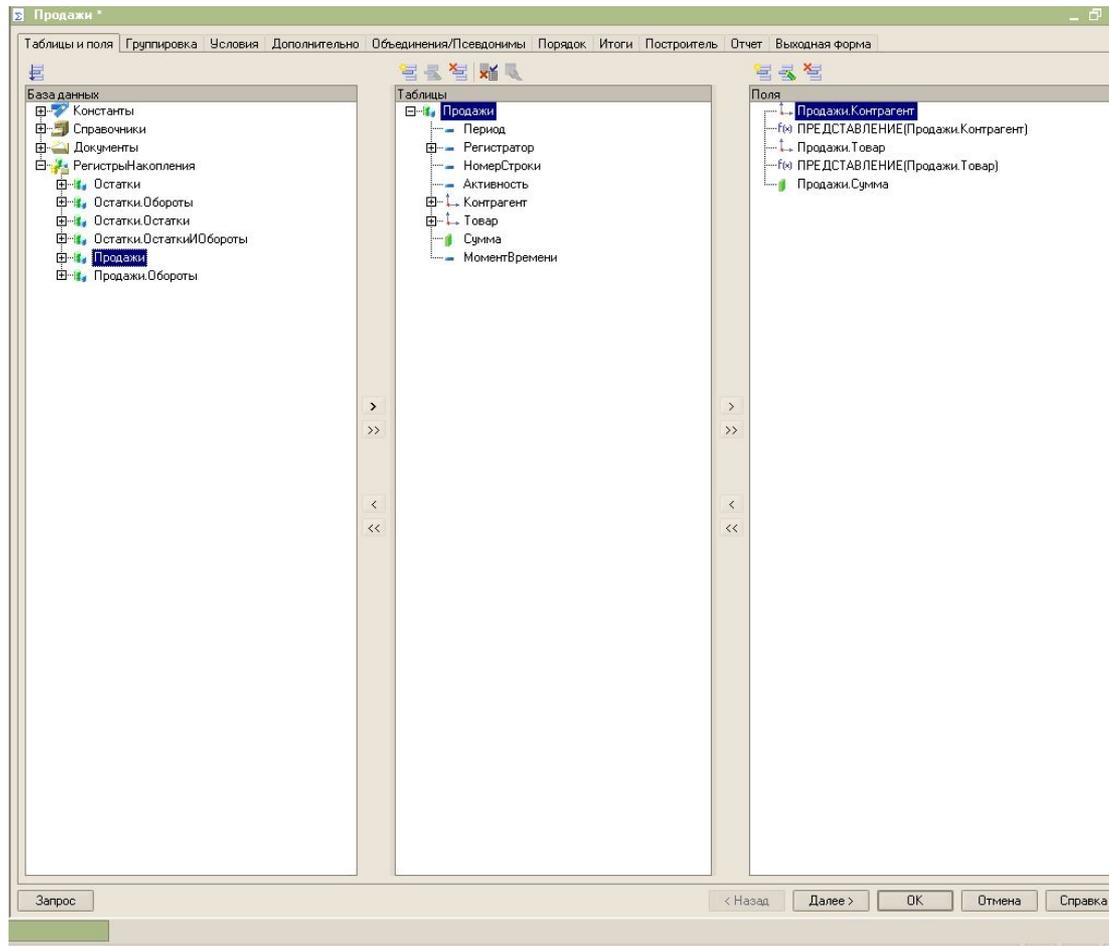
Справка

Создание отчетов

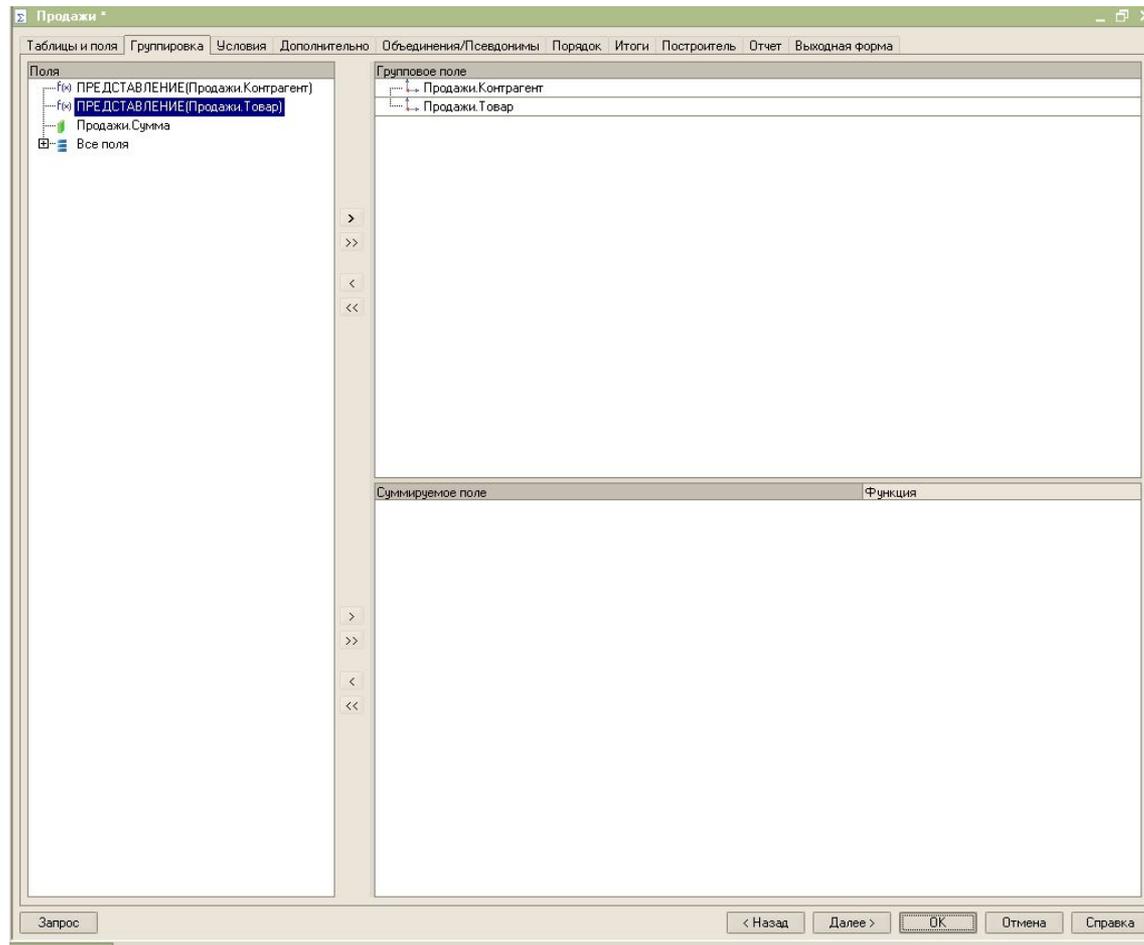
Отчеты предназначены для показа детальной и сводной информации, выбранной из базы данных. В нашем примере будет создано всего два отчета: Остатки и Продажи. Эти отчеты можно легко создан конструктором выходных форм.



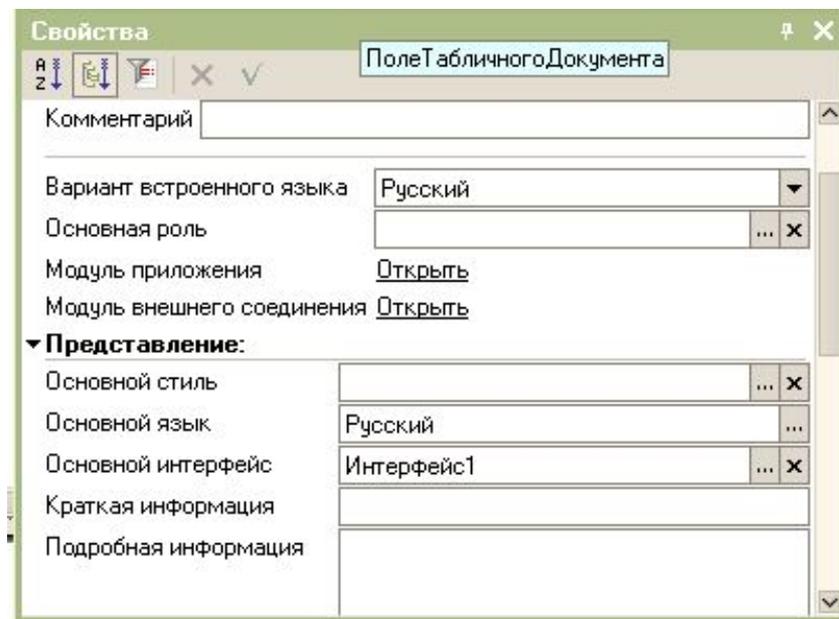
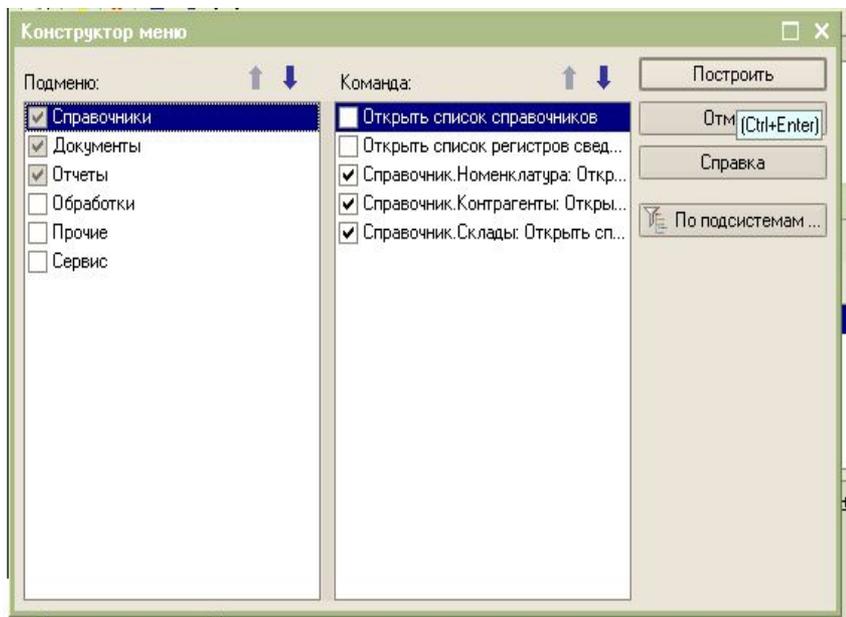
Создание отчетов



Создание отчетов



Создание интерфейса



Строки 1с

Функции работы со строками

- [-]  Функции работы со значениями типа Строка
 - ? Врег
 - ? КодСимвола
 - ? Лев
 - ? Найти
 - ? Нрег
 - ? Прав
 - ? ПустаяСтрока
 - ? Символ
 - ? СокрЛ
 - ? СокрЛП
 - ? СокрП
 - ? Сред
 - ? СтрДлина
 - ? СтрЗаменить
 - ? СтрПолучитьСтроку
 - ? **СтрЧислоВхождений**
 - ? СтрЧислоСтрок

Пример

- В поле Фамилия справочника сотрудники имеется информация о ФИО, нужно имя и отчество записать в соответствующие реквизиты и удалить из фамилии

```
□ Процедура КнопкаВыполнитьНажатие (Кнопка)  
  
Выборка = Справочники.Сотрудники.Выбрать ();  
Пока Выборка.Следующий () Цикл  
    Сот=Выборка.ПолучитьОбъект ()  
    ФИО=Трег (Сот.фамилия ());  
    флаг=Найти (ФИО, " ");  
    Сот.фамилия=Сред (Сот.фамилия, 1, флаг-1);  
    ФИО=Сред (Сот.фамилия, флаг+1);  
    флаг=Найти (ФИО, " ");  
    Сот.Имя=Сред (Сот.фамилия, 1, флаг-1);  
    Сот.Отчество=Сред (Сот.фамилия, флаг+1);  
    Сот.Записать ();  
КонецЦикла;
```

Функции работы с Датами

☐ 📅 Функции работы со значениями типа Дата

- ? Год
- ? День
- ? ДеньГода
- ? ДеньНедели
- ? ДобавитьМесяц
- ? КонецГода
- ? КонецДня
- ? КонецКвартала
- ? КонецМесяца
- ? КонецМинуты
- ? КонецНедели
- ? КонецЧаса
- ? Месяц
- ? Минута
- ? НачалоГода
- ? НачалоДня
- ? НачалоКвартала
- ? НачалоМесяца
- ? НачалоМинуты
- ? НачалоНедели
- ? НачалоЧаса
- ? Неделя Года
- ? Секунда
- ? ТекущаяДата
- ? Час

Выгрузка данных в текстовый файл

```
□ Процедура КнопкаВыполнитьНажатие (Кнопка)
Текст = Новый ТекстовыйДокумент;
Выборка = Справочники.Дисконт2009.Выбрать ();
Пока Выборка.Следующий () Цикл
    Текст.ДобавитьСтроку (Выборка.Наименование);
    Текст.ДобавитьСтроку (Выборка.Код);
    Текст.ДобавитьСтроку (Выборка.Образование);
    Текст.ДобавитьСтроку (Выборка.Профессия);
    Текст.ДобавитьСтроку (Выборка.Уровень);
    Текст.ДобавитьСтроку (Выборка.Е_mail);
    Текст.ДобавитьСтроку (Выборка.Телефон);
    Текст.ДобавитьСтроку (Выборка.ДатаР);
    Текст.ДобавитьСтроку (Выборка.ВИП);
    Текст.ДобавитьСтроку (Выборка.СНомер);
    Выборка.ПолучитьОбъект ()
КонецЦикла;

Текст.Записать ("c:\temp\text.txt");

КонецПроцедуры
```

|

Загрузка данных из текстового файла

```
ВИП = Справочники.ВидыДисконтныхКарт.НайтиПоКоду ("000000001");
Обычная= Справочники.ВидыДисконтныхКарт.НайтиПоКоду ("000000002");
Родитель=Справочники.Контрагенты.НайтиПоКоду ("000007798");
Текст.Прочитать ("c:\temp\text.txt");
Для НомерСтроки = 1 По Текст.КоличествоСтрок() Цикл
    НовыйЭлемент = Справочники.ИнформационныеКарты.СоздатьЭлемент ();
    НовыйЭлемент.Наименование = Текст.ПолучитьСтроку (НомерСтроки);
    НовыйЭлемент.ВидКарты=Перечисления.ВидыИнформационныхКарт.Штриховая;
    НовыйЭлемент.ТипКарты=Перечисления.ТипыИнформационныхКарт.Дисконтная;
    НовыйЭлемент.ТипШтрихКода=ПланыВидовХарактеристик.ТипыШтрихкодов.EAN13;
    НовыйЭлемент.КодКарты=Текст.ПолучитьСтроку (НомерСтроки+1);
    Если Число (Текст.ПолучитьСтроку (НомерСтроки+8))=1 Тогда
        НовыйЭлемент.ВидДисконтнойКарты=ВИП;
    Иначе
        НовыйЭлемент.ВидДисконтнойКарты=Обычная;
    КонецЕсли;
    НовыйЭлементК = Справочники.Контрагенты.СоздатьЭлемент ();
    НовыйЭлементК.Наименование=Текст.ПолучитьСтроку (НомерСтроки);
    НовыйЭлементК.Покупатель=Истина;
    НовыйЭлементК.Родитель=Родитель;
    НовыйЭлементК.ЮрФизЛицо=Перечисления.ЮрФизЛицо.ФизЛицо;
    НовыйЭлементК.ДополнительноеОписание=СокрЛП (Текст.ПолучитьСтроку (НомерСтроки+2))+""; "+СокрЛП (Текст.ПолучитьС
    НовыйЭлементК.ЮрФизЛицо=Перечисления.ЮрФизЛицо.ФизЛицо;
    СпКон=Справочники.КонтактныеЛица.СоздатьЭлемент ();
    СпКон.ДатаРождения=Дата (СокрЛП (Текст.ПолучитьСтроку (НомерСтроки+7)));
    СпКон.Наименование=Текст.ПолучитьСтроку (НомерСтроки);
    СпКон.Фамилия=Текст.ПолучитьСтроку (НомерСтроки);
    СпКон.Записать ();
    НовыйЭлементК.Записать ();
    СпКонК=Справочники.КонтактныеЛицаКонтрагентов.СоздатьЭлемент ();
    СпКонК.Владелец=НовыйЭлементК.Ссылка;
    СпКонК.Должность=СокрЛП (Текст.ПолучитьСтроку (НомерСтроки+3));
    СпКонК.КонтактноеЛицо=СпКон.Ссылка;
    СпКонК.Записать ();
    НовыйЭлементК.ОсновноеКонтактноеЛицо=СпКонК.Ссылка;
    НовыйЭлементК.Записать ();
    НовыйЭлемент.ВладелецКарты=НовыйЭлементК.Ссылка;
    НовыйЭлемент.Записать ();
    НомерСтроки=НомерСтроки+9;
```

Добавление информационной базы/группы



Добавление информационной базы в список:

Создание новой информационной базы

Создание информационной базы из поставляемой конфигурации, поставляемой демонстрационной базы или создание пустой информационной базы без конфигурации

Добавление в список существующей информационной базы

Включение в список ранее созданной информационной базы, расположенной на данном компьютере, в локальной сети или на сервере 1С:Предприятия

< Назад

Далее >

Отмена

Добавление информационной базы/группы



Создание информационной базы из шаблона

Выберите поставляемую конфигурацию для начала работы или демонстрационный пример для ознакомления

- [-] 1С:Управление торговлей
 - [-] Управление торговлей
 - 10.3.6.8
 - [-] Управление торговлей (демо)
 - 10.3.6.8

Создание информационной базы без конфигурации для разработки новой конфигурации или загрузки выгруженной ранее информационной базы

< Назад

Далее >

Отмена



Добавление информационной базы/группы



Укажите наименование информационной базы:

Авторизация

Выберите тип расположения информационной базы:

- На данном компьютере или на компьютере в локальной сети
- На сервере 1С:Предприятия

< Назад

Далее >

Отмена

Добавление информационной базы/группы



Укажите параметры информационной базы:

Каталог информационной базы:

C:\Documents and Settings\Admin\Мои документы\InfoBase



Язык (Страна):

русский (Россия)



< Назад

Готово

Отмена

Конфигуратор - Конфигурация

Файл Правка Конфигурация Отладка Администрирование Сервис Окна Справка

Конфигурация

Действия

Конфигурация

- Общие
- Подсистемы
- Общие модули
- Параметры сеанса
- Роли
 - Администратор
- Планы обмена
- Критерии отбора
- Подписки на события
- Регламентные задания
- Общие формы
- Интерфейсы
- Общие макеты
- Общие картинки
- Пакеты XDTO
- Web-сервисы
- WS-ссылки
- Стили
- Языки
- Константы
- Справочники
- Документы
 - Журналы документов
 - Перечисления
 - Отчеты
 - Обработки
 - Планы видов характеристик
 - Планы счетов
 - Планы видов расчета
 - Регистры сведений
 - Регистры накопления
 - Регистры бухгалтерии
 - Регистры расчета
 - Бизнес-процессы
 - Задачи

Роль Администратор: Права

Действия

Права Шаблоны ограничений

Объекты:

- Конфигурация
 - Общие
 - Константы
 - Справочники
 - Документы
 - Журналы документов
 - Отчеты
 - Обработки
 - Планы видов характеристик
 - Планы счетов
 - Планы видов расчета
 - Регистры сведений
 - Регистры накопления
 - Регистры бухгалтерии
 - Регистры расчета
 - Бизнес-процессы

Права:

- Административные функции
 - Обновление конфигурации базы данных
 - Монопольный режим
 - Активные пользователи
 - Журнал регистрации
 - Внешнее соединение
 - Automation

Ограничения доступа к данным:

Поля	Ограничение доступа
------	---------------------

Описание: Право на выполнение административных функций (Администрирование)

Устанавливать права для новых объектов

Свойства: Роль

Основные:

Имя	Администратор
Синоним	Администратор
Комментарий	

Права Открыть

Подсистемы Конфигурация

Имя объекта метаданных

Роль Администратор: Права

Для получения подсказки нажмите F1

ПУСК Конфигуратор - Кон... Microsoft PowerPoint ... RU 21:47



Конфигурация

Действия

- Конфигурация
 - Общие
 - Подсистемы
 - Общие модули
 - Параметры сеанса
 - Роли
 - Администратор
 - Планы обмена
 - Критерии отбора
 - Подписки на события
 - Регламентные задания
 - Общие формы
 - Интерфейсы
 - Полный**
 - Общие макеты
 - Общие картинки
 - Пакеты XDTO
 - Web-сервисы
 - WS-ссылки
- Стили
- Языки
- Константы
- Справочники
- Документы
 - Журналы документов
 - Перечисления
 - Отчеты
 - Обработки
 - Планы видов характеристик
 - Планы счетов
 - Планы видов расчета
 - Регистры сведений
 - Регистры накопления
 - Регистры бухгалтерии
 - Регистры расчета
 - Бизнес-процессы
- Задачи

Роль Администратор: Права

Интерфейс Полный: Интерфейс

Добавить панель Удалить панель ↑ ↓

Список панелей интерфейса:

- Главное меню

Файл **Операции** Справочники Документы Отчеты Обработки Прочие <Новая> Сервис

Окна

- ...Стандартные действия...
- <Новая>
- ...Стандартные действия...

Свойства: Элемент панели

ТипКнопки Подменю

Подменю Операции

Представление:

Текст Операции

Подсказка Операции

Пояснение Основные операции с данными

Картинка ... X

Отображение Авто

Конфигуратор - Конфигурация

Файл Правка Конфигурация Отладка Администрирование Сервис Окна Справка

Конфигурация * Список пользователей Дополнительно: Полный

Действия * Действия * Действия * П... П... И... П... В...

Конфигурация

- Общие
 - Подсистемы
 - Общие модули
 - Параметры сеанса
 - Роли
 - Администратор
 - Планы обмена
 - Критерии отбора
 - Подписки на события
 - Регламентные задания
 - Интерфейсы
 - Полный
 - Общие макеты
 - Общие картинки
 - Пакеты XDTO
 - Web-сервисы
 - WS-ссылки
- Стили
- Языки
- Константы
- Справочники
- Документы
 - Журналы документов
 - Перечисления
 - Отчеты
 - Обработки
 - Планы видов характеристик
 - Планы счетов
 - Планы видов расчета
 - Регистры сведений
 - Регистры накопления
 - Регистры бухгалтерии
 - Регистры расчета
 - Бизнес-процессы
 - Задачи

Имя Адми

Интерфейс Полный: Интерфейс

Добавить панель Удалить панель ↑ ↓

Список панелей интерфейса:

Главное меню

Файл Операции Справочники Документы Отчеты Обработки Прочие <Новая> Сервис

Окна Справка

- ...Стандартные действия...
- Удаление помеченных объектов
- Поиск ссылок на объекты
- Управление итогами...
- Управление полнотекстовым поиском
- Проведение документов...
- <Новая>
- ...Стандартные действия...

Отбор:

Количество: 1

Роль: Администратор

Права:

- Использование

Ограничения доступа к данным: >>

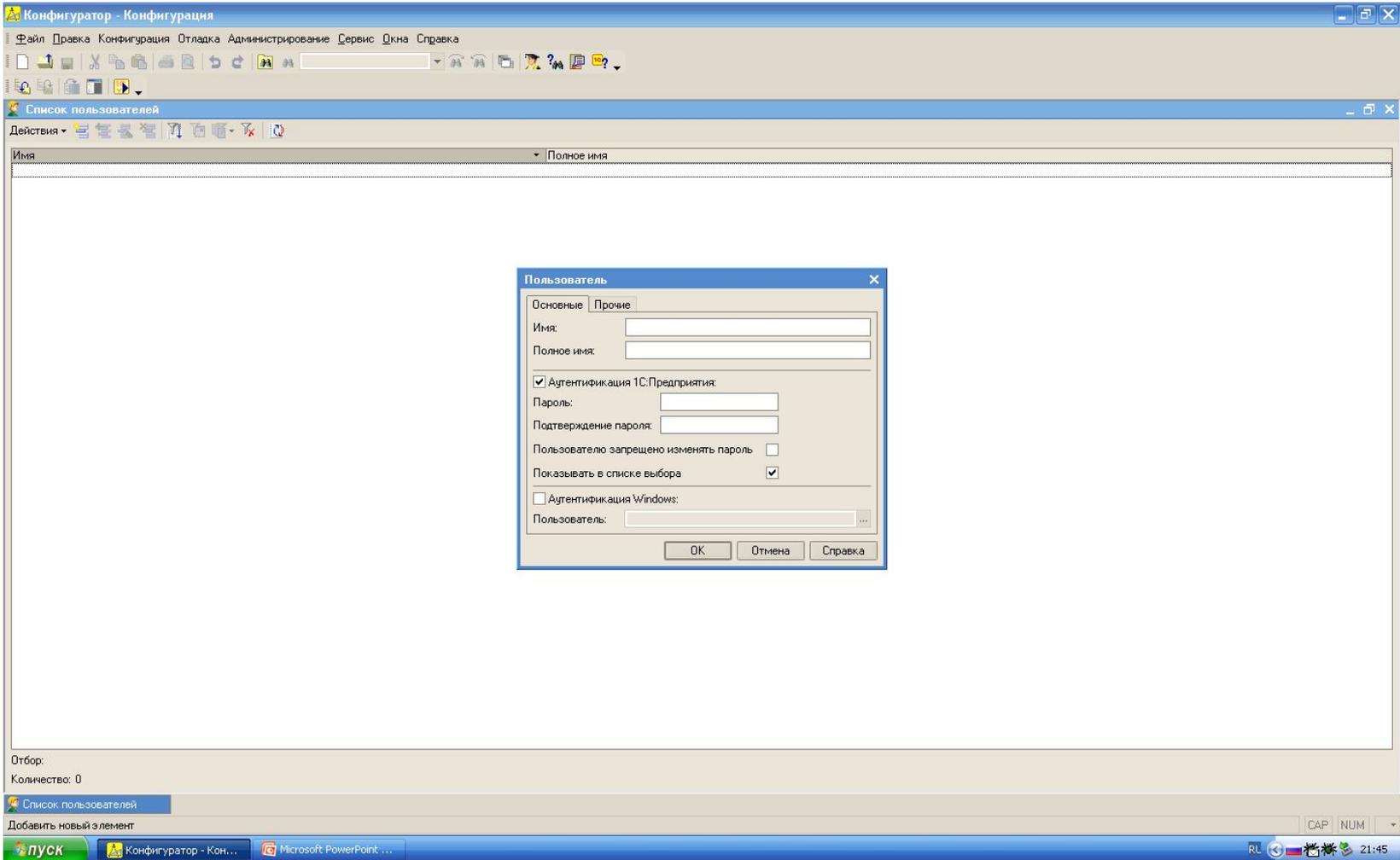
Поля: Ограничение доступа

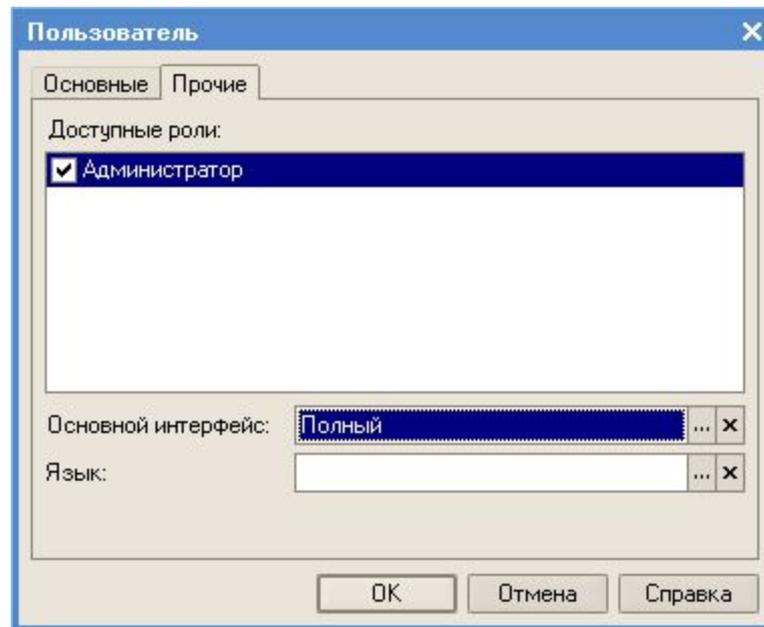
Свойства: Инте... Дополнительно...

