МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ИВАНОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Шуйский филиал ИвГУ

Кафедра математики, информатики и методики обучения

ОДОБРЕНО:

Руководитель ОП

(подпись) С.В. Белов

«21»февраля 2023 г.

Рабочая программа дисциплины

ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

Уровень высшего образования:	бакалавриат
Квалификация выпускника:	бакалавр
Направление подготовки:	44.03.01 Педагогическое образование
Направленность (профиль) образовательной программы:	Информационные технологии и робототехника

Шуя

1. Цели освоения дисциплины

Дисциплина «Прикладноепрограммирование» имеет своей цельюподготовку квалифицированных учителей информатики, владеющих познаниями и практическими навыками в области создания программ прикладного назначения; ознакомление студентов со средствами построения приложений на платформе 1С:Предприятие 8.3.

2. Место дисциплины в структуре ОП

Дисциплина «Прикладное программирование» является дисциплиной по выбору, относится к формируемой участниками образовательных отношений части блока дисциплин Б1, входит в предметно-методический модуль.

Успешное освоение данной дисциплины будет способствовать готовности обучающихся к освоению дисциплин: «Основы искусственного интеллекта», «Логическое программирование» и успешному прохождению следующих практик: «Производственная практика, ранняя преподавательская практика по предметно-методическому модулю», «Производственная практика, педагогическая (комплексная) практика по предметно-методическому модулю», «Производственная практика, научно-исследовательская работа по предметно-методическому модулю», подготовке к сдаче и сдаче государственного экзамена, выполнению и защите выпускной квалификационной работы.

Студент, приступающий к изучению данной дисциплины, должен:

Знать:

- особенности системного и критического мышления;
- основы современных технологий сбора, обработки и представления информации;
- основные понятия и термины, связанные с алгоритмизацией и программированием.

Уметь:

- анализировать источники информации с точки зрения временных и пространственных условий их возникновения;
- получать, обрабатывать и анализировать информацию из различных источников;
- использовать современные информационно-коммуникационные технологии (включая пакеты прикладных программ, локальные и глобальные компьютерные сети) для сбора, обработки и анализа информации;
- работать с математической и технической литературой;
- использовать программную поддержку курса и оценивать ее методическую целесообразность.

Владеть:

- навыками сопоставления разных источников информации с целью выявления их противоречий и поиска достоверных суждений;
- основными методами математической обработки информации;
- методами математического доказательства гипотез;
- базовыми понятиями и терминами в области математики и информатики;

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

3.1. Компетенции, формированию которых способствует данная дисциплина

При реализации дисциплины формируются следующие компетенции в соответствии с ФГОС ВОпо данному направлению подготовки:

а) универсальные компетенции (УК) и индикаторы их достижения:

Наименование категории (группы) универсальных компетенций	Код и наименование универсальной компетенции выпускника	Код и наименование индикатора достижения универсальной компетенции
Системное и критическое мышление	УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1. Демонстрирует знание особенностей системного и критического мышления и готовность к нему. УК-1.2. Применяет логические формы и процедуры, способен к рефлексии по поводу собственной и чужой мыслительной деятельности. УК-1.3. Анализирует источник информации с точки зрения временных и пространственных условий его возникновения. УК-1.4. Анализирует ранее сложившиеся в науке оценки информации. УК-1.5. Сопоставляет разные источники информации с целью выявления их противоречий и поиска достоверных суждений. УК-1.6. Аргументировано формирует собственное суждение и оценку информации, принимает обоснованное решение. УК-1.7. Определяет практические последствия предложенного решения задачи.

б) общепрофессиональные компетенции (ОПК) и индикаторы их достижения:

–

в) профессиональные компетенции (ПК) и индикаторы их достижения:

Код и наименование профессиональной компетенции выпускника	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ опыта)
Тип задач профессиональной деятельности: педагогический		
ПК-6. Способен выделять структурные элементы, входящие в систему познания предметной области (в соответствии с профилем и уровнем обучения), анализировать их в единстве содержания, формы и выполняемых	ПК-6.1. Выделяет и анализирует единицы различных уровней знаковой математической структуры в единстве их содержания, формы и функций. ПК-6.2. Выделяет и анализирует закономерности и явления информатики как науки в их структурном и функциональном единстве с информационно-коммуникационными технологиями. ПК-6.3. Знает и умеет анализировать	Профессиональный стандарт 01.001 «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель,

функций	аксиомы, определения и математические доказательства в единстве с прикладными аспектами использования.	учитель)»
ПК-7. Способен определять собственную позицию относительно дискуссионных проблем предметной области (в соответствии с профилем и уровнем обучения)	ПК-7.1. Проявляет мировоззренческую рефлексию при анализе проблем и тенденций в области информационных технологий и робототехники. ПК-7.2. Проявляет способность аргументированно, логические верно и ясно выражать свою позицию по обсуждаемым дискуссионным проблемам в сочетании с готовностью к конструктивному диалогу и толерантному восприятию иных точек зрения. ПК-7.3. Выделяет концептуальную основу дискуссий в области информационных технологий и робототехники, соотносит с нею свои мировоззренческие установки, гражданскую позицию и социальную мотивацию.	Профессиональный стандарт 01.001 «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)»

3.2. Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения формируемых компетенций

В результате освоения дисциплины обучающийся должен:

Знать:

−основы современных технологий сбора, обработки и представления информации (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

−основные понятия информатики, их системно-структурные связи в контексте развития информационно-коммуникационных технологий (ПК-6 (индикаторы – 1, 2, 3);

−теоретические сведения об архитектуре современного компьютера (ПК-6 (индикаторы – 1, 2, 3);

−важнейшие дискуссионные проблемы в научном освоении информатики (трудные вопросы развития и исторического становления научной области «Информатика», прикладные аспекты информатики) (ПК-7 (индикаторы – 1, 2, 3).

Уметь:

– использовать современные информационно-коммуникационные технологии (включая пакеты прикладных программ, локальные и глобальные компьютерные сети) для сбора, обработки и анализа информации (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– использовать программную поддержку курса и оценивать ее методическую целесообразность (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– использовать различного рода прикладные программы и языки программирования для решения учебных задач (ПК-6 (индикаторы – 1, 2, 3);

– анализировать и характеризовать готовый программный продукт в контексте решения учебных задач (ПК-6 (индикаторы – 1, 2, 3);

– выявлять в конкретной решенной задаче ошибки и недочеты, определять их природу (ПК-6 (индикаторы – 1, 2, 3);

–работать с технической и специальной литературой (ПК-7 (индикаторы – 1, 2, 3).

Владеть:

– навыками сопоставления разных источников информации с целью выявления их противоречий и поиска достоверных суждений (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– технологиями алгоритмизации и программирования (ПК-6 (индикаторы – 1, 2, 3);

– спорными и противоречивыми проблемами в области построения научных теорий информатики (ПК-7 (индикаторы – 1, 2, 3);

– механизмами оценки применимости научных положений в прикладных областях (ПК-7 (индикаторы – 1, 2, 3).

4.Объём и содержание дисциплины

Очная форма обучения

Объём дисциплины составляет 2 зачётные единицы (54 час.)

Контактная работа 22,75

Самостоятельная работа 31,5 час.

Практическая подготовка 38,75 час.

Виды учебных занятий и формы промежуточной аттестации	7 сем.	
	Контактная работа с преподавателем	Самостоятельная работа обучающегося
Лекции	10,5	31,25
Практическиеработы	12	
Зачет	0,25	-
Всего	22,75	31,25

4.1.Содержание дисциплины по разделам (темам), соотнесенное с видами и объемом учебных занятий

Очная форма обучения

№ п/п	Раздел (тема) дисциплины	Семестр	Виды учебных занятий, включая самостоятельную работу обучающихся и объем (в часах)				Формы текущего контроля успеваемости Формы промежуточной аттестации
			Занятия лекционного типа	Занятия семинарского типа		Самостоятельная работа обучающегося	
				Лаб.	Пр.		
1	Создание и настройка информационной базы данных на платформе 1С: Предприятие 8.3	7	1,5		1,5 ПП	3,5	доклад
2	Основные объекты системы	7	1,5		1,5 ПП	3	лабораторная работа
3	Разработка отчетов	7	3		3 ПП	8 ПП	лабораторная работа
4	Основы администрирования	7	1,5		3 ПП	6,25 ПП	лабораторная работа

	прикладных решений						
5	Регистры и формы	7	1,5		1,5 ПП	8 ПП	лабораторная работа
6	Основы программирования на платформе 1С: Предприятие 8.3	7	1,5		1,5 ПП	4,5 ПП	лабораторная работа
ИТОГО:			10,5		12	31,25	экзамен

4.2.Развернутое описание содержания учебного материала по разделам (темам)

Тема 1. Создание и настройка информационной базы данных на платформе 1С:Предприятие8.3

Создание новой информационной базы данных. Настройка пользовательского интерфейса. Панель разделов и подсистемы конфигурации.Работа со справочниками. Линейные, иерархические и подчиненные справочники. Предопределенные элементы. Иерархия элементов. Включение справочника в командный интерфейс.Группы панели навигации. Подчиненные подсистемы и оглавление раздела. Реквизиты и табличные части. Обязательность заполнения реквизитов. Ссылочные и примитивные типы данных. Реквизиты ссылочного типа, ссылки на справочники. Перечисления и заполнение значений по умолчанию.Документы. Интерфейсные свойства и дополнительные реквизиты. Параметры выбора и установка связей между ними. Различные виды заполнения. Копирование объектов конфигурации. Журнал документов. Константы и Функциональные опции.

Тема 2. Основные объекты системы

Классификация объектов конфигурации. Прикладные и подчиненные объекты. Концепция системы. Типы данных. Универсальные коллекции значений. Встроенный язык системы.Определение режима запуска. Командный интерфейс. Подсистемы. Роли. Константы. Определение, настройка свойств. Форма констант. Механизм работы формы.

Тема 3. Разработка отчетов

Введение в язык запросов. Источники данных и табличная модель данных. Основы синтаксиса языка запросов. Введение в компоновку данных - предыстория создания и основные возможности механизма. Формирование отчетов с помощью запросов. Конструктор запросов. Доступные поля отчета.Пользовательские настройки отчета. Выбор полей. Операции отбора и сортировки результатов. Условное оформление и группировка результатов запросов. Сохранение и восстановление настроек. Разбор примера отчета о закупках товаров. Текст запроса. Доступные поля отчета о закупках. Ресурсы запроса. Параметры компоновки. Варианты отчетов «Список», «Кросс-таблица», «Диаграмма». Стандартная расшифровка отчета. Фоновое выполнение отчета. Внешние отчеты.

