



**Некоммерческое  
акционерное  
общество**

**АЛМАТИНСКИЙ  
УНИВЕРСИТЕТ  
ЭНЕРГЕТИКИ И  
СВЯЗИ**

Кафедра  
информационных  
систем

**ВВЕДЕНИЕ В КОНФИГУРИРОВАНИЕ ПАКЕТА  
1С:ПРЕДПРИЯТИЕ**

Конспект лекций для студентов специальности  
5В070300 - Информационные системы

Алматы 2016

СОСТАВИТЕЛИ: Ж.Г. Аренбаева, М. С. Ибраев. Введение в конфигурирование пакета 1С:ПРЕДПРИЯТИЕ. Конспект лекций для студентов специальности 5В070300 - Информационные системы. Алматы: АУЭС, 2016. - 46 с.

Настоящие методические указания составлены в соответствии с учебной программой курсов для студентов специальности 5В070300 - Информационные системы.

Ил.16, библиография – 5 наименований.

Рецензент: ст.преподаватель Д.К. Абылхасенова

Печатается по плану издания некоммерческого акционерного общества «Алматинский университет энергетики и связи» на 2016 г.

© НАО «Алматинский университет энергетики и связи», 2016 г.

## Содержание

Лекция 1. Введение .....	4
Лекция 2. Объекты системы .....	5
Лекция 3. Встроенный программный язык .....	9
Лекция 4. Постановка задачи .....	12
Лекция 5. Константы .....	14
Лекция 6. Справочники .....	15
Лекция 7. Документы .....	22
Лекция 8. Журналы документов и документ «Расходная» .....	24
Лекция 9. Регистры .....	26
Лекция 10. Отчеты и обработки .....	28
Лекция 11. Администрирование .....	31
Лекция 12. Запросы .....	33
Лекция 13. Работа с конструктором запроса .....	34
Лекция 14. Хранилище .....	38
Лекция 15. Бизнес-процессы .....	40
Список литературы .....	46

## Лекция 1. Введение

**Содержание лекции:** основные особенности программного комплекса «1С:Предприятие».

**Цель лекции:** изучение понятий конфигурации и основных принципов работы в данном программном комплексе.

«1С:Предприятие» является универсальной системой автоматизации деятельности предприятия. За счет своей универсальности система «1С:Предприятие» может быть использована для автоматизации самых различных участков деятельности организаций, предприятий.

Основной особенностью системы «1С:Предприятие» является ее конфигурируемость.

Функционирование системы делится на два процесса:

- конфигурирование (описание модели предметной области средствами, предоставляемыми системой);
- исполнение (обработка данных предметной области).

На этапе конфигурирования система оперирует такими универсальными понятиями (объектами), как «Документ», «Справочник», «Реквизит», «Форма» и другие. Совокупность этих понятий и определяет концепцию системы. В свою очередь процесс конфигурирования распадается на несколько составляющих. Это «визуальное» создание структуры конфигурации, форм диалогов, выходных документов, интерфейсов, наборов прав пользователей (ролей) и написание программ на встроенном языке «1С:Предприятия» для обработки различных используемых данных.

На уровне платформы определены сами понятия объектов и стандартные операции по их обработке. Средства конфигурирования позволяют описать структуры информации, входящей в эти объекты, и алгоритмы, описывающие специфику их обработки, для отражения различных особенностей учета.

В процессе исполнения система уже оперирует конкретными понятиями, описанными на этапе конфигурирования (справочниками товаров и организаций, накладными и т.д.).

При работе пользователя в режиме «1С:Предприятие» обработка информации выполняется как штатными средствами системы, так и с использованием алгоритмов, созданных на этапе конфигурирования. Данные в базе находятся в кодировке «UNICODE».

*Основные принципы работы в программном комплексе.*

В начале работы программного комплекса его необходимо запустить, для чего в окне запуска выбираем режим: «Предприятие» или «Конфигуратор»:

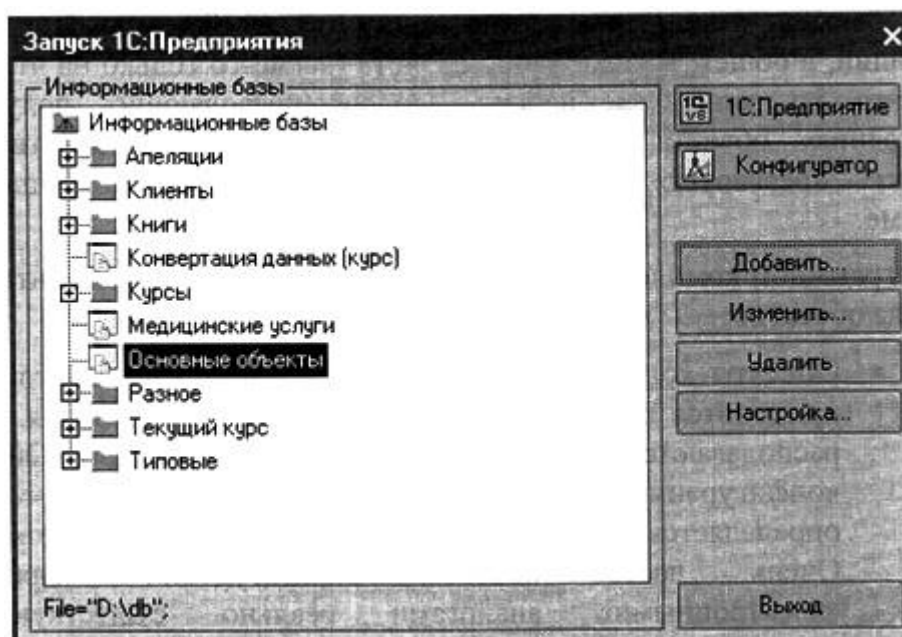


Рисунок 1 - Запуск программного комплекса

Режим «Конфигуратор» является режимом администратора системы. Начинаем процесс конфигурирования с того, что создаем объекты конфигурации, прописывая в модулях текст на встроенном языке. Для этого используем команду главного меню «Открыть конфигурацию». При этом открывается конфигурация и окно конфигурации.

Контрольные вопросы:

- 1) Для каких целей используется система «1С:Предприятие»?
- 2) Какова основная особенность системы «1С:Предприятие»?
- 3) Какими универсальными понятиями оперирует система на этапе конфигурирования?
- 4) Каковы основные принципы работы в программном комплексе?

## Лекция 2. Объекты системы

**Содержание лекции:** объекты и структура программного комплекса.

**Цель лекции:** знакомство с прикладными и подчиненными объектами системы, их классификация и назначение.

Программный комплекс «1С – Предприятие» относится к классу объектно-ориентированных систем. При этом под объектом понимается некий «черный ящик» с набором свойств и набором методов. С их помощью можно читать, записывать, а также управлять состоянием объекта или выполнять какие-либо действия. В рассматриваемом программном комплексе выделяют две основные группы объектов:

- объекты конфигурации, которые имеют различные свойства, зависящие от вида объекта. Часто эти объекты выступают в роли электронных аналогов реальных прикладных объектов;

- объекты встроенного языка, которые применяются при разработке алгоритмов подготовки информации;

Необходимо иметь в виду, что добавление объекта конфигурации позволяет оперировать с помощью встроенного языка этими объектами. Например, добавление справочника «Номенклатура» дает возможность оперировать следующими значениями:

- справочник – Ссылка – Номенклатура;
- справочник – Объект – Номенклатура;
- классификация объектов конфигурации;

Объекты конфигурации в системе «1С: Предприятие» подразделяются на три основных вида:

- общие объекты, с помощью которых создаются конфигурации, осуществляющие взаимодействия пользователей с учетными данными;

- прикладные объекты, которые находятся на первом уровне дерева метаданных;

- подчиненные объекты, к ним относятся «Реквизиты», «Табличные части» и др.

При создании объектов конфигурации в информационной базе системы добавляется таблица с определенными полями.

#### *Прикладные объекты.*

К данной группе относятся следующие объекты:

- константы, которые используются для хранения постоянных величин;

- журналы документов, в которых содержатся списки различных документов;

- перечисления, в которых содержатся списки значений, задаваемых при конфигурировании;

- планы видов расчета, в которых описаны однотипные объекты различных расчетов;

- отчеты, которые предназначены для получения выходной информации;

- обработки, в которых описаны выполнения действий для обработки информации;

- планы счетов, содержащие множество синтетических счетов;

- регистры сведений, в которых хранится информация, сформированная по определенным значениям и объединенная по времени;

- регистры накопления, в которых содержится информация, сформированная в определенных разрезах;

- регистры расчетов, в которых содержится информация о периодических расчетах;

- регистры бухгалтерии, в которых отражается информация о бухгалтерских операциях;

- бизнес-процессы, реализующие процессы автоматизации обработки и контроля операций;
- задачи, реализующие и контролирующие технологию бизнес-процессов с учетом всех необходимых задач.

#### *Подчиненные объекты.*

К подчиненным можно отнести следующие объекты:

- реквизиты — это информация, содержащая дополнительные сведения об объекте;
- табличные части - это табличная информация с дополнительными сведениями об объекте;
- реквизиты табличных частей - это доступная информация в пределах табличной части объекта;
- формы – они используются для ввода, просмотра и редактирования информации;
- макеты - табличная информация, предназначенная для создания объекта для – печати;
- графы - графы журнала в различных документах;
- измерения – это информация, содержащаяся в регистрах в различных разрезах;
- ресурсы – вся информация, содержащаяся в регистрах.

Ниже приведен пример подчиненной группы объектов в зависимости от конкретного объекта конфигурации:

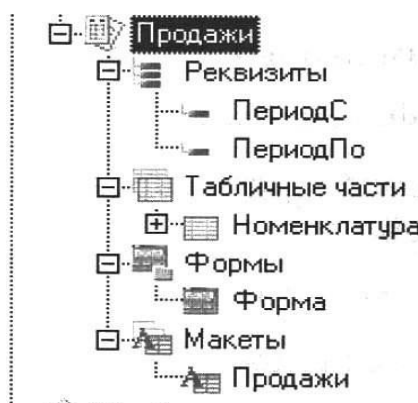


Рисунок 2 - Типы данных

Одним из основных свойств некоторых объектов конфигурации является «тип данных». Это свойство определяет, какого рода информацию может содержать объект конфигурации.

Различают следующие основные группы типов данных:

- примитивные типы (в их состав входят базовые типы данных);
- типы, не относящиеся к примитивным, но их поддержка во встроенном языке установлена с самого начала.

К примитивным типам данных относятся:

- число (десятичное число);
- строка (строка фиксированной, переменной или неограниченной длины);
- дата (дата, время, дата+время);
- булево (истина или ложь);
- тип;
- неопределенно;
- null.

#### *Универсальные коллекции значений.*

Типы, не относящиеся к примитивным, но поддержка которых во встроенном языке установлена с самого начала, являются коллекциями. Часть из этих типов входят в так называемые «Универсальные коллекции значений». Они предназначены для хранения временных наборов данных в течение сеанса работы пользователя и предназначены для вспомогательного сбора, группировки, анализа и обработки информации.

#### *Массив.*

Объекты этого типа представляют собой совокупность значений любого типа, в том числе и типа массив, что в частности позволяет организовывать многомерные массивы. Объект создается из программного кода с использованием конструктора «Новый».

Пример кода:

```
Массив "Новый Массив(Кол-во элементов)
Массив=Новый Массив;
Массив.Добавить("Первый");
Массив.Добавить(2);
// и так далее...
```

#### *Структура.*

Структура представляет собой динамический набор данных - коллекцию значений, каждый элемент которой состоит из пары «Ключ» и «Значение». Ключи структуры уникальны, и поэтому ими можно идентифицировать значения. Ключ структуры должен быть строковым и отвечать требованиям к именам переменных. К значениям структуры можно обращаться как к свойствам объекта, при этом ключ используется как имя свойства.