Роль Администратор: Права Список пользователей Интерфейс По... Интерфейс

Для получения подсказки нажмите F1

CAP NUM ru





Конфигуратор - Конфигурация

Файл Правка Конфигурация Отладка Администрирование Сервис Окна Справка

Конфигурация Список пользователей Свойства

Действия

Конфигурация

- Общие
 - Подсистемы
 - Общие модули
 - Параметры сеанса
 - Роли
 - Администратор
 - Планы обмена
 - Критерии отбора
 - Подписки на события
 - Регламентные задания
 - Общие формы
 - Интерфейсы
 - Полный
 - Общие макеты
 - Общие картинки
 - Пакеты XDTO
 - Web-сервисы
 - WS-ссылки
 - Стили
 - Языки
 - Константы
 - Справочники
- Документы
 - Журналы документов
 - Перечисления
 - Отчеты
 - Обработки
 - Планы видов характеристик
 - Планы счетов
 - Планы видов расчета
 - Регистры сведений
 - Регистры накопления
 - Регистры бухгалтерии
 - Регистры расчета
 - Бизнес-процессы
 - Задачи

Имя Полное имя

Администратор	Администратор
---------------	---------------

1С:ПРЕДПРИЯТИЕ 8.1

1С:Предприятие

Пользователь

Имя: Администратор

Пароль: xxxxxxxx

OK Отмена

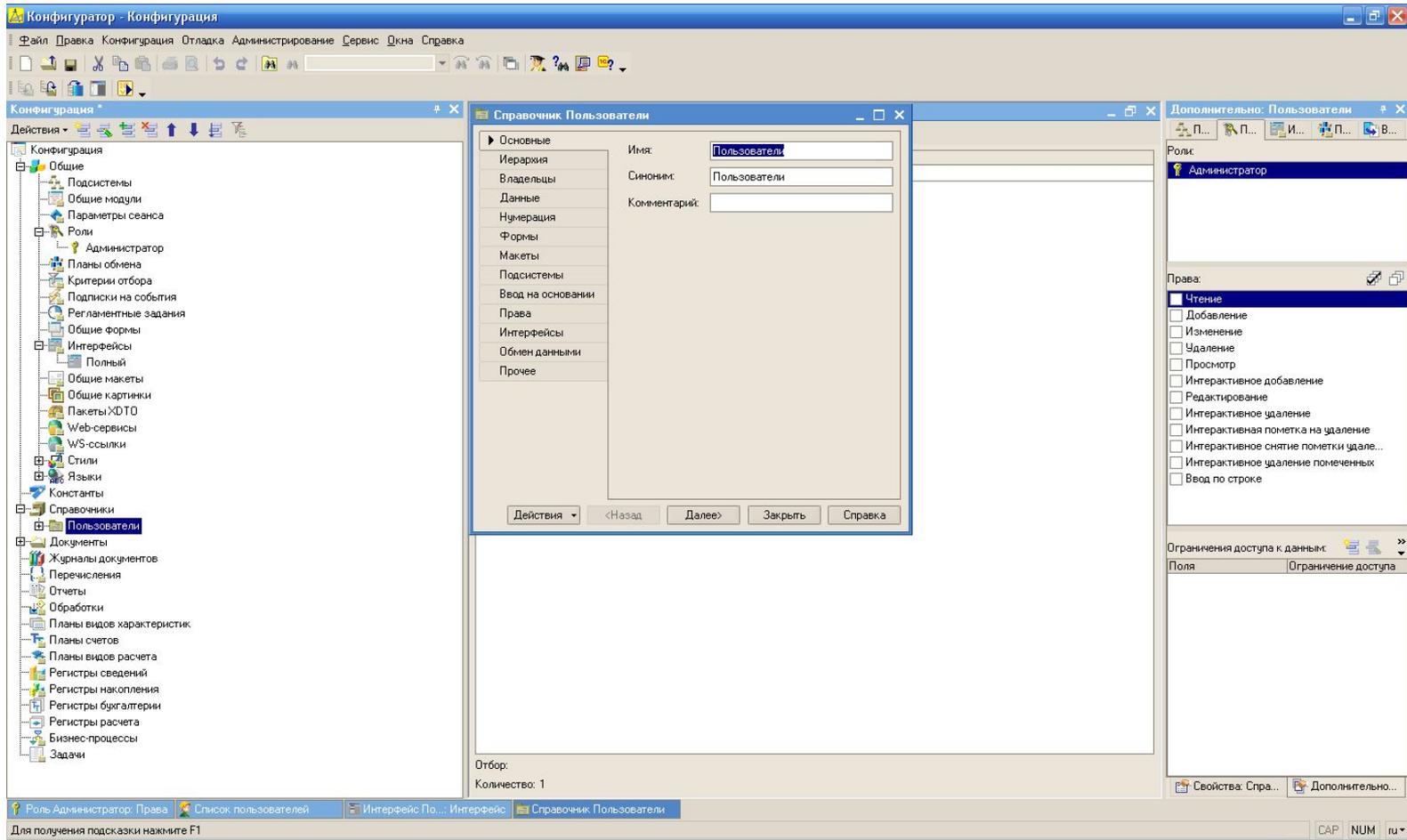
© 1996-2008 1С. Все права защищены

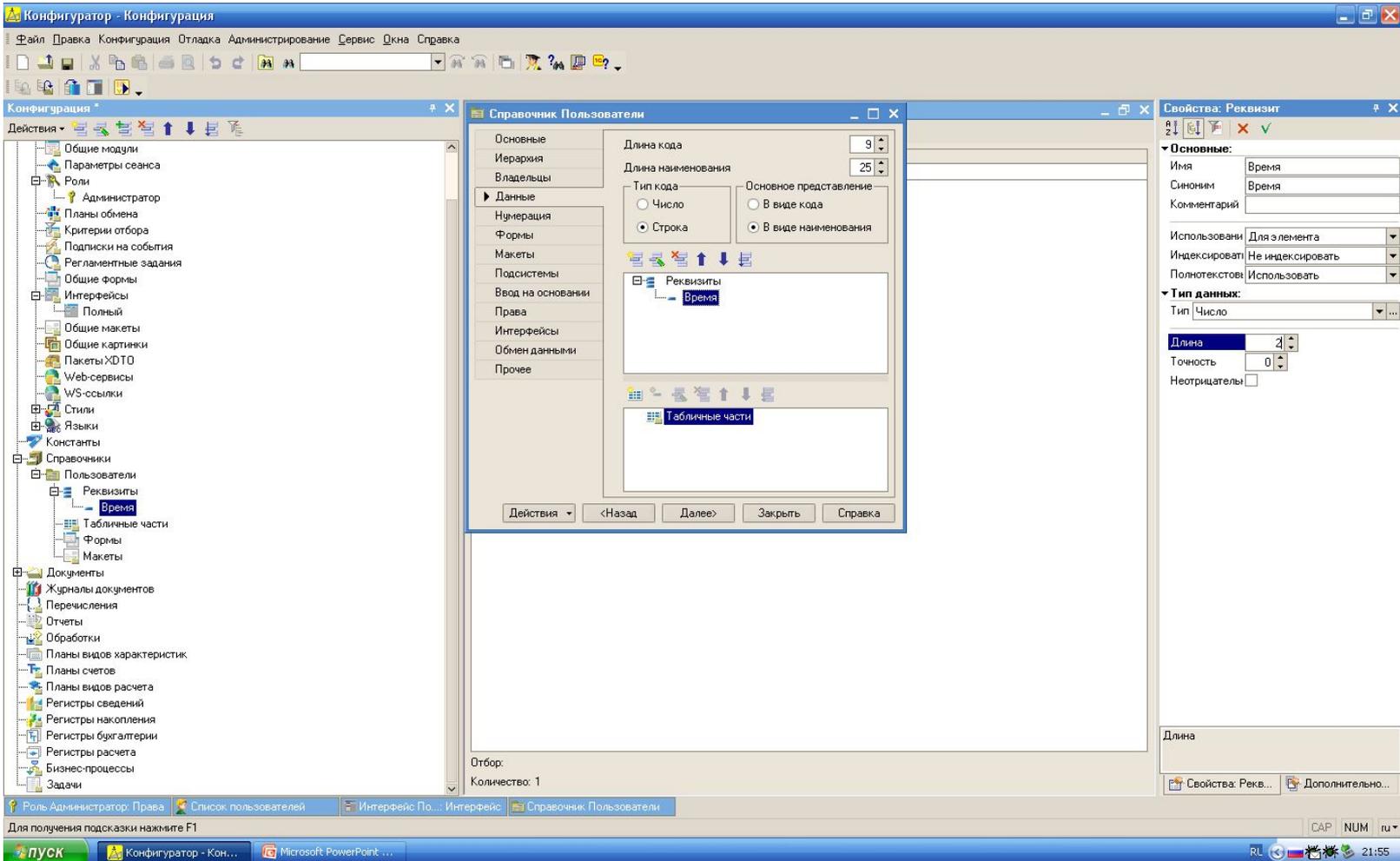
Отбор:
Количество: 1

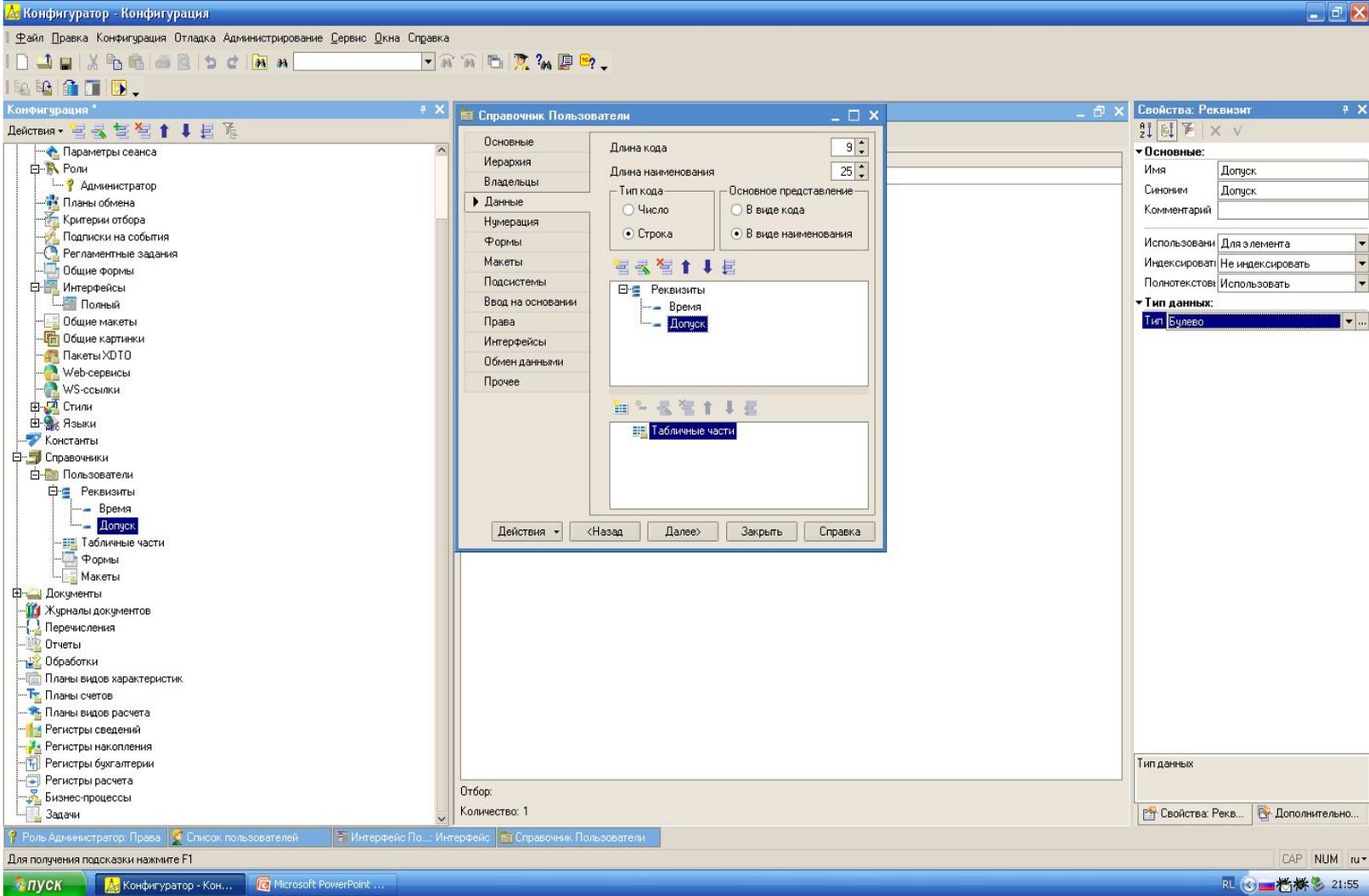
Роль Администратор: Права Список пользователей

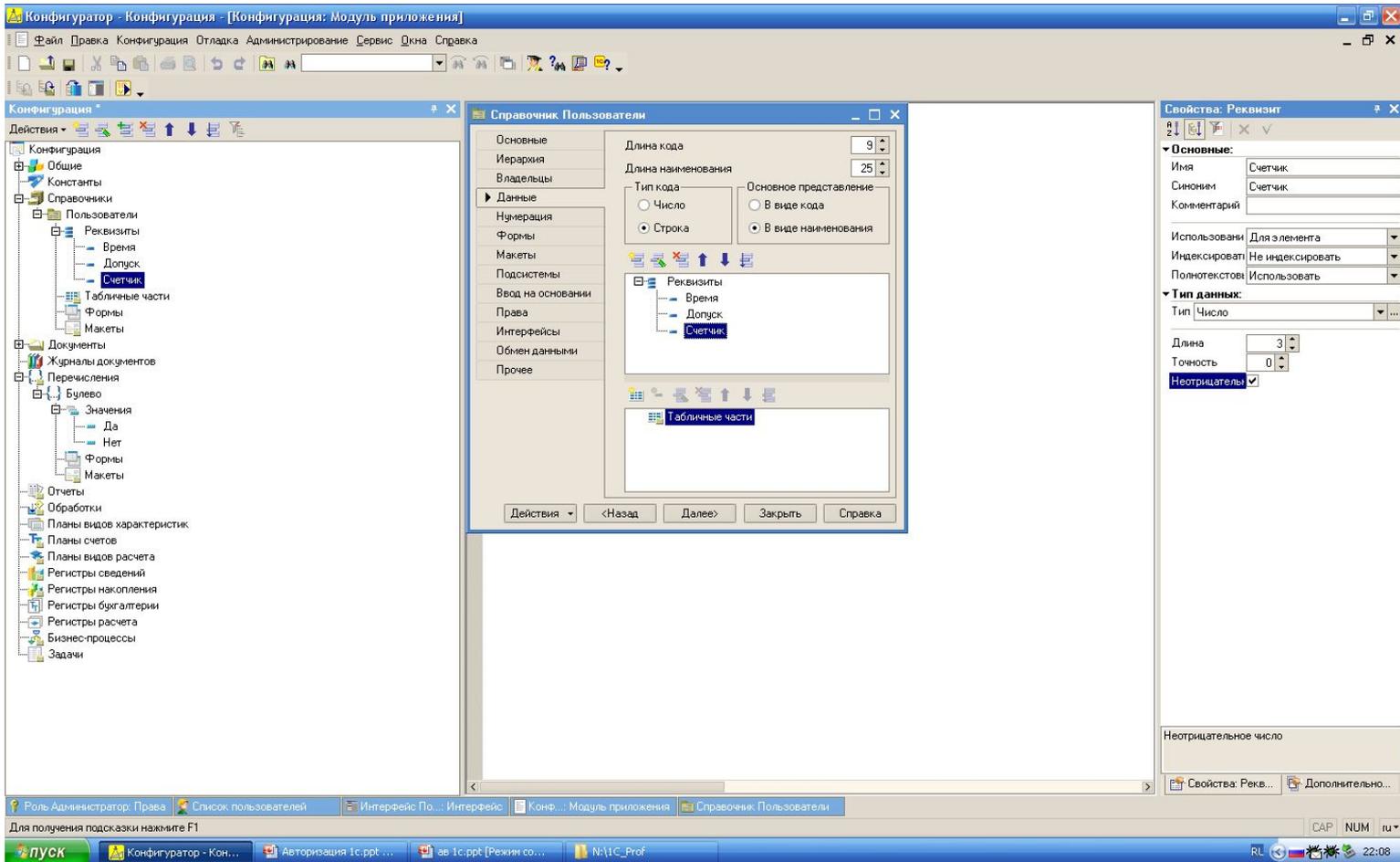
Для получения подсказки нажмите F1

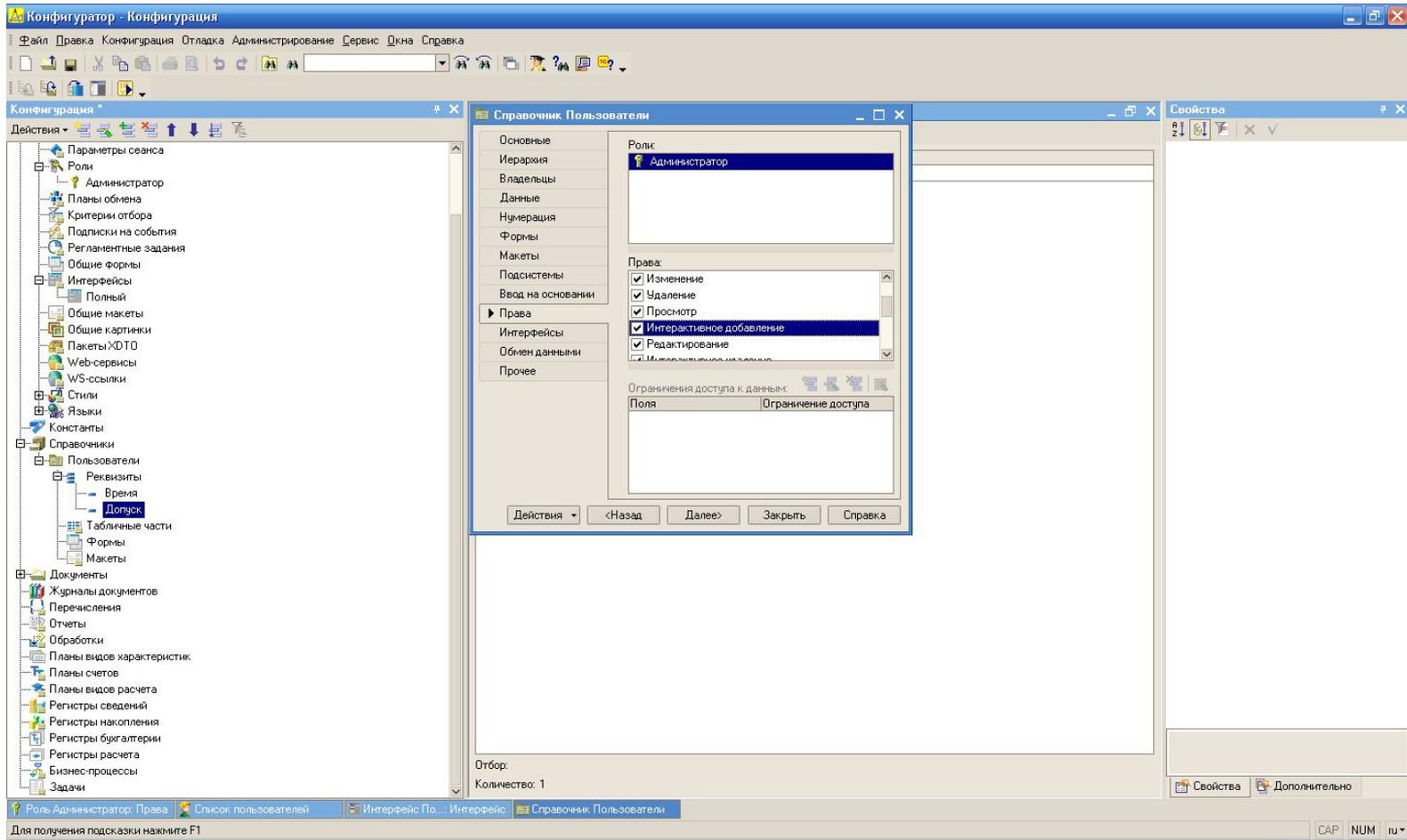
ПУСК Конфигуратор - Кон... 1Сv8 Загрузка конф... Microsoft PowerPoint ... EN 21:50