Тема 4. Основы администрирования прикладных решений

Роли и права пользователей. Добавление ролей. Основная роль конфигурации. Журнал регистрации. Выгрузка, загрузка и конфигурация базы данных.

Тема 5. Регистры и формы

Введение в Регистры. Виды регистров. Регистр сведений «Артикулы». Связи наборов данных в компоновке. Соединения источников в запросе. Пакетный запрос и временные таблицы.Формы и редактор форм. Виды форм: констант, документов, списков. Периодические регистры сведений. Курсы валют. Виртуальные таблицы регистра сведений. Динамический список с произвольным запросом. Рабочий стол.

Тема 6. Основы программирования на платформе 1С: Предприятие 8.3

Объектная модель. Понятие модуля. Конструкции и ключевые слова языка. Директивы компиляции модуля.Сервисные функции. Синтакс-помощник. Шаблоны текста. Контекстная подсказка. Синтаксический контроль. Форматирование модуля и другие полезные свойства.Обработчики событий формы. Отладчик. Программное выполнение запроса. Команды формы. Показатели производительности и сценарий «клиент-сервер». Экспортируемые процедуры и общие модули. Параметризируемая команда объекта. Поддержка других языков при создании интерфейса. Механизм объектных блокировок.

5. Образовательные технологии, в том числе перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

В соответствии с требованиями ФГОС ВОв программе данного курса используются использовать следующие образовательные технологии: лекции, практические занятия, самостоятельная работа студентов.

В преподавании дисциплины существуют темы, изучение которых должно обязательно начинаться с теоретической подготовки. Чтобы повысить результативность изучения теоретического материала, применяются следующие приемы:

- лекции сопровождаются компьютерными презентациями;
- применяются методики «мозгового штурма», разбора практических ситуаций;
- используются краткие сообщения студентов по новейшим разработкам (по материалам Интернет) с использованием демонстрационных материалов.

Основной и самой результативной формой обучения являются практические занятия. Для преподавания дисциплины создан учебно-методический комплекс, обеспечивающий предоставление информации (электронные учебники и пособия) и управление деятельностью обучаемого.

Разработанные практические задания включают в себя обучающие тексты, набор пошаговых инструкций, учебных задач и заданий, демонстрационный материал.

Такой подход к организации проведения практических занятий стимулирует и организует самостоятельную работу студента. Достигаются не только учебные, но и развивающие цели, т.к. формируется умения по самоорганизации учебной деятельности.

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю): мультимедиа технологии; технологии смешанного обучения.

6. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся по дисциплине осуществляется в следующих формах:

– конспектирование и реферирование литературных источников по тематике учебного курса;
– регистрация на сайте производителя платформы, получение учебной версии и доступа к методическим материалам, необходимым для изучения;
– установка и настройка системы 1с: предприятие 8.3;
– изучение и анализ предложенных электронных образовательных ресурсов, содержащих информацию по программированию на платформе 1с;
– повторение материала с использованием конспектов.

Полностью весь методический материал по обеспечению самостоятельной работы

студентов приводится в приложении 1 к настоящей программе. Доступ к методическим материалам осуществляется через сайт Шуйского филиала ИвГУ, в том числе через ЭБС Шуйского филиала ИвГУ на платформе «Руконт».

7. Характеристика оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине

Текущая аттестация осуществляется с использованием накопительного рейтингового оценивания. Критерии оценки деятельности обучающихся разрабатываются в зависимости от уровня их информационно технологической и дидактической подготовки обучающихся.

Для текущего контроля успеваемости применяются:

- лабораторные работы на компьютере: 8 практических работ по 7 баллов (56 баллов);
- доклад по предложенной теме (14 баллов).

Помимо этого учитывается:

- степень активности и самостоятельности обучающегося на занятиях – 15 баллов,
- посещаемость (отсутствие пропусков занятий) – 15 баллов.

Промежуточная аттестация по дисциплине (зачет) проводится с использованием рейтинговой шкалы оценки. На зачете по дисциплине «Прикладное программирование» контролируемые знания, умения и владения, характеризующие этапы формирования компетенций в процессе освоения образовательной программы. Зачет проводится в форме устного собеседования по темам (10 баллов), представленным в приложении, а также выполнения практических заданий (20 баллов).

Общая накопительная оценка складывается из баллов текущей успеваемости в семестре по учебной дисциплине и баллов, полученных студентами на экзамене, таким образом, чтобы результирующий балл не превышал значения 100.

Шкала оценки результатов по дисциплине«Прикладноепрограммирование» при рейтинговой системе оценивания

Повышенный уровень	Базовый уровень	Пороговый уровень	Ниже порогового уровня
зачтено	зачтено	зачтено	не зачтено
85-100 баллов	71-84 балла	61-70 баллов	менее 61 балла

8. Учебно-методическое и информационное обеспечение дисциплины

Учебная и учебно-методическая литература:

- ***основная литература:***

1. Заика, А.А. Основы разработки прикладных решений для 1С:Предприятие 8.1 / А.А. Заика. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 208 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=429116

2. Скороход, С. В. Программирование на платформе 1С:Предприятие 8.3 : учебное пособие : [16+] / С. В. Скороход ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2019. – 136 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=577921

- ***дополнительная литература:***

1. Агафонов, Е.Д. Прикладное программирование : учебное пособие / Е.Д. Агафонов, Г.В. Ващенко ; Министерство образования и науки Российской Федерации, Сибирский Федеральный университет. – Красноярск : Сибирский федеральный университет, 2015. – 112 с. : табл., граф., ил. – Режим доступа: по подписке. – URL:

http://biblioclub.ru/index.php?page=book&id=435640
2. Бартеньев, О.В. 1С:Предприятие 8.0: опыты программирования / О.В. Бартеньев. – Москва : Диалог-МИФИ, 2004. – 389 с. : ил.,табл., схем. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=89373
3. Бартеньев, О.В. 1С:Предприятие: программирование для всех: базовые объекты и расчеты на одной дискете / О.В. Бартеньев. – Москва : Диалог-МИФИ, 2005. – 460 с. : ил., табл., схем. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=89374
4. Заика, А.А. Разработка прикладных решений для платформы "1С:Предприятие 8.1" / А.А. Заика. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 252 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=429017
5. Марченко, И. О. Разработка системы управления предприятием на платформе «1С: Предприятие 8.3» : учебно-методическое пособие : [16+] / И. О. Марченко, М. Л. Перевертайло. – Новосибирск : Новосибирский государственный технический университет, 2018. – 116 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=574864
6. Смирнов, А. А. Прикладное программное обеспечение : учебное пособие : [16+] / А. А. Смирнов. – Москва ; Берлин : Директ-Медиа, 2017. – 358 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=457616

Интернет-ресурсы:

- Веб-сервис для учебного тестирования по платформе «1С:Предприятие 8» – http://edu.1c.ru/dist-training;
- Раздел информационной системы 1С:ИТС «Разработка и администрирование» -- http://its.1c.ru/#dev

Современные профессиональные базы данных и информационно-справочные системы:

- ЭБС «Университетская библиотека онлайн» - http://biblioclub.ru;
- Коллекция Шуйского филиала ИвГУ на национальном цифровом ресурсе «Руконт» - https://www.rucont.ru;
- Консультант Плюс http://www.consultant.ru.

Программное обеспечение:

В процессе изучения дисциплины используются офисные пакеты MicrosoftOffice, OpenOffice, программа для просмотра и чтения файлов PDFAdobeAcrobatReaderDC, программа для воспроизведения флэш-анимации в браузерах AdobeFlashPlayer, вэб-обозреватели Google Chrome, Opera, Антивирус Касперского, программный продукт 1С8:Университет ПРОФ, учебная версия 1С:Предприятие 8.3,программа для просмотра и чтения файлов PDFAdobeAcrobatReaderDC, веб-обозреватели, антивирусное ПО.

9. Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения занятий лекционного типа, укомплектованная специализированной мебелью и техническими средствами обучения, служащими для предоставления учебной информации большой аудитории.
2. Учебная аудитория для проведения занятий семинарского типа, укомплектованная специализированной мебелью и техническими средствами обучения.

3. Учебная аудитория для проведения текущего контроля и промежуточной аттестации, укомплектованная специализированной мебелью и техническими средствами обучения.
4. Помещение для самостоятельной работы, оснащенное специализированной мебелью, компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.
5. Учебно-наглядные пособия для занятий лекционного типа, обеспечивающие тематические иллюстрации, видеопрезентации и видеоуроки по теме «Прикладное программное обеспечение».

Автор рабочей программы дисциплины:А.В.Ермошин, доцент кафедры математики, информатики и методики обучения, кандидат технических наук, доцент.

Программа рассмотрена на заседании кафедры математики, информатики и методики обучения «16» февраля2023г., протокол № 7

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ С.В. Белов
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ С.В. Белов
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ С.В. Белов
(подпись)

Приложение 1.

Методические указания для обучающихся по освоению дисциплины

На самостоятельную работу студентов по курсу «Прикладное программирование» отводится более 60% учебного времени от общей трудоемкости курса. Сопровождение самостоятельной работы студентов по данному курсу может быть организовано в следующих формах:

– согласование индивидуальных заданий (виды и темы заданий, сроки представления результатов) самостоятельной работы студента в пределах часов, отведенных на самостоятельную работу;

– консультации (индивидуальные и групповые), в том числе с применением дистанционных технологий взаимодействия;

– промежуточный контроль хода выполнения заданий, строящихся на основе различных способов самостоятельной информационной деятельности в открытой информационной среде.

Преподаватель определяет содержание самостоятельной работы, график её выполнения; создает сетевую информационную и коммуникационную среду для организации самостоятельной работы.

В процессе организации самостоятельной работы особое внимание уделяется не только развитию информационной культуры, но и формированию готовности к кооперации, к работе в коллективе в сетевом пространстве, для чего целесообразно стимулировать студентов к совместной деятельности в малых группах.

Одной из важнейших особенностей заданий для самостоятельной работы по курсу «Прикладное программирование» является их ориентация на формирование способности оценивать приоритеты и ограничения при выборе средств информационных технологий для решения управленческих задач, а также на мотивирование самообразовательной деятельности с использованием ИКТ.

К типовым заданиям для самостоятельной работы студентов относятся:

– конспектирование и реферирование литературных источников по тематике учебного курса;
– установка и настройка среды разработки;
– решение задач, связанных с разработкой систем на платформе 1С;
– изучение и анализ предложенных электронных образовательных ресурсов, содержащих информацию по прикладному программированию;
– повторение материала с использованием конспектов.