#### *Соответствие.*

Соответствие представляет собой динамический набор данных - коллекцию значений, каждый элемент которой состоит из пары «Ключ» и «Значение». Ключи соответствия уникальны, и поэтому ими можно идентифицировать значения. В отличие от ключа структуры, ключи соответствия могут



быть произвольных типов. Рекомендуется, чтобы в качестве ключа выступало значение неизменяемого типа или другого типа, значение которого может только присваиваться, но не может менять свое содержимое.

#### *Список значений.*

Список значений — это объект позволяющий строить динамические наборы значений и манипулировать ими. Может быть наполнен значениями любых типов. Условно список значений можно представить как таблицу из четырех колонок: пометка, значение, представление, картинка. Каждое из значений характеризуется позицией в списке (индексом):

СПЗ=Новый СписокЗначений();

#### *Таблица значений.*

Таблица значения – это объект, который позволяет создавать динамические наборы значений и работать с ними. Он содержит значения различных типов и связан с элементом «табличное поле», например:

*Таблица Значений. Колонки. Добавить ("Количество" "Количество товара");*

*Строка Таблицы Значений = Таблица Значений. Добавить);*

*Строка Таблицы Значений, Количество= 11;*

#### *Дерево значений.*

Объект похожий на таблицу значений. Но в отличие от нее строки дерева значений могут образовывать иерархические структуры: каждая строка дерева может иметь набор подчиненных строк и т.д., например:

ДЗ=Новый ДеревоЗначений();

Контрольные вопросы:

- 1) Для каких целей используются прикладные объекты?
- 2) Какова основная особенность использования универсальных коллекций значений?
- 3) Объяснить назначение массива, структуры, соответствия и списка значений?
- 4) Чем отличается дерево значений от таблицы значений?

### **Лекция 3. Встроенный программный язык**

**Содержание лекции:** введение во встроенный программный язык программного комплекса «1С:Предприятие».

**Цель лекции:** знакомство с описанием встроенного языка, концепциями, процедурами языка и видами модулей.

Необходимость наличия встроенного языка определена концепцией «настраиваемые системы». Язык относится к группе предметно ориентированных. Он поддерживает специализированные типы данных предметной области, определяемые конфигурацией системы. Работа с этими типами данных в языке организована с использованием объектной техники.

Язык поддерживает конструкции, позволяющие определять переменные, процедуры, функции. Операторы разделяются символом «;». Встроенный язык не чувствителен к регистру, допускается двуязычное описание конструкций, например: Если или If. Однако предпочтение отдается написанию на языке типовых конфигураций, например:

```
Переменная ИмяПеременной;  
Процедура ИмяПроцедуры(Имяпараметра 1,...)  
// текст комментария // тело процедуры  
КонецПроцедуры  
Функция ИмяФункции(ИмяПараметра 1,...)  
//тело функции  
Возврат (ВозвращаемоеЗначение);  
Конец Функции
```

Имя переменной, процедуры, функции может состоять из букв, цифр и символов подчеркивания. Начинаться имя должно либо с буквы, либо с символа подчеркивания.

Порядок описания процедур, функций между собой значения не имеет. Как и в любом другом языке существуют конструкции, реализующие ветвление, циклы:

```
Если Условие Тогда  
//код  
ИначеЕсли Условие Тогда  
//код  
Иначе  
//код  
КонецЕсли;  
Для ПеременнаяСчетчик=Начальное Значение По Конечное Цикл  
// тело цикла  
КонецЦикла;  
Для Каждого ПеременнаяЦикла Из ИмяКоллекции Цикл  
//тело цикла  
КонецЦикла;  
Пока УсловиеЦикла Цикл  
//тело цикла  
КонецЦикла;
```

Часто во встроенном языке приходится иметь дело с объектами, имеющими набор свойств, методов. При этом используют следующие подходы:

Наим=Спр. Наименование;

Наим=Спр [«Наименование»];

Вызов методов объектов производится «через точку»

Спр.Печать();

Допускаются следующие конструкции:

Док.Контрагент.ПолучитьОбъект().ПечатьКарточкиКлиента();

Платформа «1С:Предприятие 8» содержит визуальные и языковые средства конфигурирования. Языковые модули, как правило, используются в конкретных местах для работы с отдельными алгоритмами и настраиваются в процессе конфигурации. Программный код всегда помещается в «модули».

Место размещения конкретного программного модуля предоставляется конфигуратором в тех точках конфигурации, которые требуют описания специфических алгоритмов функционирования. Эти алгоритмы оформляются в виде процедур или функций и вызываются автоматически в необходимых ситуациях.

#### *Виды модулей.*

Существуют модули следующих видов:

- модуль приложения, который располагается в корневом разделе конфигурации и содержит процедуры - обработчики событий, определения (в том числе с ключевым словом «Экспорт») переменных, процедур, функций, доступных в любых точках конфигурации (исключение составляют общие модули). При этом не рекомендуется реализация процедур и функций, выполняющих обработку данных;

- модуль внешнего соединения, в котором располагаются экспортируемые переменные, процедуры и функции, а также процедуры - обработчики событий «ПриНачалеРаботыСистемы()» и «ПриЗавершенииРаботыСистемы()», используемые в режиме внешнего соединения;

- общие модули, которые находятся в отдельной ветви дерева метаданных и содержат процедуры и функции. Те из них, которые определены с использованием ключевого слова «Экспорт» доступны из всех модулей конфигурации, в том числе из модуля приложения. При определении общих модулей можно проводить их дополнительную логическую классификацию (по подсистемам). Если используется клиент-серверный вариант системы «1С:Предприятие», то с помощью свойств «Клиент» и «Сервер», а также указаний препроцессору (*директивы #Если Сервер Тогда ... И #Если Клиент Тогда,..*) можно организовать выполнение различных процедур и функций общих модулей на сервере приложения или на клиентском месте;

- модули объектов находятся в той ветви конфигурации, где находятся следующие объекты (справочники, документы, отчеты, обработки и т.д.);

- модули набора записей содержатся в регистрах любого вида и определяют процедуры «ПриЗаписи», «ПередЗаписью»;
- модули форм находятся в формах конфигурации и содержат переменные, процедуры, функции, реализующие алгоритмы формы.

#### *Контекст выполнения модуля.*

Каждый программный модуль связан с остальной частью конфигурации. Эта связь называется контекстом выполнения модуля. Различают два вида контекста:

- глобальный контекст, который составляется из значений свойств и методов глобального контекста функциями встроенного языка и языковыми конструкциями, а именно: переменными, процедурами и функциями программного модуля приложения и общего модулей, при этом используется функция «Экспорт»;

- локальный контекст модуля, который составляется при конфигурации, при этом используется программный модуль. При этом используется набор доступных этому модулю объектов.

Контрольные вопросы:

- 1) Для каких целей используется система «1С:Предприятие»?
- 2) Какова основная особенность системы «1С:Предприятие»?
- 3) Какие модули существуют в системе «1С:Предприятие»?
- 4) Каковы отличия глобального контекста выполнения модуля от локального?

## **Лекция 4. Постановка задачи**

**Содержание лекции:** введение в постановку задачи.

**Цель лекции:** демонстрация сценария типичных бизнес-процессов, параметров, создание информационной базы данных.

Содержание задачи состоит в том, что некоторая фирма закупает товары у отдельных поставщиков и продает затем эти товары своим покупателям. В рассматриваемой задаче фигурируют следующие элементы:

- торговля товарами;
- бухгалтерский учет;
- кадры;
- заработная плата.

Необходимо обработать и сохранить информацию по следующим параметрам:

- по номенклатуре товаров;
- по контрагентам фирмы;
- по сотрудникам компании.

Требуется организовать документооборот таким образом, чтобы пользователь вводил информацию один раз и при работе с программой информация обрабатывалась в наиболее удобном и оптимальном виде. В любое время можно было получить информацию в следующих разрезах:

- по покупкам (у каких поставщиков, какой товар и в каком количестве куплен);
- по продажам (в разрезе покупателей, товаров и количестве проданного товара);
- по работникам фирмы.

Для простоты решения задачи определим, что у фирмы имеется один склад, фирма представляет собой одно юридическое лицо, фирма работает с национальной валютой.

#### *Создание информационной базы данных.*

Для создания информационной базы необходимо на жестком диске компьютера создать рабочую папку. После запуска платформы в окне запуска необходимо воспользоваться кнопкой «Добавить», после чего откроется «Мастер подключения».

На первом этапе необходимо выбрать вариант добавления.

На втором этапе можно выбрать создание новой базы, или создать пустую информационную базу. Для реализации нашей задачи необходимо создать пустую информационную базу.

На третьем этапе выберем тип расположения и имя информационной базы. Платформа «1С:Предприятие 8» может работать в двух режимах: *Файловом* - когда выбирается тип расположения информационной базы «На данном компьютере или на компьютере в локальной сети» или же *Клиент-серверном* - когда выбирается тип расположения «На сервере 1С предприятия».

Для решения задачи выберем первый вариант - *Файловый режим*.

На последнем этапе указываем каталог, где будет размещаться созданная база данных. После создания информационной базы данных необходимо открыть ее в режиме «Конфигуратор», для чего воспользуемся соответствующей кнопкой в окне запуска программы.

Рассмотрим объекты конфигурации:

- *подсистемы* - для добавления подсистемы необходимо внутри ветви «Общие» щелкнуть правой клавишей мыши и в открывшемся контекстном меню выбрать «Добавить». В создаваемой конфигурации необходимо будет выбрать следующие направления учета: торговля, ведение персонала, бухгалтерский учет. С помощью подсистем можно организовать и дополнительную классификацию объектов конфигурации по этим направлениям учета. Эта классификация может использоваться, например, для организации выборки, для просмотра дерева объектов конфигурации, для определения интерфейсов и т.п. В соответствии с нашей задачей добавим три подсистемы: «Бухгалтерия», «Торговля», «Кадры»;

- *константы* - в любой организации имеются значения данных, которые меняются очень редко, например: название организации, юридический адрес, фамилии руководства и т.д. Для хранения таких значений и существуют объекты «Константы».

Контрольные вопросы:

- 1) Приведите другие примеры постановки задачи для системы «1С:Предприятие».
- 2) По какой схеме организуется документ?
- 3) Как создается информационная база данных?

## **Лекция 5. Константы**

**Содержание лекции:** введение в работу с константами в комплексе «1С:Предприятие».

**Цель лекции:** изучение понятий констант, как типа данных, взаимодействий с другими объектами, правил пользования конструктором.

Создадим константу «*Название Организации*», для чего щелкнем правой кнопкой мыши на ветви «*Константы*» и выберем пункт «*Добавить*». В открывшемся окне свойств заполним необходимые значения.

Значением свойства «*Подсистемы*» выберем «*Учебная Конфигурация*».

Теперь создадим константу «*Юридический адрес*». Тип данных константы выберем - «*Строка*» длиной в 60 символов. Одним из основных интерфейсных средств «1С:Предприятия» для взаимодействия с пользователем служит объект Форма, которую создадим для просмотра и редактирования. Для ее создания необходимо щелкнуть правой клавишей мыши на ветви «*Константы*» и выбрать пункт контекстного меню «*Создать форму констант*», после чего откроется Конструктор форм. Пользователю будет предложено задать Имя, Синоним, Комментарий, выбрать тип формы и выбрать варианты расположения панелей управления формой (для нашей задачи необходимо снять флажок «*Командная панель формы снизу*»).

Созданная форма будет находиться для ее корректировки в ветви дерева метаданных «*Общие*», а затем в ветви «*Общие формы*».

Следует отметить, что, несмотря на существенные различия типов объектов, для которых может производиться создание формы, их конструкторы имеют много общего. Отличие будет заключаться в количестве типов форм. В нашем случае у константы их два. По каждому типу форм может быть создано сколько угодно форм. Если у объекта несколько форм одного типа, то одну из них можно сделать основной. Сколько бы типов форм не было представлено в Конструкторе, один из них обязательно называется «*Произвольная форма*».

Выбор типа «*Произвольная форма*» не приводит к созданию основного реквизита формы. Выбор другого типа влечет за собой создание основного реквизита, который определяет основную функцию формы. Основной реквизит

формы можно увидеть на закладке уже созданной формы «Реквизиты», он отмечен жирным шрифтом.