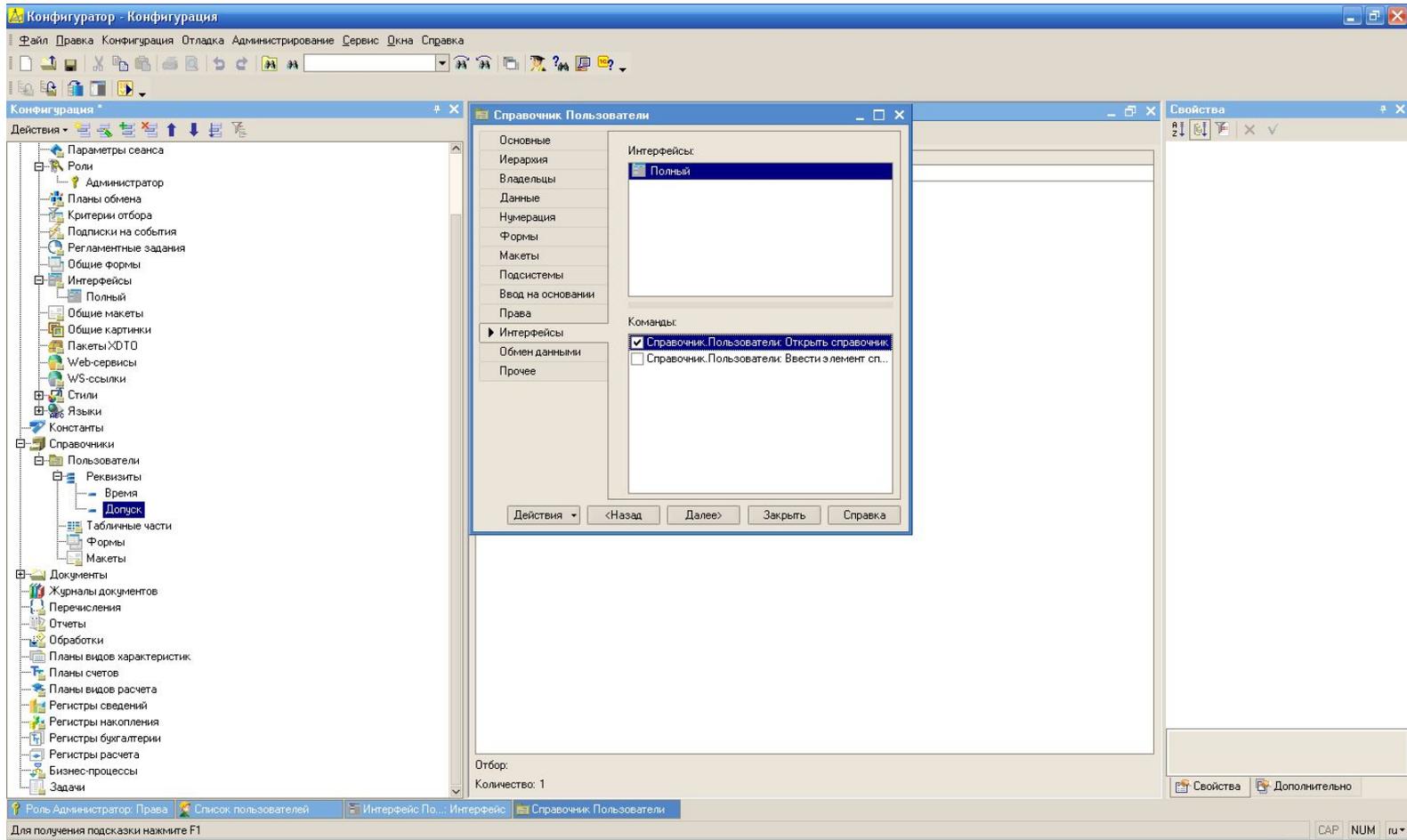


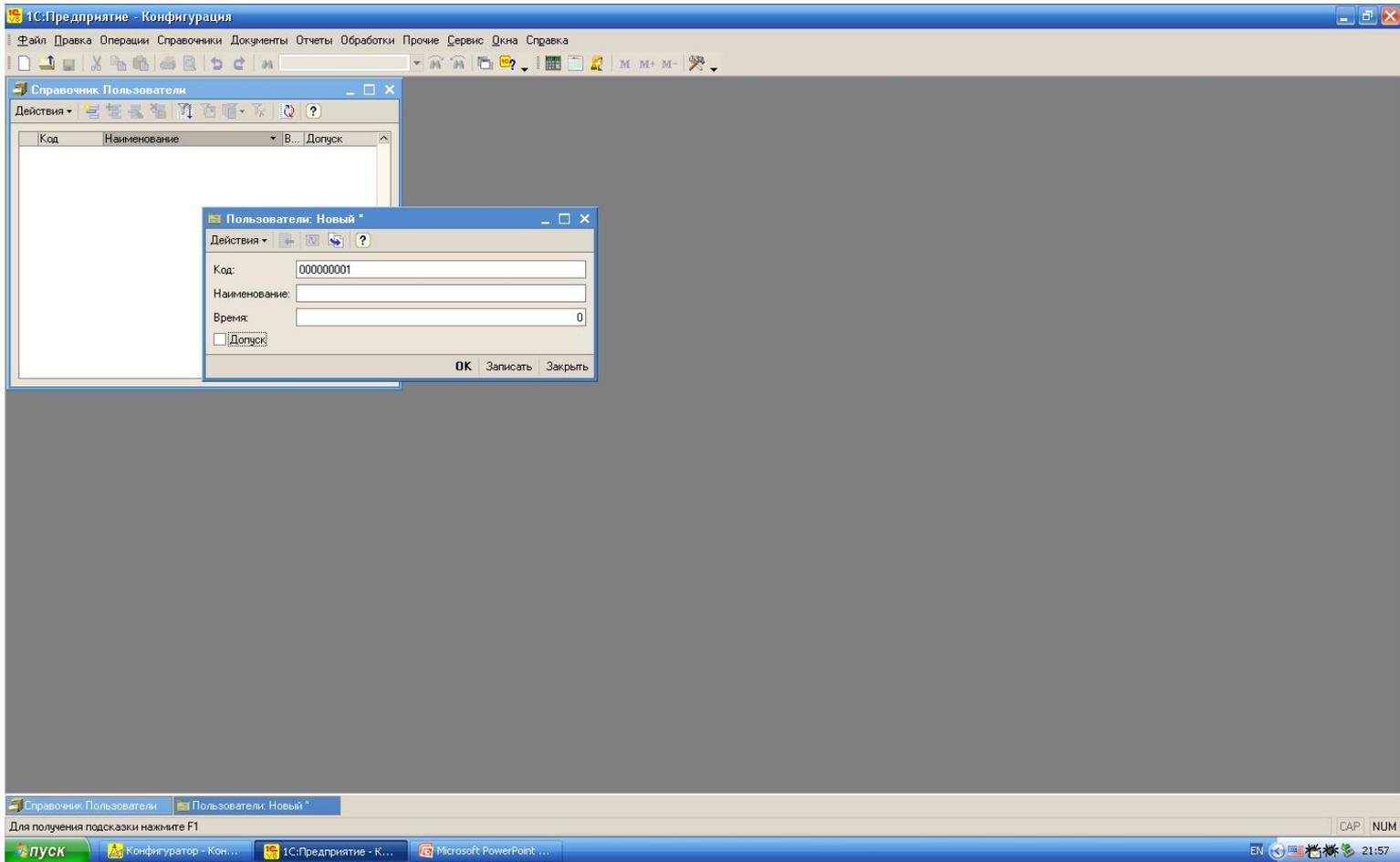


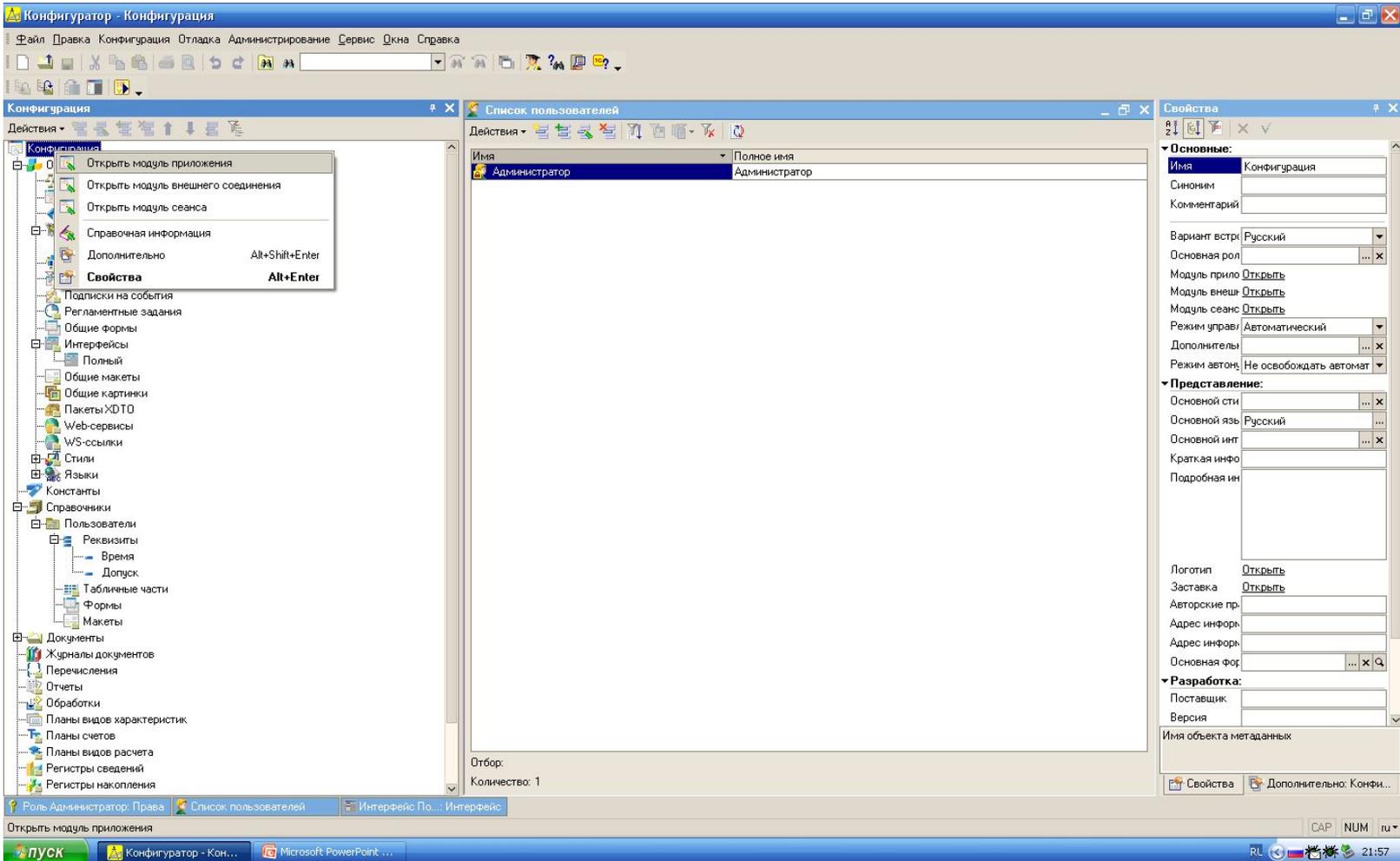




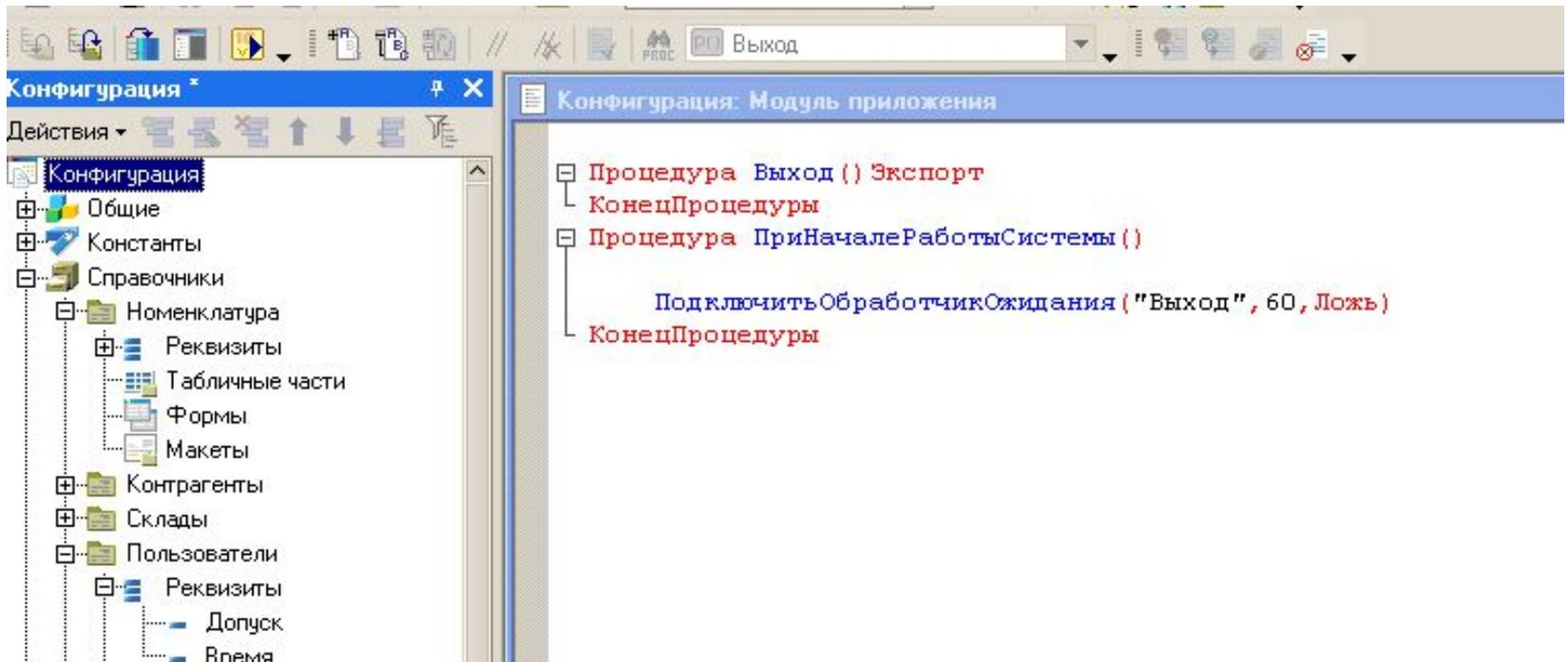








Модуль приложения



```
Процедура Выход () Экспорт
  КонецПроцедуры
Процедура ПриНачалеРаботыСистемы ()
  ПодключитьОбработчикОжидания ("Выход", 60, Ложь)
  КонецПроцедуры
```

Модуль приложения

```
⊕ Процедура Выход () Экспорт ...  
⊖ Процедура ПриНачалеРаботыСистемы ()  
    .....  
    Если Пол.Время<=Вр Тогда  
        ЗавершитьРаботуСистемы ();  
    КонечЕсли;  
  
    ПодключитьОбработчикОжидания ("Выход", 60, Ложь)  
КонецПроцедуры
```

Модуль приложения. Процедура ПриНачалеРаботыСистемы()

Процедура ПриНачалеРаботыСистемы()

Если Справочники.Пользователи.НайтиПоНаименованию(ИмяПользователя())=Справочники.Пользователи.ПустаяСсылка() Тогда

ПолН=Справочники.Пользователи.СоздатьЭлемент();

ПолН.Время=17;

ПолН.Допуск=Перечисления.Булево.Да;

ПолН.Наименование=ИмяПользователя();

ПолН.Счетчик=0;

ПолН.Записать();

Иначе

Вр=Число(Сред(ТекущаяДата(), 12, 1));

Пол=Справочники.Пользователи.НайтиПоНаименованию(ИмяПользователя()).ПолучитьОбъект();

Если (Пол.Время<=Вр) или (Пол.Допуск=Перечисления.Булево.Нет) Тогда

ЗавершитьРаботуСистемы();

КонецЕсли;

КонецЕсли;

ПодключитьОбработчикОжидания("Выход", 60, Ложь)

КонецПроцедуры

Поле счетчик

Справочник Пользователи

Поиск текста (Ctrl+F)

Действия

Код	Наименование	Допуск	Время	Счетчик
000000001	Администратор	Да		17
000000003	Администратор1	Да		17
000000002	Один	Нет		17

Модуль приложения. Процедура Выход

Процедура Выход () Экспорт

Вр=Число (Сред (ТекущаяДата (), 12, 1));

Пол=Справочники.Пользователи.НайтиПоНаименованию (ИмяПользователя ()) .ПолучитьОбъект ();

Если (Пол.Время<=Вр) или (Пол.Допуск=Перечисления.Булево.Нет) Тогда

 Если Пол.Счетчик=0 Тогда

 Пол.Счетчик=300;

 Пол.Записать ();

 Предупреждение ("Завершение работы системы через 5 минут", 5);

 Иначе

 ТСчетчик=Пол.Счетчик;

 ТСчетчик=ТСчетчик-60;

 Пол.Счетчик=ТСчетчик;

 Пол.Записать ();

 Если ТСчетчик=0 Тогда

 Предупреждение ("Пока", 1);

 ЗавершитьРаботуСистемы ();

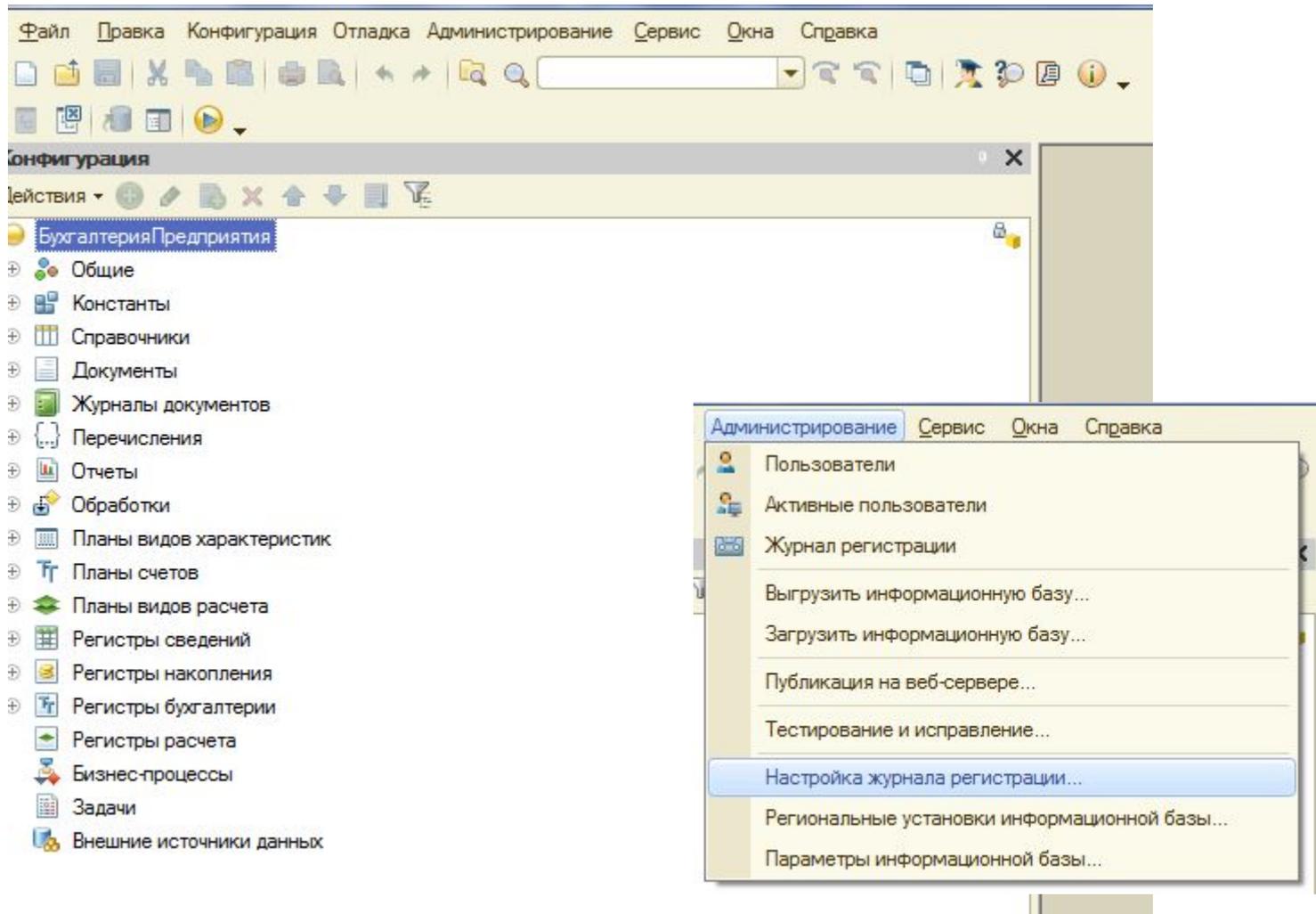
 КонецЕсли;

 КонецЕсли;

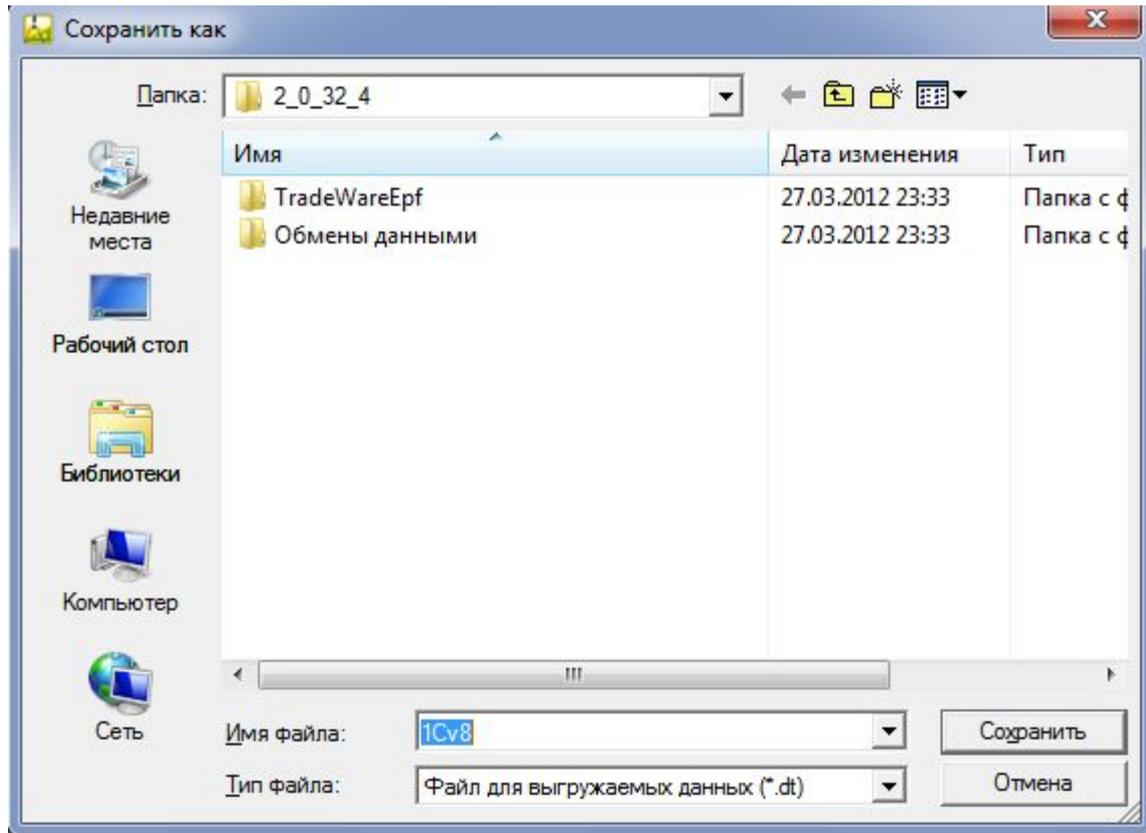
КонецЕсли;

КонецПроцедуры

Администрирование системы 1с: Предприятие



Создание резервной копии



Журнал регистрации

Журнал регистрации <1>					
Действия					
Дата, время	Пользователь		Событие	Статус транз...	Метаданные
	Компьютер		Комментарий	Транзакция	Данные
	Приложение	Сеанс			Представление данных
27.04.2011 18:28:18	MICROSOFT-C7A30B Толстый клиент...		Данные. Изменение	Зафиксирова...	Регистр сведений. Настрой...
27.04.2011 18:28:18	MICROSOFT-C7A30B Толстый клиент...		Данные. Добавлен...	Зафиксирова...	Справочник. Валюты
27.04.2011 18:28:19	MICROSOFT-C7A30B Толстый клиент...		Данные. Изменение	Зафиксирова...	Регистр сведений. Курсы в ...
27.04.2011 18:28:19	MICROSOFT-C7A30B Толстый клиент...		Данные. Изменение	Зафиксирова...	Константа. Валюта реглам...
27.04.2011 18:28:19	MICROSOFT-C7A30B Толстый клиент...		Данные. Изменение	Зафиксирова...	Константа. Применяемые с...

Отбор:

Тестирование ИБ

Тестирование и исправление информационной базы

Проверки и режимы :

- Реиндексация таблиц информационной базы
- Проверка логической целостности информационной базы
- Проверка ссылочной целостности информационной базы
- Пересчет итогов
- Сжатие таблиц информационной базы
- Реструктуризация таблиц информационной базы

Только тестирование

Тестирование и исправление

При наличии ссылок на несуществующие объекты :

- Создавать объекты
- Очищать ссылки
- Не изменять

При частичной потере данных объектов :

- Создавать объекты
- Удалять объект
- Не изменять

Продолжить прерванное ранее тестирование

Прервать выполнение проверки через часов мин.

Выполнить

Закреть

Справка

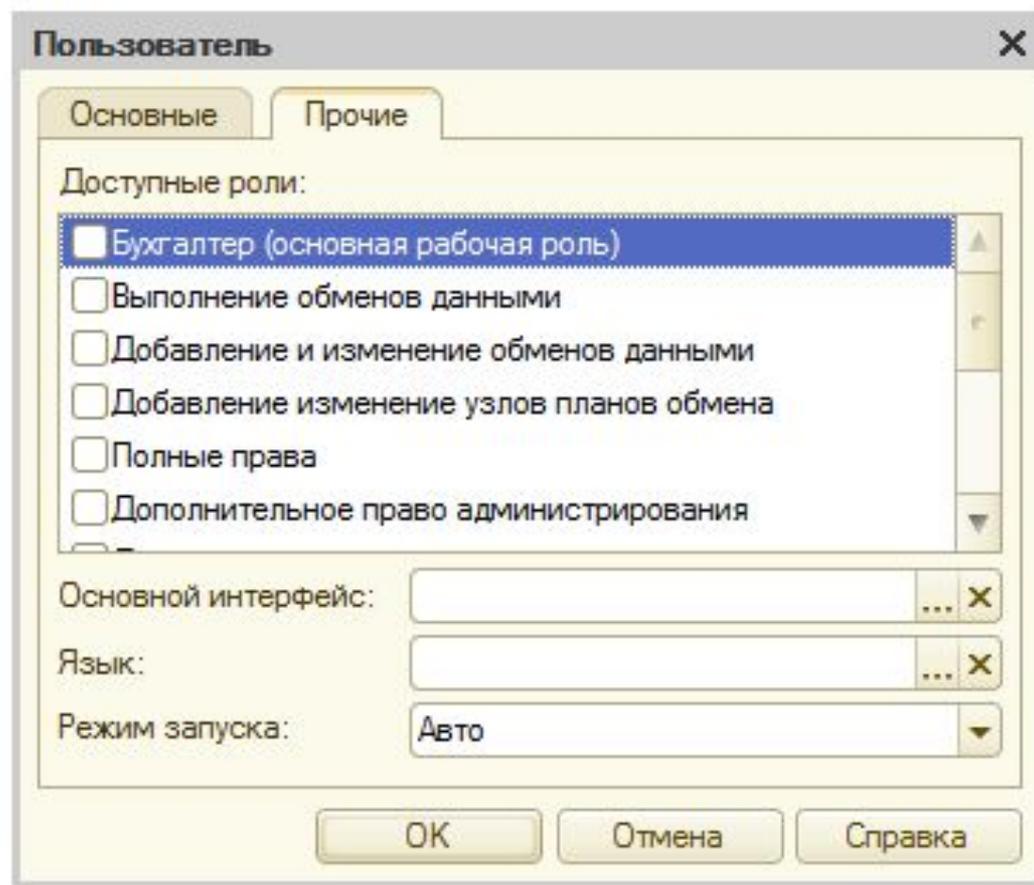
Создание учетных записей

The image shows a screenshot of a user management application. The main window is titled "Список пользователей" (List of users) and has a toolbar with various icons for adding, deleting, and editing users. Below the toolbar, there is a header row with "Имя" (Name) and "Полное имя" (Full name) columns. A dialog box titled "Пользователь" (User) is open, allowing the creation of a new user. The dialog has two tabs: "Основные" (Basic) and "Прочие" (Other). The "Основные" tab is active and contains the following fields and options:

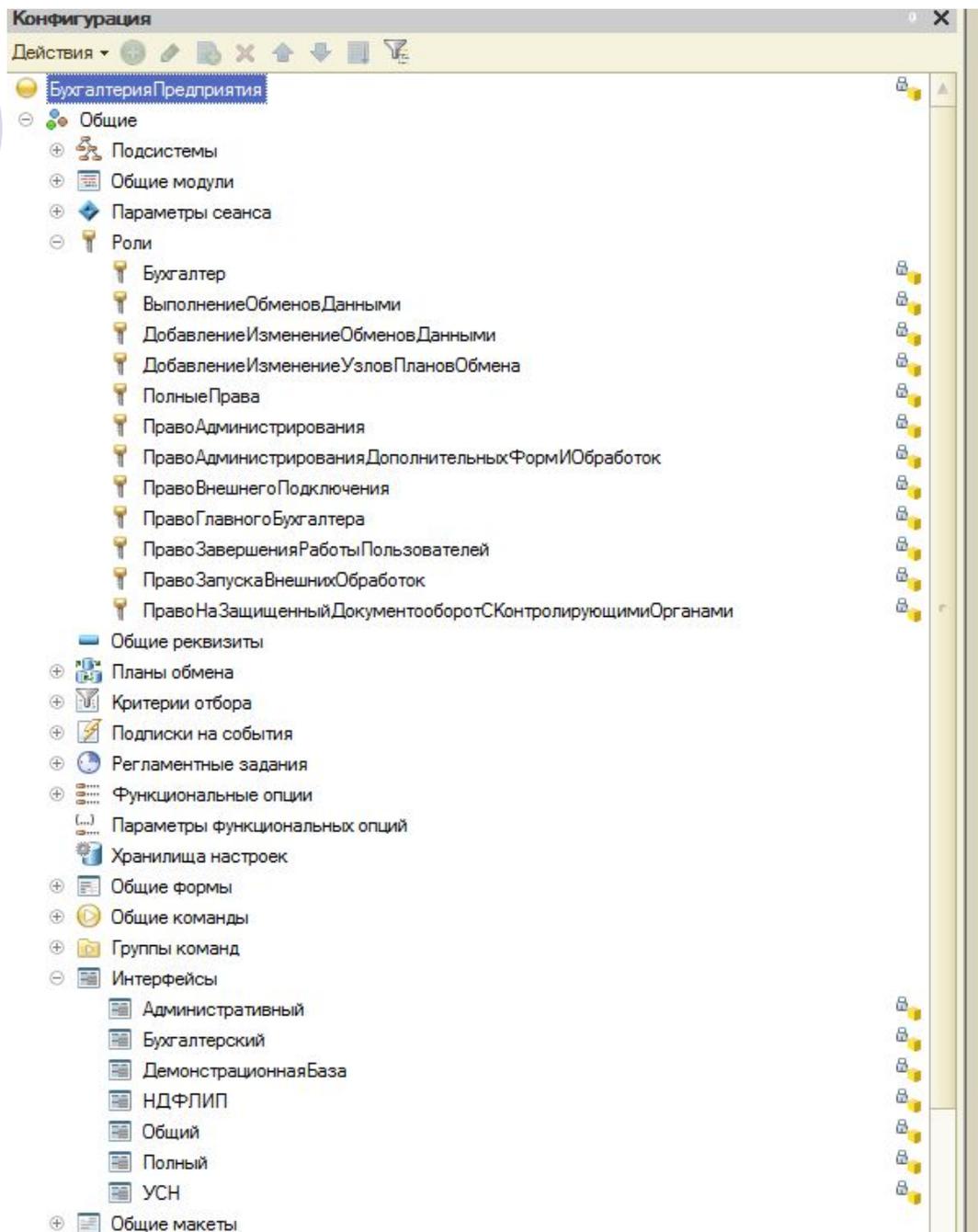
- Имя:** Text input field.
- Полное имя:** Text input field.
- Аутентификация 1С:Предприятия:** (1C authentication)
- Пароль:** Password input field.
- Подтверждение пароля:** Password confirmation input field.
- Пользователю запрещено изменять пароль** (Password change is prohibited for the user).
- Показывать в списке выбора** (Show in selection list).
- Аутентификация операционной системы:** (Operating system authentication)
- Пользователь:** Text input field with a dropdown arrow.

At the bottom of the dialog, there are three buttons: "ОК", "Отмена" (Cancel), and "Справка" (Help).

Создание учетных записей



Создание учетных записей



Роли (права)

Роль ПравоАдминистрирования: Права [Только для чтения]

Действия ▾

Права | Шаблоны ограничений

Объекты:

- Бухгалтерия Предприятия
- Общие
- Константы
- Справочники
- Документы
- Журналы документов
- Перечисления
- Отчеты
- Обработки
- Планы видов характеристик
- Планы счетов
- Планы видов расчета
- Регистры сведений
- Регистры накопления

Права:

- ✓ Администрирование данных
- ✓ Обновление конфигурации базы данных
 - Монопольный режим
 - Активные пользователи
 - Журнал регистрации
- ✓ Тонкий клиент

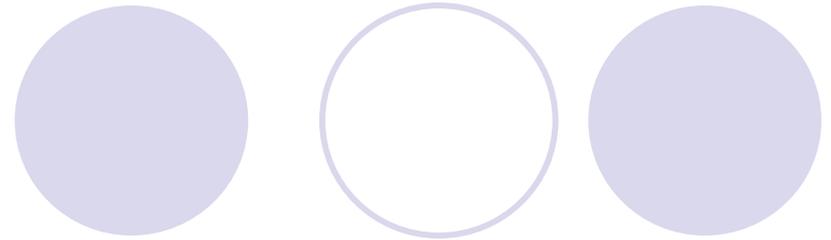
Ограничения доступа к данным:

Поля	Ограничение доступа
------	---------------------

Описание: Право на выполнение административных функций (Администрирование)

- Устанавливать права для новых объектов
- Устанавливать права для реквизитов и табличных частей по умолчанию
- Независимые права подчиненных объектов

Роли (права)



Пользователь

Основные Прочие

Имя: Администратор

Полное имя: Администратор

Аутентификация 1С:Предприятия:

Пароль: [маска] (простой)

Подтверждение пароля: [маска]

Пользователю запрещено изменять пароль

Показывать в списке выбора

Аутентификация операционной системы:

Пользователь: [поле]

OK Отмена Справка

Пользователь

Основные Прочие

Доступные роли:

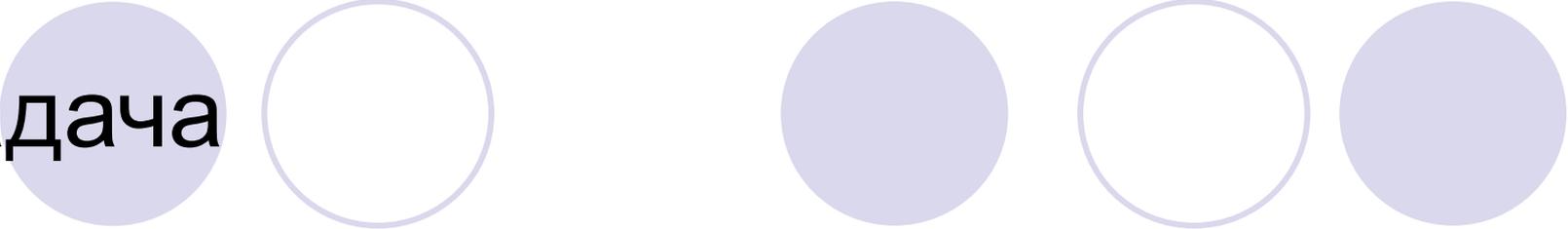
- Добавление и изменение обменов данными
- Добавление изменение узлов планов обмена
- Полные права
- Дополнительное право администрирования
- Дополнительное право администрирования подключен...
- Дополнительное право внешнего подключения (соедин...