Перечень вопросов для самостоятельного изучения

1 Расширенная работа со справочниками

Справочники. Иерархия элементов. Перечисления. Иерархия групп. Подчиненные справочники. Табличные части. Расширение функциональности формы. Работа с данными справочника. Реквизиты формы, объекты базы. Создание печатных форм.

2. Расширенная работа с документами

Создание документов. Доступ к данным документа. Модуль объекта. Создание объектов копированием. Журналы документов. Регистры сведений. Создание регистра сведений. Работа с данными регистра. Форма списка регистра. Режим записи «Подчинение регистратору». Планы видов характеристик. Функциональные опции. Учетные объекты.

3. Углубленное изучение языка запросов

Источники данных. Структура запроса (описание запроса). Использование конструктора запросов. Особенности работы с виртуальными таблицами. Построение запросов по нескольким таблицам. Работа с временными таблицами. Использование предопределенных данных. Пакетные запросы.

4.Дополнительные разделы по администрированию прикладных решений

Отчеты. Рабочий стол. Критерии отбора. Обработка заполнения данных и установка значений по умолчанию. Хранилище значений (работа с изображениями). Механизм полнотекстового поиска. Регламентные задания. Бизнес-процессы и задачи.

ЗАДАНИЯ ДЛЯ ОРГАНИЗАЦИИ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

В курсе предусмотрено 8 практических работ. Каждая отдельная работа максимально оценивается в 7 баллов. Максимально студент может получить 56 баллов.

Критерии оценки:

За верно выполненное задание и правильные ответы студент получает максимум 7 баллов.

Критерии	Баллы
Задания лабораторной работы выполнены в полном объеме	4
Самостоятельность выполнения	1
Полнота ответа на контрольные вопросы	2
Итого	**7**

Практическая работа №1. Знакомство, создание информационной базы.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1С:Предприятие является универсальной системой автоматизации экономической и организационной деятельности предприятия. Поскольку такая деятельность может быть довольно разнообразной, система 1С:Предприятие может «приспосабливаться» к особенностям конкретной области деятельности, в которой она применяется. Для обозначения такой способности используется термин конфигурируемость, то есть возможность настройки системы на особенности конкретного предприятия и класса решаемых задач.

Логически всю систему можно разделить на две большие части, которые тесно взаимодействуют друг с другом: конфигурацию и платформу, которая управляет работой конфигурации.

Существует одна платформа (1С:Предприятие 8) и множество конфигураций. Для функционирования какого-либо прикладного решения всегда необходима платформа и какая-либо конфигурация (рис. 1.1).



Рис. 1.1. Конфигураций много, а платформа одна

Сама по себе платформа не может выполнить никаких задач автоматизации, так как она создана для обеспечения работы какой-либо конфигурации. То же самое с конфигурацией: чтобы выполнить те задачи, для которых она создана, необходимо наличие платформы, управляющей ее работой.

Итак, поскольку задачи автоматизации могут быть самыми разными, фирма «1С» и ее партнеры выпускают прикладные решения, каждое из которых предназначено для автоматизации

одной определенной области человеческой деятельности. В качестве примера существующих прикладных решений можно перечислить следующие типовые решения:

- 1С:Управление небольшой фирмой,
- 1С бухгалтерия,
- 1С:Предприятие. Управление торговлей,
- 1С: Зарплата и Управление Персоналом,
- 1С:Предприятие. Управление производственным предприятием,
- 1С:Консолидация.

Режимы работы системы:

1С: Предприятие имеет различные режимы работы: 1С: Предприятие и Конфигуратор.

Режим 1С Предприятие является основным и служит для работы пользователей системы. В этом режиме пользователи вносят данные, обрабатывают их и получают итоговые результаты.

Режим Конфигуратор используется разработчиками и администраторами информационных баз. Именно этот режим и предоставляет инструменты, необходимые для модификации существующей или создания новой конфигурации.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Создание новой информационной базы

Запустите 1С Предприятие. В открывшемся диалоге вы увидите список информационных баз, с которыми вы работаете. Если этот список пуст, система сама предложит вам создать новую базу. Если же в списке информационных баз содержится какая-либо база, например, у вас установлена демонстрационная конфигурация, то для создания новой базы нажмите кнопку***Добавить*** (рис. 1.2).

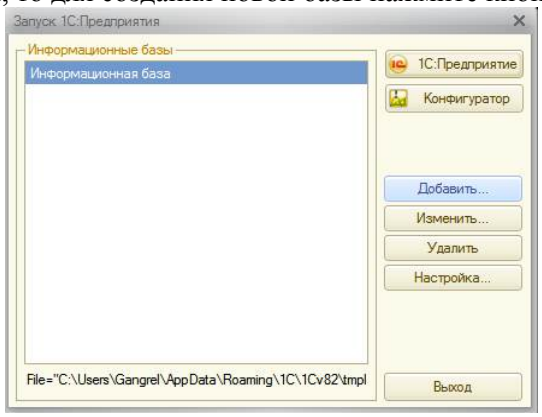


Рис. 1.2. Создание новой информационной базы

- В открывшемся диалоге выберите пункт «Создание новой информационной базы»;
- Нажмите кнопку «Далее» На следующем шаге выберите пункт «Создание информационной базы без конфигурации»;
- Нажмите кнопку «Далее». На следующем шаге задайте наименование вашей информационной базы и выберите тип ее расположения «На данном компьютере»;
- Нажмите кнопку «Далее». На следующем шаге укажите каталог для расположения вашей информационной базы. Язык по умолчанию установлен в значение Русский;
- Нажмите кнопку «Далее». На следующем шаге нажмите кнопку «Готово»;
- В диалоге запуска 1С:Предприятия, в списке информационных баз, вы увидите созданную вами новую пустую базу.

Знакомство с конфигуратором:

Итак, запустим 1С:Предприятие в режиме Конфигуратор. Для этого нажмем кнопку «Конфигуратор» в диалоге запуска системы.

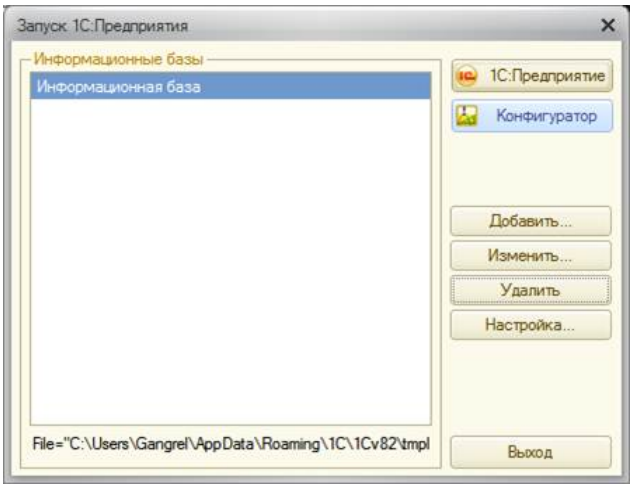


Рис. 1.3. Конфигуратор

Перед вами окно конфигуратора:

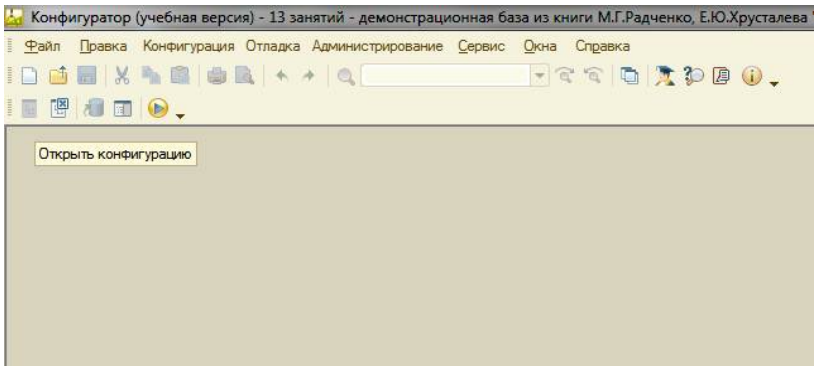


Рис. 1.4. Окно конфигуратора

Именно с помощью этого инструмента мы будем создавать нашу конфигурацию. Сразу под заголовком окна находится главное меню конфигуратора, содержащее пункты **Файл, Правка, Конфигурация, Администрирование** и т.п. В каждом из этих пунктов содержится много подпунктов, вызов которых обеспечивает выполнение различных действий конфигуратора.

Ниже находится панель инструментов конфигуратора, в которую в виде кнопок-пиктограмм помещены наиболее часто используемые действия, вызываемые из меню.

Таким образом, одни и те же действия можно выполнить двумя разными способами: вызвав определенный пункт меню или нажав соответствующую ему кнопку на панели инструментов.

Выполним первую команду, с которой начинается работа с любой конфигурацией, - откроем конфигурацию с помощью пункта меню **Конфигурация> Открыть конфигурацию** или соответствующей кнопки на панели инструментов.

На экране *откроется дерево объектов конфигурации:*

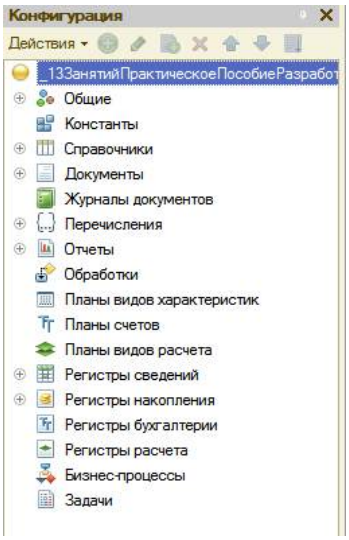


Рис. 1.5. Дерево объектов конфигурации

Можно сказать, что дерево объектов конфигурации - основной инструмент, с которым работает разработчик. Дерево объектов конфигурации содержит в себе практически всю информацию о том, из чего состоит конфигурация.

Что такое объекты конфигурации:

Конфигурация представляет собой описание. Она описывает структуру данных, которые пользователь будет использовать в режиме работы 1С:Предприятие.

Кроме этого, конфигурация описывает всевозможные алгоритмы обработки этих данных, содержит информацию о том, как эти данные должны будут выглядеть на экране и на принтере, и т.д.

С одной стороны, объекты конфигурации представляют собой детали «конструктора», из которого собирается конфигурация. Обычно в конструкторе существует некоторый набор деталей. Детали могут быть разного вида. Теперь представьте, что деталей каждого вида мы можем создавать столько, сколько нам нужно. Мы можем соединять детали между собой различными способами.