На следующем шаге предлагается выбрать перечень констант, помещаемых на форму, и проставить соответствие между ними и элементами формы. Список возможных вариантов зависит от типа константы.

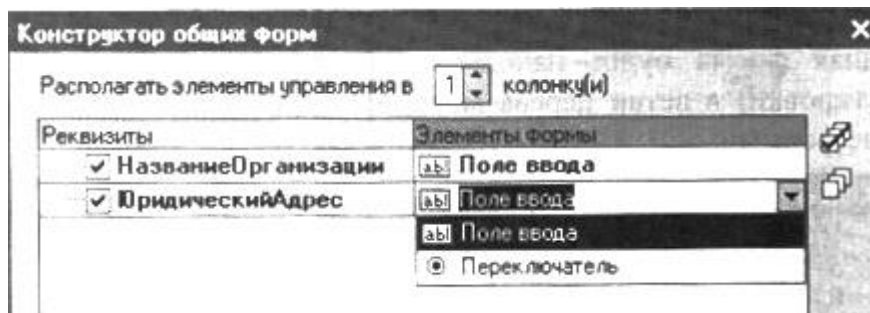


Рисунок 3 - Содержимое Конструктора общих форм

После выполнения всех действий необходимо нажать на кнопку «Готово» и получить окончательный результат.

Кнопки «ОК», «Записать», «Закрывать» определены как кнопки командной панели. Это объясняется тем, что на первой закладке Конструктора форм был отмечен флажок: «Командная панель формы снизу».

Контрольные вопросы:

- 1) Почему константы выделены в отдельный тип данных в система «1С:Предприятие»?
- 2) Для чего применяется произвольная форма?
- 3) Как пользоваться конструктором общих форм?

## Лекция 6. Справочники

**Содержание лекции:** введение в работу со справочниками в комплексе «1С:Предприятие».

**Цель лекции:** изучение понятий справочников как объектов конфигурации во взаимодействии с другими объектами и примеров работы со справочниками, их компонентами, формами из кода.

В рассматриваемой системе используются объекты типа «Справочник», представляющие собой списки материалов, товаров, организаций, работников и т.д. Структура того или иного справочника определяется при его создании в конфигураторе. Справочники имеют иерархическую структуру, при их настройке определяется состав реквизитов, необходимые колонки, различные дополнительные свойства, а также другие настройки.

*Справочник «Единицы измерения».*

При создании справочника «ЕдиницыИзмерения» на закладке «Основные» указывается его имя и синоним, как указано на рисунке.

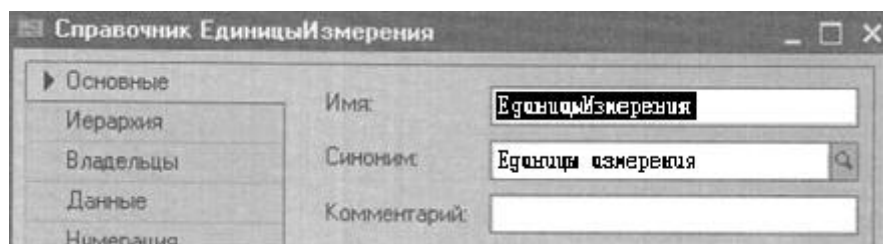


Рисунок 4 - Закладка «Основные»

На закладке «Подсистемы» определяются основные подсистемы, например, подсистема «Торговля».

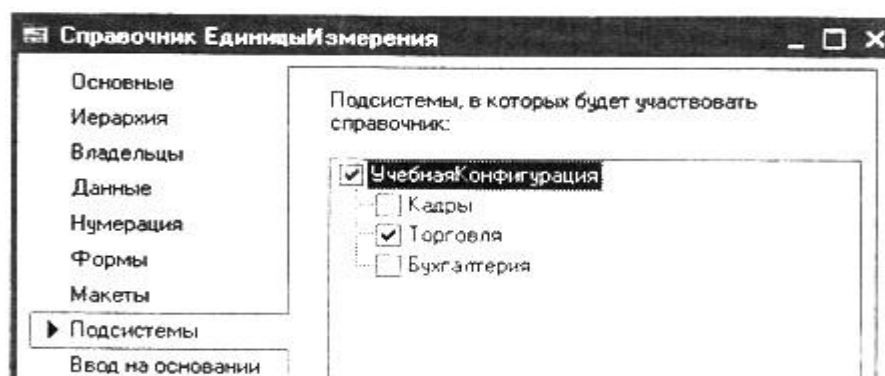


Рисунок 5 - Закладка «Подсистемы»

На закладке «Прочее», с помощью кнопки «Предопределенные», можно ввести предопределенные элементы справочника. При введении предопределенного элемента указывается имя, код, наименование, которые могут отличаться от подобных записей справочника в информационной базе.

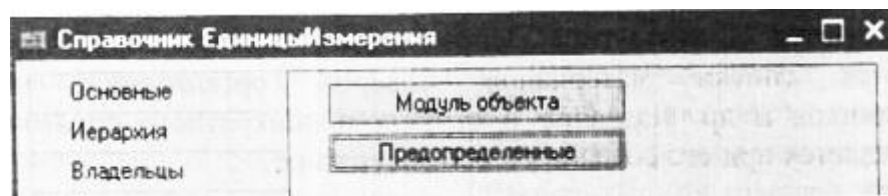


Рисунок 6 - Кнопка «Предопределенные»



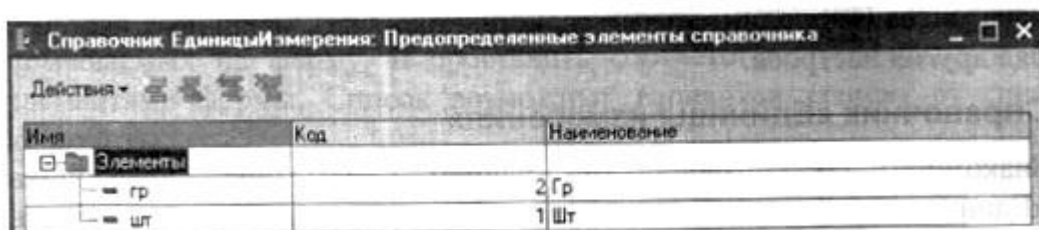


Рисунок 7 - Введение предопределенных элементов

Для обращения к предопределенному элементу после его введения необходимо использовать следующую конструкцию:

Справочники.Имя справочника.Имя предопределенного элемента; например: СсылкаНаЭлемент=Справочники.ЕдиницыИзмерения.гр.

*Справочник «Подразделения».*

Рассматриваемый справочник имеет иерархическую структуру элементов.

Например, для создания справочника «Номенклатура» используем иерархическую структуру с неограниченным количеством уровней и следующими реквизитами:

Реквизиты справочника:

«ЦенаПокупки» - тип «Число»;

«ЦенаПродажи» - тип «Число»;

«ОснЕдиницаИзмерения» тип;

«СправочникСсылка.ЕдиницыИзмерения».

При добавлении реквизитов необходимо указать состав свойств. Редактирование происходит в диалоговом режиме с настройкой формы: списка, группы, элемента. Справочник принадлежит подсистеме «Торговля»:

«Отчество» - тип «Строка» длина 30;

«ДатаРождения» - тип «Дата» состав : «Дата».

У этого справочника имеется табличная часть «ТрудоваяДеятельность», реквизиты которой следующие:

«НачалоРаботы» - тип «Дата» состав : «Дата»;

«ОкончаниеРаботы» - тип «Дата» состав : «Дата»;

«Организация» - тип «Строка»;

«Должность» - тип «Строка».

Редактирование справочника производится следующим образом: требуется настроить форму списка и форму элемента, при этом в форме списка необходимо определить только колонки «Код», «Наименование» (в данное поле будет записываться фамилия с инициалами), «Дата Рождения». Используя свойство колонки «Текст шапки» заменим заголовок «Наименование» на «Фамилия Имя Отчество». Затем, используя элемент диалога «Список», свой-

ство «Способ редактирования» выставим в значение: «В списке». При этом необходимо учесть свойство «Изменять способ редактирования».

При настройке формы элемента справочника разместим реквизиты и табличную часть на разных страницах, используя непосредственно форму или используя элемент управления «Панель».

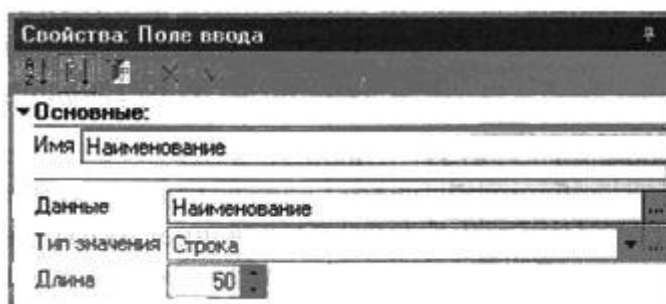


Рисунок 8 - Свойства: Поле ввода

При настройке расположим реквизиты табличной части «НачалоРаботы», «ОкончаниеРаботы», «Должность» на одной строке, а «Организация» на другой. Для этого необходимо выбрать в свойстве «Положение» значение «В той же колонке» для «ОкончаниеРаботы» и «Должность». Для колонки же «Организация» в свойстве «Положение» выберем значение «На следующей строке».

Рассмотрим элементы управления, расположенные на первой странице панели (например «Наименование»), при этом необходимо учесть, что в данной форме существует:

- наименование - как элемент управления «поле ввода»;
- наименование — как наименование элемента справочника (в контексте которого открывается данная форма элемента).

Любой элемент управления через свойство «Данные» связан с данными формы. Состав данных формы можно просмотреть на закладке «Реквизиты» соответствующей формы. Для обращения к элементу управления необходимо использовать свойство формы «ЭлементыФормы»:

*ЭлементУправления=ЭлементыФормы.Наименование;*

В общем случае несколько элементов управления могут быть связаны через свойство «Данные» с одним элементом данных формы.

#### *Основной реквизит формы.*

Основной реквизит формы выделен жирным шрифтом на закладке формы «Реквизиты» или в свойствах самой формы. Этот реквизит характеризует основное предназначение формы, которую можно неограниченно расширять в случае необходимости. Это достигается за счет добавления реквизитов. Рассмотрим форму элемента справочника «Физические лица». На закладке «Реквизиты» добавим новый реквизит и определим его тип «СправочникСписок.Номенклатура». Имя реквизита определим как «СписокНоменк-

латуры». Создадим в элементе управления «Панель» новую страницу. Добавим элемент управления «табличное поле» и через свойство «Данные» свяжем его с реквизитом «СписокНоменклатуры». Затем добавим элемент управления «командная панель» и настроим свойства:

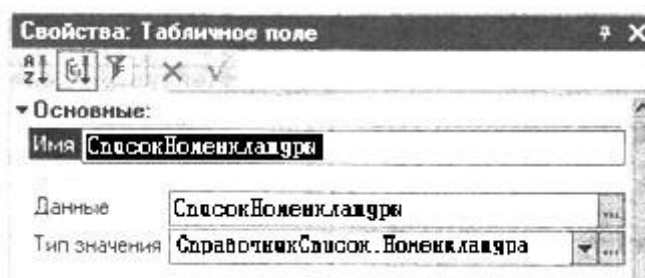


Рисунок 9 - Свойства: Табличное поле

Таким образом, на одной из страниц формы элемента справочника «Физические лица» мы сможем работать со списком элементов справочника «Номенклатура».

#### *Работа со справочниками из кода.*

Рассмотрим справочник «ФизическиеЛица», описав алгоритм, который позволит открывать форму элемента. В форме списка необходимо определить свойства кнопки командной панели управления. Добавим в автоматически созданную панель управления свою кнопку).

В модуле формы запишем следующий текст процедуры:

Процедура ОткрытьПодробно(Кнопка)

ТекСтрока=ЭлементыФормы. Справочникелисок.ТекущаяСтрока;

Форма=ТекСтрока.ПолучитьФорму("ФормаЭлемента",,);

Форма. Открыть();

КонецПроцедуры.