Основной интерфейс: Административный

Язык: [поле]

Режим запуска: Авто

OK Отмена Справка



Задача

- Ограничить время работы пользователей с 9.00-17.00
- Организовать «выгонялку»
- Реализация через справочник

Определение имени пользователя

```
⊞ // Обработчик события УстановкаПараметровСеанса() [//...]
⊞ Процедура УстановкаПараметровСеанса (ТребуемыеПараметры)

    Если ТребуемыеПараметры = Неопределено Тогда
        // Раздел установки параметров сеанса при начале сеанса (ИменаПараметровСеанса = Неопределено)
        // Выполняется установка параметров сеанса, которые можно инициализировать
        // при начале работы системы

        // инициализация параметра сеанса "ТекущийПользователь"
        ПолныеПрава.УстановитьПараметрСеансаТекущийПользователь ();
        ПолныеПрава.УстановитьИзменяемыеПараметрыСеансаПользователя ();

    Иначе
        // Установка параметров сеанса "по требованию"

        // Параметры сеанса, инициализация которых требует обращения к одним и тем же данным
        // следует инициализировать сразу группой. Для того, чтобы избежать их повторной инициализации,
        // имена уже установленных параметров сеанса сохраняются в массиве УстановленныеПараметры
        УстановленныеПараметры = Новый Массив;
        Для каждого ИмяПараметра Из ТребуемыеПараметры Цикл
            УстановитьЗначениеПараметраСеанса (ИмяПараметра, УстановленныеПараметры);
        КонечныйЦикл;
    КонечныйЕсли;
    Сообщить (ИмяПользователя ());
КонечныйПроцедуры // УстановкаПараметровСеанса

⊞ // Установить значения параметров сеанса и вернуть имена установленных параметров сеанса [//...]
⊞ Процедура УстановитьЗначениеПараметраСеанса (Знач ИмяПараметра, УстановленныеПараметры)

    // {ОбменДаннымиБСП
    ОбменДаннымиСервер.УстановкаПараметровСеанса (ИмяПараметра, УстановленныеПараметры);
    // }ОбменДаннымиБСП
КонечныйПроцедуры
```

Изменения справочника

- НомераГТД
- ОбработкиОбслуживанияТО
- ОбщероссийскийКлассификаторОсновныхФондов
- ОбъектыСтроительства
- Организации
- ОрганыПФР
- ОснованияВыслугиЛет
- ОснованияИсчисляемогоТрудовогоСтаж
- ОснованияУвольненийИзОрганизаций
- ОсновныеСредства
- ОсобыеУсловияТруда
- ОтправкиФСС
- ПараметрыВыработкиОС
- ПерепискаСКонтролирующимиОрганами
- ПодразделенияОрганизаций
- Пользователи
 - Реквизиты
 - ФизЛицо
 - Время
 - Допуск
 - Табличные части
 - Формы
 - ФормаСписка
 - ФормаЭлемента
 - ФормаГруппы
 - ФормаВыбораГруппы
 - ФормаВыбора
 - ФормаНастройкиГруппПользователя
 - Команды
 - Макеты
- ПричиныСписанияОС
- ПрочиеДоходыИРасходы
- РасходыБудущихПериодов
- РегистрацияВИФНС
- РегламентированныеОтчеты

БухгалтерияПредприятия: Модуль сеанса

Справочник Пользователи: ФормаЭлемента

Настройки пользователя

Действия | Перейти | Группы пользователя

Группа: []

Краткое имя: []

Полное имя: []

Физическое лицо: []

Время [0]

Допуск

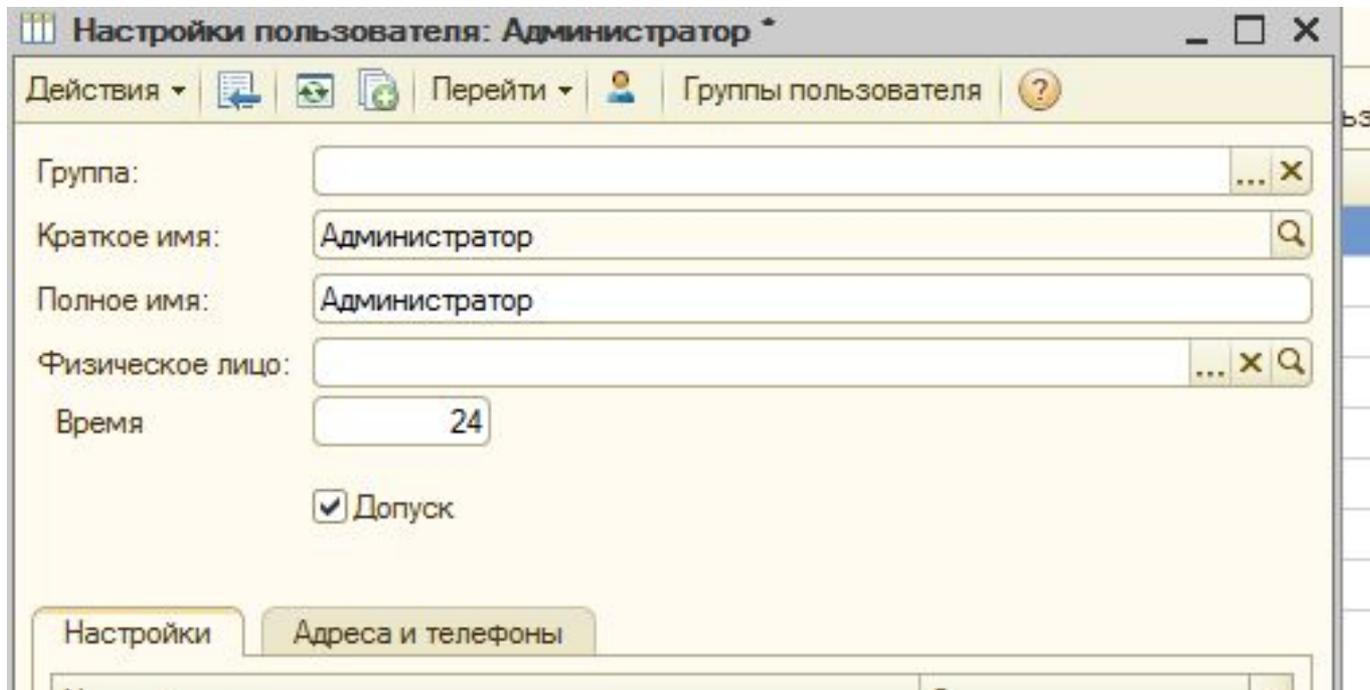
Настройки | Адреса и телефоны

Надпись КИ1 []

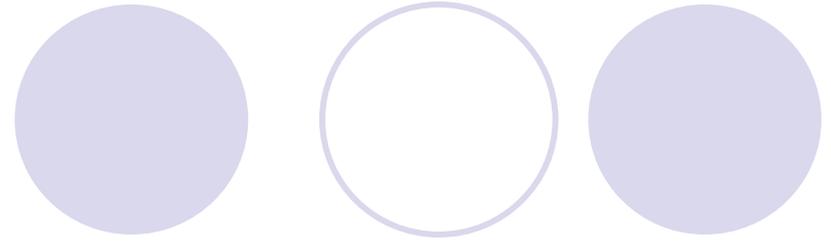
Другое (любая другая контактная информация): []

Диалог | Модуль | Реквизиты

Результат

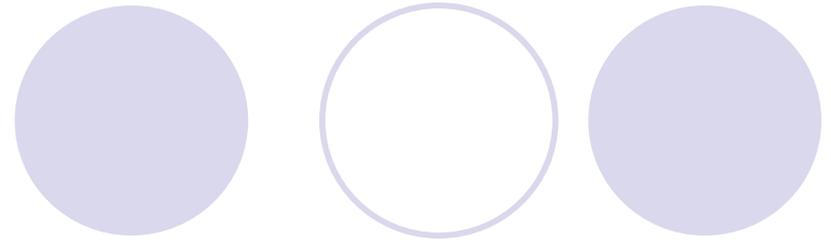


Перечисление



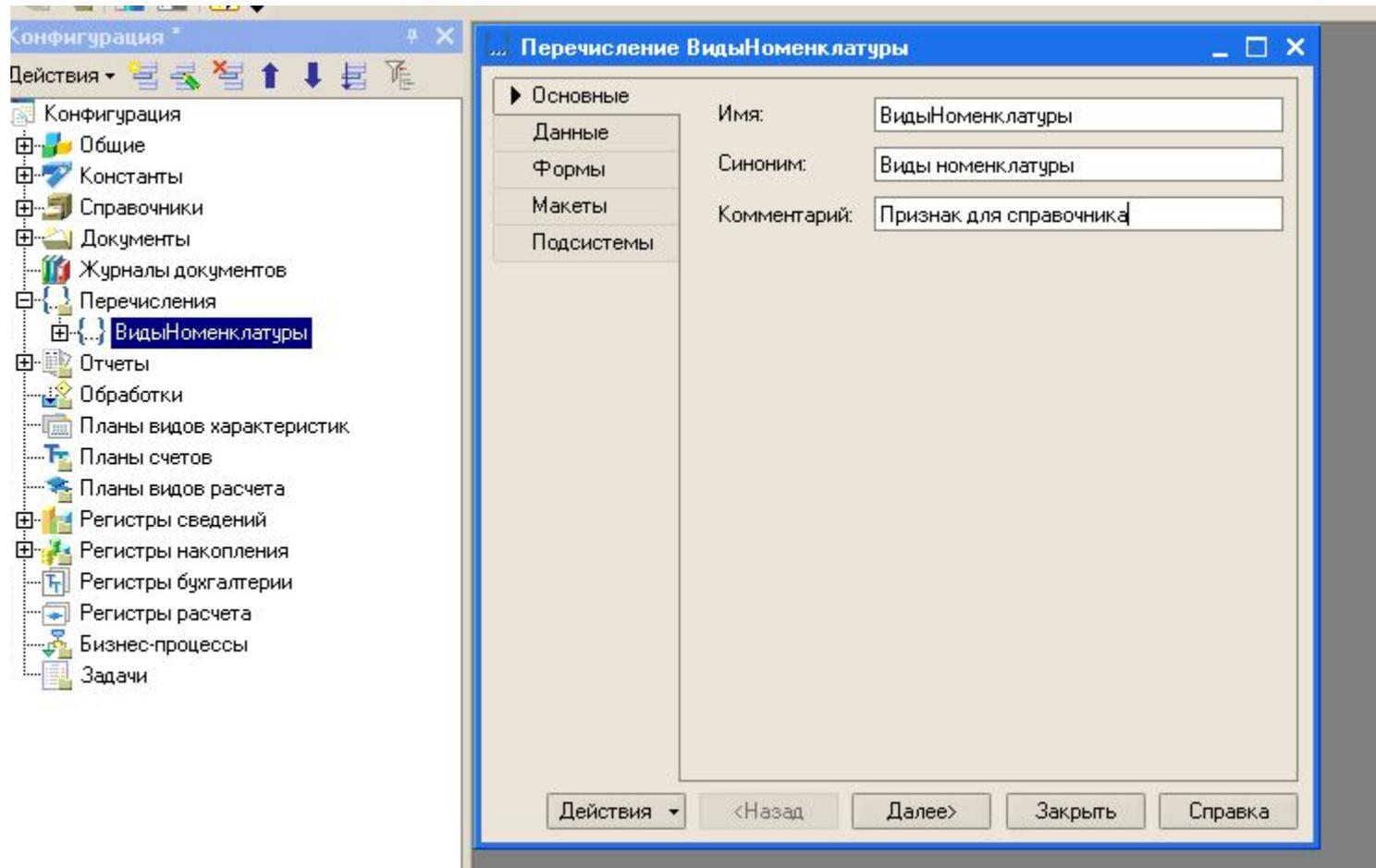
- Признаки для номенклатуры:
- Товар
- Услуга
- Продукция

Перечисление



- Создадим сначала новый объект
- конфигурации Перечисление с
- именем «ВидыНоменклатуры».

Перечисление



Перечисление

The screenshot displays the 'Конфигуратор - Конфигурация' application interface. The main window shows a tree view on the left with 'Перечисления' expanded to 'ВидыНоменклатуры' and 'Значения' selected. The 'Значения' list contains 'Услуга', 'Товар', and 'Продукция'. A dialog box titled 'Свойства: Значение перечисления' is open, showing the following properties:

Основные:	
Имя	Продукция
Синоним	Продукция
Комментарий	

The application window includes a menu bar (Файл, Правка, Конфигурация, Отладка, Администрирование, Сервис, Окна, Справка), a toolbar, and a status bar at the bottom with the text 'Для получения подсказки нажмите F1' and system tray icons.

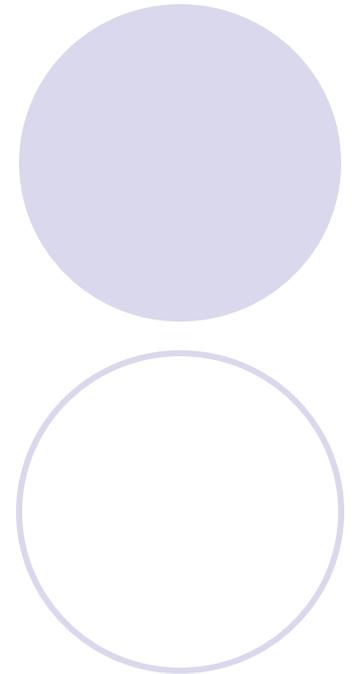
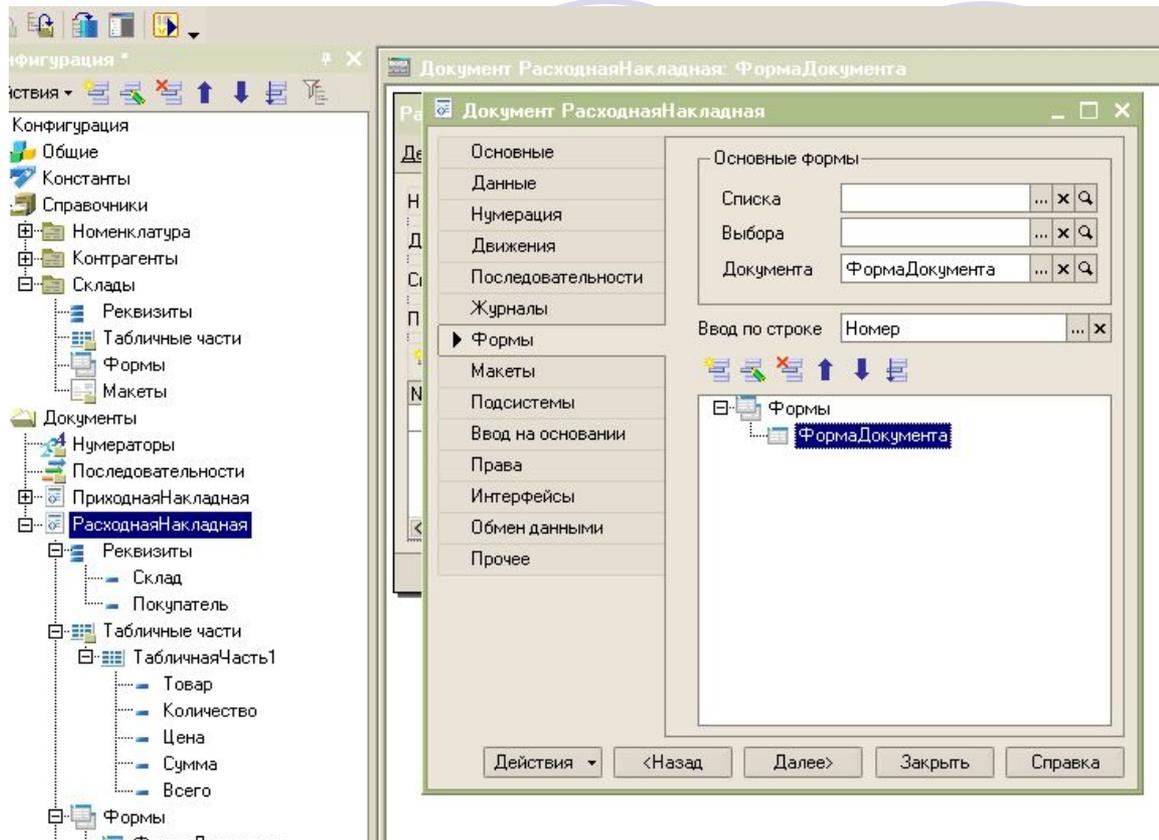
Новый реквизит справочника Номенклатура

The screenshot displays the 'Конфигуратор - Конфигурация' (Configurator - Configuration) application. The main window shows a tree view of the configuration structure, with 'Номенклатура' (Inventory) expanded to show 'Реквизиты' (Attributes). The 'Свойства: Реквизит' (Attribute Properties) dialog box is open, showing the following fields:

- Имя (Name): Тип
- Синоним (Synonym): Тип
- Комментарий (Comment):
- Использование (Usage): Для элемента (For element)
- Индексировать (Index): Не индексировать (Do not index)
- Тип данных (Data Type): ПеречислениеСсылка.ВидыНоменклатуры (EnumerationLink.InventoryTypes)

The background window shows the 'Перечисление ВидыНоменклатуры' (Inventory Types Enumeration) dialog, with the 'Значения' (Values) list containing 'Услуга' (Service), 'Товар' (Goods), and 'Производство' (Production). The 'Производство' value is currently selected.

Создание формы документа



Создание формы документа

Документ РасходнаяНакладная: ФормаДокумента

Расходная накладная

Действия Перейти

Номер:

Дата:

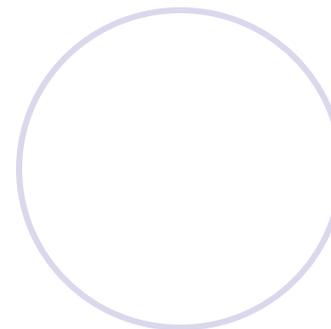
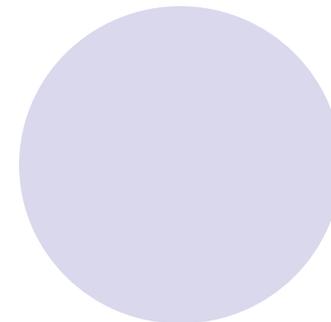
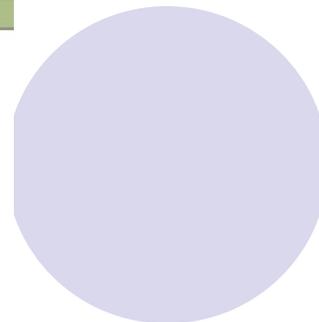
Склад:

Покупатель:

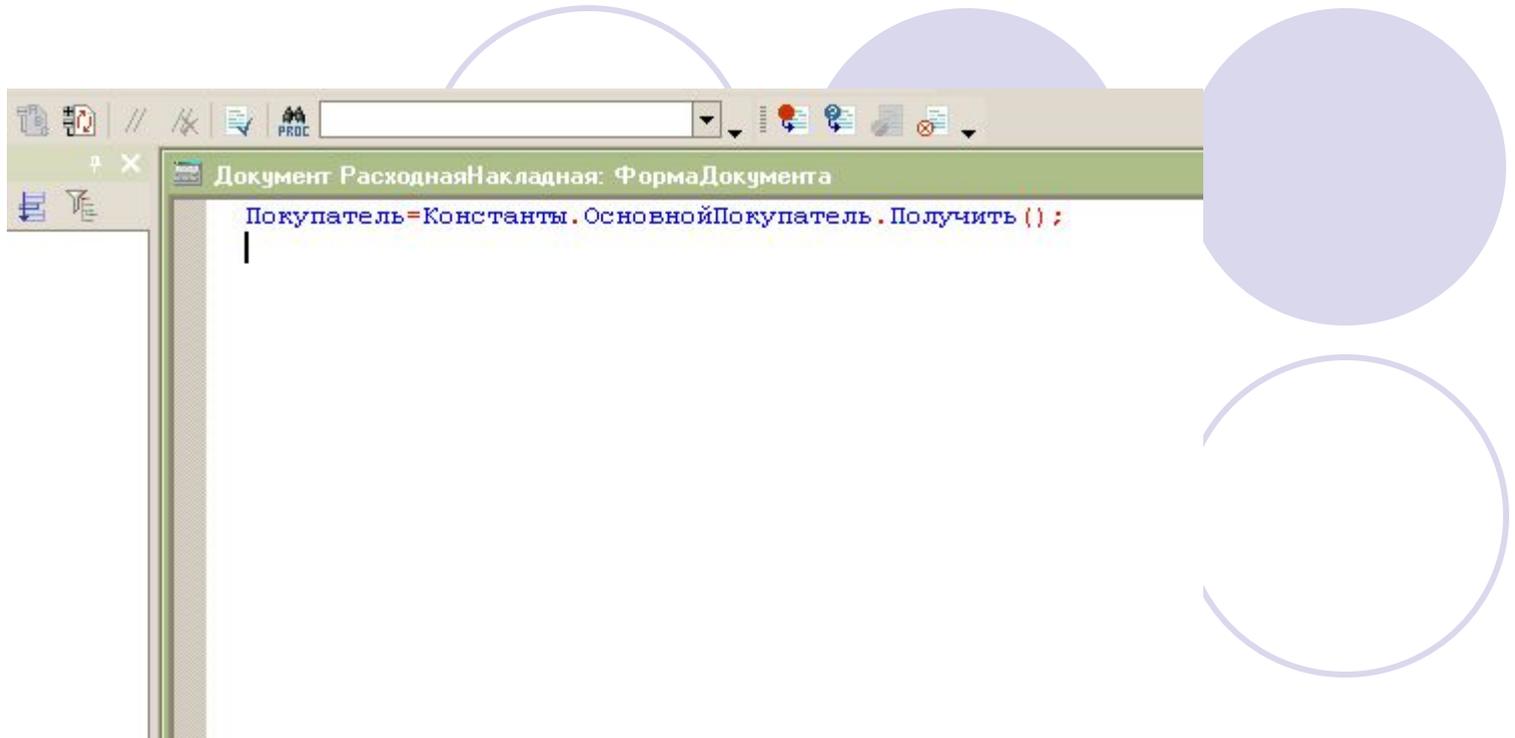
N	Товар	Количество	Цена	Сумма
<input type="text" value="ab"/>	<input type="text" value="<Поле ввода>"/>	<input type="text" value="ab"/>	<input type="text" value="<Поле ввода>"/>	<input type="text" value="ab"/>

ОК Записать Закреть

Диалог | Модуль | Реквизиты



Определение значения по умолчанию



Создание процедуры обработки события

The image shows a software interface for creating a procedure to handle an event. The main window is titled "Расходная накладная" (Expense Receipt) and contains fields for "Номер:" (Number), "Дата:" (Date), "Склад:" (Warehouse), and "Покупатель:" (Buyer). Below these fields is a table with columns: "N", "Товар" (Goods), "Количество" (Quantity), "Цена" (Price), and "Сумма" (Sum). The first row of the table contains the text "<Поле ввода>" (Input Field) in each column.

An open dialog box titled "Свойства: ТабличнаяЧасть1 Товар Поле ввода" (Properties: TablePart1 Goods Input Field) is overlaid on the table. It contains settings for the selected cell, including "Высота списка выбора" (Selection list height) and "Минимальная ширина списка выбора" (Minimum selection list width), both set to 0. Below these are the "События:" (Events) section, which lists several events with dropdown menus for selection:

- При изменении (Selected)
- Начало выбора
- Начало выбора из списка
- Очистка
- Регулирование
- Открытие (редактирование)
- Обработка выбора
- Автоподбор текста
- Окончание ввода текста

Создание процедуры обработки события



```
Процедура ТабличнаяЧасть1ТоварПриИзменении ( Элемент )
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные;
    СтрокаТабличнойЧасти.Цена=СтрокаТабличнойЧасти.Товар.Цена;
КонецПроцедуры
Покупатель=Константы.ОсновнойПокупатель.Получить ();
```

Создание процедуры обработки события

```
Документ РасходнаяНакладная: ФормаДокумента
```

```
[-] Процедура ТабличнаяЧасть1ТоварПриИзменении (Элемент)  
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные;  
    СтрокаТабличнойЧасти.Цена=СтрокаТабличнойЧасти.Товар.Цена;  
[-] КонецПроцедуры
```

```
[-] Процедура ТабличнаяЧасть1КоличествоПриИзменении (Элемент)  
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные;  
    СтрокаТабличнойЧасти.Сумма=СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена;  
[-] КонецПроцедуры |
```

```
[-] Процедура ТабличнаяЧасть1ЦенаПриИзменении (Элемент)  
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные;  
    СтрокаТабличнойЧасти.Сумма=СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена;  
[-] КонецПроцедуры  
Покупатель=Константы.ОсновнойПокупатель.Получить();
```

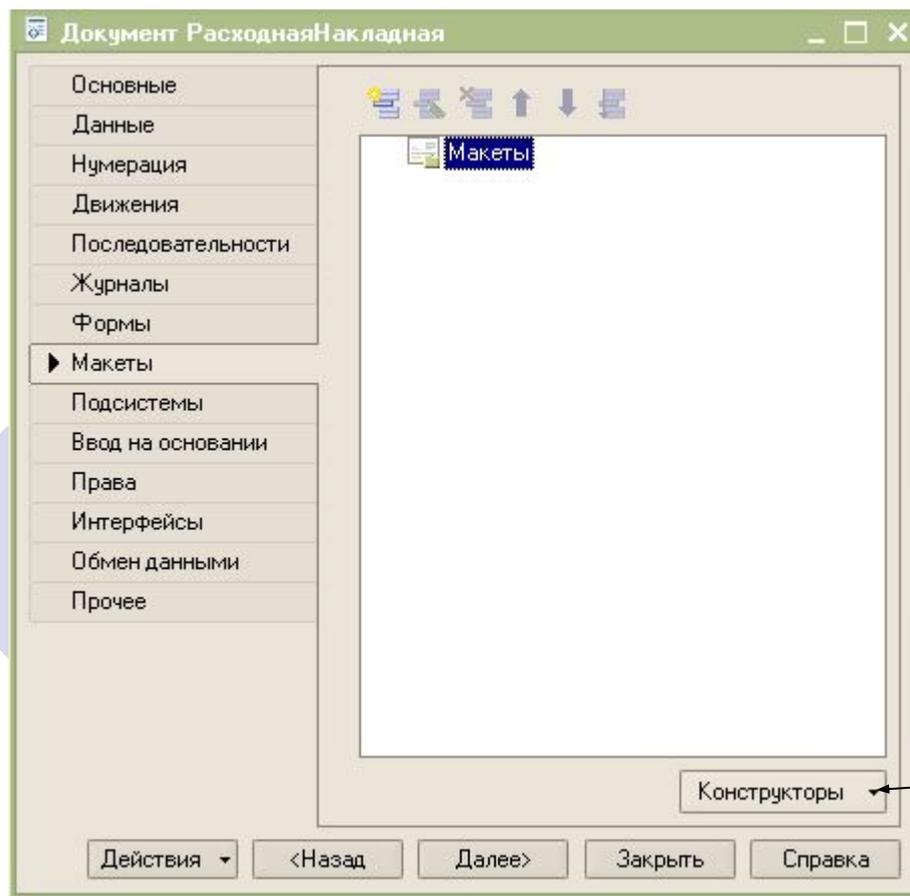
Создание процедуры обработки события

```
Процедура ТабличнаяЧасть1ТоварПриИзменении ( Элемент )
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные ;
    СтрокаТабличнойЧасти.Цена=СтрокаТабличнойЧасти.Товар.Цена ;
    СтрокаТабличнойЧасти.Сумма=СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена ;
    СтрокаТабличнойЧасти.Всего=СтрокаТабличнойЧасти.Сумма ;
    |
КонецПроцедуры

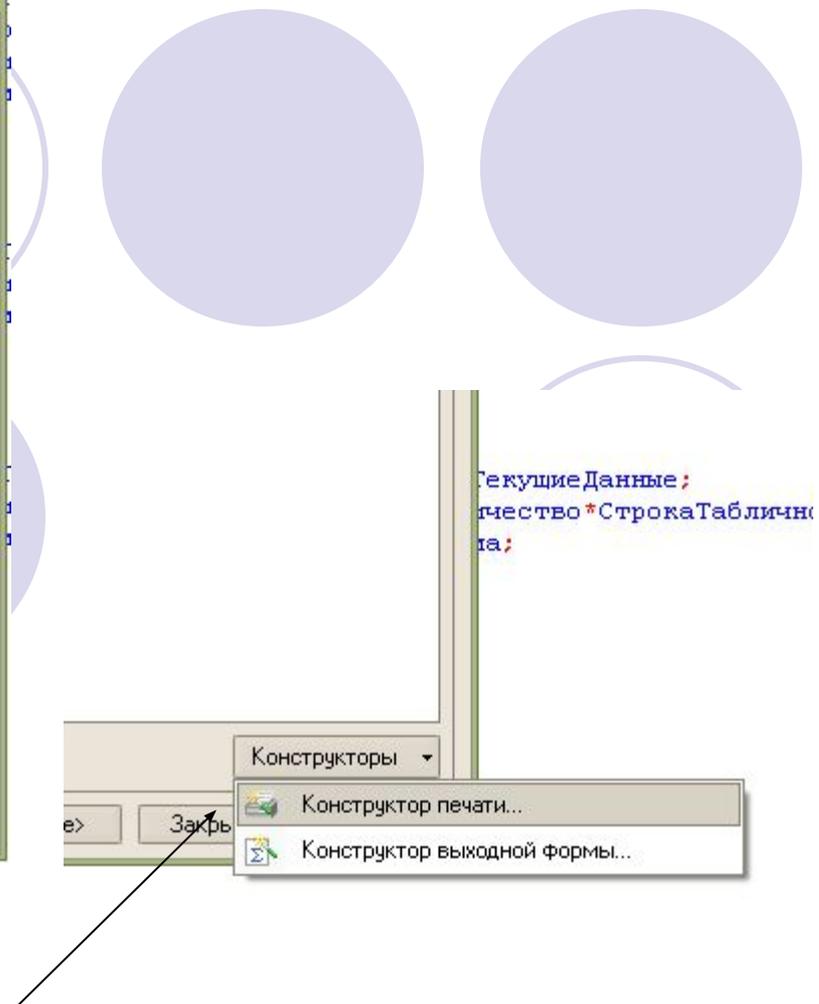
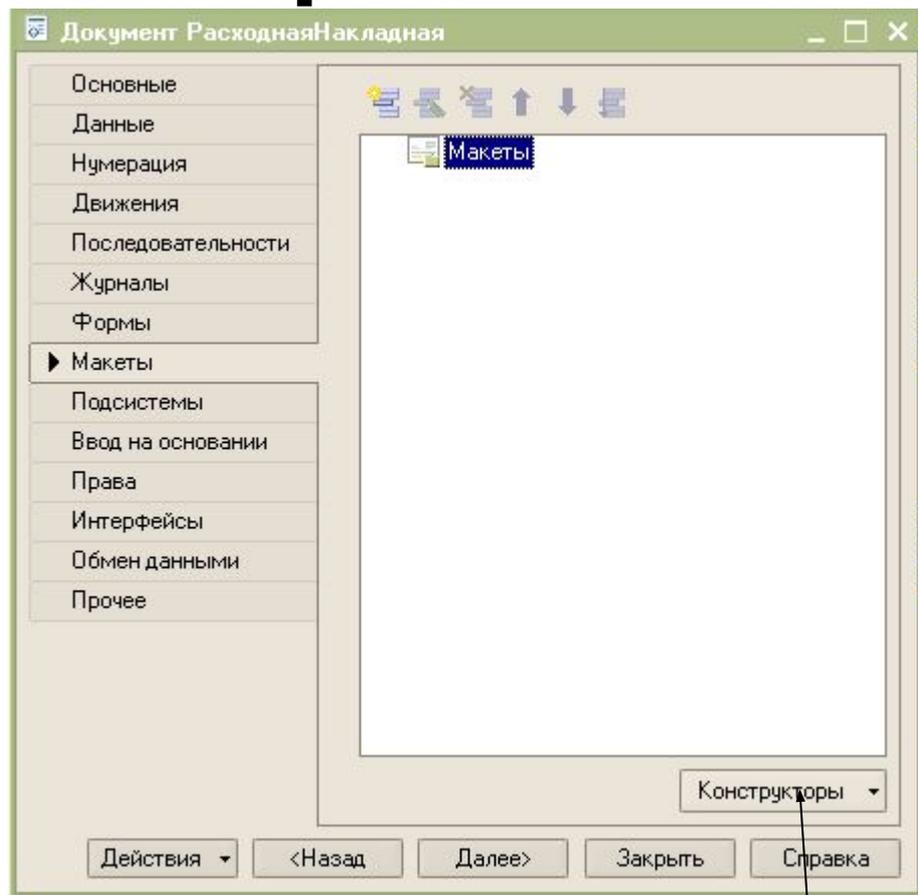
Процедура ТабличнаяЧасть1КоличествоПриИзменении ( Элемент )
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные ;
    СтрокаТабличнойЧасти.Сумма=СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена ;
    СтрокаТабличнойЧасти.Всего=СтрокаТабличнойЧасти.Сумма ;
    |
КонецПроцедуры

Процедура ТабличнаяЧасть1ЦенаПриИзменении ( Элемент )
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные ;
    СтрокаТабличнойЧасти.Сумма=СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена ;
    СтрокаТабличнойЧасти.Всего=СтрокаТабличнойЧасти.Сумма ;
    |
КонецПроцедуры
Покупатель=Константы.ОсновнойПокупатель.Получить ( ) ;
```