То же и с объектами конфигурации. Мы можем создавать только объекты определенных видов. Но каждого вида объектов мы можем создать столько, сколько нам нужно. Объекты одного вида отличаются от объектов другого вида тем, что имеют разные свойства. Объекты могут взаимодействовать друг с другом, и мы можем описать такое взаимодействие.

Объекты конфигурации также обладают различным поведением, и оно зависит от вида объекта. Одни объекты могут выполнять какие-то действия, другие этих действий выполнять не могут, зато у них есть свой собственный набор действий.

«Сложные» объекты конфигурации состоят из более «простых», и одни и те же «простые» объекты могут входить в состав сложных объектов. Такая структура позволяет упростить работу с объектами конфигурации, поскольку если мы знаем, как работать с каким-либо «простым» объектом, то в любом «сложном» объекте, в состав которого он входит, мы будем работать с ним все тем же образом.

Самое важное качество объектов конфигурации - это их прикладная направленность. Объекты конфигурации не просто некие абстрактные конструкции, при помощи которых разработчик пытается описать поставленную перед ним задачу. Они представляют собой аналоги реальных объектов, которыми оперирует предприятие в ходе своей работы.

Как добавить объект конфигурации:

Перед началом работы следует объяснить некоторые приемы работы с конфигуратором:

Для того чтобы открыть и закрыть конфигурацию, следует использовать пункты меню **Конфигурация> Открыть конфигурацию** и **Конфигурация > Закрыть конфигурацию** или соответствующие им кнопки на панели инструментов.

После того как конфигурация открыта, ее состав появляется в окне дерева конфигурации (см. рис. 1.5). Это окно вы можете закрыть, как любое другое окно Windows, при этом конфигурация останется открытой. Чтобы снова отобразить на экране окно дерева конфигурации, следует

воспользоваться командой меню **Конфигурация > Окно конфигурации.**

Добавить новый объект конфигурации можно несколькими способами, и вы можете использовать наиболее понятный и удобный для вас.

Первый способ: Необходимо установить курсор на ту ветку объектов конфигурации, которая вас интересует, и в командной панели окна конфигурации нажать кнопку **Действия> Добавить.**

Второй способ. Вы можете воспользоваться контекстным меню, которое вызывается при нажатии на правую клавишу мыши. Установите курсор на интересующую вас ветку объектов конфигурации и нажмите правую клавишу мыши. В появившемся меню выберите пункт Добавить.

Третий способ. Установите курсор на интересующую вас ветку объектов конфигурации и в командной панели окна конфигурации нажмите кнопку **«Добавить»** (с пиктограмой «+»).

Палитра свойств:

Палитра свойств - это специальное служебное окно, которое позволяет редактировать все свойства объекта конфигурации и другую связанную с ним информацию. Поскольку разные объекты конфигурации имеют самые разные свойства, содержимое этого окна будет меняться в зависимости от того, какой объект является текущим (на каком объекте конфигурации установлен курсор).

- Выделим в дереве объектов конфигурации корневой элемент **«Конфигурация**» и двойным щелчком мыши откроем его палитру свойств.
- Зададим имя конфигурации **«Фамилия».** Соответствующий ему синоним устанавливается автоматически, но его можно изменить по своему усмотрению. В дальнейшем именно его мы будем видеть в рабочем окне 1С предприятия.

В этом случае, как и сейчас, палитра свойств откроется и будет закреплена на рабочей области конфигуратора. То есть при выделении какого-либо объекта конфигурации окно его свойств всегда будет открыто. Однако есть удобная возможность «открепить» палитру свойств, используя символ кнопки в заголовке окна палитры свойств.

Запуск отладки в режиме 1С:Предприятие:

Проверим первые изменения в режиме 1С:Предприятие.

Для этого выполним пункт меню **Отладка> Начать отладку** или нажмем соответствующую кнопку > на панели инструментов конфигуратора. Система сама анализирует наличие изменений в конфигурации и выдает соответствующий вопрос об обновлении конфигурации базы данных.

На вопрос конфигуратора ответим **«ДА»**, и на экране появится окно 1С:Предприятия.

В режиме 1С:Предприятие:

Внешний вид интерфейса прикладного решения

В заголовке окна видим название нашей конфигурации. Пустое пространство - это рабочая область приложения, которая пока ничем не заполнена.

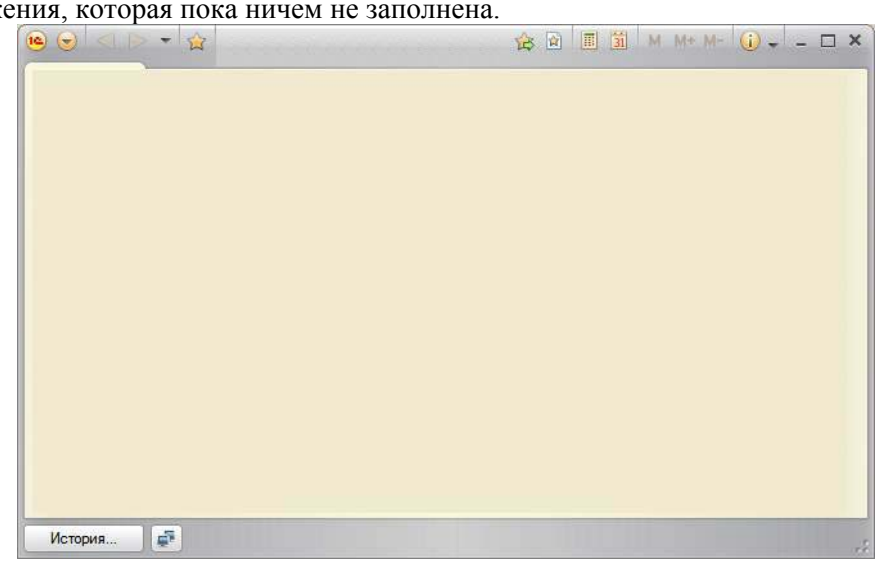


Рис. 1.6. Внешний вид интерфейса прикладного решения

Кроме заголовка конфигурации в окне 1С:Предприятия ничего не появилось. И этого следовало ожидать.

Контрольные вопросы:

Что такое конфигурируемость системы 1С:Предприятие.
Из каких основных частей состоит система.
Что такое платформа и что такое конфигурация.
Для чего используются разные режимы запуска системы 1С .'Предприятие.
Что такое дерево объектов конфигурации.
Что такое объекты конфигурации.
Что создает система на основе объектов конфигурации.
Какими способами можно добавить новый объект конфигурации.
Зачем нужна палитра свойств.
Как запустить 1С:Предприятие в режиме отладки

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Что такое подсистема:

Подсистемы - это основные элементы для построения интерфейса 1С:Предприятия. Поэтому первое, с чего следует начинать разработку конфигурации, - это проектирование состава подсистем.

При этом перед разработчиком стоит важная и ответственная задача - тщательно продумать состав подсистем, и затем аккуратно и осмысленно привязать к подсистемам те объекты конфигурации, которые он будет создавать.

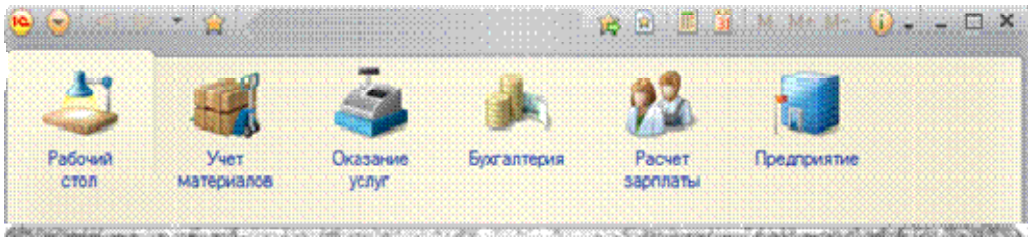


Рис. 2.1. Подсистемы

ПРАКТИЧЕСКАЯ ЧАСТЬ

Каждый объект конфигурации может быть включен в одну или сразу несколько подсистем, в составе которых он будет отображаться.

Сейчас мы создадим в нашей конфигурации пять новых объектов конфигурации Подсистема, которые будут иметь имена: «**Бухгалтерия», «РасчетЗарплаты», «УчетМатериалов», «ОказаниеУслуг»** и **«Предприятие».** Чтобы это сделать, выполним следующие действия.

Добавление подсистемы в режиме «Конфигуратор»:

Чтобы создать новые подсистемы, раскроем ветвь «**Общие»** в дереве объектов конфигурации, нажав на+ слева от нее.

Затем выделим ветвь «**Подсистемы»**, вызовем ее контекстное меню и выберем пункт «**Добавить»** или нажмем соответствующую кнопку в командной панели окна конфигурации.

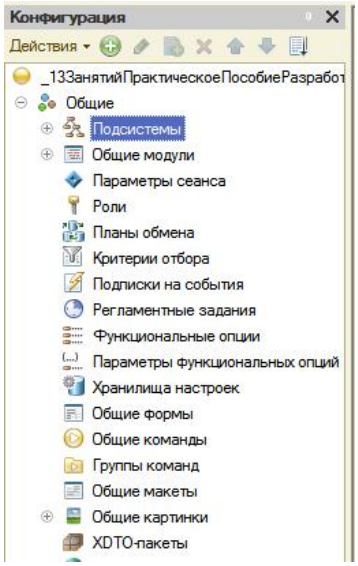


Рис. 2.2. Дерево объектов конфигурации

После этого система откроет окно редактирования объекта конфигурации.

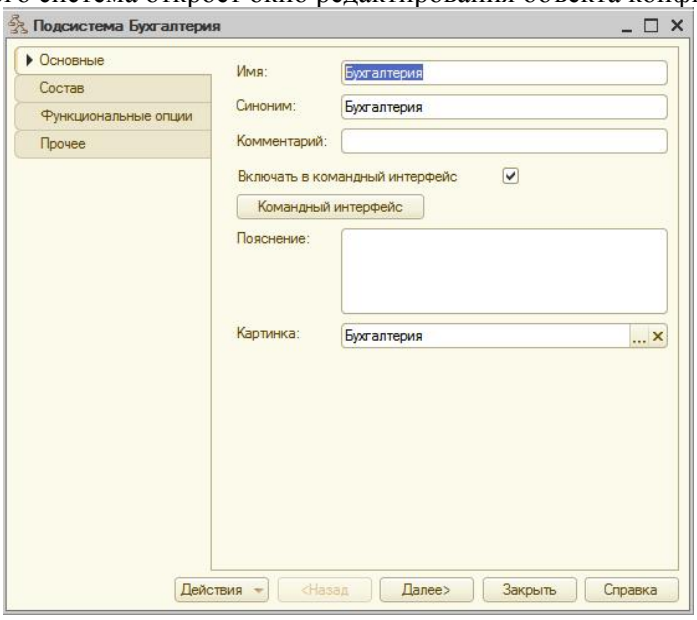


Рис. 2.3. Подсистема Бухгалтерия

Зададим имя подсистемы – «**Бухгалтерия»**. На основании имени платформа автоматически создаст синоним – «**Бухгалтерия»**.