В свойствах элемента формы «СправочникСписок» необходимо снять флажок «Автоконтекстное меню» и в свойстве «Контекстное меню» выбрать значение «ДействияФормы». Необходимо отметить, что при работе с формой следует различать данные и элементы управления на форме. В данной форме списка находится основной реквизит формы «СправочникСписок», который имеет тип «СправочникСписок, ФизическиеЛица», а в форме находится элемент управления «СправочникСписок», который имеет тип «ТабличноеПоле». Элементы формы через свойство «Данные» могут быть связаны с такими данными, как реквизиты, свойства справочника, реквизиты форм и т.д. При использовании свойства формы «ЭлементыФормы» обращаемся к элементу диалога.

### *Печатные формы.*

Для формирования печатных форм в системе используются два объекта: «Макет» и «Табличный документ». Объект «Макет» является объектом метаданных и, как правило, содержит в себе объект «Табличный документ».

Табличный документ содержит ячейки, составляющие строки и колонки, которые обозначаются уникальными номерами. Табличный документ представляет собой готовую форму для заполнения. Элемент управления типа «Поле табличного документа» располагается в этой форме. Также в табличный документ можно вставить и другие элементы управления.

Макет предназначен для определения составных элементов, на основании которых строится готовая выходная форма. Проектирование макета заключается в определении этих составных элементов - именованных областей. Макет располагается внутри конфигурации. Имя определяется через свойство «Имя» после выделения необходимой области. Содержимое ячейки зависит от значения свойства «Заполнение». Процесс заполнения печатной формы производится по следующей технологии: вначале создается пустой табличный документ. Затем при обработке алгоритма из макета извлекаются именованные области, заполняются значениями параметров и копируются в табличный документ готового отчета. Средства языка позволяют заполнять отчет как вниз, так и вправо.

Рассмотрим печать «Прайс-листа» из формы списка справочника «Номенклатура». Определим макет «Прайс-листа», а затем создадим алгоритм формирования печатной формы. Для этого в модуле формы запишем следующую процедуру:

Процедура Печать()

Макет = Справочники.Номенклатура.ПолучитьМакет("ПрайсЛист");

ОбластьШапки = Макет.ПолучитьОбласть(" Шапка");

ТабДок = Новый ТабличныйДокумент;

ТабДок.Вывести(ОбластьШапки);

Выборка = Справочники.Номенклатура.Выбрать();

Пока Выборка.Следующий() Цикл

Если Выборка.ЭтоГруппа=Истина Тогда

ОбластьТабЧасти= Макет.ПолучитьОбластьС"СтрокаГруппы");

ОбластьТабЧасти.Параметры.Код=Выборка.Код;

ОбластьТабЧасти.Параметры.Наименование=Выборка.Наименование;

Иначе

ОбластьТабЧасти.Макет.ПолучитьОбласть("Строка");

ОбластьТабЧасти.Параметры.Код=Выборка.Код;

ОбластьТабЧасти.Параметры.Наименование=Выборка.Наименование;

ОбластьТабЧасти.Параметры.РозничнаяЦена=Выборка.ЦенаПродажи;

ОбластьТабЧасти.Параметры.РасшифровкаЭлем=Выборка.Ссылка;

КонецЕсли;

ТабДок.Вывести(ОбластьТабЧасти);

КонецЦикла;  
ТабДок.Защита = Истина;  
ТабДок.ТолькоПросмотр = Истина;  
ТабДок.Показать();  
КонецПроцедуры

Последним шагом будет помещение в диалог формы кнопки и указание в свойстве «Нажатие» имени процедуры «Печать».

События: Нажатие Печать

Подобные печатные формы объектов можно создавать с помощью конструктора печатных форм. Вызвать его можно либо сделав щелчок правой клавишей мыши на заголовке объекта в дереве метаданных, либо с помощью кнопки «Конструкторы», расположенной в основной форме настроек объекта на закладке «Макеты».

#### *Подчиненные справочники.*

Создадим справочники «Контрагенты» и «Контактные Лица». Справочник «Контрагенты» имеет реквизит «Полное наименование», редактируется в списке, используя иерархию групп и элементов. Справочник «Контактные лица» подчинен справочнику «Контрагенты», поэтому в свойстве «Использование подчинения» мы выбираем значение элементам.

В справочнике необходимо иметь реквизит «Телефон», тип «Строка», осуществим редактирование, которое производится в списке.

При настройке формы списка справочника «Контрагенты» в диалог формы необходимо поместить элемент диалога «Табличное поле», определив имя элемента в «Подчиненный», тип «СправочникСписок. Контактные Лица». После настройки отобразятся колонки, включающие код и наименование. Для просмотра колонки «Телефон», щелчком правой кнопкой мыши, можно также выбрать пункт «Добавить колонку» или «Размещение данных».

Затем в свойстве «Связь по владельцу» табличного поля «Подчиненный» выбираем событие «СправочникСписок. ПриАктивизацииСтроки».

В заключении добавим командную панель управления для работы со справочником, добавив элемент управления «командная панель» в диалог, а в качестве источника укажем табличное поле «Подчиненный» и отметим «Автозаполнение».

#### Контрольные вопросы:

- 1) Для чего используются справочники и какую структуру они имеют?
- 2) Почему имя справочника используется обязательно без пробела, а в синониме можно использовать пробел?
- 3) Для каких целей используются predetermined элементы?
- 4) Для чего в программном комплексе предусмотрена работа со справочниками из кода и как пользоваться печатными формами?

## Лекция 7. Документы

**Содержание лекции:** введение в работу с документами в комплексе «1С:Предприятие».

**Цель лекции:** знакомство с понятием документа как важным объектом конфигурации программного комплекса. Объединение ключевых характеристик бизнес-процессов ERP и MES - систем.

Документ является основным понятием системы «1С:Предприятие», с помощью которых осуществляется ввод в систему первичной информации обо всех хозяйственных операциях, совершаемых на предприятии.

Большинство документов, которые создаются при настройке конфигурации, являются электронными аналогами стандартных документов, однако электронные документы не только фиксируют информацию о хозяйственных операциях.

Важными характеристиками всех документов являются дата и время, которые устанавливают точную последовательность совершения всех без исключения операций. Необходимо иметь в виду, что у любого документа может быть неограниченное число табличных частей.

*Документ «Приходная».*

Для наглядности создадим документ «Приходная», который будет являться электронным аналогом расходной накладной поставщика. В этом документе имеется четыре реквизита:

- «Контрагент» (тип «СправочникСсылка.Контрагенты»);
- «Контакт» (тип «СправочникСсылка.КонтактныеЛица»);
- «Сотрудник» (тип «СправочникСсылка.ФизическиеЛица»);
- «СуммаДокумента» (тип «Число» длина 15, точность 2).

Определим одну табличную часть «Товары», ее состав будет следующий:

- номенклатура (тип СправочникСсылка.Номенклатура);
- количество (тип «Число» длина 10, точность 0);
- цена (тип «Число» длина 10, точность 2);
- сумма (тип «Число» длина 10, точность 2).

В свойстве «Проведение» выберем значение «Запретить». Создадим форму документа, форму списка документа. При формировании формы документа реквизит «СуммаДокумента» вставим как надпись, в свойстве «Связь по владельцу» для элемента «Контакт» установим значение «Контрагент».

Полученный документ можем заполнять, сохранять в нем данные, но необходимо, чтобы при выборе из справочника номенклатурной позиции в документ автоматически переносилась цена, а после проставления количества рассчитывалась общая сумма за позицию. Кроме этого, перечень выбираемых контактных лиц должен определяться выбранным контрагентом. Чтобы это

выполнить, необходимо написать процедуры на встроенном языке по следующей схеме:

```
Процедура ТоварыНоменклатураПриИзменении(Элемент)
СтрокаТЧ = ЭлементыФормы.Товары.ТекущиеДанные;
СтрокаТЧ.Цена=СтрокаТЧ.Номенклатура.ЦенаПокупки;
ТоварыКоличествоПриИзменении(Элемент);
КонецПроцедуры
```

Текст процедуры расчета суммы в строке в табличной части будет иметь следующую схему:

```
Процедура ТоварыКоличествоПриИзменении(Элемент)
СтрокаТЧ = ЭлементыФормы.Товары.ТекущиеДанные;
СтрокаТЧ.Сумма = СтрокаТЧ.Цена*СтрокаТЧ.Количество;
КонецПроцедуры
```

Текст процедуры расчета общей суммы по всему документу имеет следующий вид:

```
Процедура ПередЗаписью(Отказ,РежимЗаписи,РежимПроведения)
СуммаДокумента=Товары.Итог("Сумма");
КонецПроцедуры
```

Указанные процедуры необходимо «привязать» к соответствующим событиям элементов диалога формы. Обратите внимание на то, что в реальных конфигурациях обработчиков события даже для такого простого документа должно быть больше, чтобы организовать защиту, проверить заполнение реквизитов и т.п. Например, при смене контрагента необходимо проверить правильность заполнения данных на нового контактного сотрудника:

```
Процедура КонтрагентПриИзменении(Элемент)
Если Не КонтактноеЛицо.Пустая() Тогда
Если Контрагент <>КонтактноеЛицо.Владелец Тогда
КонтактноеЛицо=Справочники.КонтактныеЛица.ПустаяСсылка();
КонецЕсли;
КонецЕсли;
КонецПроцедуры
```

Необходимо обратить внимание на то, что ряд обработчиков событий могут быть описаны в модуле объекта, например - обработчик события «ПередЗаписью».

Кстати, в нашем случае общая сумма документа считается только после нажатия кнопки «Записать» в форме документа. Если желательно отображать сумму табличной части при ее заполнении данными, то в свойствах табличного поля «Товары» выделяют флажок «Подвал».

Контрольные вопросы:

- 1) Почему документ является основным понятием в системе «1С:Предприятие»?
- 2) Какими ключевыми характеристиками обладает документ в системе «1С:Предприятие»?
- 3) Какие наиболее важные реквизиты имеет документ?
- 4) Для чего необходимы свойства «Проведение», «Запись»?

## **Лекция 8. Журналы документов и документ «Расходная»**

**Содержание лекции:** объединение списка документов в программном продукте «1С:Предприятие».

**Цель лекции:** изучение понятия Журнала документов и как примера в нем документа «Расходная».

Для работы с уже выписанными документами мы применяли формы списка, которые не позволяли просматривать в одном списке документы разных видов. Если же необходимо выполнять эту процедуру, то в этом случае используют журналы.

В журнале может быть любое количество граф. При этом, создавая различные графы, необходимо из каждого документа выбрать по одному реквизиту различных типов. Эти графы и будут отображаться в созданном документе.

Создадим журнал «Складские», для чего зарегистрируем документ «Приходная»; создадим документ «Расходная» (см. ниже) и зарегистрируем его на закладке «Данные».

*Документ «Расходная».*

Новый документ в конфигурации можно сгенерировать следующим образом:

- 1) Создать новый документ.
- 2) Скопировать уже созданный документ.

Документ «Расходная» отличается от документа «Приходная» тем, что содержит две табличные части. Первая табличная часть - «Товары» - применяется для хранения перечня продаваемой номенклатуры, а вторая табличная часть - для учета издержек при отпуске, например, при доставке или погрузке, подъеме и т.д.

Если перечень издержек определен на продолжительное время, то для хранения таких долговременных значений применяется объект «перечисление».

Если предполагается создать перечень «Услуги», то необходимо щелкнуть мышкой на ветви «перечисление» и выбрать объект «Добавить».

Для создания документа «Расходная» напишем следующее:



*Реквизиты документа:*

- «Контрагент» (тип «СправочникСсылка.Контрагенты»);
- «Контакт» (тип «СправочникСсылка.КонтактныеЛица»);
- «Сотрудник» (тип «СправочникСсылка.ФизическиеЛица»);
- «СуммаДокумента» (тип «Число»).