Разработка макета документа



Разработка макета документа



Разработка макета документа

Конструктор печати документа

Создать новую процедуру

в модуле:

в модуле формы: ФормаДокумента

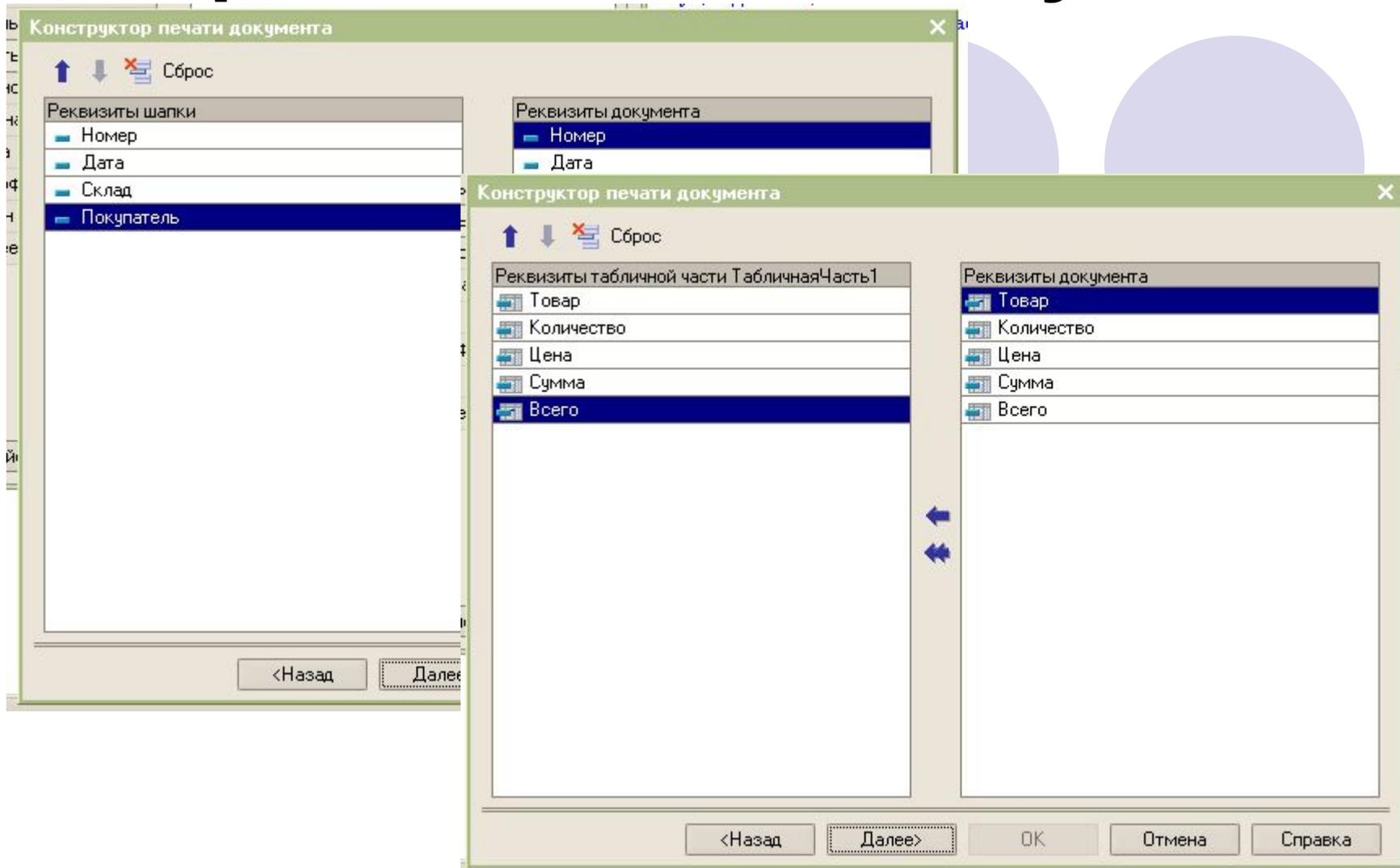
Имя процедуры: Печать

Использовать найденную процедуру

Местоположение	Процедуры
----------------	-----------

<Назад Далее> ОК Отмена Справка

Разработка макета документа



Разработка макета документа

Конструктор печати документа

Вызов процедуры

Не вставлять вызов процедуры

Новая кнопка

Выбрать кнопку

Печатать без предварительного просмотра

Таблица "только просмотр"

Защита таблицы

<Назад Далее> ОК Отмена Справка

Разработка макета документа

	1	2	3	4	5	6	7	8	9	10	11	12
Заголовок	1	Расходная накладная										
	2											
	3											
Блок	4											
	5	Номер	<Номер>									
	6	Дата	<Дата>									
	7	Склад	<Склад>									
	8	Покупатель	<Покупатель>									
	9											
	10											
Общая	11											
	12	Товар	Количество	Цена	Сумма							
Общая	13	<Товар>	<Количество>	<Цена>	<Сумма>							
	14											
	15											
	16											
	17											
	18											
	19											
	20											
	21											
	22											
	23											
	24											
	25											
	26											
	27											
	28											
	29											
	30											
	31											
	32											
	33											
	34											
	35											
	36											

Свойства: Ячейки

Ячейки

▼ Основные:

Текст

Размещение Авто

Имя R18C6

Защита

Гиперссылка

▼ Макет:

Заполнение Текст

Парам. расшифровки

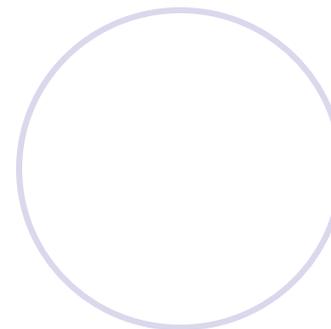
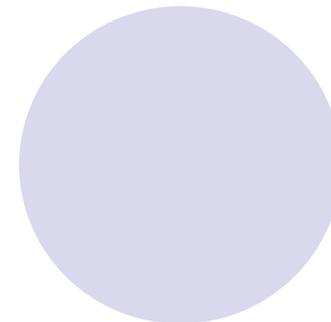
Использование расшифровки Ячейка

Разработка подвала

The image shows a spreadsheet application window with a form for 'Расходная накладная' (Expense Receipt). The form is structured as follows:

1	2	3	4	5	6	7	8	
Заголовок	1	Расходная накладная						
	2							
	3							
Шапка	4							
	5	Номер	<Номер>					
	6	Дата	<Дата>					
	7	Склад	<Склад>					
	8	Покупатель	<Покупатель>					
	9							
	10							
Табличная	11							
	12	Товар	Количество	Цена	Сумма	Всего		
Табличная	13	<Товар>	<Количество>	<Цена>	<Сумма>	<Всего>		
	14							
Подвал	15	Всего на сумму:	<Итог>					
	16							
	17							
	18							
	19							
	20							
	21							
	22							
	23							
	24							
	25							
	26							
	27							
	28							
	29							
	30							
	31							

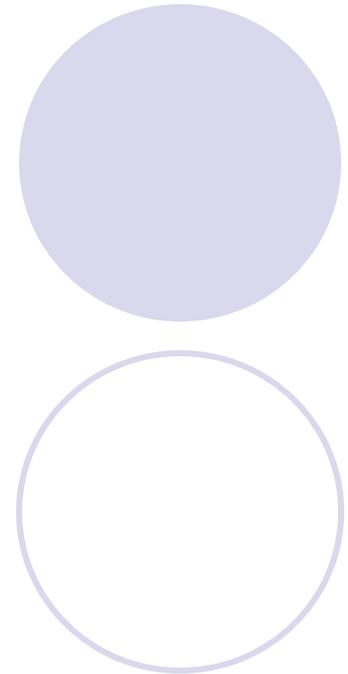
A dialog box titled 'Имя' (Name) is open, showing a text input field with 'Область1' and buttons for 'OK' and 'Отмена' (Cancel).



Разработка подвала

```
□ Процедура Печать (Элемент)
    ТабДок = Новый ТабличныйДокумент;
    Макет = Документы.РасходнаяНакладная.ПолучитьМакет ("Печать");
    // Заголовок
    Область = Макет.ПолучитьОбласть ("Заголовок");
    ТабДок.Вывести (Область);
    // Шапка
    Область = Макет.ПолучитьОбласть ("Шапка");
    Область.Параметры.Номер = Номер;
    Область.Параметры.Дата = Дата;
    Область.Параметры.Склад = Склад;
    Область.Параметры.Покупатель = Покупатель;
    ТабДок.Вывести (Область);
    // ТабличнаяЧасть1
    Область = Макет.ПолучитьОбласть ("ТабличнаяЧасть1Шапка");
    ТабДок.Вывести (Область);
    Итог=0;
    Для Каждого ТекСтрокаТабличнаяЧасть1 Из ТабличнаяЧасть1 Цикл
        Область = Макет.ПолучитьОбласть ("ТабличнаяЧасть1");
        Область.Параметры.Товар = ТекСтрокаТабличнаяЧасть1.Товар;
        Область.Параметры.Количество = ТекСтрокаТабличнаяЧасть1.Количество;
        Область.Параметры.Цена = ТекСтрокаТабличнаяЧасть1.Цена;
        Область.Параметры.Сумма = ТекСтрокаТабличнаяЧасть1.Сумма;
        Область.Параметры.Всего = ТекСтрокаТабличнаяЧасть1.Всего;
        ТабДок.Вывести (Область);
        Итог=Итог+ТекСтрокаТабличнаяЧасть1.Всего;
    КонецЦикла;
    Область = Макет.ПолучитьОбласть ("Подвал");
    Область.Параметры.Итог = Итог;
    ТабДок.Вывести (Область);

    ТабДок.ОтображатьСетку = Ложь;
    ТабДок.Защита = Ложь;
    ТабДок.ТолькоПросмотр = Истина;
    ТабДок.ОтображатьЗаголовки = Ложь;
    ТабДок.Показать ();
```



Редактирование формы документа

Расходная накладная: Расходная накладная 1 ...:42

Действия ▾ Перейти ▾ ?

Номер:

Дата: 29.05.2008 15:57:4

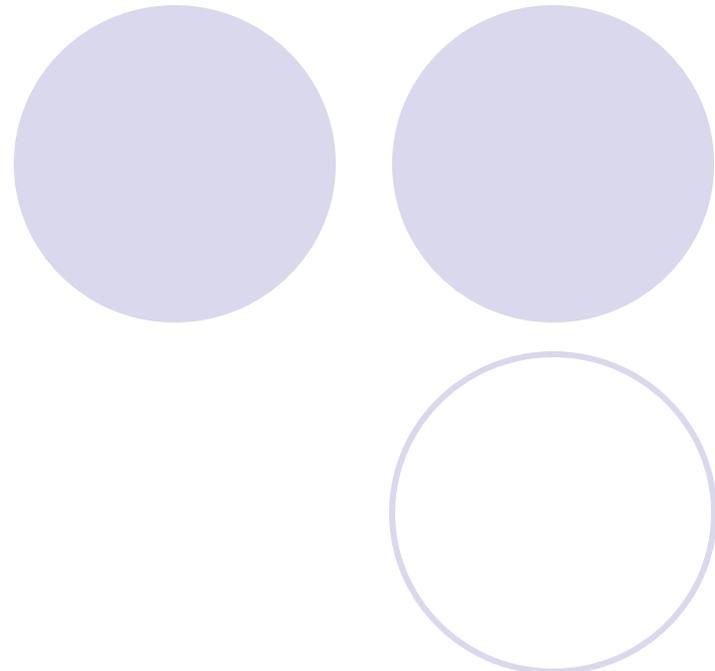
Склад: Склад №1 ... x

Покупатель: Покупатель ... x

N	Товар	Количество	Цена	Сумма	Всего
1	Кефир	1	45,00	45,00	45,
2	Молоко	1	45,00	45,00	45,
Всего:				90,00	

< >

OK Записать Закреть Печать



Редактирование формы документа

Конфигуратор - Конфигурация

Файл Правка Форма Конфигурация Отладка Администрирование Сервис Окна Справка

Конфигурация

Действия

Конфигурация

- Общие
- Константы
- Справочники
- Документы
 - Нумераторы
 - Последовательности
 - ПриходнаяНакладная
 - РасходнаяНакладная
 - Реквизиты
 - Табличные части
 - ТабличнаяЧасть1
 - Формы
 - ФормаДокумента**
- Макеты
- Журналы документов
- Перечисления
- Отчеты
- Обработки
- Планы видов характеристик
- Планы счетов
- Планы видов расчета
- Регистры сведений
- Регистры накопления
- Регистры бухгалтерии
- Регистры расчета
- Бизнес-процессы
- Задачи

Документ РасходнаяНакладная: ФормаДокумента

Расходная накладная

Действия

Номер: 0

Дата: . . . :

Склад: ... X

Попкупатель: ... X

N	Товар	Количество	Цена	Сумма
	<Поле ввода>	<Поле ввода>	<Поле ввода>	<Поле

Всего:

OK Записать Закрыть Печать

Свойства: Табличное поле

Основные:

Имя: ТабличнаяЧасть1

Данные: ТабличнаяЧасть1

Тип значения: ДокументТабличнаяЧастьРасходнаяНакладн...

Изменяет данные:

Пропускать при вводе:

Первый в группе:

Видимость:

Доступность:

Авто контекстное меню:

Контекстное меню: Отсутствует

Использование:

Только просмотр:

Изменять состав строк:

Разрешить начало перетаскивания:

Разрешить перетаскивание:

Изменять настройку колонок:

Изменять позицию колонок:

Изменять порядок строк:

Проверка отображения новой стр: Сообщать о несоответствии

Характеристики:

Режим ввода строк: В конце списка

Режим выделения: Единичный

Режим выделения строки: Ячейка

Шапка:

Подвал:

Высота шапки: 1

Высота подвала: 1

Гор. полоса прокрутки: Использовать автоматически

Верт. полоса прокрутки: Использовать автоматически

Фиксация слева: 0

Фиксация справа: 0

Авто - ввод новой строки:

Нач. отображение списка: Начало

Подсказка: Попкупатель

Оформление:

Цвет фона поля: Авто

Цвет текста поля: Авто

Для получения подсказки нажмите F1

ПУСК | Конфигуратор - Кон... | 1С:Предприятие - К... | 1с8 | Microsoft PowerPoint ... | Corel PHOTO-PAINT ... | CAP NUM ru +8.157 384 x 102 | 6:39

Редактирование формы документа

Текст шапки Сумма

Текст подвала

Гиперссылка

▼ **Использование:**

Автоотметка незаполненного

Выделять отрицательные

▼ **Характеристики:**

Положение Новая колонка

Подсказка в шапке

Горизонт. положение в шапке Прижать влево

Горизонтальное положение Автоматически

Горизонт. положение в подвале Прижать вправо

Отображать в шапке

Отображать в подвале

Режим редактирования Ввод

Активизировать по умолчанию

Изменение размера колонки Изменять

Изменять видимость

Изменять настройку

Изменять позицию

Показывать итог в подвале

Пропускать при вводе

Формат

Высота ячейки 1

Автовысота ячейки

▼ **Оформление:**

Ширина 100

Цвет фона поля Авто

Цвет текста поля Авто

Цвет фона шапки Авто

Цвет текста шапки Авто

Цвет фона подвала Авто

Цвет текста подвала Авто

Шрифт текста Авто

Шрифт шапки Авто

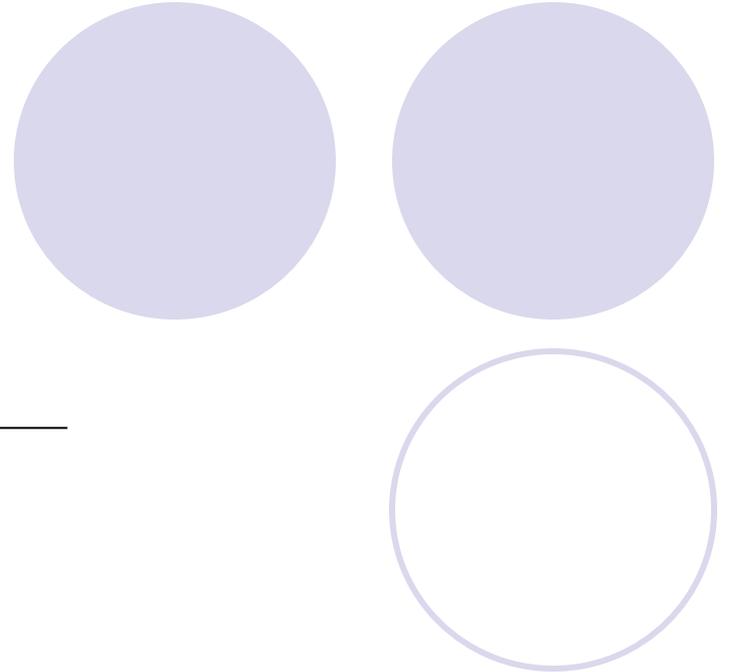
Шрифт подвала стиль: Шрифт текста

Картинка шапки

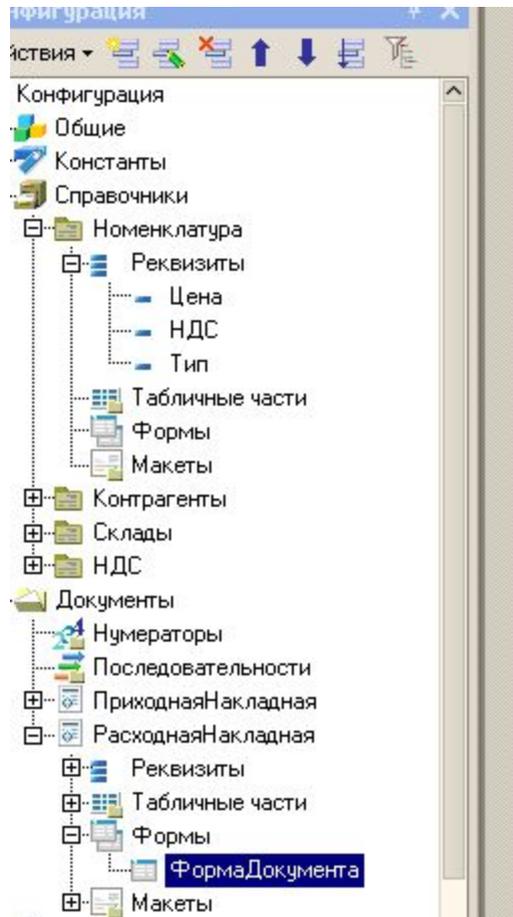
Картинки строк

Картинка подвала

документа



Изменение модуля документов

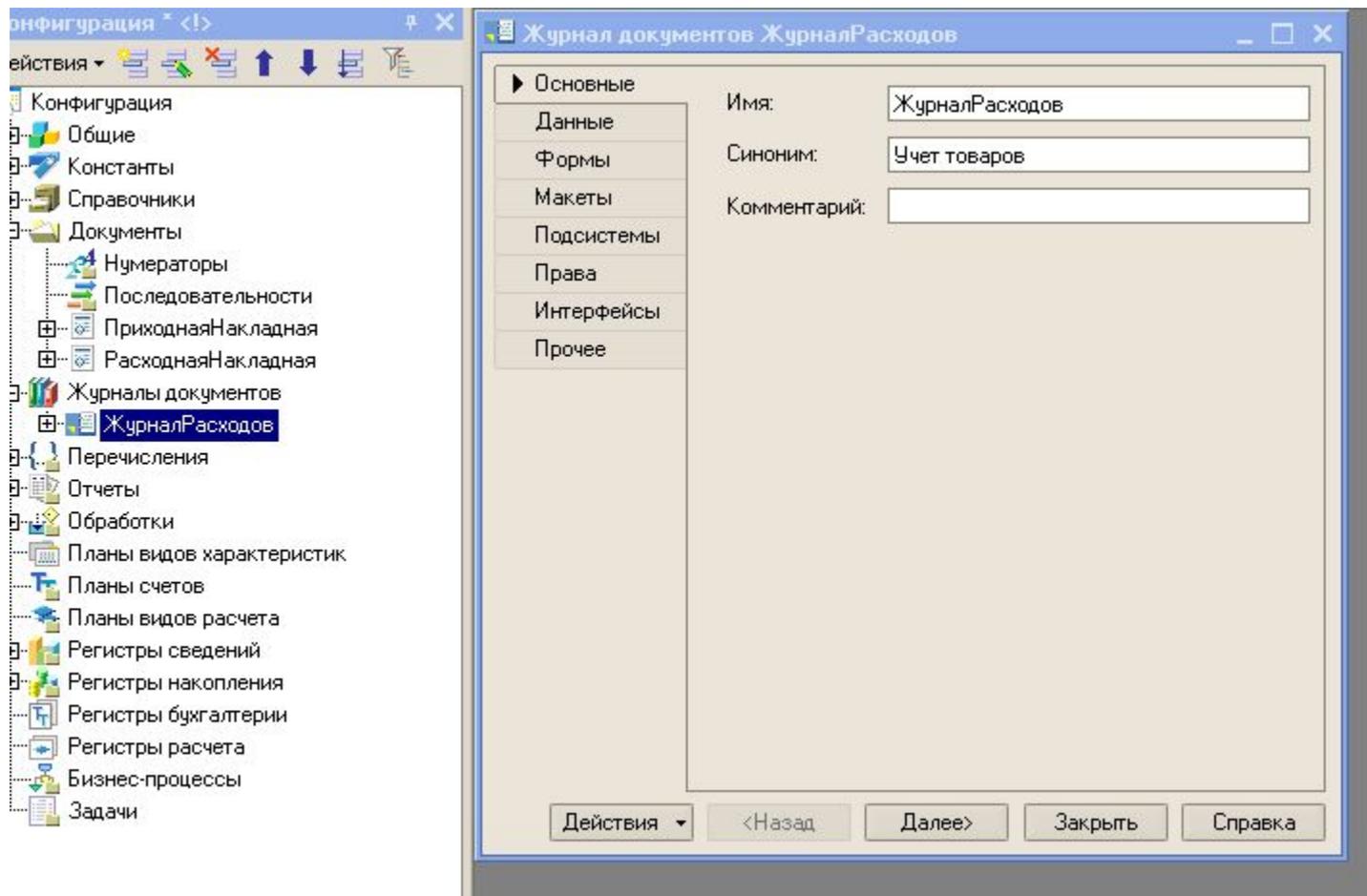


```
Процедура Пересчет(СТЧ) [...]
Процедура ТабличнаяЧасть1ТоварПриИзменении(Элемент)

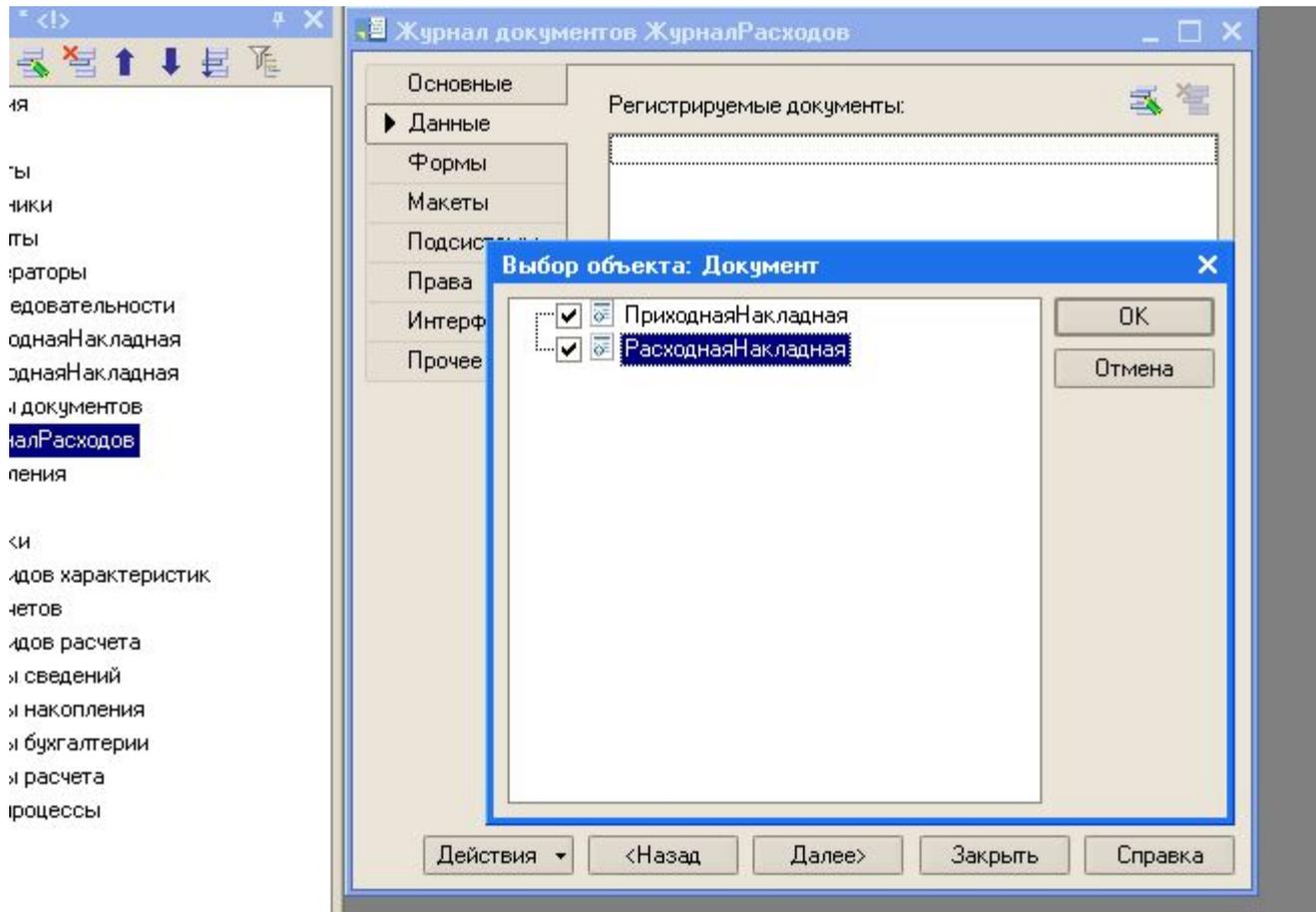
    СтрокаТабличнойЧасти=ЭлементыФормы.ТабличнаяЧасть1.ТекущиеДанные;
    Если СтрокаТабличнойЧасти.Товар.Тип<>Перечисления.ВидыНоменклатуры.Товар Тогда
        Предупреждение(СтрокаТабличнойЧасти.Товар.Наименование+"-это не товар",3);
        СтрокаТабличнойЧасти.Товар="";
        Возврат;
    КонецЕсли;
    СтрокаТабличнойЧасти.Цена=РозничнаяЦена(Дата,Элемент,Значение);
    СтрокаТабличнойЧасти.СтавкаНДС=СтрокаТабличнойЧасти.Товар.НДС;
    Пересчет(СтрокаТабличнойЧасти);
КонецПроцедуры

Процедура ТабличнаяЧасть1КоличествоПриИзменении(Элемент) [...]
Процедура ТабличнаяЧасть1ЦенаПриИзменении(Элемент) [...]
Процедура Печать(Элемент) [...]
    Покупатель=Константы.ОсновнойПокупатель.Получить();
```