Имя и синоним объекта конфигурации:

Имя является основным свойством любого объекта конфигурации. При создании нового объекта система автоматически присваивает ему некоторое имя.

Свойство **Синоним** также есть у любого объекта конфигурации. Оно предназначено для хранения «альтернативного» наименования объекта конфигурации, которое будет использовано в элементах интерфейса нашей программы, то есть будет показано пользователю. Поэтому на синоним практически нет никаких ограничений.

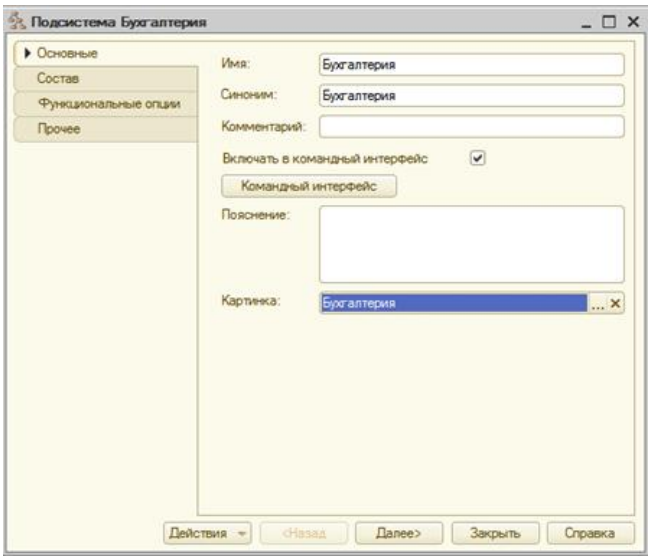


Рис. 2.4. Подсистема Бухгалтерия

В целях усовершенствования интерфейса приложения мы можем также задать картинку для отображения подсистемы.

Система создаст объект конфигурации Общая картинка и откроет окно редактирования его свойств.

Картинке можно присвоить свое имя, а так же выбрать свое изображение с локального диска. Для просмотра изображений поставим флажок «**Просмотр».** Выбранная нами картинка появится в окне редактирования общей картинки. Закроем окно редактирования объекта конфигурации

Итак, мы вернулись в окно редактирования объекта конфигурации Подсистема Бухгалтерия. Мы видим, что выбранная нами одноименная картинка установилась в качестве картинки для подсистемы.

Таким образом, в интерфейсе 1С Предприятия в качестве названия раздела будет показан синоним подсистемы, и над ним будет выводиться указанная картинка.

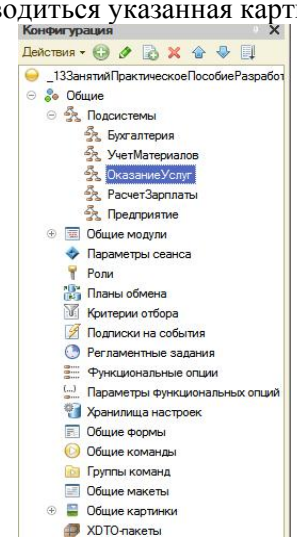


Рис. 2.5. Дерево объектов конфигурации

Снова выделим ветвь «**Подсистемы**», нажмем кнопку «**Добавить**» в дереве объектов конфигурации и создадим подсистемы с именами «**УчетМатериалов»** и «**ОказаниеУслуг»**. Установим для них картинки.

Теперь воспользуемся другим способом для добавления подсистем. Вызовем контекстное меню одной из созданных подсистем. Выберем в нем пункт «**Добавить»**. Он разбивается на два подпункта. Выбор подпункта «**Подсистема»** позволяет добавить подсистему того же уровня иерархии, что и выделенная. Выбор подпункта «**ПодчиненнаяПодсистема»** позволяет добавить

подсистему, подчиненную выделенной.Поскольку в нашей конфигурации не планируется сложной многоуровневой структуры, выберем первый вариант и добавим подсистему «**РасчетЗарплаты».** Установим для нее в качестве картинки общую картинку «**Зарплата»**, добавив ее из файла «**Зарплата»**.

В заключение добавим подсистему Предприятие для доступа к административным и сервисным функциям.

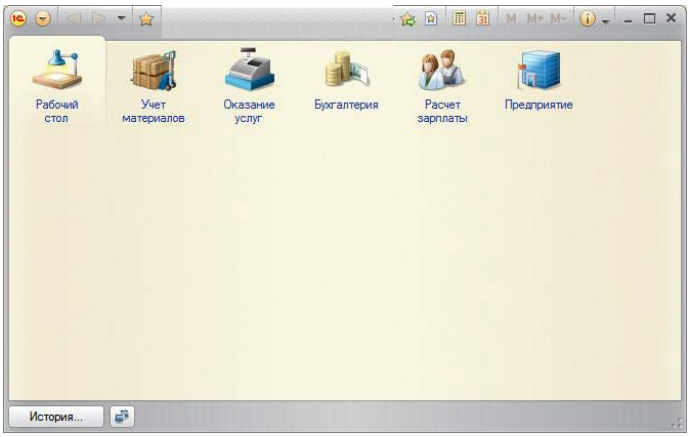


Рис. 2.6. Подсистемы

Панель разделов прикладного решения в режиме 1С:Предприятие:

Запустим 1С Предприятие в режиме отладки и увидим результат наших изменений. Вид разрабатываемого нами приложения изменился. Сразу под главным меню располагается панель разделов приложения, где и отражены созданные нами подсистемы. Причем все разделы выводятся с выбранными в их свойствах картинками. Разделы представлены в форме гиперссылок, нажав на которые пользователь может открыть связанные с ними документы, справочники, отчеты и т. п.

Обратите внимание, что раздел **Рабочий стол** формируется платформой по умолчанию. Он предназначен для размещения наиболее часто используемых пользователем документов, отчетов и т. п.

Порядок разделов в режиме Конфигуратор:

Однако порядок расположения подсистем нас не совсем устраивает. Изменим его.

Закроем приложение и вернемся в конфигуратор. Выделим корень дерева объектов конфигурации «**Фамилия»,** нажатием правой кнопки мыши вызовем контекстное меню и выберем пункт «**Открыть командный интерфейс конфигурации»**

В открывшемся окне «**Командный интерфейс»** вы увидите список созданных вами подсистем (разделов приложения). Расположим сначала подсистемы, отражающие производственную деятельность нашей фирмы: «**Учет материалов**» и «**Оказание услуг**», затем бухгалтерскую деятельность и расчет зарплаты сотрудников: «**Бухгалтерия**» и «**Расчет зарплаты**», а затем подсистему «**Предприятие**».

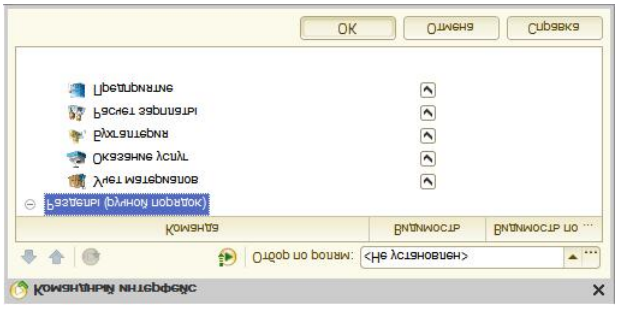


Рис. 2.7. Командный интерфейс

В режиме 1С:Предприятие:

Запустим 1С Предприятие в режиме отладки и увидим, что порядок расположения подсистем в панели разделов приложения изменился так, как мы его задали.

Примечание:

После успешного завершения каждого занятия рекомендуется сохранять конфигурацию, выполнив команду главного меню **Администрирование> Выгрузить информационную базу.** Для загрузки БД выполните: **Администрирование> Загрузить информационную базу.**

Окно редактирования объекта конфигурации и палитра свойств

На первый взгляд окно редактирования объекта и палитра свойств дублируют друг друга. *Окно редактирования объекта конфигурации предназначено в первую очередь для быстрого создания новых объектов.* Окно редактирования объекта помогает быстро создать незнакомый объект конфигурации и обеспечивает удобный доступ к нужным свойствам.

Что же касается палитры свойств, то она предоставляет одну абсолютно незаменимую возможность. Дело в том, что она не привязана по своей структуре к какому-то конкретному виду объектов конфигурации. Ее содержимое меняется в зависимости от того, какой объект является текущим. За счет этого она может «запоминать», какое свойство объекта в ней выбрано, и при переходе в дереве к другому объекту будет подсвечивать у себя все то же свойство, но уже другого объекта.

Контрольные вопросы:

Для чего используется объект конфигурации Подсистема.

Как описать логическую структуру конфигурации при помощи объектов Подсистема.

Как управлять порядком вывода и отображением подсистем в конфигурации.

Что такое окно редактирования объекта конфигурации и в чем его отличие от палитры свойств.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Что такое справочник

Объект конфигурации Справочник предназначен для работы со списками данных. Свойства и структура этих списков описываются в объектах конфигурации Справочник, на основе которых платформа создает в базе данных таблицы для хранения информации из этих справочников.

Справочник состоит из элементов. Пользователь в процессе работы может самостоятельно добавлять новые элементы в справочник.

В базе данных каждый элемент справочника представляет собой отдельную запись в основной таблице, хранящей информацию из этого справочника.

Каждый элемент справочника, как правило, содержит некоторую дополнительную информацию, которая подробнее описывает этот элемент. Поскольку эти объекты конфигурации логически связаны с объектом Справочник, они называются подчиненными этому объекту.

В БД справочник хранится в виде таблицы, в строках которой расположены элементы списка, а каждому реквизиту (стандартному или созданному разработчиком) в этой таблице соответствует отдельный столбец.

Система скрывает от разработчика всю «техническую» часть, связанную с хранением данных: в базе данных для справочника создаются несколько таблиц, эти таблицы связываются по уникальному полю **(Ссылка),** поля таблиц имеют определенные типы и т.д. Все это система делает сама. Нам лишь нужно добавить в объект конфигурации Справочник подчиненный ему объект «**Табличная часть».**

Для удобства использования элементы справочника могут быть сгруппированы по какому-либо принципу. Группировка называется иерархией, выделяют два вида иерархии:

- иерархия групп и элементов *(в качестве родителя выступает группа элементов справочника)*
- иерархия элементов (родитель один из элементов справочника)

Элементы одного справочника могут быть подчинены элементам или группам другого справочника. В системе 1С: Предприятие это достигается путем указания списка владельцев справочника для каждого объекта конфигурации Справочник.