*Табличная часть «Товары»:*

- номенклатура (тип СправочникСсылка.Номенклатура);
- количество (тип «Число» длина 10, точность 0);
- цена (тип «Число» длина 10, точность 2);
- сумма (тип «Число» длина 10, точность 2).

*Табличная часть «Услуги»:*

- услуга (тип ПеречислениеСсылка.Услуги);
- сумма (тип «Число» длина 10, точность 2).

Для печати созданного документа создаем с помощью конструктора печати печатную форму. Для этого используем закладки «Макеты». При этом необходимо выбрать следующие начальные установки:

- для установки способов отображения различных величин использовать свойства объекта «Формат»;
- для корректировки текста использовать процедуры;
- для дополнительного создания реквизита в форме документа «СуммаПрописью» и связыванию его с надписью в диалоге, необходимо в обработчик события «ПередЗаписью()» добавить следующую строчку:

*СуммаПрописью=ЧислоПрописью(СуммаДокумента,, "тенге,тенге, тенге, тиын, тиын, тиын")*

Создадим, настроив форму списка, две графы с названиями «Контрагент» и «ОбщСумма».

*Контрольные вопросы:*

- 1) Как в одном списке объединяются однотипные локументы в системе «1С:Предприятие»?
- 2) В каких случаях применяется объект «перечисление» в системе «1С:Предприятие»?
- 3) Можно ли в коде использовать знаки «» вместо "" ?

## Лекция 9. Регистры

**Содержание лекции:** основы работы со списками объектов для оперативного сохранения в программном продукте «1С:Предприятие».

**Цель лекции:** знакомство с описанием возможностей регистров для оперативного сохранения управления данными.

Основной задачей регистра сведений является сохранение важной информации, хранящейся в регистре. Регистры сведений, информация в которых содержит временную информацию, называются периодическими.

*Регистр сведений «Цены поставщиков».*

В предыдущем примере мы рассмотрели основы работы с регистром сведений, у которого был «Независимый» режим записи. Теперь рассмотрим пример, когда запись ведется регистратором. Создадим механизм, позволяющий отслеживать приходные цены номенклатуры по поставщикам.

Воссоздадим регистр сведений «ЦеныПоставщиков». Регистр сведений периодический, режим записи «Подчинение регистратору». У него два измерения «Поставщик», «Номенклатура» и один ресурс «Цена». Настройте форму списка и форму набора записей.

Так как запись ведется регистратором (по смыслу документом «Приходная»), необходимо модифицировать данный документ. А именно: разрешить проведение и вставить в модуль документа (объекта) процедуру проведения, текст которой приводится ниже.

```
Процедура ОбработкаПроведения(Отказ, Режим)
Для Каждого ТекСтрокаТовары Из Товары Цикл
Движение=Движения.ЦеныПоставщиков.Добавить();
Движение.Период=Дата;
Движение.Поставщик=Контрагент;
Движение.Номенклатура=ТекСтрокаТовары.Номенклатура;
Движение.Цена=ТекСтрокаТовары.Цена;
КонецЦикла;
Движения.ЦеныПоставщиков.Записать();
КонецПроцедуры
```

Текст процедуры можно ввести вручную, а можно воспользоваться конструктором. Для его вызова на закладке «Движения» необходимо нажать кнопку «Конструктор движений».

Проверьте работоспособность созданного вами механизма.

Выполните следующее задание:

- найти регистр сведений, содержащий данные о количестве рабочего времени в указанный день. Данному объекту дайте имя: «График». Он не должен быть периодическим.

### *Планы видов характеристик.*

Данный вид объекта имеет очень много схожих характеристик со справочником. Самое существенное его отличие (от справочника) в том, что для каждого элемента существует такое свойство как «Тип».

Создайте регистр сведений «СвойстваНоменклатуры». Определите у него два измерения:

- позиция тип «СправочникСсылка.Номенклатура»;
- свойство тип «ПланВидовХарактеристикСсылка.СвойстваОбъектов».

Определите один ресурс: значение тип «Характеристика.СвойстваОбъектов».

Настройте формы созданных вами объектов и проверьте их работу. При настройке формы для «включения» контроля типов (в самом простом варианте) потребуется определить свойства: «связь по типу» и «связь по владельцу» для поля ввода «Значение» в форме списка регистра сведений.

### *Регистры накопления.*

Различают два вида регистров накопления:

- регистры остатков;
- регистры оборотов.

### *Регистр «Остатки номенклатуры».*

Создадим регистр «ОстаткиНоменклатуры» (регистр остатков), определите у него одно измерение «Номенклатура», два ресурса «Количество», «Сумма», настройте формы списка.

Далее изменим документ «Приходная». Необходимо разрешить ему изменять данный регистр и, используя конструктор движений, переопределить обработку проведения.

### *Регистр накопления «Продажи».*

Создадим регистр оборотов «Продажи». Определите у него два измерения: «Номенклатура», «Контрагент», два ресурса: «Количество», «Сумма». Следует отметить, что наилучшего результата можно добиться в том случае, если структура измерений регистра расчета соответствует определенным в нем же измерениям перерасчетов.

Для получения полной картины (в рамках довольно простого примера) необходимо создать документ «НачислениеЗарплаты».

Далее потребуется определить процедуру «ОбработкаПроведения»:

Контрольные вопросы:

- 1) Что является основной задачей регистра сведений в системе «1С:Предприятие»?
- 2) Что общего и какие отличия между справочником и планом вида характеристик в системе «1С:Предприятие»?
- 3) Какими универсальными понятиями оперирует система на этапе конфигурирования?
- 4) Какие виды регистров накопления существуют и порядок работы с ними?

## Лекция 10. Отчеты и обработки

**Содержание лекции:** способы вывода имеющейся в «1С:Предприятие» информации.

**Цель лекции:** знакомство с приемами вывода данных, содержащихся в объектах «1С:Предприятие».

Для вывода информации используются «Отчеты». Обработки же могут использоваться для работы с данными и могут входить в конфигурацию или в виде отдельных файлов (внешние обработки).

Приемы работы с отчетами и обработками похожи и мало чем отличаются от приемов работы с другими объектами системы.

Знакомиться с порядком работы с отчетами и обработками будем на примере обработки «Печать расходных».

При создании данной обработки для нее необходимо определить два реквизита: «ДатаНачала» (тип «Дата», состав «Дата») и «ДатаОкончания» (тип «Дата», состав «Дата»).

В свойствах основной формы необходимо поставить флажок «Сохранять значения» и в соответствующем свойстве отметить, какие из значений будут сохраняться. Для того чтобы соответствующие кнопки (сохранения и восстановления значений) появились в командной панели, необходимо снять отметку и заново отметить в свойствах командной панели флажок «Автозаполнение»:

```
Процедура КнопкаВыполнитьНажатие(Элемент)
Выборка.Документы.Расходная.Выбрать(ДатаНачалаДонецДня (ДатаОкончания));
Пока Выборка.Следующий().Цикл
Выборка.ПолучитьОбъект().Печать();
КонецЦикла;
КонецПроцедуры
```

Чтобы данный пример стал работоспособным, необходимо произвести небольшую модификацию документа «Расходная».

Требуется определение процедуры «Печать()» из модуля формы документа перенести в модуль документа (модуль объекта), при этом потребуется указать ключевое слово «Экспорт». Также в модуле формы необходимо написать процедуру:

```
Процедура ОбработчикПечати(Элемент)
Печать();
КонецПроцедуры
```

Остается переопределить действие по нажатию на кнопку «Печать» (в диалоге формы документа).

Когда реализация какого-либо механизма строится на двух процедурах (процедура в модуле объекта и в модуле формы), очень важно определить какие части алгоритма, в какой процедуре реализуются. Исправьте реализованный выше пример таким образом, чтобы печатные формы документа открывались в одном окне табличного документа.

*Ввод на основании.*

Одним из режимов ввода новых объектов (элементов, групп элементов, справочников, документов, планов видов характеристик и т.д.) является режим, называемый «ввод на основании».

Во время работы пользователя с системой режим «ввод на основании» дает возможность вводить документы или элементы справочника, заполняя их реквизиты из других объектов информационной базы. При этом можно вводить документ одного вида на основании данных из документа другого вида, можно также заполнить элементы справочника на основании выбранного документа и т.д.

В частности, на закладке «Ввод на основании» документа «Расходная» следует указать, что основанием для данного документа будет являться документ «Приходная».

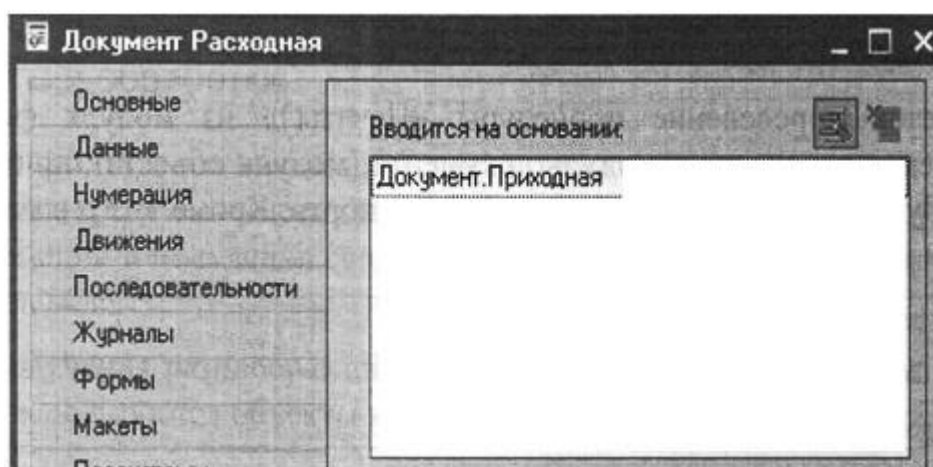


Рисунок 10 - Документ: Расходная

После этого для определения процедуры «ОбработкаЗаполнения()» можно воспользоваться подходящим конструктором (вызов находится на закладке «Ввод на основании»).

#### *Критерии отбора.*

Критерий отбора представляет собой правило «поиска» информации в справочниках, документах. Создание критерия отбора расширяет список predeterminedенных отборов в системе.

Поставим себе задачу: необходимо отбирать документы, в которых встречается интересующая нас номенклатурная позиция.

Для достижения этой цели создадим критерий отбора «Номенклатура». Определим тип критерия: «СправочникСсылка.Номенклатура».

На закладке «Состав» определим какие объекты (по значениям каких реквизитов) будут входить в результат отбора.

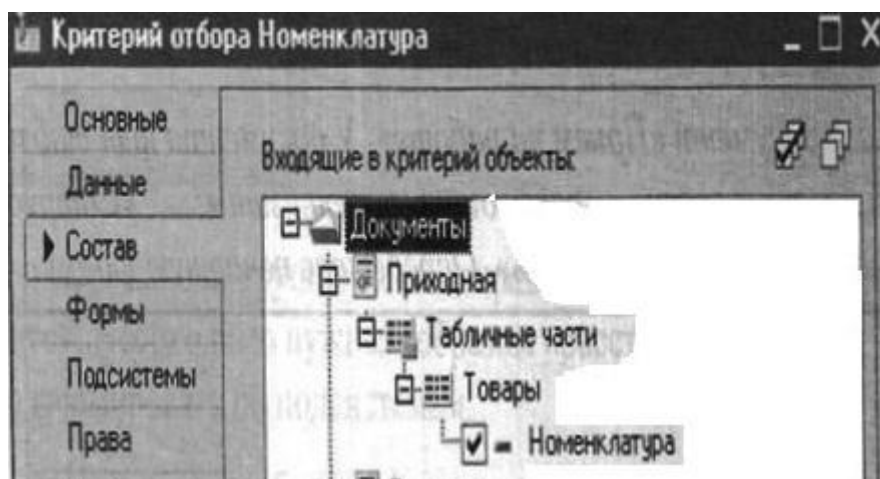


Рисунок 11 - Критерий отбора: Номенклатура

Для использования критерия можно в списке справочника «Номенклатура» воспользоваться кнопкой «Перейти» (с выбором критерия отбора), или в журнале документов «Торговые» организовать отбор по номенклатуре, используя появившийся predeterminedенный отбор.