Разработка журнала учета документов



Разработка журнала учета документов



Разработка журнала учета документов

Основные
Данные
▶ Формы
Макеты
Подсистемы
Права
Интерфейсы
Прочее

Основная форма журнала:
... x Q

Конструктор формы журнала документов

Выберите тип формы:

- Форма журнала документов
- Произвольная форма

Назначить форму основной

Имя: ФормаСписка

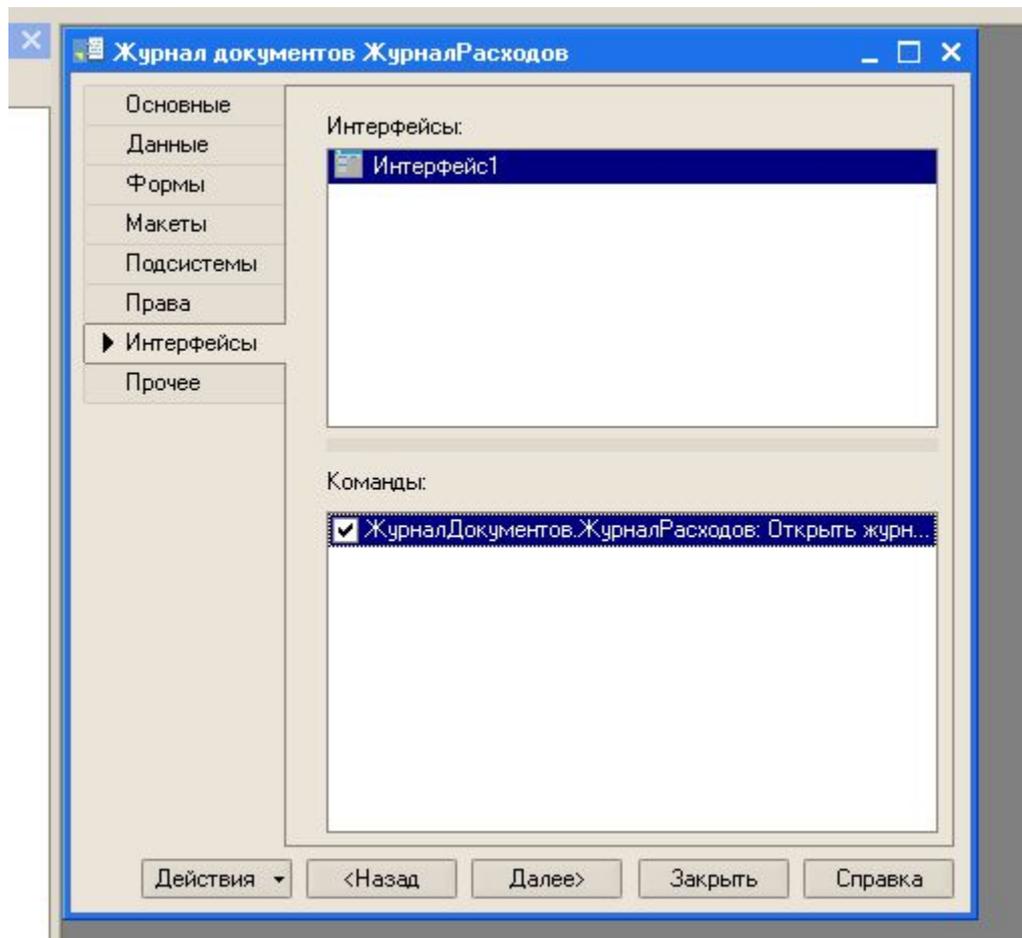
Синоним: Форма списка

Комментарий:

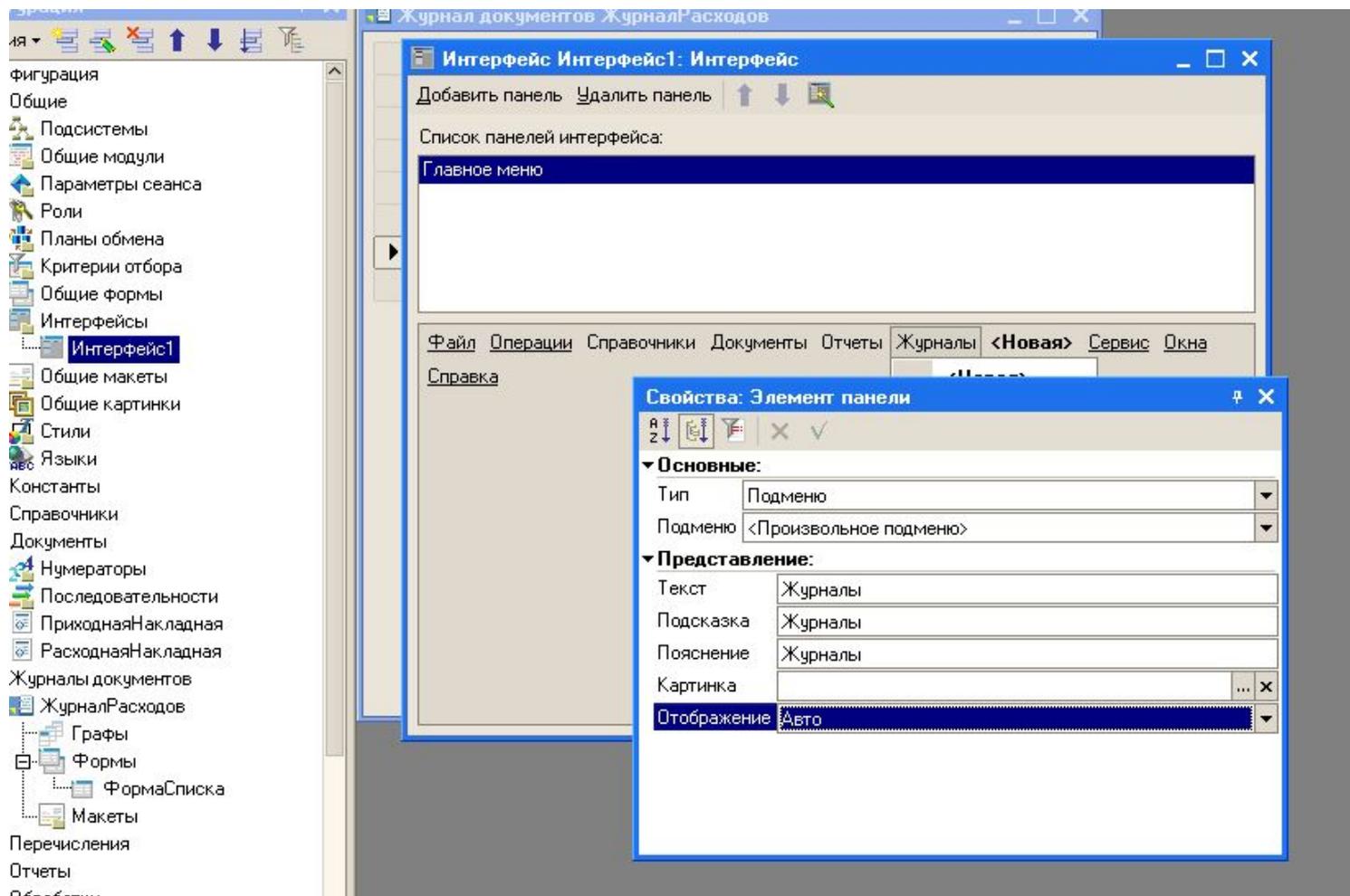
Командная панель формы сверху
 Командная панель формы снизу

< Назад Далее > Готово Отмена Справка

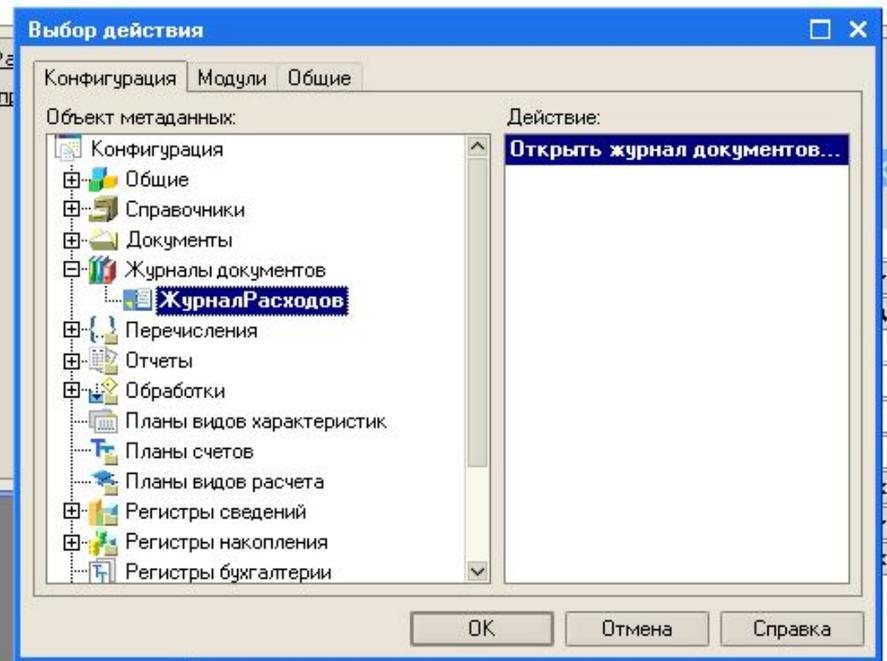
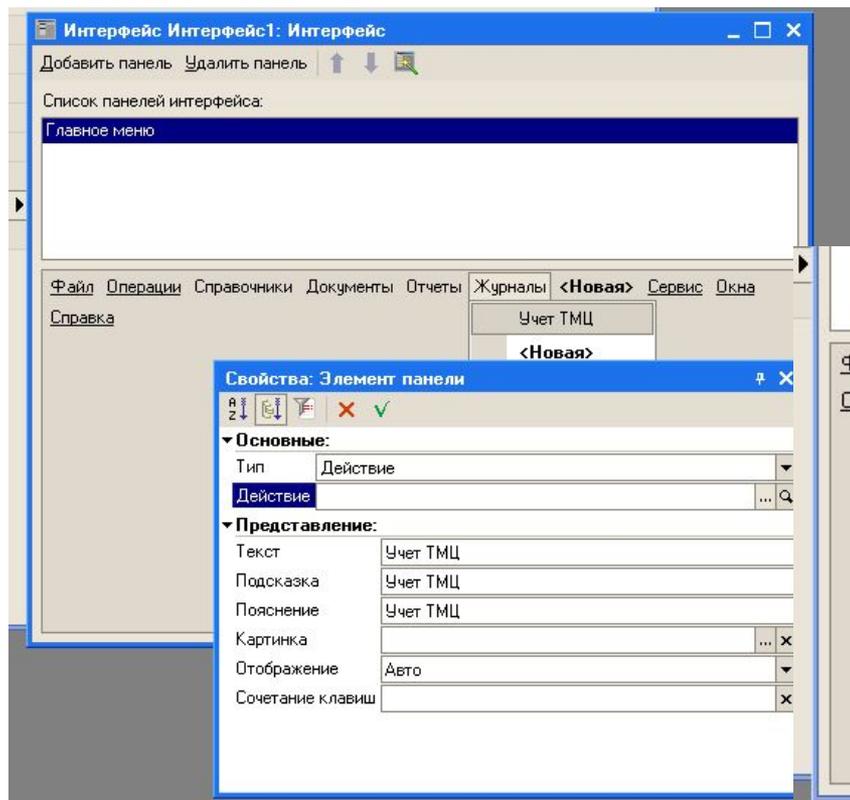
Разработка журнала учета документов



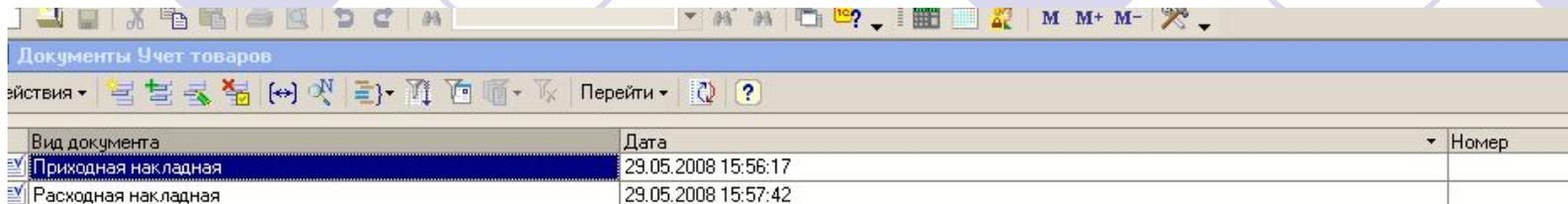
Разработка журнала учета документов



Разработка журнала учета документов

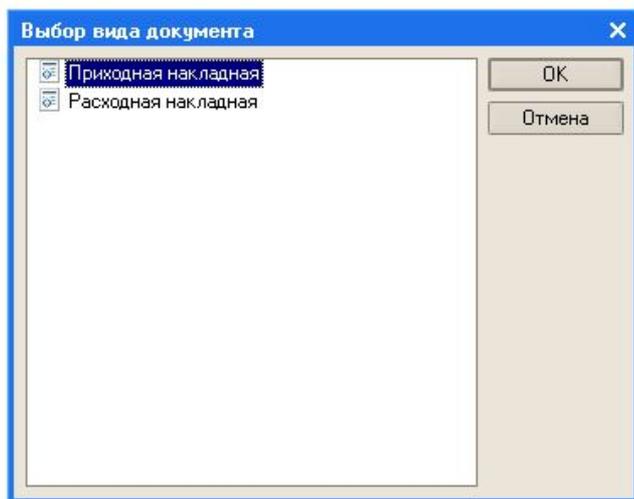


Проверка работы журнала

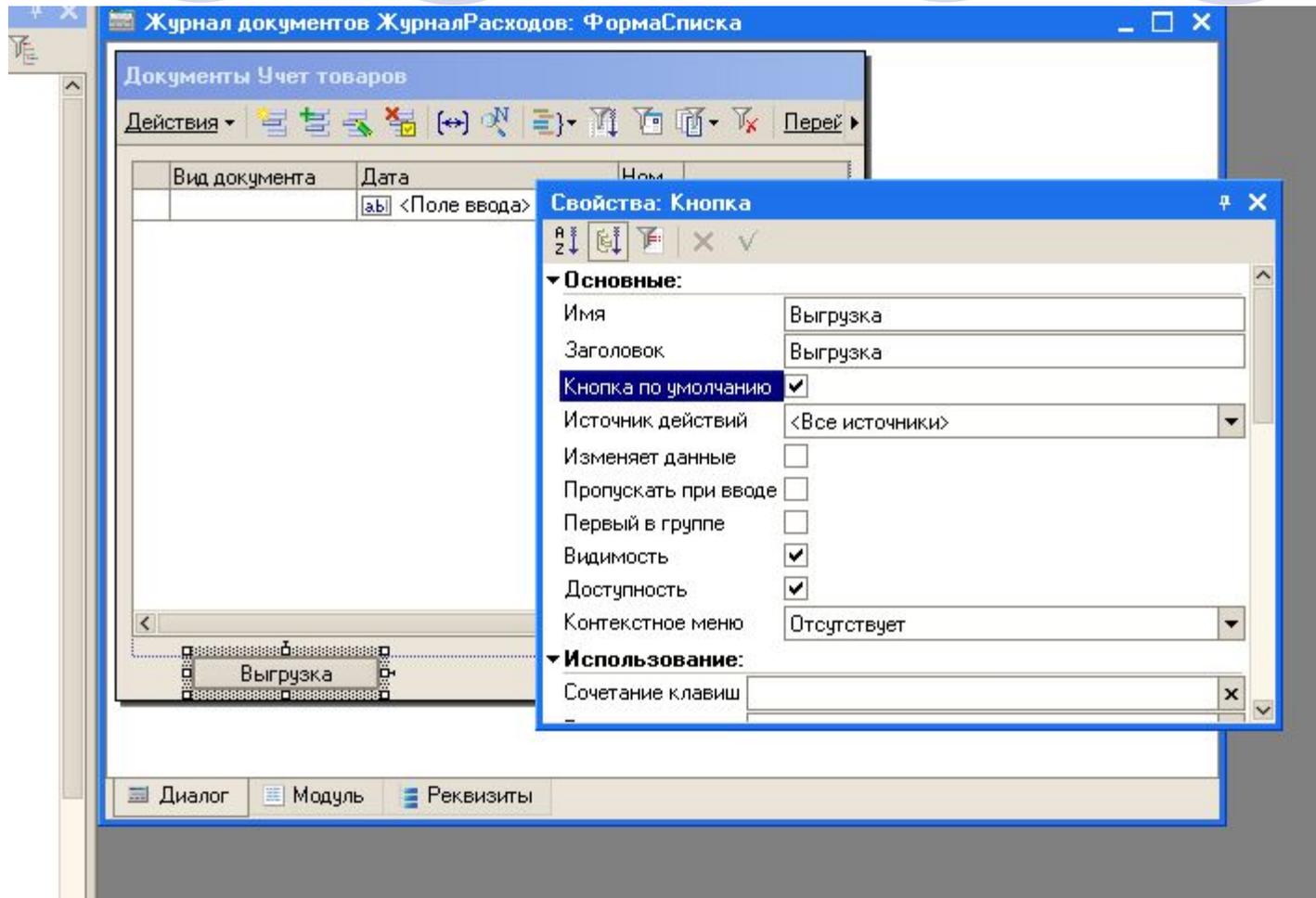


The screenshot shows a software window titled "Документы Учет товаров". Below the title bar is a menu bar with "ействия" and a "Перейти" button. Below the menu bar is a table with the following data:

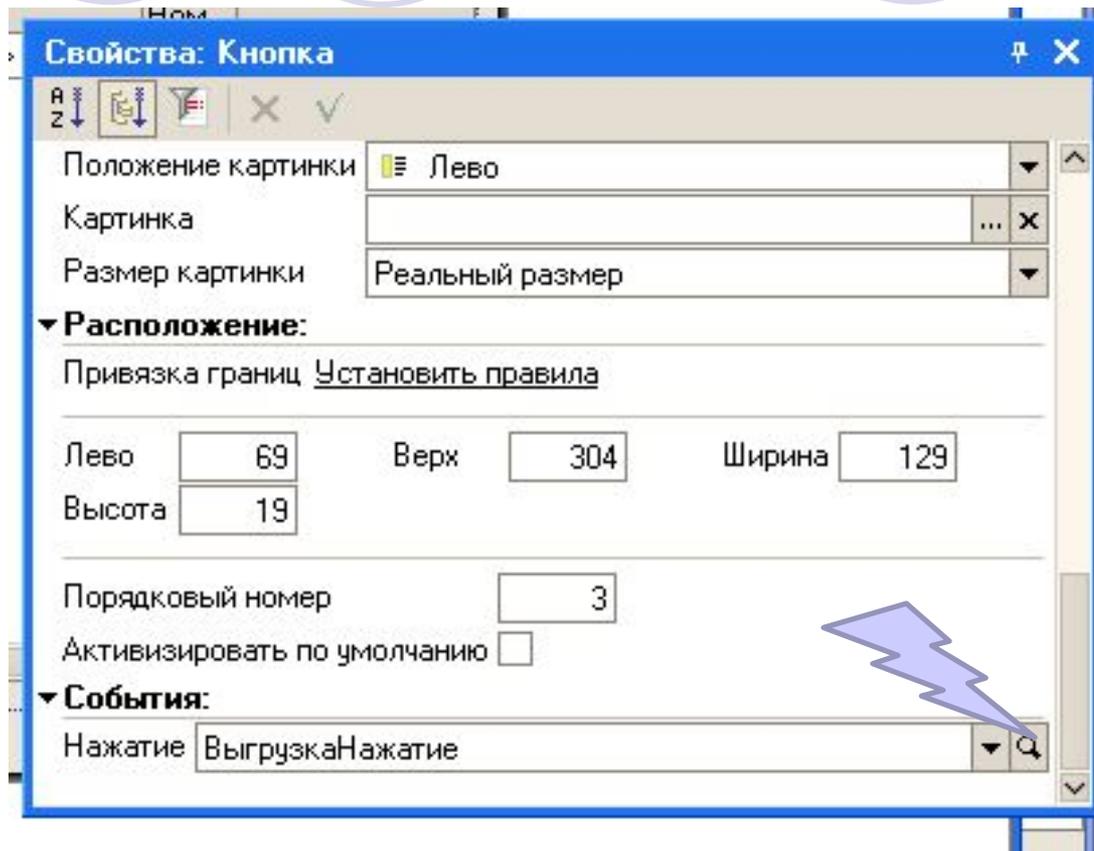
Вид документа	Дата	Номер
Приходная накладная	29.05.2008 15:56:17	
Расходная накладная	29.05.2008 15:57:42	



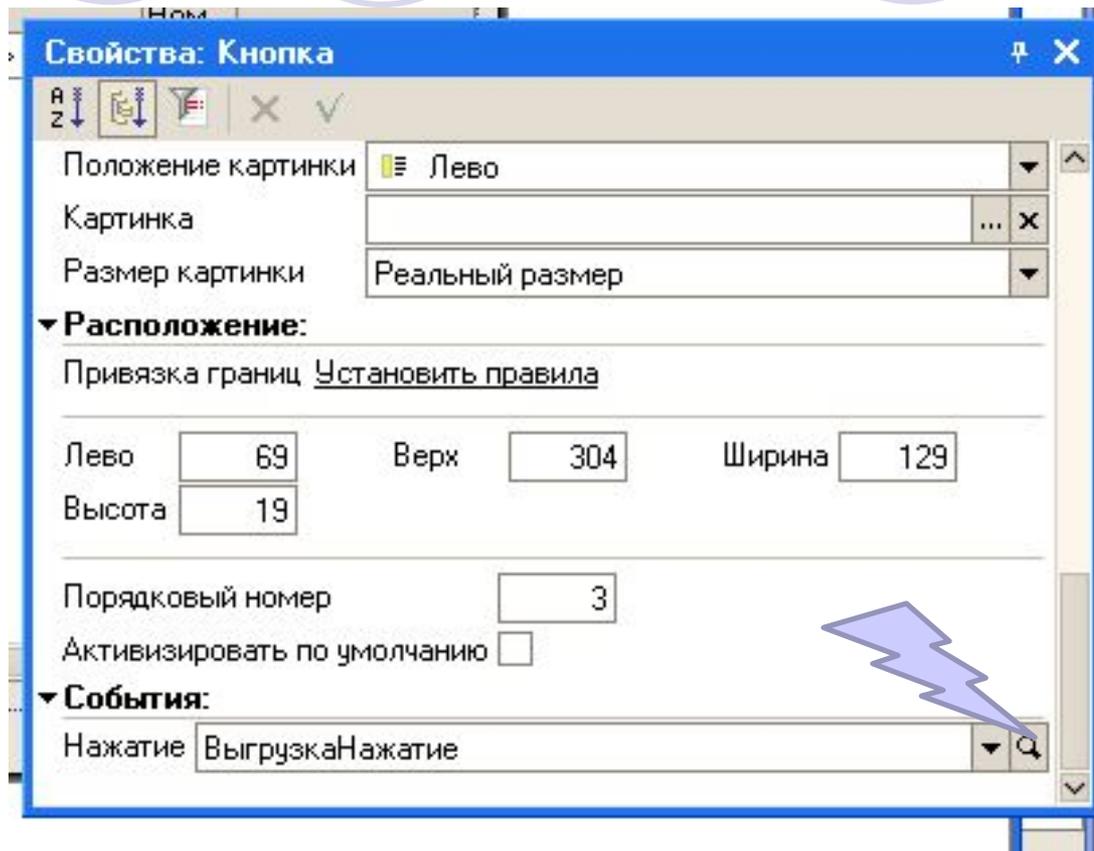
Выгрузка документов



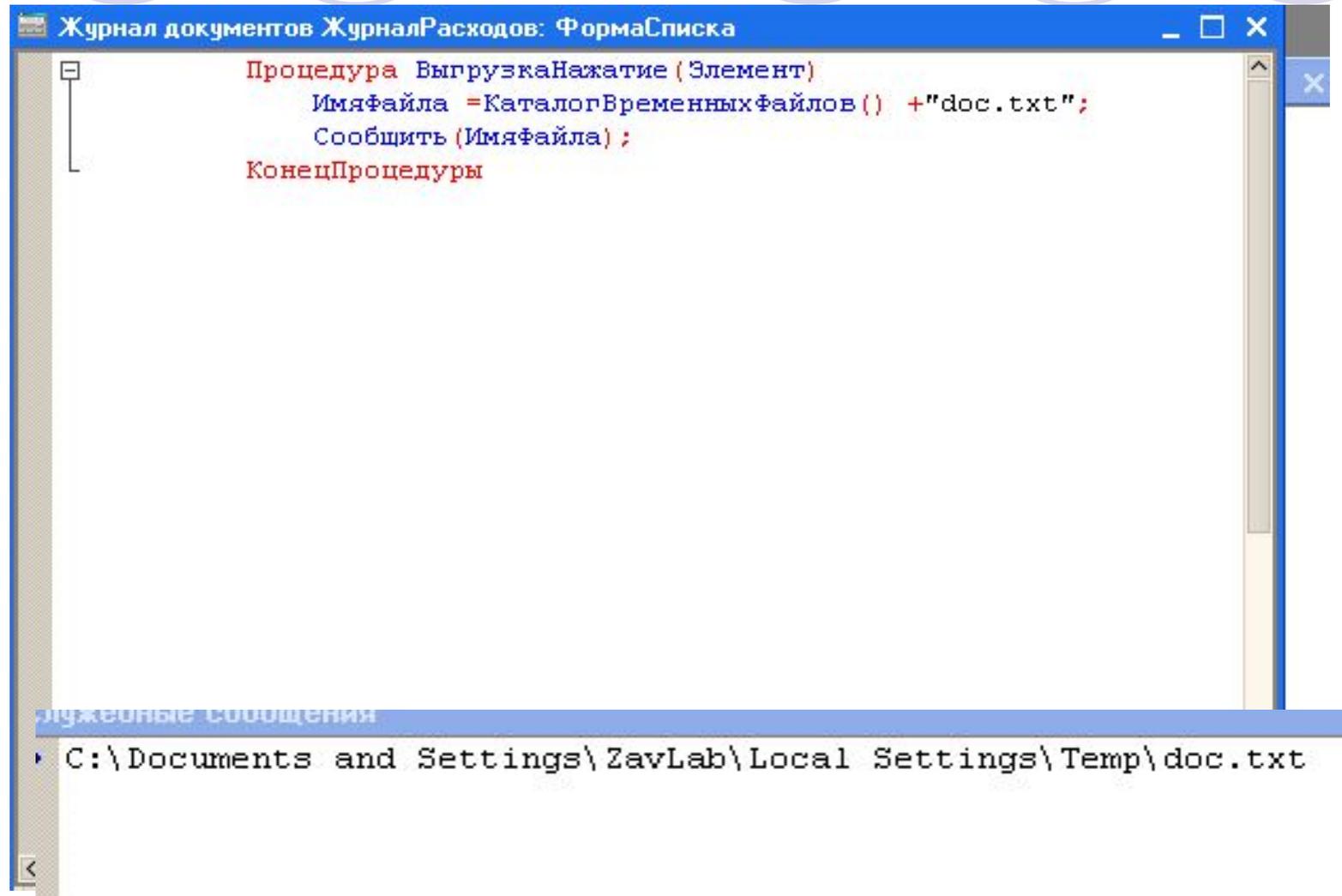
Выгрузка документов



Выгрузка документов



Выгрузка документов



```
Процедура ВыгрузкаНажатие (Элемент)  
    Имяфайла =КаталогВременныхфайлов() +"doc.txt";  
    Сообщить (Имяфайла) ;  
КонiecПроцедуры
```