Порой возникают ситуации, когда необходимо, чтобы в справочнике некоторые элементы существовали всегда, независимо от действий пользователя. Объект конфигурации Справочник позволяет описать любое количество таких элементов справочника. Они называются предопределенными *элементами справочника.*

Предопределенные элементы отличаются от обычных тем, что они создаются в

конфигураторе и что пользователь не может их удалить. Все остальные действия с ними он делать может, в том числе и переименовывать. В интерфейсе предопределенные элементы справочника помечены специальной пиктограммой.

Предопределенные элементы

Выделяют следующие особенности предопределенных элементов:

- на предопределенные элементы могут опираться алгоритмы работы конфигурации (т. к. возможно обращение к ним из встроенного языка по имени);
- предопределенные элементы являются объектами базы данных, которые нельзя удалить в режиме 1С Предприятия.

Обычные элементы «непостоянны» для конфигурации. В процессе работы пользователя они могут появиться, исчезнуть. Поэтому конфигурация хоть и может отличить их друг от друга, но рассчитывать на них в выполнении каких-либо алгоритмов она не может в силу их «непостоянства». Предопределённые элементы, напротив, «постоянны». В процессе работы пользователя они находятся всегда на своих местах и исчезнуть не могут.

Основная конфигурация и конфигурация базы данных

Конфигурация, предназначенная для разработчика, называется Основная конфигурация. Конфигурация, с которой работают пользователи, называется Конфигурация базы данных. Основную конфигурацию можно редактировать. Конфигурацию базы данных редактировать нельзя, можно только произвести обновление конфигурации базы данных на основе основной конфигурации. Хранилище конфигурации содержит конфигурацию, предназначенную для групповой разработки. Она хранится не в виде единой конфигурации, а в виде отдельных объектов в разрезе версий конфигурации. Таким образом, мы можем получить из хранилища конфигурацию любой версии - для этого она «собирается» из объектов нужной версии.

Теперь представьте, что между всеми этими видами конфигураций существует возможность сравнения и обновления. В этом случае очень легко запутаться, и название Основная конфигурация как нельзя лучше отражает конечную цель всех изменений.

Если эти изменения не затрагивают структуру базы данных (например, если не нужно изменять таблицы, если поменялся только программный код в каком-то модуле), то обновить конфигурацию базы данных можно не прерывая работы пользователей - **динамическое обновление**.

Но если изменения касаются структуры базы данных, например, добавился новый реквизит у справочника или изменился тип существующего реквизита, то тогда нужно завершить работу всех пользователей. Разработчик всегда может сравнить основную конфигурацию и конфигурацию базы данных, может вернуться к исходному состоянию основной конфигурации, используя конфигурацию базы данных.

Формы справочника:

В зависимости от того, какие действия мы хотим выполнять со справочником, нам требуется изображать справочник в «разном виде».

Система может самостоятельно сгенерировать все формы, которые нужны для представления данных, содержащихся в справочнике. Причем система «знает», какие именно формы нужно использовать в каких ситуациях.

Вообще говоря, для отображения справочника в различных ситуациях требуется максимум пять форм для справочника.

Таблица 3.1. Формы справочника

В контекстном меню и в палитре свойств (Рис.3.3)	В конструкторе форм (Рис. 3.2)	На закладке формы (Формы…) (Рис.3.1)
Форма объекта	Форма элемента справочника	Элемента
Форма группы	Форма группы справочника	Группы
Форма списка	Форма списка справочника	Списка
Форма для выбора	Форма выбора справочника	Выбора
Форма для выбора группы	Форма выбора группы справочника	Выбора группы

Обратите внимание, что в различных местах конфигуратора одни и те же формы называются немного по-разному. Дело в том, что в контекстном меню и палитре свойств отображаются свойства

объектов конфигурации. Они одинаковые для всех объектов конфигурации. А в конструкторе форм и на закладке формы отображаются представления этих свойств в более понятном виде.

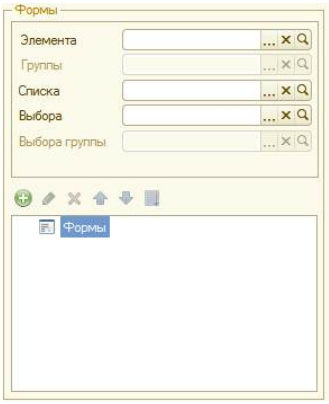


Рис.3.1 Формы

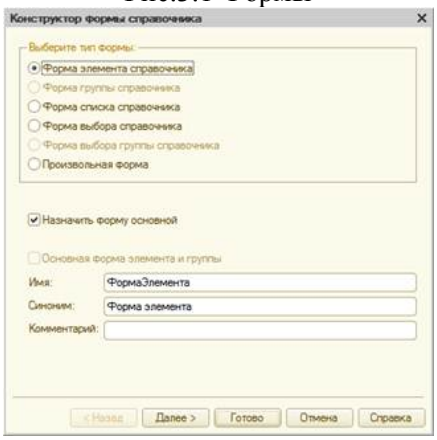


Рис.3.2 Форма элемента справочника

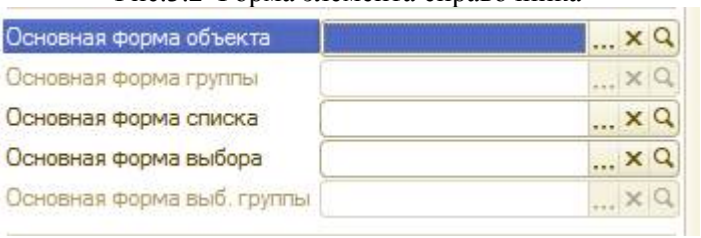


Рис.3.3 Основная форма объекта

«Форма элемента» используется для редактирования или создания элемента справочника.

«Форма группы» используется для редактирования или создания группы справочника. Группа, как правило, содержит гораздо меньше информации, чем сам элемент справочника. Поэтому для нее нужна отдельная форма, отличная от формы элемента.

«Форма списка» используется для отображения списка элементов справочника.

«Форма выбора» используется для того, чтобы в поле некоторой формы выбрать один из элементов справочника. При этом форма выбора проще, чем форма списка, так как в форме списка может показываться много реквизитов. А при выборе элемента (в документе, например), нам нужно знать только наименование. Поэтому можно для выбора использовать отдельную, более простую форму.

«Форма выбора группы» используется, когда в поле некоторой формы нужно выбрать не просто элемент справочника, а одну из его групп. При этом форма выбора группы проще, чем форма выбора элемента, так как группа, как правило, содержит гораздо меньше информации, чем сам элемент справочника.

Любая форма может быть описана в конфигураторе. Для создания такого описания существует подчиненный объект конфигурации Форма, она подчинена одному из прикладных

объектов, но может существовать и самостоятельно.

Форма служит для «визуализации» данных, находящихся в базе данных. Она представляет эти данные в удобном для пользователя виде и позволяет описать алгоритмы, которые будут сопровождать работу пользователя с данными, показанными в форме.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Для начала нам понадобится список сотрудников предприятия, которые будут оказывать услуги. Затем нам будет нужен список клиентов, с которыми работает наше предприятие. После этого нам понадобится перечень услуг, которые может оказывать наше предприятие, и список материалов, которые могут быть израсходованы.

Кроме этого, нам потребуется список складов, на которых могут находиться материалы.

1. Создание простого справочника

Создадим справочник, в котором будут храниться наименования наших клиентов

- **В режиме Конфигуратор:**

Откроем в конфигураторе конфигурацию, выделим в дереве объектов конфигурации ветвь «**Справочники»** и нажмем кнопку «**Добавить»** в командной панели окна конфигурации.

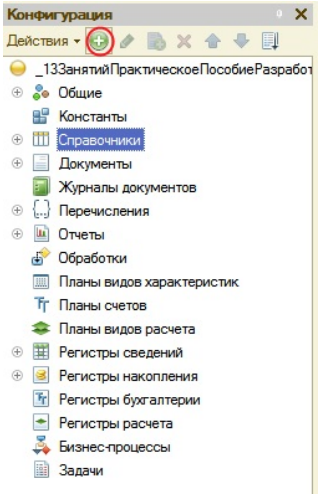


Рис. 3.4. Дерево объектов конфигурации

В открывшемся окне редактирования объекта конфигурации зададим имя справочника - **Клиенты**. На основании имени платформа автоматически создаст синоним - **Клиенты**. Напомним, что свойство **Синоним** служит для представления объекта в интерфейсе нашей программы.

Также у разработчика есть возможность установки дополнительных свойств, определяющих пользовательское представление объектов. Эти свойства задавать не обязательно.

- **Представления объекта в конфигурации:**

Представление объекта определяет название объекта в единственном числе и используется в названии стандартной команды, например, команды создания объекта - **Клиент: создать**. Представление объекта нужно задавать тогда, когда синоним объекта конфигурации задан во множественном числе, или когда он описывает множество объектов. Потому что в интерфейсе автоматически формируются команды открытия списка справочника и команды создания нового элемента справочника.

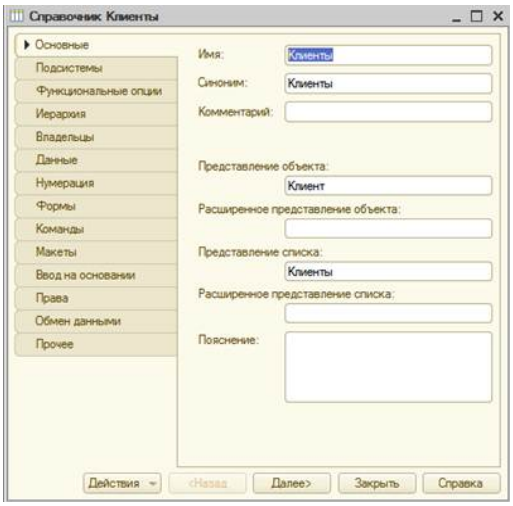


Рис. 3.5. Справочник Клиенты

Представление объекта используется для того, чтобы описать, как будет выглядеть в интерфейсе команда добавления нового клиента. Также оно будет использовано в заголовке формы и в представлении ссылки на клиента.

Расширенное представление объекта определяет заголовок формы объекта, например формы для создания нового элемента справочника. Если это свойство не задано, то вместо него используется свойство **Представление объекта**.