#### *Организация отбора документов по сотруднику.*

Для организации отбора документов по сотруднику выполним следующие задания:

Задание: создайте критерий отбора «Сотрудник». Тип значения определите «СправочникСписок.ФизическиеЛица». В отбор должны входить документы «ПриемНаРаботу» по реквизиту «Сотрудник». После того как вы определите критерий отбора, модифицируйте форму элемента справочника «ФизическиеЛица». Добавьте новую страницу (заголовок «Документы»), на нее поместите элемент управления «ТабличноеПоле» (свойство «тип значения» у элемента установите в «КритерийОтбораСписок.Сотрудник»). В свой-

стве табличного поля «Связь по значению отбора» необходимо поставить значение «СправочникОбъект.Ссылка»;

Задание: создайте справочник «Должности», этот документ не содержит реквизиты, в нем отсутствуют табличные части, этот справочник создается без иерархии. Редактирование производится в списке. Создайте документ «Прием на работу». У данного документа нет табличных частей, но есть следующие реквизиты: «Сотрудник», «Подразделение», «Должность». Необходимо также определить печатную форму.

Контрольные вопросы:

- 1) Для чего используется объект «отчеты» в программном комплексе?
- 2) Как выполняется процедура ввод на основании в система «1С:Предприятие»?
3. Что представляет собой критерий отбора в системе «1С:Предприятие»?
- 4) Как осуществляется организация отбора документов по сотруднику?

## **Лекция 11. Администрирование**

**Содержание лекции:** введение в интеграцию и разделение доступа.

**Цель лекции:** изучение интерфейсов, ролей, пользователей с целью разделения доступов.

При работе пользователей с системой «1С:Предприятие» в сети важное значение приобретают два вопроса - интеграция и разделение доступа.

Конфигуратор содержит развитые средства администрирования, предназначенные для решения указанных задач. Другой, также не менее важной задачей, является формирование резервных копий. В клиент-серверном варианте могут использоваться брендовые системы баз данных, как то: MS SQL, Oracle и так далее.

*Определение интерфейсов, ролей и пользователей.*

Прежде всего, в процессе создания конфигурации, создается необходимое число типовых ролей и интерфейсов. Также необходимо в начале разработать карты систем доступа к различным объектам одной конфигурации или ряда конфигураций. Роль описывает полномочия определенной категории пользователей на доступ к информации, обрабатываемой системой.

Настройка интерфейса связана с определением состава команд главного меню и панелей инструментов. Для определения ролей и интерфейсов внутри ветви «Общие» существуют одноименные объекты. Порядок работы с ними довольно простой.

После создания нового интерфейса (щелчок правой клавишей мыши на соответствующей ветви дерева метаданных) необходимо определить состав

главного меню программы (меню является «одной из панелей» интерфейса).

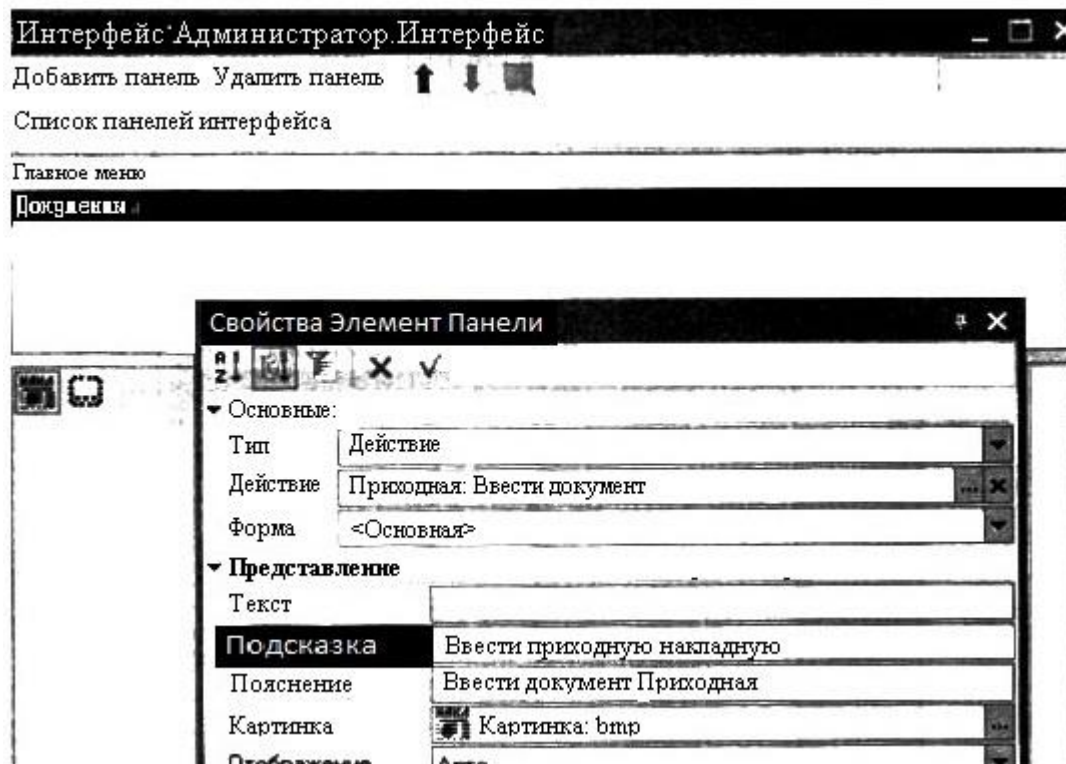


Рисунок 12 - Интерфейс: Администратор

Конструктор меню запускается автоматически после создания интерфейса. Слева отображаются группы меню, справа состав входящих в группу элементов. Необходимо нужным образом «расставить галочки». Меню можно формировать по подсистемам.

После того, как меню будет сформировано, можно добавлять любое количество панелей управления.

При определении интерфейса следует обратить внимание на его важное свойство «Переключаемый». В случае, если свойство отмечено, интерфейс будет относиться к переключаемым.

Контрольные вопросы:

- 1) Чем отличается друг от друга интеграция и разделение доступа в системе «1С:Предприятие»?
- 2) Для чего перед началом работы администрирования нужно разрабатывать карту доступа?
- 3) Каким образом производится настройка интерфейса?
- 4) Для каких целей используется свойство «переключаемый»?



## Лекция 12. Запросы

**Содержание лекции:** введение в язык запросов программного продукта «1С:Предприятие».

**Цель лекции:** изучение начальных понятий о языке запросов, в том числе на практическом примере.

Для формирования запросов, выборки и обработки результатов запросов в языке предусмотрен специальный набор объектов, с помощью которых выполняется формирование запроса, обход записей запроса и т.д. Рассмотрим в качестве первого примера отчет «Знакомство с Запросом». При создании отчета определим у него следующие реквизиты:

НачПериода тип «Дата» состав «Дата+Время»  
КонПериода тип «Дата» состав «Дата+Время»  
ЗначениеСсылка тип «СправочникСсылка»  
ЗначениеЧисло тип «Число»  
ЗначениеСтрока тип «Строка»  
ЗначениеБулево тип «Булево»  
Определена табличная часть «ТЧ»:  
Реквизит табличной части «ТЧСсылка», тип «СправочникСсылка»

При настройке диалога основной формы отчета необходимо разместить элементы управления, связанные с реквизитами и табличной частью отчета на второй закладке. На первую необходимо поместить поле текстового документа «ПолеТекстаЗапроса» и табличное поле «ПолеРезультатаЗапроса».

У поля текстового документа в свойстве «Расширение» проставить значение «Язык запросов». Установка данного свойства в указанное значение позволит из контекстного меню данного элемента управления вызывать конструктор запроса и кроме этого при просмотре текста запроса в данном элементе управления выделяться синтаксические конструкции языка запросов.

Текст процедуры, которая выполняется при нажатии на кнопку «Сформировать» приводится ниже (рассмотрим не весь текст модуля):

```
Процедура КнопкаСформироватьНажатие(Элемент)  
Запрос=Новый Запрос;  
Запрос.Текст = ЭлементыФормы.ПолеТекстаЗапроса.ПолучитьТекст();  
Запрос.УстановитьПараметрС"ТЧ",  
ТЧ.ВыгрузитьКолонку("ТЧСсылка");  
Запрос.УстановитьПараметр("НачПериода".НачПериода);  
Запрос.УстановитьПараметр("КонПериода",КонПериода);  
Запрос.УстановитьПараметр("ЗначениеСсылка",ЗначениеСсылка);  
Запрос.УстановитьПараметр("ЗначениеЧисло",ЗначениеЧисло);  
Запрос.УстановитьПараметр("ЗначениеБулево",ЗначениеБулево);
```

```

Запрос.УстановитьПараметр("ЗначениеСтрока",ЗначениеСтрока);
РезультатЗапроса = Запрос.Выполнить();
ЭлементыФормы.ПолеРезультатаЗапроса.Значение=
РезультатЗапроса.Выгрузить();
ЭлементыФормы.ПолеРезультатаЗапроса.
ЧередованиеЦветовСтрок=Истина;
ЭлементыФормы.ПолеРезультатаЗапроса.СоздатьКолонки();
Для Каждого Колонка из ЭлементыФормы.ПолеРезультатаЗапроса,
Колонки Цикл Колонка. Ширина= 15;
КонецЦикла;
КонецПроцедуры

```

Контрольные вопросы:

- 1) Для чего нужен язык запросов в системе «1С:Предприятие»?
- 2) Какие компоненты используются при настройке основной формы?
- 3) Можно ли в коде использовать английские команды вперемешку с русскими?

### **Лекция 13. Работа с конструктором запроса**

**Содержание лекции:** автоматизация работы с языком запросов в программном продукте «1С:Предприятие».

**Цель лекции:** знакомство с понятием работы с конструктором языка запросов.

Создавать тексты запросов можно с помощью конструктора запросов или же «вручную».

В рамках данного курса знакомиться как с таблицами базы данных, так и с конструкциями языка запросов будем с использованием конструктора запросов. Данный конструктор может вызываться как в режиме конфигурирования, так и в пользовательском режиме.

Для открытия конструктора запросов в пользовательском режиме необходимо выполнить соответствующую команду в контекстном меню поля текстового документа. На первой закладке конструктора можно просмотреть существующие на данный момент таблицы в системе (левая часть озаглавленная «База данных»).

Таблицы, данные из которых будут получаться запросом переносятся в область «Таблицы», необходимые запросу поля переносятся в область «Поля».

В любой момент работы с конструктором можно смотреть получаемый текст запроса. Для выполнения этой задачи необходимо нажать кнопку «Запрос», расположенную в левом нижнем углу формы конструктора.

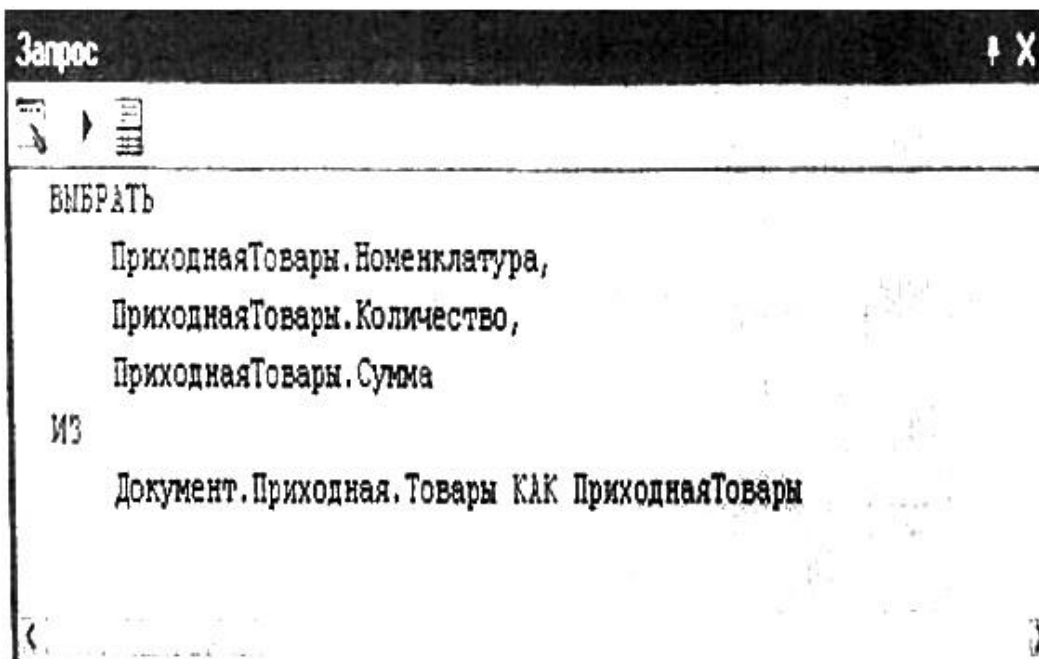


Рисунок 13 - Запрос

Кроме этого в режиме «Предприятие» можно посмотреть результаты выполнения «предварительного» запроса. Для этого необходимо нажать на кнопку «Выполнить запрос».