лучебные сообщения

- C:\Documents and Settings\ZavLab\Local Settings\Temp\doc.txt

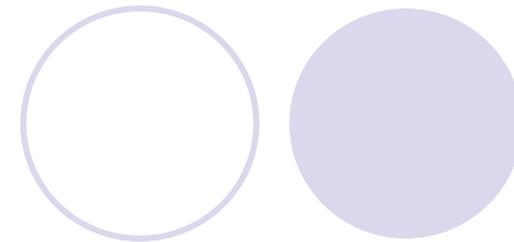
Выгрузка документов

```
Процедура ВыгрузкаНажатие (Элемент)  
    Имяфайла =КаталогВременныхфайлов() +"doc.txt";  
    Сообщить (Имяфайла) ;  
КонiecПроцедуры
```

лучебные сообщения

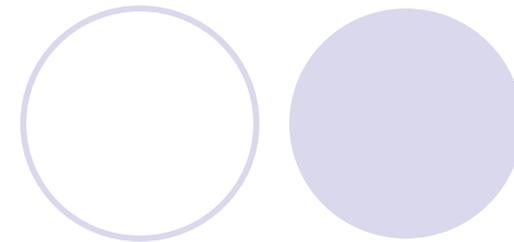
▸ C:\Documents and Settings\ZavLab\Local Settings\Temp\doc.txt

Выгрузка документов



```
Процедура ВыгрузкаНажатие (Элемент)
    Имяфайла =КаталогВременныхфайлов() +"doc.txt";
    Выбфайл = Новый ТекстовыйДокумент;
    Выбфайл.УстановитьТипфайла (КодировкаТекста.UTF8);
    Выборка=Документы.РасходнаяНакладная.Выбрать (НачалоНедели (РабочаяДата) ,КонецНедели (РабочаяДата) );
    Пока Выборка.Следующий () Цикл
        Док=Выборка.ПолучитьОбъект ();
        Сообщить (Док.Дата)
    КонецЦикла
КонецПроцедуры
```

Выгрузка документов



```
Процедура ВыгрузкаНажатие (Элемент)
    Имяфайла =КаталогВременныхфайлов() +"doc.txt";
    Выбфайл = Новый ТекстовыйДокумент;
    Выбфайл.УстановитьТипфайла (КодировкаТекста.UTF8);
    Выборка=Документы.РасходнаяНакладная.Выбрать (НачалоНедели (РабочаяДата) ,КонецНедели (РабочаяДата) );
    Пока Выборка.Следующий () Цикл
        Док=Выборка.ПолучитьОбъект ();
        Сообщить (Док.Дата)
    КонецЦикла
КонецПроцедуры
```

Выгрузка документов

Процедура ВыгрузкаНажатие (Элемент)

```
Имяфайла =КаталогВременныхФайлов() +"doc.txt";
```

```
Выбфайл = Новый ТекстовыйДокумент;
```

```
Выбфайл.УстановитьТипфайла(КодировкаТекста.UTF8);
```

```
Выборка=Документы.РасходнаяНакладная.Выбрать(НачалоНедели(РабочаяДата),КонецНедели(Рс
```

```
Пока Выборка.Следующий() Цикл
```

```
    Док=Выборка.ПолучитьОбъект();
```

```
    Если Док.Проведен=Истина Тогда
```

```
        Сообщить ("Ø");
```

```
        Сообщить (Док.Дата);
```

```
        Сообщить (Док.Номер);
```

```
        Сообщить (Док.Склад.Наименование);
```

```
        Сообщить (Док.Склад.Код);
```

```
        Сообщить (Док.Покупатель.Наименование);
```

```
        Сообщить (Док.Покупатель.Код);
```

```
    КонецЕсли;
```

```
КонецЦикла;
```

```
КонецПроцедуры
```

Выгрузка документов

Процедура ВыгрузкаНажатие (Элемент)

```
Имяфайла =КаталогВременныхФайлов() +"doc.txt";
```

```
Выбфайл = Новый ТекстовыйДокумент;
```

```
Выбфайл.УстановитьТипфайла(КодировкаТекста.UTF8);
```

```
Выборка=Документы.РасходнаяНакладная.Выбрать(НачалоНедели(РабочаяДата),КонецНедели(Рс
```

```
Пока Выборка.Следующий() Цикл
```

```
    Док=Выборка.ПолучитьОбъект();
```

```
    Если Док.Проведен=Истина Тогда
```

```
        Сообщить ("@" );
```

```
        Сообщить (Док.Дата);
```

```
        Сообщить (Док.Номер);
```

```
        Сообщить (Док.Склад.Наименование);
```

```
        Сообщить (Док.Склад.Код);
```

```
        Сообщить (Док.Покупатель.Наименование);
```

```
        Сообщить (Док.Покупатель.Код);
```

```
    КонецЕсли;
```

```
КонецЦикла;
```

```
КонецПроцедуры
```

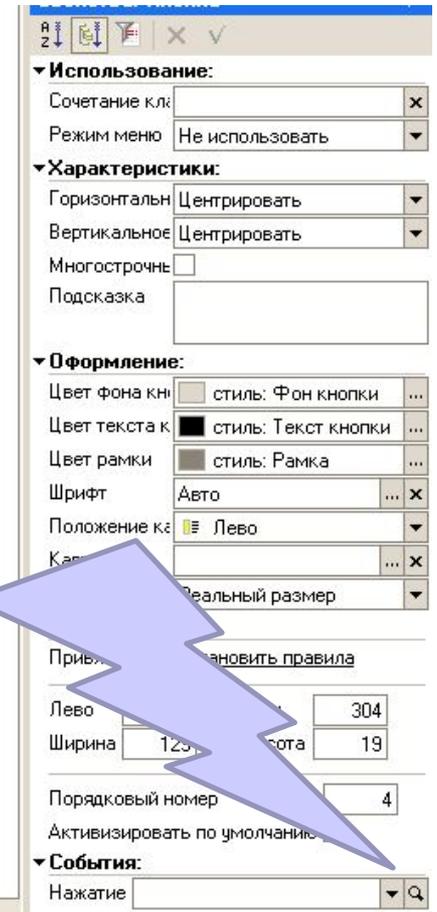
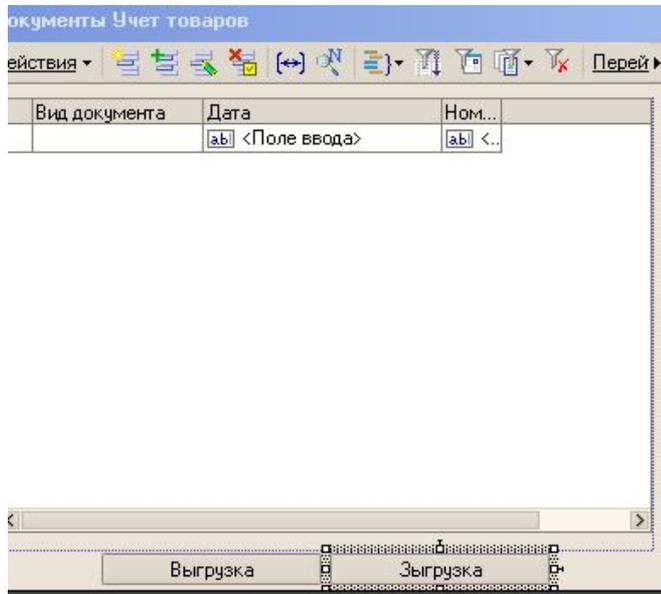
Выгрузка документов

```
□ Процедура ВыгрузкаНажатие (Элемент)
    Имяфайла =КаталогВременныхфайлов() +"doc.txt";
    Выбфайл = Новый ТекстовыйДокумент;
    Выбфайл.УстановитьТипФайла(КодировкаТекста.UTF8);
    Выборка=Документы.РасходнаяНакладная.Выбрать(НачалоНедели(РабочаяДата),КонецНедели(Ра
Пока Выборка.Следующий() Цикл
    Док=Выборка.ПолучитьОбъект();
    Если Док.Проведен=Истина Тогда
        Сообщить("␣");
        Сообщить(Док.Дата);
        Сообщить(Док.Номер);
        Сообщить(Док.Склад.Наименование);
        Сообщить(Док.Склад.Код);
        Сообщить(Док.Покупатель.Наименование);
        Сообщить(Док.Покупатель.Код);
        Для Каждого ТекСтрока Из Док.ТабличнаяЧасть1 Цикл
            Сообщить(ТекСтрока.Товар.Наименование);
            Сообщить(ТекСтрока.Товар.Код);
            Сообщить(ТекСтрока.Количество);
            Сообщить(ТекСтрока.Сумма);
            Сообщить(ТекСтрока.Всего);
        КонецЦикла;
    КонецЕсли;
КонецЦикла;
КонецПроцедуры
```

Выгрузка документов

```
Процедура ВыгрузкаНажатие (Элемент)
Имяфайла =КаталогВременныхфайлов() +"doc.txt";
Выбфайл = Новый ТекстовыйДокумент;
Выбфайл.УстановитьТипфайла(КодировкаТекста.UTF8);
Выборка=Документы.РасходнаяНакладная.Выбрать(НачалоНедели(РабочаяДата),КонецНедели(РабочаяДата));
Пока Выборка.Следующий() Цикл
    Док=Выборка.ПолучитьОбъект();
    Если Док.Проведен=Истина Тогда
        Выбфайл.ДобавитьСтроку("");
        Выбфайл.ДобавитьСтроку(Док.Дата);
        Выбфайл.ДобавитьСтроку(Док.Номер);
        Выбфайл.ДобавитьСтроку(Док.Склад.Наименование);
        Выбфайл.ДобавитьСтроку(Док.Склад.Код);
        Выбфайл.ДобавитьСтроку(Док.Покупатель.Наименование);
        Выбфайл.ДобавитьСтроку(Док.Покупатель.Код);
        Для Каждого ТекСтрока Из Док.ТабличнаяЧасть1 Цикл
            Выбфайл.ДобавитьСтроку(ТекСтрока.Товар.Наименование+";"+ТекСтрока.Товар.Код+";"+ТекСтрока.Количество);
        КонецЦикла;
    КонецЕсли;
КонецЦикла;
Сообщить(Имяфайла);
Выбфайл.Записать(Имяфайла);
КонецПроцедуры
```

Загрузка документов



Загрузка документов

```
□ Процедура ЗыгрузкаНажатие (Элемент)
  Имяфайла = КаталогВременныхФайлов() + "doc.txt";
  Выбфайл = Новый ТекстовыйДокумент;
  Выбфайл.Прочитать (Имяфайла);
  КолСтр=Выбфайл.КоличествоСтрок ();
  Док=Документы.РасходнаяНакладная;
  Позиция=0; флаг=0;
  Для к=1 По КолСтр Цикл
    Врстр=Выбфайл.ПолучитьСтроку (к);
    Позиция=Найти (Врстр, "0");
    Если Позиция<>0 Тогда
      Если флаг=1 Тогда
        Сообщить ("Документ записан");
        флаг=0;
      КонецЕсли;
      ДатаД=Дата (Выбфайл.ПолучитьСтроку (к+1));
      Сообщить (ДатаД);
      НомерД=Выбфайл.ПолучитьСтроку (к+2);
      Сообщить (НомерД);
      Если Док.НайтиПоНомеру (НомерД, ДатаД) = Док.ПустаяСсылка () Тогда
        Сообщить ("Новый документ");
        флаг=1;
      КонецЕсли;
      к=к+7;
    Иначе
      Если флаг=1 Тогда Сообщить ("Новый документ"); КонецЕсли;
    КонецЕсли;
  КонецЦикла;
  Если флаг=1 Тогда Сообщить ("Документ записан"); флаг=0; КонецЕсли;
КонецПроцедуры
```

Загрузка документов

Процедура ЗыгрузкаНажатие (Элемент)

Перем ДокНов;

Имяфайла = КаталогВременныхФайлов() + "doc.txt";

Выбфайл = Новый ТекстовыйДокумент;

Выбфайл.Прочитать (Имяфайла);

КолСтр=Выбфайл.КоличествоСтрок();

Док=Документы.РасходнаяНакладная;

Позиция=0; флаг=0;

Для к=1 По КолСтр Цикл

 Врстр=Выбфайл.ПолучитьСтроку(к);

 Позиция=Найти(Врстр, "@");

 Если Позиция<>0 Тогда

 Если флаг=1 Тогда

 ДокНов.Записать();

 Сообщить("Документ записан");

 флаг=0;

 КонецЕсли;

 ДатаД=Дата(Выбфайл.ПолучитьСтроку(к+1));

 Сообщить(ДатаД);

 НомерД=Выбфайл.ПолучитьСтроку(к+2);

 Сообщить(НомерД);

 Если Док.НайтиПоНомеру(НомерД, ДатаД) = Док.ПустаяСсылка() Тогда

 ДокНов= Док.СоздатьДокумент();

 ДокНов.Дата=ДатаД;

 ДокНов.Номер=НомерД;

 Сообщить("Новый документ");

Загрузка документов

```
Если док.найтипономеру(номерд, датад) = док.пустаяссылка() Тогда  
ДокНов = Док.СоздатьДокумент();  
ДокНов.Дата = ДатаД;  
ДокНов.Номер = НомерД;  
Сообщить ("Новый документ");  
флаг = 1;  
КонецЕсли;  
к = к + 7;
```

Иначе

```
Если флаг = 1 Тогда Сообщить ("Новый документ"); КонецЕсли;
```

КонецЕсли;

КонецЦикла;

```
Если флаг = 1 Тогда ДокНов.Записать(); Сообщить ("Документ записан"); флаг = 0; КонецЕ
```

КонецПроцедуры

Изменения модуля

```
} Процедура Выход ()
ТекПол=Справочники.Пользователи.НайтиПоНаименованию(ИмяПользователя());
Вр=Число(Сред(ТекущаяДата(),10,2));
    Если (ТекПол.Время<Вр) ИЛИ (ТекПол.Допуск=Ложь) Тогда
        Предупреждение("Завершение работы!",3);
        ЗавершитьРаботуСистемы(Истина);
    КонечЕсли;
КонечПроцедуры

} // Обработчик события УстановкаПараметровСеанса ()
//
} Процедура УстановкаПараметровСеанса(ТребуемыеПараметры)

    Если ТребуемыеПараметры = Неопределено Тогда
        // Раздел установки параметров сеанса при начале сеанса (ИменаПараметровСеанс
        // Выполняется установка параметров сеанса, которые можно инициализировать
        // при начале работы системы

        // инициализация параметра сеанса "ТекущийПользователь"
        ПолныеПрава.УстановитьПараметрСеансаТекущийПользователь();
        ПолныеПрава.УстановитьИзменяемыеПараметрыСеансаПользователя();

    Иначе
        // Установка параметров сеанса "по требованию"

        // Параметры сеанса, инициализация которых требует обращения к одним и тем ж
        // следует инициализировать сразу группой. Для того, чтобы избежать их повтс
        // имена уже установленных параметров сеанса сохраняются в массиве Установле
        УстановленныеПараметры = Новый Массив;
        Для каждого ИмяПараметра Из ТребуемыеПараметры Цикл
            УстановитьЗначениеПараметраСеанса(ИмяПараметра, УстановленныеПараметры);
        КонечЦикла;
    КонечЕсли;
// Сообщить(ИмяПользователя());
ТекПол=Справочники.Пользователи.НайтиПоНаименованию(ИмяПользователя());
Если ТекПол=Справочники.Пользователи.ПустаяСсылка() Тогда
    Возврат;
КонечЕсли;
Вр=Число(Сред(ТекущаяДата(),10,2));
Если (ТекПол.Время<Вр) ИЛИ (ТекПол.Допуск=Ложь) Тогда
    Предупреждение("Допуск закрыт!",3);
    ЗавершитьРаботуСистемы(Ложь);
КонечЕсли;
ПодключитьОбработчикОжидания("Выход",3);
КонечПроцедуры // УстановкаПараметровСеанса
```

Загрузка документов

```
Если      Док.НайтиПоНомеру(НомерД,ДатаД)=Док.ПустаяСсылка() Тогда
    ДокНов= Док.СоздатьДокумент();
    ДокНов.Дата=ДатаД;
    ДокНов.Номер=НомерД;
    // Чтение информации для шапки
    СкладНаименование=ВыбФайл.ПолучитьСтроку(к+3);
    СкладКод=ВыбФайл.ПолучитьСтроку(к+4);
    ПокупательНаименование=ВыбФайл.ПолучитьСтроку(к+5);
    ПокупательКод=ВыбФайл.ПолучитьСтроку(к+6);
    флаг=1;
КонецЕсли;
к=к+6;
```

Загрузка документов

```
ПокупательНаименование=ВыбФайл.ПолучитьСтроку (k+5);
ПокупательКод=ВыбФайл.ПолучитьСтроку (k+6);
//Выбор склада
Ск=Справочники.Склады;
ПС=Ск.НайтиПоКоду(Число(СкладКод));
Если ПС=Ск.ПустаяСсылка() Тогда
    СкНов= Справочники.Склады.СоздатьЭлемент();
    СкНов.Код=СкладКод;
    СкНов.Наименование=СкладНаименование;
    СкНов.Записать();
    ДокНов.Склад=СкНов;
Иначе
    ДокНов.Склад=ПС;
КонецЕсли;
//Выбор поставщика
Пок=Справочники.Контрагенты;
ПП=Пок.НайтиПоКоду(Число(ПокупательКод));
Если ПП=Пок.ПустаяСсылка() Тогда
    ПокНов= Справочники.Контрагенты.СоздатьЭлемент();
    ПокНов.Код=ПокупательКод;
    ПокНов.Наименование=ПокупательНаименование;
    ПокНов.Записать();
    ДокНов.Покупатель=ПокНов;
Иначе
    ДокНов.Покупатель=ПП;
КонецЕсли;

Флаг=1;
КонецЕсли;
k=k+6;
Иначе
```

Обработка табличной части

```
к=к+6;
Иначе
Если флаг=1 Тогда
//обработка табличной части
ВрСтр=ВыбФайл.ПолучитьСтроку(к);
Сообщить(ВрСтр);
Позиция=Найти(ВрСтр,";");
ТовНаименование=Сред(ВрСтр,1,Позиция-1); Сообщить(ТовНаименование);
ВрСтр=Сред(ВрСтр,Позиция+1);
Позиция=Найти(ВрСтр,";");
ТовКод=Сред(ВрСтр,1,Позиция-1); Сообщить(ТовКод);
ВрСтр=Сред(ВрСтр,Позиция+1);
Позиция=Найти(ВрСтр,";");
Кол=Число(Сред(ВрСтр,1,Позиция-1)); Сообщить(Кол);
ВрСтр=Сред(ВрСтр,Позиция+1);
Позиция=Найти(ВрСтр,";");
Сумма=Число(Сред(ВрСтр,1,Позиция-1)); Сообщить(Сумма);
Всего=Число(Сред(ВрСтр,Позиция+1)); Сообщить(Всего);
НоваяСтрока=ДокНов.ТабличнаяЧасть1.Добавить();

КонецЕсли;
КонецЕсли;
КонецЦикла;
```

Служебные сообщения

- ▶ Молоко;3;2;8;8,8
- ▶ Молоко
- ▶ 3
- ▶ 2
- ▶ 8
- ▶ 8,8
- ▶ Документ записан

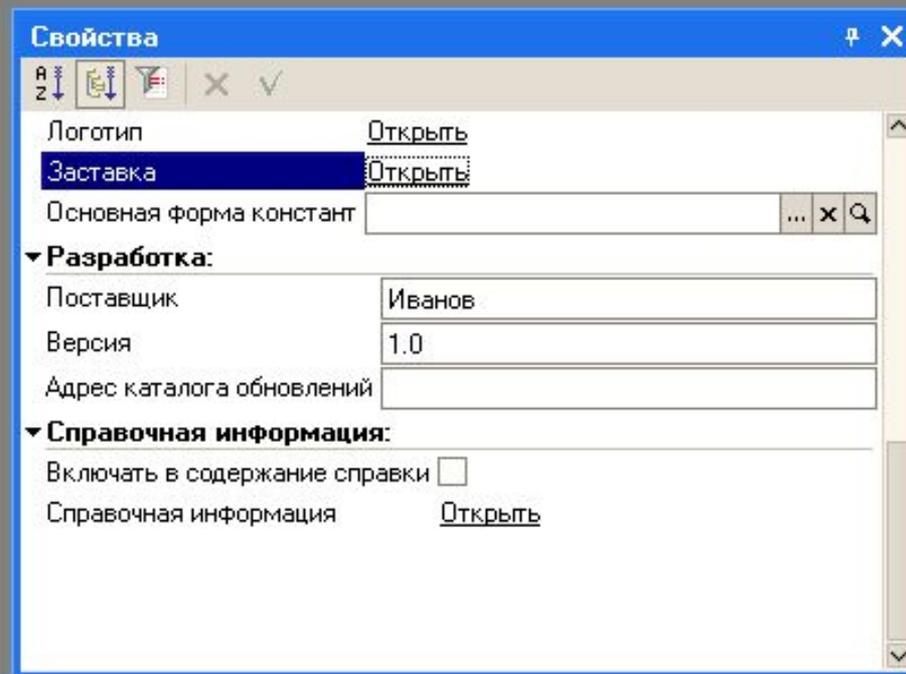
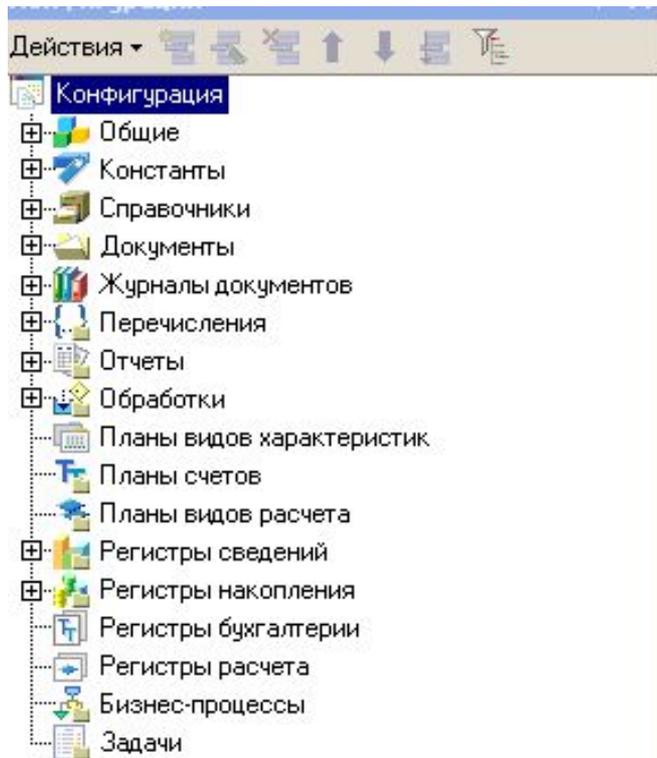
Обработка табличной части

```
ТовНаименование=Сред (ВрСтр, 1, Позиция-1) ; //Сообщить (ТовНаименов  
ВрСтр=Сред (ВрСтр, Позиция+1) ;  
Позиция=Найти (ВрСтр, ";" ) ;  
ТовКод=Число (Сред (ВрСтр, 1, Позиция-1) ) ; //Сообщить (ТовКод) ;  
ВрСтр=Сред (ВрСтр, Позиция+1) ;  
Позиция=Найти (ВрСтр, ";" ) ;  
Кол=Число (Сред (ВрСтр, 1, Позиция-1) ) ; //Сообщить (Кол) ;  
ВрСтр=Сред (ВрСтр, Позиция+1) ;  
Позиция=Найти (ВрСтр, ";" ) ;  
Сумма=Число (Сред (ВрСтр, 1, Позиция-1) ) ; //Сообщить (Сумма) ;  
Всего=Число (Сред (ВрСтр, Позиция+1) ) ; //Сообщить (Всего) ;  
НоваяСтрока=ДокНов.ТабличнаяЧасть1.Добавить ( ) ;  
НоваяСтрока.Количество=Кол ;  
НоваяСтрока.Сумма=Сумма ;  
НоваяСтрока.Всего=Всего ;  
ТабНДС=Справочники.НДС ;  
ТекНДС=ТабНДС.НайтиПоКоду (1) ;  
НоваяСтрока.СтавкаНДС=ТекНДС ;  
НоваяСтрока.НДС=Сумма*ТекНДС.Ставка/100 ;  
Тов=Справочники.Номенклатура ;  
ТТ=Тов.НайтиПоКоду (ТовКод) ;  
Если ТТ=Тов.ПустаяСсылка ( ) Тогда  
    ТовНов= Справочники.Номенклатура.СоздатьЭлемент ( ) ;  
    ТовНов.Код=ТовКод ;  
    ТовНов.Наименование=ТовНаименование ;  
    ТовНов.Записать ( ) ;  
    НоваяСтрока.Товар=ТовНов ;  
Иначе  
    НоваяСтрока.Товар=ТТ ;  
КонецЕсли ;  
КонецЕсли ;  
КонецЕсли ;
```

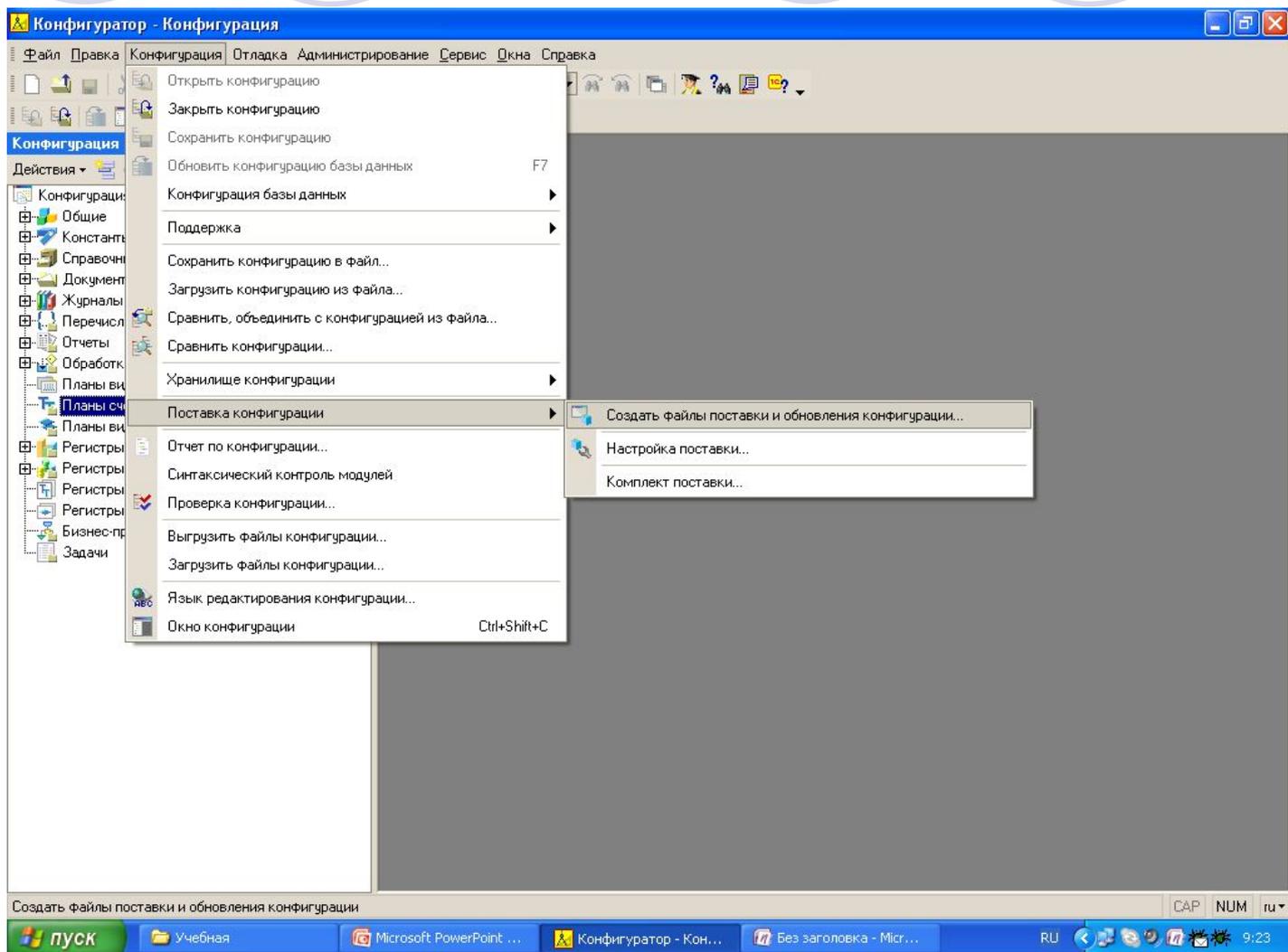
Небольшая оптимизация кода

```
□ Процедура ЗыгрузкаНажатие ( Элемент)  
    Перец ДокНов;  
    Тов=Справочники.Номенклатура;  
    Пок=Справочники.Контрагенты;  
    Ск=Справочники.Склады;  
    ТабНДС=Справочники.НДС;  
    ТекНДС=ТабНДС.НайтиПоКоду (1);  
  
    Имяфайла = КаталогВременныхФайлов() + "doc.txt";  
    Выбфайл = Новый ТекстовыйДокумент;  
    Выбфайл.Прочитать (Имяфайла);  
    КолСтр=Выбфайл.КоличествоСтрок ();  
    Док=Документы.РасходнаяНакладная;  
    Позиция=0; флаг=0;  
    Для к=1 По КолСтр Цикл  
        Врстр=Выбфайл.ПолучитьСтроку (к);  
        Позиция=Найти (Врстр, "@" );  
        Если Позиция<>0 Тогда  
            Если флаг=1 Тогда  
                ДокНов.Записать ();
```