Представление списка определяет название списка объектов и используется в названии стандартной команды, например, команды открытия списка объектов - **Клиенты: открыть**. Представление списка нужно задавать тогда, когда синоним задан в единственном числе.

Расширенное представление списка определяет заголовок формы списка, например формы списка справочника. Если это свойство не задано, то вместо него используется свойство **Представление списка.**

В лабораторной работе зададим два свойства **Представление объекта - Клиент** и **Представление списка - Клиенты**. Последнее можно было и не задавать, так как синоним справочника совпадает со свойством **Представление списка.**

- **Принадлежность объекта к подсистемам:**

Нажмем кнопку**Далее** и перейдем на закладку **Подсистемы**. На этой закладке определяется, в каких подсистемах будет отображаться данный справочник.

В списке подсистем мы видим подсистемы, созданные ранее при определении структуры приложения. Логично предположить, что список клиентов должен быть доступен в разделе **Оказание услуг**. Бухгалтерская отчетность, формируемая в разделе **Бухгалтерия**, также может быть представлена в разрезе клиентов.

Поэтому отметим в списке подсистемы **Бухгалтерия и ОказаниеУслуг.**

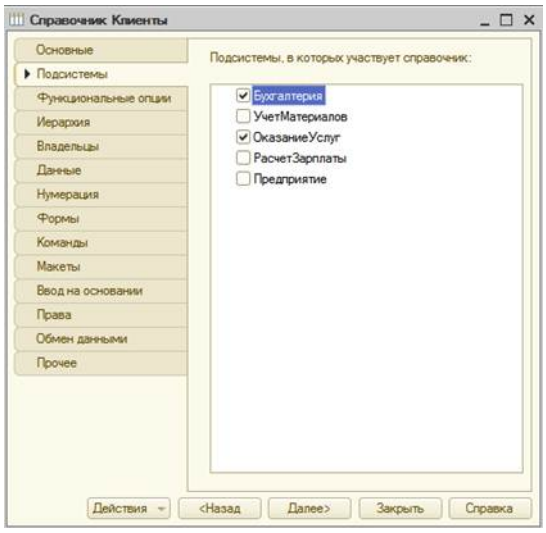


Рис. 3.6. Справочник Клиенты

- **Код и наименование справочника**

В окне редактирования объекта конфигурации Справочник нажмем на вкдадку**Данные**. Здесь для нас представляют интерес длина кода и длина наименования.

Длина кода - количество элементов, содержащихся в справочнике.. Как правило, код справочника используется для идентификации элементов справочника и содержит уникальные для каждого элемента справочника значения. Платформа может сама контролировать уникальность кодов и поддерживать автоматическую нумерацию элементов справочника.

Зададим длину кода - 9 символов, а длине наименования увеличим длину до 50.

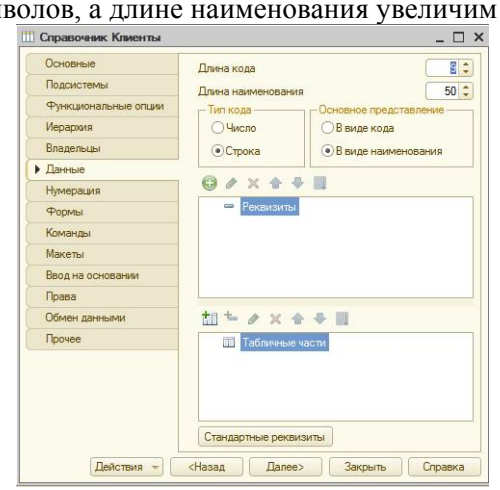


Рис. 3.7. Справочник Клиенты

- **Команда добавления нового элемента**

Прежде чем запускать 1С предприятие, настроим интерфейс приложения, чтобы было удобнее вводить новые элементы справочника.

Команда для открытия списка справочника, как и команда для создания его новых элементов, добавляется в интерфейс тех разделов (подсистем), в которых будет отображаться справочник. Но команда создания новых элементов по умолчанию невидима в интерфейсе приложения.

Сделаем доступной в панели действий раздела **ОказаниеУслуг** стандартную команду для создания новых клиентов.

Для этого в дереве объектов конфигурации выделим ветвь **Подсистемы**, вызовем ее контекстное меню и выберем пункт**Все подсистемы,** выделим подсистему **ОказаниеУслуг.** В списке **Командный интерфейс** отразятся все команды выбранной подсистемы. В группу **Панель действий.Создать** добавилась команда **Клиент: создать** для создания нового элемента справочника, включим видимость у этой команды.

*Для подсистемы **Бухгалтерия**никаких команд добавлять в панель действий не будем, так как это определяется прикладной логикой работы.*

Закроем окно редактирования справочника **Клиенты** и запустим 1С предприятие в режиме отладки. Ответим утвердительно на запрос конфигуратора об обновлении конфигурации и нажмем кнопку**Принять**

- **В режиме 1С: Предприятие. Панель навигации и панель разделов**

Перед нами откроется окно системы в режиме 1С:Предприятие. Мы видим, что если перейти в раздел **Оказание услуг** или **Бухгалтерия**, то слева в вертикальной области окна появится панель навигации (см рис).

Панель навигации отображает структуру выбранного раздела, предназначена для быстрого перехода к различным спискам в пределах выбранного раздела программы.

Также в разделе **Оказание услуг** появилась панель действий. Панель действий содержит команды, которые соответствуют текущему разделу, выбранному в панели разделов.

Сейчас в панели действий раздела **Оказание услуг** в группе**Создать** доступна команда для создания элементов нашего первого справочника **Клиенты**. Этой командой мы и воспользуемся для создания новых элементов справочника, не открывая при этом списка клиентов.



Рис. 3.8. Панель навигации и панель разделов

- **Создание элементов справочника**

Пока наш справочник пуст, поэтому добавим в него несколько элементов. Для этого выполним команду **Клиент** в панели действий раздела **Оказание услуг.**

Перед нами откроется форма для создания элемента справочника.

Внесем последовательно в справочник 3-х новых клиентов. Код вносить не нужно, так как он генерируется автоматически. После последовательного добавления клиентов нажмем**Записать и закрыть**.

Чтобы открыть существующий элемент справочника для редактирования, нужно дважды щелкнуть на нем мышью. А кроме этого, нажав на ссылку в информационной панели, мы тоже можем открыть для редактирования один из последних измененных элементов справочника.

2. Создание справочника с табличной частью

Теперь мы можем перейти к созданию второго справочника, который будет использоваться в нашей конфигурации, - справочника **Сотрудники**.

Этот справочник будет устроен несколько сложнее, чем справочник **Клиенты.** Дело в том, что в нем мы будем хранить не только фамилию, имя и отчество сотрудника, но и информацию о его прошлой трудовой деятельности.

- **В режиме Конфигуратор**

Добавим новый объект конфигурации Справочник. Назовем его **Сотрудники**.

На основании имени платформа автоматически заполнит его синоним.

Зададим **Представление объекта** как **Сотрудник**. **Представление списка** устанавливать не будем, а **Расширенное представление списка** зададим как Список сотрудников.

Нажмем кнопку**Далее** и перейдем на закладку **Подсистемы**.

По логике нашей конфигурации список сотрудников должен быть доступен в разделах **Оказание услуги и Расчет зарплаты.**

Перейдем на закладку **Данные**. Оставим по умолчанию длину и тип кода, длину

наименования справочника зададим равной 50 символам.

- **Табличная часть**

Наша задача - создать справочник, имеющий табличную часть. Поэтому добавим в справочник новую табличную часть с именем **ТрудоваяДеятельность.**

Для этого нажмем кнопку**Добавить табличную часть** над списком табличных частей справочника. Зададим имя – **ТрудоваяДеятельность.**

После чего добавим реквизиты табличной части, для этого нажмем на кнопку**Добавить реквизит.** Добавим следующие реквизиты:

- **Организация** - тип **Строка**, длина 100;
- **НачалоРаботы** - тип **Дата**, состав даты - **Дата**;
- **ОкончаниеРаботы** - тип **Дата**, состав даты - **Дата**;
- **Должность** - тип **Строка**, длина 100.

В заключение отредактируем командный интерфейс, чтобы нам было удобнее вводить новые элементы справочника. Сделаем видимой в панели действий подсистемы **РасчетЗарплаты** стандартную команду для создания новых сотрудников.

Сделаем это в дереве объектов конфигурации в ветви **Подсистемы\Все подсистемы:**

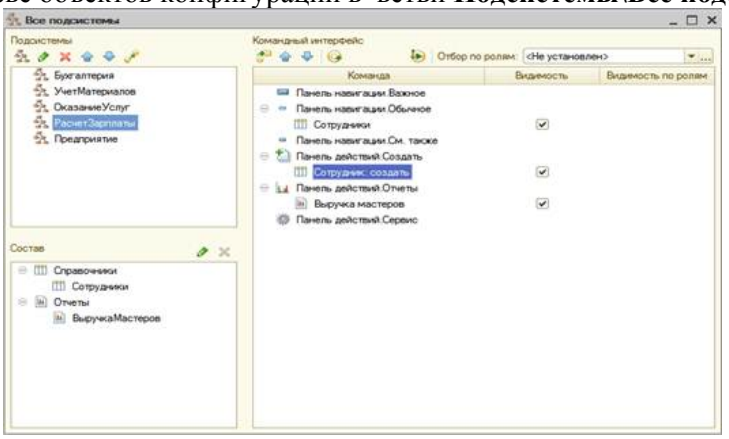


Рис. 3.9. Все подсистемы

Для подсистемы **ОказаниеУслуг** никаких команд добавлять в панель действий не будем, так как вряд ли понадобится пополнять список сотрудников в этом разделе. На этом создание справочника **Сотрудники** завершено.

Обновим конфигурацию, подтвердим обновление и запустим программу в режиме Предприятие.

Заполнение табличной части в режиме 1С: Предприятие

В панели действий раздела **Расчет зарплаты** появилась команда **Сотрудник** для создания новых сотрудников. Этой командой мы и воспользуемся для создания новых элементов справочника, не открывая при этом списка сотрудников. Выполним команду **Сотрудник.** Перед нами откроется форма для создания элемента справочника -основная форма объекта. Заголовок этой формы определяется свойством **Представление объекта**.