Запрос

ВЫБРАТЬ  
 ПриходнаяТовары.Номенклатура,  
 ПриходнаяТовары.Количество,  
 ПриходнаяТовары.Сумма  
 ИЗ  
 Документ.Приходная.Товары КАК ПриходнаяТовары

Номенклатура	Количество	Сумма
--- Kohinor тм	10,00	70,00
--- Паркер "Golg"	10,00	1 000,00
--- Kohinor тм	60,00	390,00
--- Паркер "Golg"	40,00	3 800,00
--- BOSCH KGS 3760 IE	10,00	200 000,00
--- ELECTROLUX ER 9007 B	10,00	300 000,00
--- Атлант МХМ 1704-00	20,00	240 000,00
--- Ardo TL 1000 EX-1	5,00	49 000,00

Рисунок 14 - Выполнение запроса

Если проанализировать данные, полученные при выполнении запроса, то можно встретить «повторы» номенклатурных позиций. Если требуется «свернуть» результат запроса, то для этой цели можно использовать закладку «Группировка» конструктора запроса.

При определении группировок необходимо придерживаться следующего правила: все поля выборки запроса делятся на поля по которым производится группировка (свертка), поля вложенных таблиц (относительно тех, по которым ведется группировка) и на агрегатные функции.

Если данные, получаемые запросом должны выбираться по какому-либо условию, то в этом случае может потребоваться применить закладку «Условия».

Обратите внимание на тот факт, что в случае если условие определено таким же образом как на рисунке, то запрос выполняться не будет.

Исправить ситуацию можно двумя способами:

- переопределив условие, отметив флажок «П...»;
- воспользовавшись возможностью изменения самого текста запроса, получаемого при нажатии на кнопку «Текст запроса».

Само изменение будет заключаться в том, что необходимо перед числом «10» убрать символ «&». Этим символом в тексте запроса определяются параметры запроса, в которые в дальнейшем (но перед выполнением запроса) должны быть записаны какие-либо значения. На закладке «Дополнительно» можно отметить ряд флагов (касающихся ключевого слова «Выбрать» языка запросов) и определить состав таблиц предназначенных для изменения.

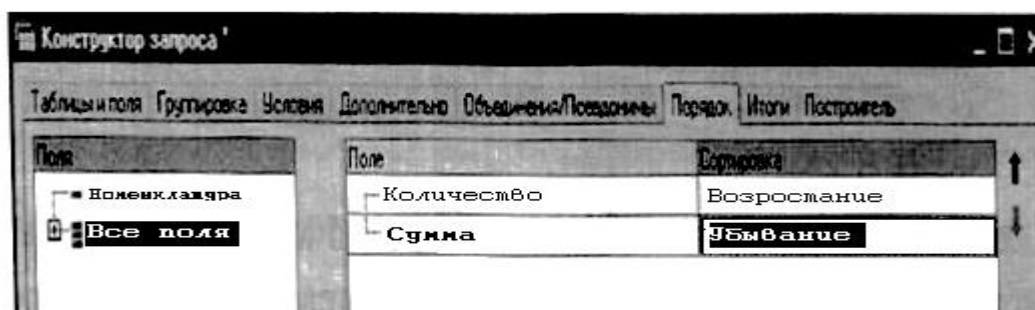


Рисунок 15 - Конструктор запроса

На закладке «Порядок» можно определить порядок сортировки записей в результате запроса. При определении раздела «Итоги» следует быть готовым к тому, что в результате запроса появятся дополнительные итоговые записи. Вместе с этими записями результат запроса становится иерархическим. Допустимо указание нескольких типов итогов.

#### *Особенности работы с виртуальными таблицами.*

Работа с виртуальными таблицами имеет некоторую особенность. Заключается она в том, что в общем случае необходимо обязательное использование параметров виртуальных таблиц. Правильное использование данных

параметров в некоторых случаях напрямую сказывается на оптимальности и быстродействии выполнения запроса, например можно использовать следующие параметры: Таблицы и поля (Группировка Условия\_ Дополнения Объявления Псевдонимы Порядок Итог).

Для задания параметров необходимо нажать на кнопку «Параметры виртуальной таблицы» и в открывшемся диалоговом окне указать их значения.

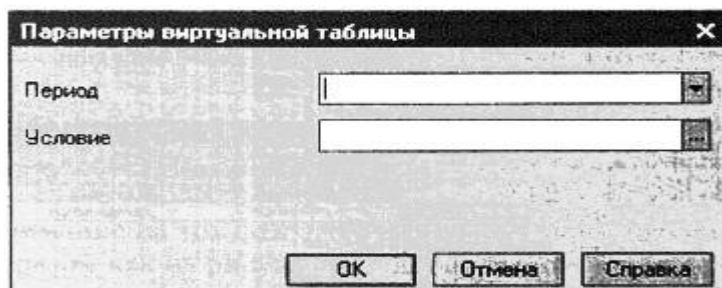


Рисунок 16 - Параметры виртуальной таблицы

Состав параметров напрямую зависит от выбранной виртуальной таблицы. При определении условия можно повторно (вложено) вызывать конструктор запроса для определения подзапроса.

#### *Построение запросов по нескольким таблицам.*

В реальной практике очень редко бывает ситуация, когда данные, которые необходимо получить определены в одной таблице. Чаще всего они находятся в различных таблицах, но должны быть получены вместе (при этом должна соблюдаться «связность» данных).

На закладке «Таблицы и поля» можно одновременно указывать несколько таблиц-источников данных. Для их увязки можно использовать раздел «Условия», либо можно использовать закладку «Связи». Данная закладка появляется в конструкторе запроса в случае, если в качестве источников данных выбрана более чем одна таблица.

Если никакой связи между явно определенными источниками не определять, то в результат запроса войдет полная комбинация записей из определенных таблиц.

#### Контрольные вопросы:

- 1) Какие способы существуют для создания текста запроса?
- 2) Для чего предусмотрена работа конструктора в пользовательском режиме?
- 3) Для чего необходимо создавать виртуальную таблицу?
- 4) Как построить запрос по нескольким таблицам?

## Лекция 14. Хранилище

**Содержание лекции:** введение в технологию разделения доступа к объектам конфигурации.

**Цель лекции:** изучение работы с хранилищем информации в программном комплексе «1С: Предприятие».

Хранилище конфигурации создается для разделения доступа к объектам конфигурации. Оно создается в общедоступном каталоге (общем сетевом ресурсе). Для его создания необходимо выбрать пункт «Конфигурация - Хранилище конфигурации - Создать хранилище». В появившемся диалоге необходимо указать путь к каталогу (он не должен содержать другого хранилища), указать имя и пароль администратора хранилища. После нажатия кнопки «ОК» конфигуратор производит анализ данных, формирование структуры объектов и запись данных в указанный каталог. По окончании формирования хранилища конфигуратор предлагает пользователю подключиться к хранилищу.

Если по указанному адресу (каталогу) уже зарегистрировано хранилище конфигурации, то конфигуратор выводит сообщение о невозможности создания хранилища.

### *Администрирование хранилища.*

Для того чтобы несколько разработчиков могли работать с хранилищем, они должны быть зарегистрированы, то есть определены администратором.

Для решения задач администрирования необходимо выбрать пункт меню «Конфигурация - Хранилище конфигураций - Администрирование хранилища». В открывшемся окне можно определять пользователей, просматривать текущие подключения, отменять захват объектов пользователями.

### *Работа с хранилищем конфигурации.*

Для работы с хранилищем к нему необходимо подключиться. С этой целью необходимо выбрать пункт меню «Конфигурация - Подключиться к хранилищу».

В диалоге подключения требуется указать путь к хранилищу, имя пользователя и пароль. С одним именем можно произвести только одно подключение.

После открытия хранилища, Конфигуратор выводит окно конфигурации. В правой части этого окна выводятся пиктограммы, указывающие статус объекта. Если требуется добавить/удалить объект любого вида (справочник, документ), то предварительно необходимо захватить конфигурацию (необязательно со всеми подчиненными объектами).

Если объект модифицирован другим пользователем хранилища, его новую версию можно получить, указав нужные объекты и выполнив команду «Получить из хранилища». Для внесения модифицированного объекта в хранилище требуется выполнить команду «Поместить в хранилище». Если не

требуется сохранять проведенные изменения необходимо выполнить команду «Отменить захват в хранилище».

#### *Поставка конфигурации, поддержка.*

Время от времени конфигурации могут меняться из-за различных причин, например: изменения в законодательстве, внесение новых возможностей и т.д. В связи с этим возникают понятия: «поддержка» и «поставка» конфигурации.

*Поставка.* Различают полную поставку и поставку обновлений. Полная поставка представляет собой файл конфигурации следующего формата «\*.cf». Поставка обновлений - файл обновлений следующего формата «\*.cfu».

*Поддержка.* Поддержка конфигурации заключается в способности конфигурации быть обновляемой средствами конфигуратора с использованием файлов поставки.

Настройка поставки заключается в указании правил поставщика на изменение объектов конфигурации разработчиками, осуществляющими поддержку конфигураций конечных пользователей.

Для настройки правил поставки необходимо выбрать пункт меню «Конфигурация — Поставка — Настройка поставки». В открывшемся окне для каждого объекта следует указать правило изменения. Кроме этого можно выбрать: включать исходные тексты модулей (т.е. делать их доступными) или включать их в скомпилированном виде. Флажок «Файл поставки» может использоваться для обновления и его можно снимать, когда предполагается использовать полученный файл как промежуточный (требуется произвести ряд последовательных обновлений).

После определения правил поставки необходимо сформировать соответствующие файлы. Для этого необходимо выбрать пункт меню «Конфигурация - Поставка конфигурации — Создать файлы поставки и обновления конфигурации». Перед этим требуется установка значений свойств конфигурации «Поставщик» и «Номер релиза». В открывшемся окне можно указать каталог файлов поставки, имена файлов. Для формирования файла обновления требуется включить в формируемый файл предыдущие обновления конфигурации. Файлы будут созданы после нажатия кнопки «Выполнить».

После установки конфигурации из файлов поставки она автоматически становится на режим поддержки. Иногда требуется изменить типовую конфигурацию, чтобы учесть требования конкретного пользователя. В этих случаях необходимо полностью или частично отказаться от поддержки некоторых объектов. Для проведения необходимых изменений потребуется выбрать в меню: «Конфигурация/Поддержка/Настройка поддержки». В открывшемся окне можно включить возможность внесения изменений. В этом случае появится сообщение о том, что обновление конфигурации автоматически производиться не будет.

Если по каким-либо обстоятельствам необходимо удалить режим поддержки всей конфигурации, то вновь включить режим поддержки можно лишь

загрузив конфигурацию из созданных файлов поставки и обновления конфигурации. Далее необходимо заполнить регистр адресации и проверить используемый механизм на практике.

Контрольные вопросы:

- 1) Для чего и как создается хранилище конфигурации?»?
- 2) Как несколько разработчиков работают с хранилищем конфигурации?
- 3) Как осуществляется поставка и поддержка конфигурации?