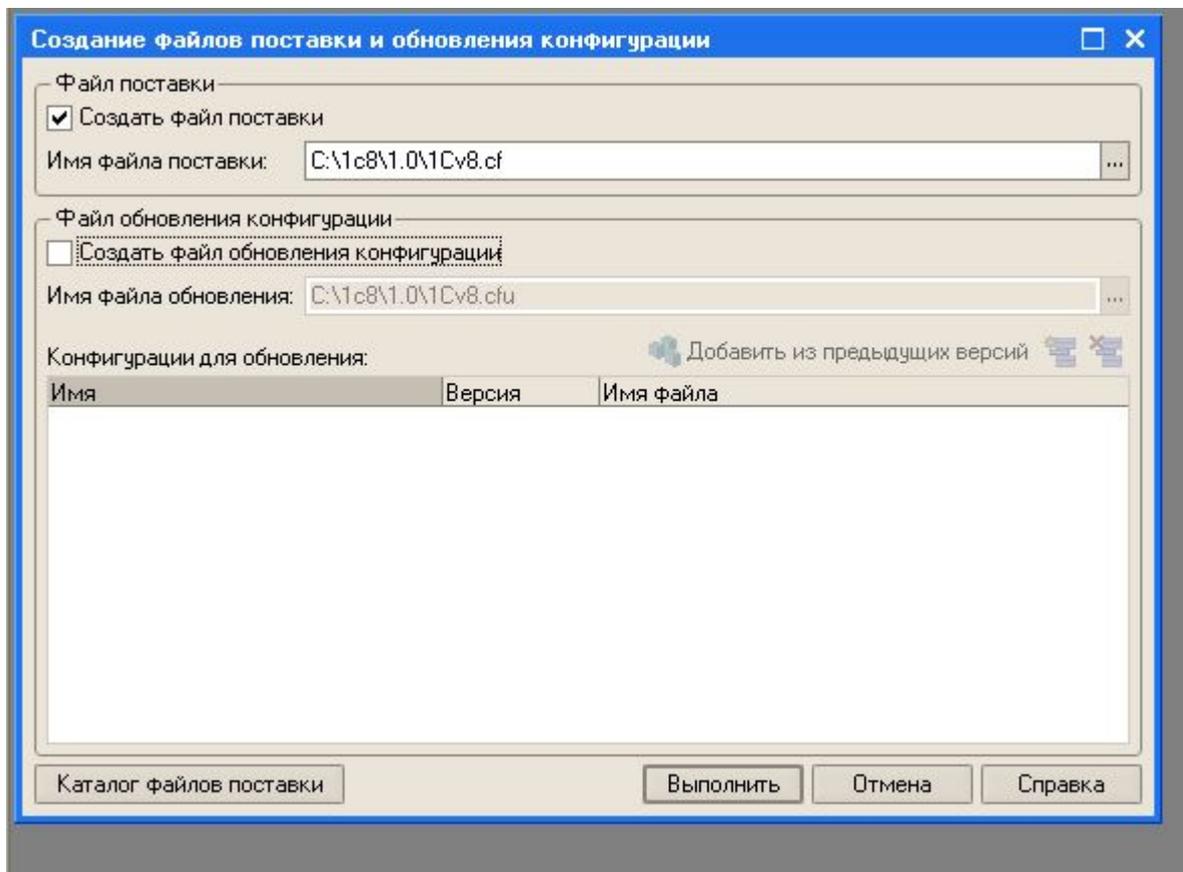
Создание файла поставки



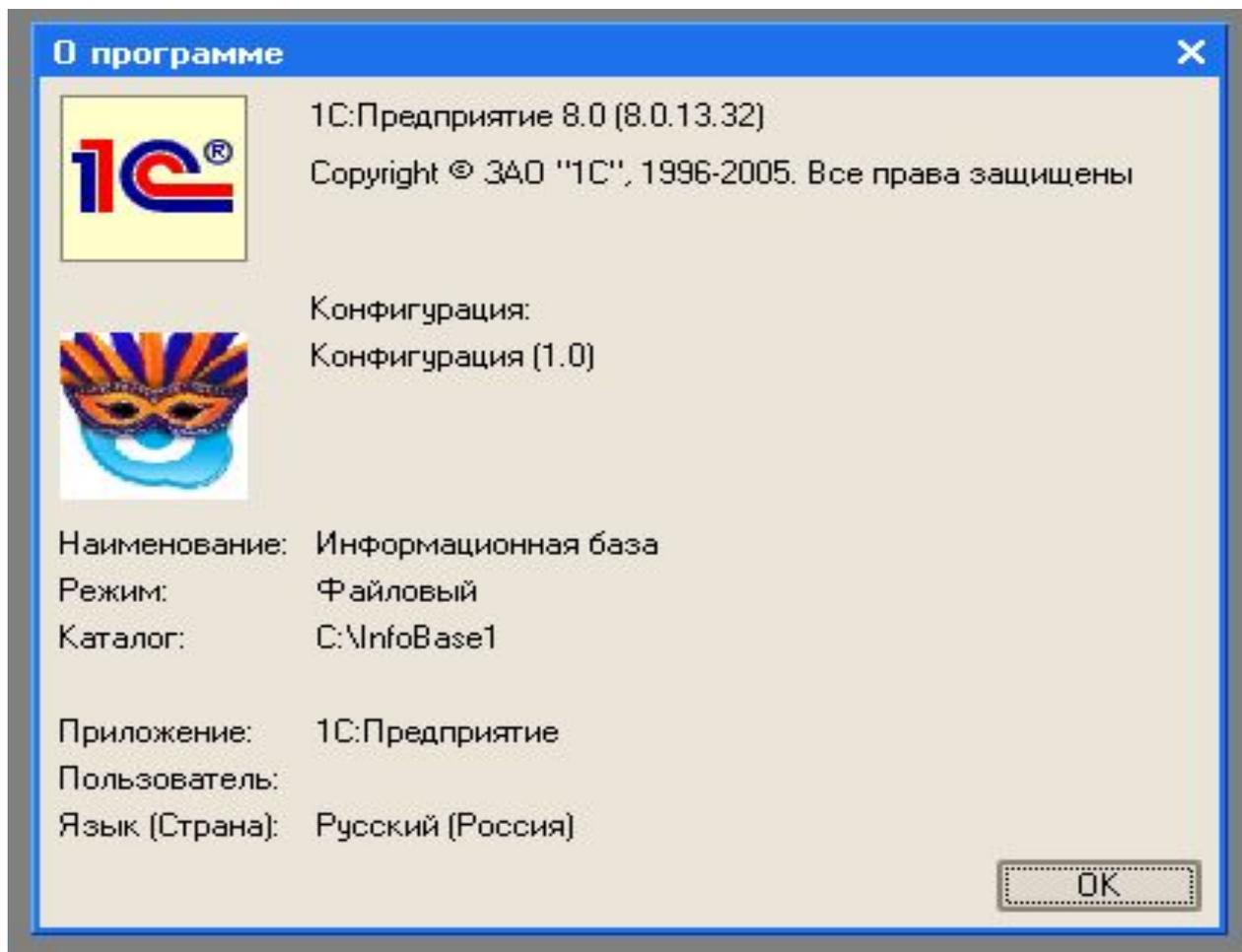
Создание файла поставки



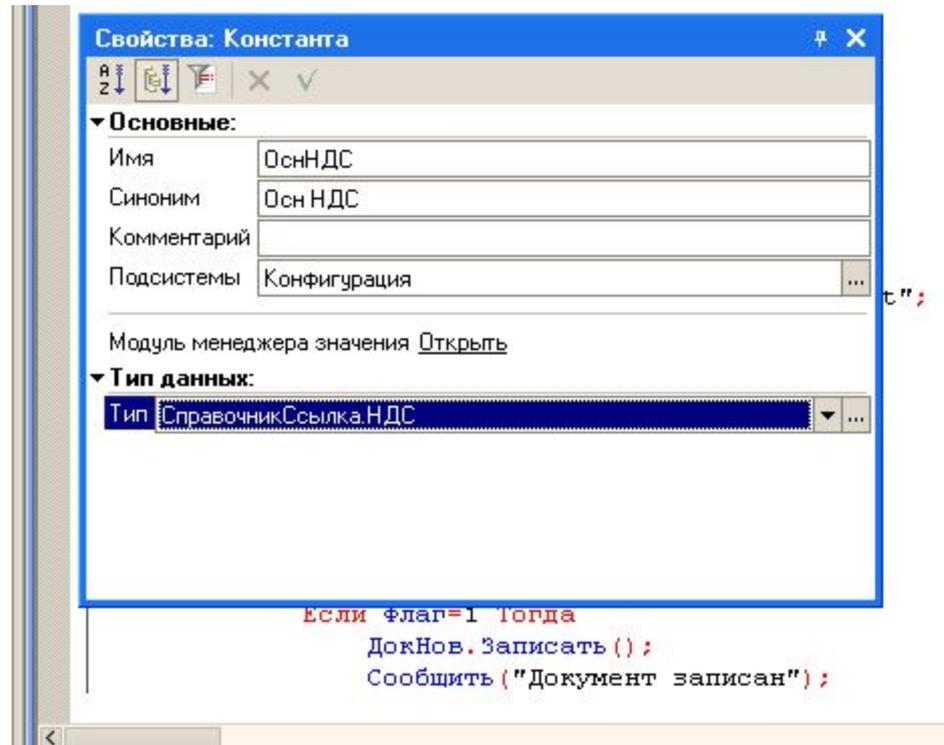
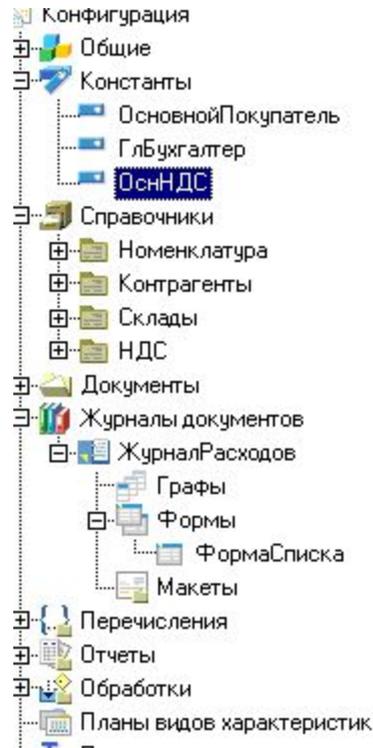
Создание файла поставки



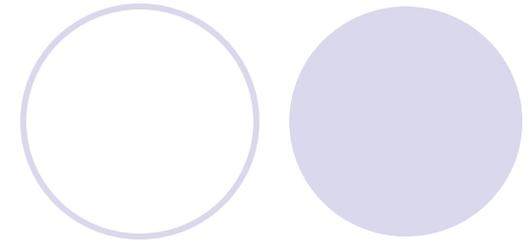
Создание файла поставки



Константа ОснНДС

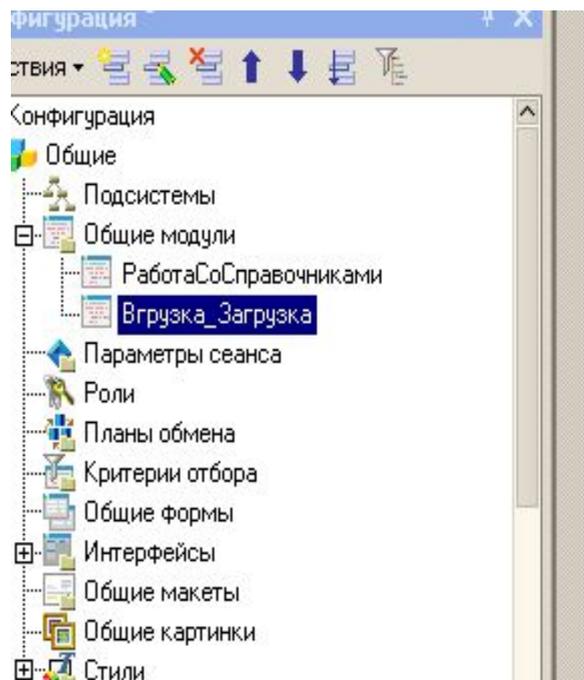


Изменение процедуры «ЗыгрузкаНажатие»,



```
□ Процедура ЗыгрузкаНажатие ( Элемент)  
    Пере м ДокНов;  
    Тов=Справочники.Номенклатура;  
    Пок=Справочники.Контрагенты;  
    Ск=Справочники.Склады;  
    //ТабНДС=Справочники.НДС;  
    //ТекНДС=ТабНДС.НайтиПоКоду(1);  
    ТекНДС=Константы.ОснНДС.Получить();  
    Имяфайла = КаталогВременныхФайлов() +"doc.txt";  
    Выбфайл = Новый ТекстовыйДокумент;  
    Выбфайл.Прочитать(Имяфайла);  
    КолСтр=Выбфайл.КоличествоСтрок();  
    Док=Документы.РасходнаяНакладная;  
    Позиция=0; флаг=0;  
    Для к=1 По КолСтр Цикл
```

Общая процедура Поиск(Вход, Выход)



```
Процедура Поиск(Вход, Выход) Экспорт
    Позиция=Найти(Вход, ";");
    Если Позиция<>0 Тогда
        Выход=Сред(Вход, 1, Позиция-1); //Сообщить(ТовНаименование);
        Вход=Сред(Вход, Позиция+1);
    Иначе
        Выход=Вход;
    КонецЕсли;
КонецПроцедуры
```

Использование процедуры Поиск(Вход,Выход)

Иначе

```
Если флаг=1 Тогда
  //обработка табличной части
  ВрСтр=ВыбФайл.ПолучитьСтроку(к);
  Поиск(ВрСтр,ТовНаименование);
  Поиск(ВрСтр,ТовКод);
  Поиск(ВрСтр,Кол);
  Поиск(ВрСтр,Сумма);
  Поиск(ВрСтр,Всего);
  НоваяСтрока=ДокНов.ТабличнаяЧасть1.Добавить();
  НоваяСтрока.Количество=Кол;
  НоваяСтрока.Сумма=Сумма;
  НоваяСтрока.Всего=Всего;
  НоваяСтрока.СтавкаНДС=ТекНДС;
  НоваяСтрока.НДС=Сумма*ТекНДС.Ставка/100;
  ТТ=Тов.НайтиПоКоду(ТовКод);
  Если ТТ=Тов.ПустаяСсылка() Тогда
    ТовНов=Справочники.Номенклатура.СоздатьЭлемент();
    ТовНов.Кол=ТовКол;
```

Макет

	1	2	3	4
	1			
Заголовок	2	Реестр документов		
	3			
Реквизиты	4			
	5	№	Дата	Документ
Реквизиты	6	<Номер>	<Дата>	<Ссылка1>
	7			
	8			
	9			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			

Модуль

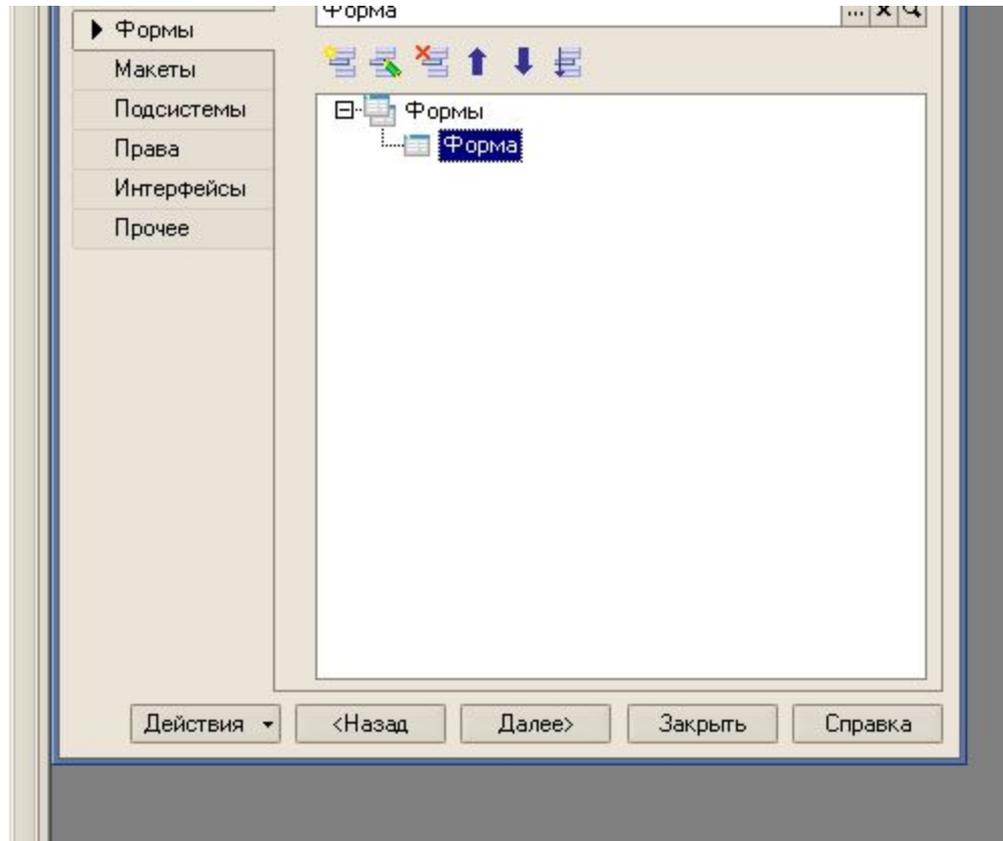
Процедура Печать (Элемент)

```
ТабДок = Новый ТабличныйДокумент;  
Макет = ЖурналыДокументов.ЖурналРасходов.ПолучитьМакет ("Печать");  
// Заголовок  
Область = Макет.ПолучитьОбласть ("Заголовок");  
ТабДок.Вывести(Область);  
Область = Макет.ПолучитьОбласть ("РеквизитыШапка");  
ТабДок.Вывести(Область);  
Выборка = ЖурналыДокументов.ЖурналРасходов.Выбрать ();  
Номер=0;  
Пока Выборка.Следующий() = Истина Цикл  
    Область = Макет.ПолучитьОбласть ("Реквизиты");  
    Поз=Найти(Строка(Выборка.Ссылка), "от");  
    Область.Параметры.Ссылка1 = Сред(Строка(Выборка.Ссылка), 1, Поз-1);  
    Область.Параметры.Дата = Сред(Строка(Выборка.Дата), 1, 10);  
    Номер=Номер+1;  
    Область.Параметры.Номер = Номер;  
    ТабДок.Вывести(Область);  
КонецЦикла;
```

Обработки. Работа со справочниками



Обработки. Работа со справочниками



Обработки. Работа со справочниками

The image displays a software interface for editing forms. The main window, titled "Обработка Изцены: Форма", shows a form design area with a table containing a single row with the value "0". The table has a header row with a question mark icon. Below the table are "Выполнить" and "Закрыть" buttons. The left sidebar lists various object types: "Формы" (Forms), "Макеты" (Layouts), "Табличные части" (Table parts), "Реквизиты" (Fields), "Диалог" (Dialog), "Модуль" (Module), and "Реквизиты" (Fields). The "Форма" (Form) item is selected. The right sidebar, titled "Свойства: Поле ввода" (Properties: Input Field), shows the configuration for a field named "Процент".

Свойства: Поле ввода

Основные:

Имя:

Данные:

Тип значения:

Длина:

Точность:

Неотрицательный:

Изменяет данные:

Пропускать при вв:

Первый в группе:

Видимость:

Доступность:

Формат:

Мин. значение:

Макс. значение:

Авто контекстное:

Контекстное меню:

Использование:

Кнопка списка выкл:

Кнопка выбора:

Кнопка очистки:

Кнопка регулировк:

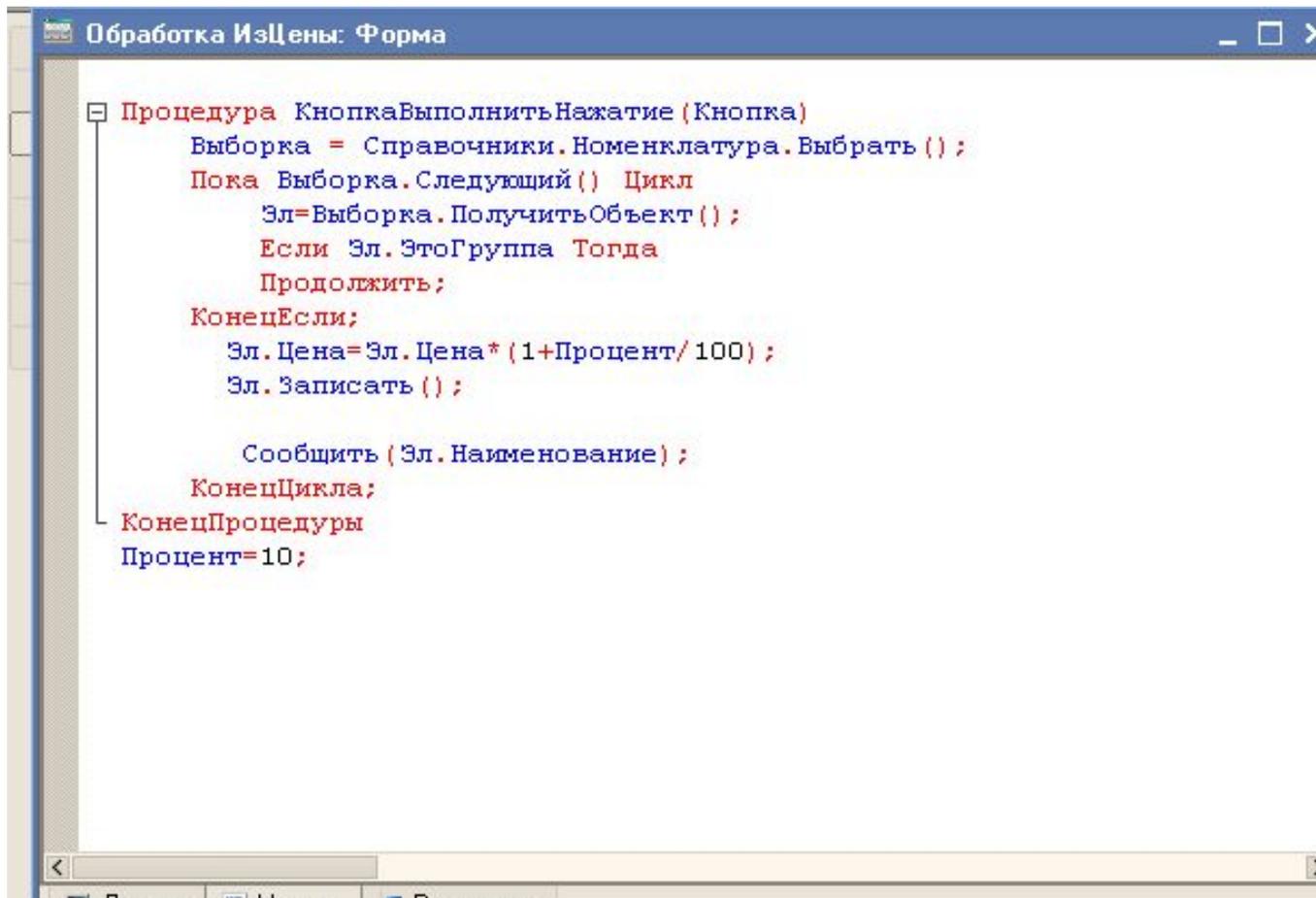
Кнопка открытия:

Автоотметка незаг:

Вывод реквизитов справочника

```
Обработка Изцены: Форма
└─ Процедура КнопкаВыполнитьНажатие (Кнопка)
    Выборка = Справочники.Номенклатура.Выбрать ();
    Пока Выборка.Следующий() Цикл
        Эл=Выборка.ПолучитьОбъект ();
        Если Эл.ЭтоГруппа Тогда
            Продолжить;
        КонецЕсли;
        Сообщить (Эл.Наименование) ;
    КонецЦикла;
КонецПроцедуры
Процент=10;
```

Изменение реквизитов справочника



```
Обработка Изцены: Форма
Процедура КнопкаВыполнитьНажатие (Кнопка)
    Выборка = Справочники.Номенклатура.Выбрать ();
    Пока Выборка.Следующий () Цикл
        Эл=Выборка.ПолучитьОбъект ();
        Если Эл.ЭтоГруппа Тогда
            Продолжить;
        КонецЕсли;
        Эл.Цена=Эл.Цена* (1+Процент/100);
        Эл.Записать ();

        Сообщить (Эл.Наименование);
    КонецЦикла;
КонецПроцедуры
Процент=10;
```

Выгрузка справочника



- Фиксированная или переменная длина полей;
- Какой используется разделитель;
- Порядок следования данных;

Выгрузка справочника

```
Процедура КнопкаВыполнитьНажатие (Кнопка)
    Имяфайла = "C:\Test\readme.txt";
    Выбфайл = Новый ТекстовыйДокумент;
    Выбфайл.УстановитьТипфайла (КодировкаТекста.UTF8);
    Выборка = Справочники.Номенклатура.Выбрать();
    Пока Выборка.Следующий() Цикл
        Эл=Выборка.ПолучитьОбъект();
        Если Эл.ЭтоГруппа Тогда
            Продолжить;
        КонецЕсли;
        Выбфайл.ДобавитьСтроку (Строка (Эл.Код) + ";" + Эл.Цена + ";" + Эл.Наименование + ";" + Эл.Тип);
    КонецЦикла;
    Выбфайл.Записать (Имяфайла);
КонецПроцедуры
```

Загрузка справочника

```
Процедура КнопкаВыполнитьНажатие (Кнопка)
Имяфайла = "C:\Test\readme.txt";
Выбфайл = Новый ТекстовыйДокумент;
Выбфайл.Прочитать (Имяфайла);
КолСтр=Выбфайл.КоличествоСтрок();
Для к=1 По КолСтр Цикл
    Врстр=Выбфайл.ПолучитьСтроку (к);
    Позиция=Найти (Врстр, ";");
    КодН=Сред (Врстр, 1, Позиция-1);
    Сообщить (КодН);
    Врстр=Сред (Врстр, Позиция+1);
КонецЦикла;
КонецПроцедуры
```

|

Загрузка справочника

□ Процедура КнопкаВыполнитьНажатие (Кнопка)

```
Имяфайла = "C:\Test\readme.txt";
```

```
Выбфайл = Новый ТекстовыйДокумент;
```

```
Выбфайл.Прочитать (Имяфайла);
```

```
КолСтр=Выбфайл.КоличествоСтрок();
```

```
Ном=Справочники.Номенклатура;
```

```
Для к=1 По КолСтр Цикл
```

```
Врстр=Выбфайл.ПолучитьСтроку (к);
```

```
Позиция=Найти (Врстр, ";");
```

```
КодН=Сред (Врстр, 1, Позиция-1);
```

```
Если Ном.НайтиПоКоду (Число (КодН))=Ном.ПустаяСсылка () Тогда
```

```
НомНов= Справочники.Номенклатура.СоздатьЭлемент();
```

```
НомНов.Код=КодН;
```

```
НомНов.Наименование="Пример";
```

```
НомНов.Записать();
```

```
Сообщить (КодН);
```

```
Врстр=Сред (Врстр, Позиция+1);
```

```
//
```

```
КонецЕсли;
```

```
КонецЦикла;
```

```
КонецПроцедуры
```