## **Лекция 15. Бизнес-процессы**

**Содержание лекции:** введение в бизнес-процессы в разрезе программного продукта «1С:Предприятие».

**Цель лекции:** изучение понятий жесткой, параллельной, свободной и условной маршрутизации бизнес-процессов.

Автоматизация бизнес-процессов может быть реализована в ERP - и MES - системах. Бизнес-процессы в «1С:Предприятии» допускают следующие виды маршрутизации:

- жесткая; бизнес-процесс имеет строгую карту маршрута, не включающую в себя условных и параллельных переходов, с жестко определенными адресатами для каждой точки маршрута. Данный вид не допускает свободной и условной маршрутизации;

- параллельная; карта маршрута предусматривает разделение бизнес-процесса на параллельные ветви с возможностью последующего слияния (ожидания). Продвижение бизнес-процесса по каждой из параллельных ветвей происходит независимо по мере выполнения соответствующих задач;

- свободная; адресаты точки карты маршрута бизнес-процесса не установлены и определяются программно или интерактивно в течение жизненного цикла бизнес-процесса;

- условная; карта маршрута предусматривает проверку условий и переход по соответствующим ветвям. Переходы могут быть как бинарными (условие), так и множественными (выбор варианта).

Как правило, в реальных картах бизнес-процессов встречаются все эти типы маршрутизации.

Другим важным понятием является «Адресация». Основное назначение системы адресации - обеспечить возможность не только персональной, но и ролевой адресации задач участникам бизнес-процессов. Для хранения данных об адресации используется регистр сведений (он не должен быть периодическим – это правило обязательно!).

Рассмотрим следующий пример: необходимо проконтролировать технологию действий по выдаче денежных средств подотчетному лицу. Этапы этой технологии следующие:



1) Подотчетное лицо подает заявку, в которой указывается сумма и цель получения денежных средств.

2) Заявка рассматривается главным бухгалтером.

3) В случае утверждения заявки главным бухгалтером, кассир выдает денежные средства из кассы, при этом выписывается документ «Расходный кассовый ордер».

4) Подотчетное лицо должно предоставить в бухгалтерию отчет об использованных денежных средствах.

Реализацию механизма начнем с создания регистра адресации (регистр сведений «СведенияПоАдресации»). В справочник «Должности» необходимо добавить два обязательных значения – «ГлавныйБухгалтер» и «Кассир». В справочнике «Подразделения» должно быть значение «Бухгалтерия».

Далее необходимо настроить адресацию у объекта «Задача» - после создания соответствующего объекта конфигурации. В свойстве «Текущий исполнитель» необходимо выбрать параметр сеанса, имеющий тип «СправочникСсылка.ФизическиеЛица».

В реквизитах формы дополнительно определены:

- ФизЛицо (тип «СправочникСсылка.ФизическиеЛица»)

- ПоИсполнителю (тип «Булево»)

Текст модуля формы списка:

Процедура ФизЛицоПриИзменении(Элемент)

ПараметрыСеанса.Пользователь=ФизЛицо;

ЗадачаСписок.Обновить();

КонецПроцедуры

Процедура ПриОткрытии()

ФизЛицо=ПараметрыСеанса.Пользователь;

КонецПроцедуры

Процедура ФлажокПриИзменении(Элемент)

Если ПоИсполнителю Тогда

ЭлементыФормы.ЗадачаСписок.ОтображениеЗадач=

РежимСпискаЗадач.ПоИсполнителю;

Иначе

ЭлементыФормы.ЗадачаСписок.ОтображениеЗадач=

РежимСпискаЗадач.ВсеЗадачи;

КонецЕсли;

КонецПроцедуры

ПоИсполнителю=Истина;

После определения задачи можно приступать к определению бизнес-процесса.

Реквизиты бизнес процесса:

- Назначение (тип «Строка»)
- Сумма (тип «Число»)
- Согласовано (тип «Булево»)
- Комментарий (тип «Строка»)
- РКО (тип «ДокументСсылка.РасходныйКассовыйОрдер»)
- Подотчетник (тип «СправочникСсылка.ФизическиеЛица»)

С помощью редактора определим карту маршрута (на закладке <Прочие> конструктора бизнес-процесса). Текст обработчиков событий модуля бизнес - процесса имеет следующий вид:

Процедура Условие1ПроверкаУсловия(ТочкаМаршрутаБизнесПроцесса, Результат) Результат=Согласовано;

КонецПроцедуры

Процедура СтартПередСтартом(ТочкаМаршрутаБизнесПроцесса, Отказ)

Если Сумма=0 Тогда

Сообщить("Сумма не может равняться нулю!!!");

Отказ=Истина;

КонецЕсли;

КонецПроцедуры

Процедура

Действие1

ОбработкаИнтерактивнойАктивации

(ТочкаМаршрутаБизнесПроцесса, Задача,

СтандартнаяОбработка)

Форма=ПолучитьФорму("ФормаДляСогласования");

Форма.Задача=Задача;

Форма.Открыть();

СтандартнаяОбработка=Ложь;

КонецПроцедуры

Процедура

Действие20обработкаИнтерактивнойАктивации

(ТочкаМаршрутаБизнесПроцесса, Задача, СтандартнаяОбработка)

Если РКО.Пустая() Тогда

Док=Документы.РасходныйКассовыйОрдер.СоздатьДокумент();

Док.УстановитьНовыйНомер();

Док.Дата=РабочаяДата;

Док.Сумма=Сумма;

Док.Валюта=Справочник.Валюты.НайтиПоКоду(1810);

Док.Записать();

РКО=Док.Ссылка;

КонецЕсли;

Форма=РКО. ПолучитьФорму());  
Форма.Задача=Задача;  
Форма.Открыть());  
СтандартнаяОбработка=Ложь;  
КонецПроцедуры

Для обеспечения работоспособности бизнес-процесса потребуется настроить ряд форм. В форме документа расходный кассовый ордер необходимо добавить реквизит «Задача». В модуле формы определить следующий обработчик событий:

Процедура ПриЗаписи(Отказ)  
Если Не Задача.Выполнена Тогда  
ОбъектЗадачи=Задача.ПолучитьОбъект());  
Попытка  
ОбъектЗадачи.ВыполнитьЗадачу ();  
Исключение  
Сообщить("Не удалось выполнить задачу!!!");  
Отказ=Истина;  
КонецПопытки;  
КонецЕсли;  
КонецПроцедуры

Кроме этого у самого бизнес-процесса необходимо определить две формы: основная форма (в ней закрыты от редактирования поля ввода «Подотчетник», «Комментарий», «Согласовано» и «РКО»).

На второй закладке расположен элемент управления «Поле графической схемы» (имя «ПолеГС») и в модуле формы определен следующий обработчик события:

Процедура ПриОткрытии()  
Если ЭтоНовый() Тогда  
Подотчетник=ПараметрыСеанса.Пользователь;  
КонецЕсли;  
ЭлементыФормы.ПолеГС.УстановитьСхему  
(ЭтотОбъект.ПолучитьКартуМаршрута());  
КонецПроцедуры

Кроме этого у бизнес-процесса определена форма «ФормаДляСогласования» (она будет открываться при активизации задачи согласования с главным бухгалтером).

Модуль данной формы бизнес-процесса имеет следующий вид:  
Перем ФлагЗаписи;

Процедура ПриОткрытии()  
Если Не Задача.Выполнена Тогда Согласовано=Истина;  
КонецЕсли;  
КонецПроцедуры  
Процедура ПриЗакрытии()  
Если ФлагЗаписи Тогда  
ОбъектЗадачи=Задача ,ПолучитьОбъект();  
Попытка  
ОбъектЗадачи.ВыполнитьЗадачу();  
Исключение  
Сообщить("Не удалось выполнить задачу!!! ");  
Отказ=Истина;  
КонецПопытки;  
КонецЕсли;  
КонецПроцедуры  
Процедура ПередЗаписью(Отказ)  
Если Не Задача.Выполнена Тогда ФлагЗаписи=Истина;  
КонецЕсли;  
КонецПроцедуры  
ФлагЗаписи=Ложь;

Далее необходимо заполнить регистр адресации и проверить используемый механизм на практике. В качестве самостоятельного задания следует реализовать бизнес-процесс, определяющий и реализовывающий следующую технологию действий:

- работник отдела кадров вводит данные о кандидате на определенную должность в определенном подразделении;
- кандидат проходит собеседование у руководителя отдела, в который он желает устроиться на работу;
- в случае успешного прохождения собеседования, работник отдела кадров производит оформление документа «Прием на работу» с внесением данных кандидата на должность;
- работник отдела кадров вводит данные о кандидате на должность в определенном подразделении кандидата в справочник «Физические лица».

Контрольные вопросы:

- 1) Какие виды маршрутизации используются в бизнес-процессах в системе «1С:Предприятие»?
- 2) Каково основное назначение процедуры «Адресация», ее свойство и виды в системе «1С:Предприятие»?
- 3) Какими универсальными понятиями оперирует система на этапе конфигурирования?
- 4) Привести другой пример бизнес-процесса в свете программного комплекса «1С:Предприятие»?



## Список литературы

- 1 Дэниел О'Лири. ERP-системы. Современное планирование и управление ресурсами предприятия. - М: Вершина, 2004.
- 2 Рюдигер Кречмер, Вольфганг Вейс. Разработка приложений SAP R/3 на языке АВАР/4.
- 3 С.В.Питеркин, Н.А.Оладов, Д.В.Исаев. Точно вовремя для России. Практика применения ERP-систем. Альпина Пабlishер, 2010.
- 4 Баронов В.В. Автоматизация управления предприятием. - М., Инфа-М, 2000. - 293 с.
- 5 Харитонов С.А. 1С: Бухгалтерия 8 для начинающих: пособие / С.А. Харитонов. - Санкт-Петербург : Питер, 2009. - 384 с.

Жанат Галимовна Аренбаева,  
Мажит Сулейменович Ибраев

ВВЕДЕНИЕ В КОНФИГУРИРОВАНИЕ ПАКЕТА 1С:ПРЕДПРИЯТИЕ

Конспект лекций для студентов специальности  
5В070300 - Информационные системы

Редактор Н.М.Голева  
Специалист по стандартизации Н.К. Молдабекова

Подписано в печать \_\_. \_\_. \_\_.  
Тираж 30 экз.  
Объем 2,9. уч.-изд. л.

Формат 60X84 1/16  
Бумага типографская №1  
Заказ \_\_\_\_\_ Цена 1450 тг.

Копировально-множительное бюро  
некоммерческого акционерного общества  
«Алматинский университет энергетики и связи»  
050013 Алматы, Байтурсынова, 126

Некоммерческое акционерное общество  
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ  
Кафедра Информационные системы

УТВЕРЖДАЮ  
Проректор по УМР  
\_\_\_\_\_ С. В. Коньшин  
" \_\_\_\_ " \_\_\_\_\_ 2016 г.

**Введение в конфигурирование пакета 1С:Предприятие**

Конспект лекций для студентов специальности  
5В070300 -Информационные системы

СОГЛАСОВАНО

Начальник УМО

\_\_\_\_\_ М.А. Мустафин  
" \_\_\_\_ " \_\_\_\_\_ 2016г

Рассмотрено и одобрено на  
заседании кафедры ИС  
протокол № 9  
от 02.03.2016 г.  
зав. кафедрой ИС

\_\_\_\_\_ Ш.И. Имангалиев

Председатель ОУМК

\_\_\_\_\_ Б.К. Курпенов  
" \_\_\_\_ " \_\_\_\_\_ 2016г

Составители:

\_\_\_\_\_ Ж.Г. Аренбаева,  
\_\_\_\_\_ М. С. Ибраев

Редактор АУЭС

\_\_\_\_\_ Н.М. Голева  
" \_\_\_\_ " \_\_\_\_\_ 2016г

Специалист по стандартизации

\_\_\_\_\_ Н.К. Молдабекова  
" \_\_\_\_ " \_\_\_\_\_ 2016г

Алматы 2016