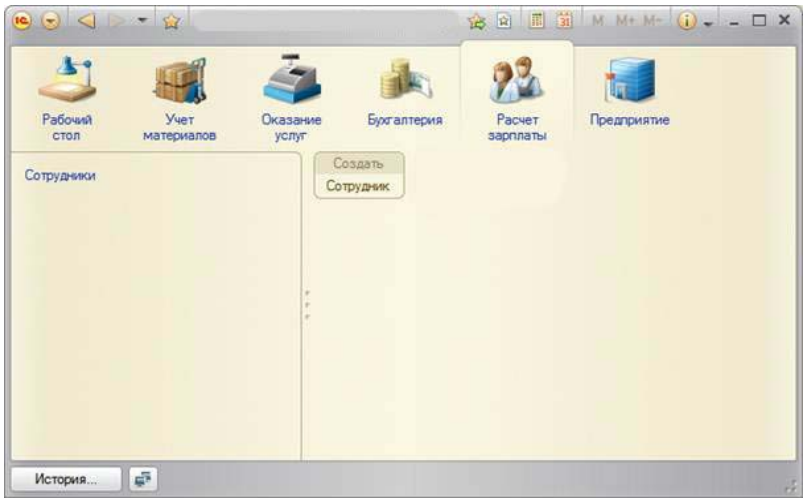


Рис. 3.10. Заполнение табличной части

Эта форма содержит табличную часть с реквизитами, которые мы описали в конфигураторе для этого справочника. Теперь приступим к созданию сотрудников, в этом вам поможет ваша фантазия (для корректной работы заполните все требующиеся поля таблицы)

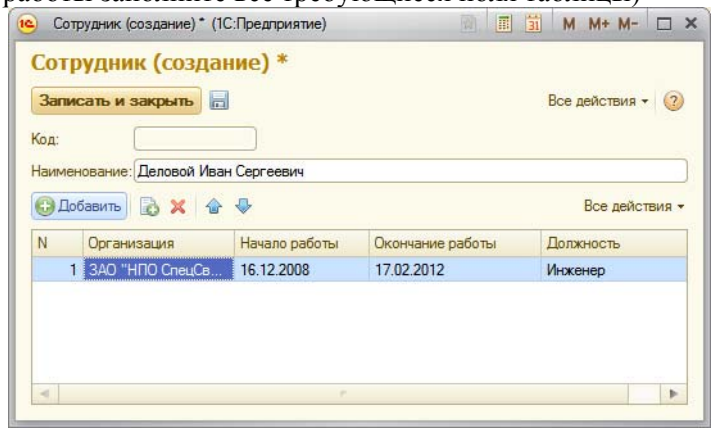


Рис. 3.11. Заполнение элемента справочника "Сотрудники"

3. Создание иерархического справочника

Справочник **Номенклатура** будет содержать информацию об услугах, которые оказывает наше предприятие, и о тех материалах, которые при этом могут быть использованы.

Этот справочник не будет сложным. Единственная особенность, которой он будет обладать, - наличие иерархической структуры. Для того чтобы справочником было удобно пользоваться, мы сгруппируем услуги в одну группу, а материалы - в другую.

- **В режиме Конфигуратор**

Создадим новый объект конфигурации Справочник и назовем его **Номенклатура**. На основании имени платформа автоматически заполнит его синоним.

Поскольку понятие **Номенклатура** не имеет единственного числа, больше никаких свойств, определяющих представление объекта в интерфейсе приложения, задавать не будем. Вместо **Представления объекта** и **Представления списка** будет использоваться **Синоним** объекта - **Номенклатура**.

*- На закладке **Подсистемы**:* По логике нашей конфигурации список номенклатуры должен быть доступен в разделах **Учет материалов. Оказание услуг и Бухгалтерия**, поэтому отметим в списке подсистем эти подсистемы.

*- На закладке **Иерархия:** установим флажок **Иерархический справочник**.*

*- На закладке **Данные:*** оставим по умолчанию длину и тип кода, длину наименования справочника зададим равной 100 символам.

*- В дереве объектов конфигурации **(Подсистемы\Все подсистемы)**: в* открывшемся окне слева в списке **Подсистемы** выделим подсистему **УчетМатериалов.** В группе **Панель**

действий.Создать включим видимость у команды **Номенклатура: создать.**

Выделив в списке подсистем **ОказаниеУслуг**, проделаем те же действия.

А для подсистемы **Бухгалтерия** никаких команд добавлять в панель действий не будем, так как вряд ли понадобится пополнять список номенклатуры в этом разделе.

Теперь заполним справочник **Номенклатура**. В процессе заполнения вы научитесь создавать группы и переносить элементы из одной группы в другую.

- **В режиме 1С:Предприятие**

В открывшемся окне 1С:Предприятия мы видим, что в панели навигации разделов **Учет материалов, Оказание услуг и Бухгалтерия** появилась команда **Номенклатура**.

Название этой команды определяется синонимом объекта, так как других представлений мы для этого справочника не задавали.

Выполним команду **Номенклатура** в панели навигации раздела **Учет материалов.** *Справа от панели навигации в рабочей области окна приложения откроется основная форма списка.*

- Создадим две группы в корне справочника: Материалы и Услуги.

Для этого нажмем кнопку Создать группу в командной панели формы списка.

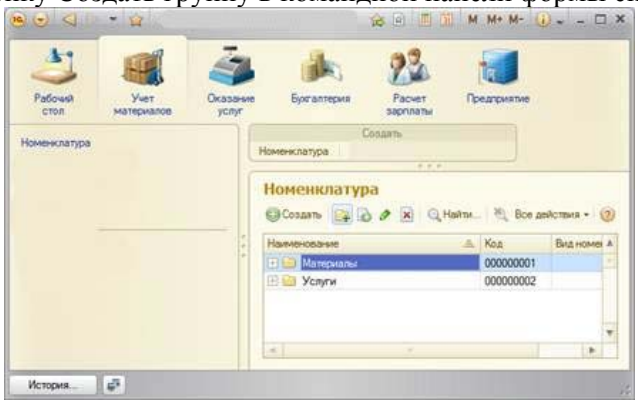


Рис. 3.12.Создание групп в корне справочника

А) Зададим наименование групп: **Материалы**, Услуги. Поля Родитель и Код заполнять не будем.

Б) Раскроем группу **Материалы** (нажатием на +) и создадим в ней пять элементов. После этого раскроем группу **Услуги** и тоже создадим в ней несколько элементов (не более 3).

- Перенос элементов в другие группы

Разнесите услуги по двум смысловым группам: Для этого в группе Услуги необходимо создать еще две подчиненных группы, которые по смыслу относятся к предоставляемым услугам.

*Для того чтобы переместить услуги в соответствующие группы, в окне списка установим курсор на ту услугу, которую мы хотим переместить, и выполним команду**Все действия > Переместить в группу**. В открывшемся окне выберем новую группу.*

4. Создание справочника с предопределенными элементами

В заключение мы создадим справочник **Склады**. Этот справочник будет включать в себя один предопределенный элемент - склад **Основной**, на который будут поступать все материалы. *Наша задача - создать справочник, содержащий предопределенные элементы.*

- **В режиме Конфигуратор**

Откроем конфигуратор и создадим новый объект конфигурации Справочник с именем **Склады**. На основании имени платформа автоматически заполнит его синоним.

*- На вкладке **Основные***: Зададим **Представление объекта - Склад. Представления списка - Склады**.

*- На вкладке **Подсистемы:***Отметим в списке подсистем подсистемы **Оказание услуг** и **Учет материалов.**

*- На вкладке **Формы**:* установим флажок в поле **Быстрый выбор**.

*Свойство **Быстрый выбор**как раз позволяет выбирать элементы не из отдельной формы, а из небольшого выпадающего списка, заполненного элементами этого справочника. Этот вариант наиболее удобен для списка складов, так как их, вероятно, будет немного.*

- На вкладке ***Прочее:*** нажмем кнопку **Предопределенные**.

Система откроет список предопределенных элементов справочника.

Сейчас он пуст, поэтому нажмем кнопку**Добавить** и создадим предопределенный элемент с именем **Основной**.

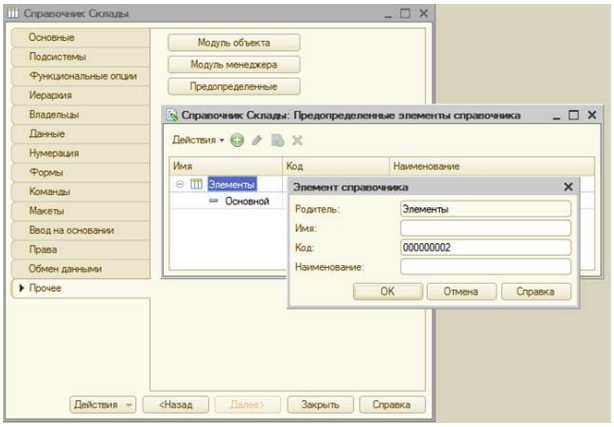


Рис. 3.13. Создание справочника с предопределенными элементами

*- В дереве объектов конфигурации (**Подсистемы\Все подсистемы**): в* открывшемся окне слева в списке **Подсистемы** выделим подсистему **УчетМатериалов.** В группе **Панель действий.Создать** включим видимость у команды Склад: создать.

- **В режиме 1С: Предприятие**

В открывшемся окне 1С:Предприятия мы видим, что в панели действий раздела **Учет материалов** появилась команда **Склад** для создания новых складов. Также в панели навигации разделов **Оказание услуг** и **Учет материалов** появилась команда **Склады** для открытия списка складов.

- Выполним команду **Склады***в панели навигации раздела* **Учет материалов.**

В списке складов уже есть один элемент с наименованием **Основной**. Это предопределенный элемент, который мы создали в конфигураторе.

Выполнив команду**Склад** в панели действий, добавим в справочник еще один склад, который назовем **Розничный**.

Контрольные вопросы

- *Для чего предназначен объект конфигурации Справочник.*
- *Каковы характерные особенности справочника.*
- *Для чего используются реквизиты и табличные части справочника.*
- *Зачем нужны иерархические справочники и что такое родитель.*
- *Зачем нужны подчиненные справочники и что такое владелец.*
- *Какие основные формы существуют у справочника.*
- *Что такое предопределенные элементы.*
- *Чем с точки зрения конфигурации отличаются обычные элементы справочника от предопределенных элементов.*
- *Как пользователь может отличить обычные элементы справочника от предопределенных элементов.*
- *Как создать объект конфигурации Справочник и описать его структуру.*
- *Как добавить новые элементы в справочник.*
- *Как создать группу справочника.*
- *Как переместить элементы из одной группы справочника в другую.*
- *Зачем нужна основная конфигурация и конфигурация базы данных.*
- *Как изменить конфигурацию базы данных.*
- *Как связаны объекты конфигурации и объекты базы данных.*
- *Что такое подчиненные объекты конфигурации.*
- *Зачем нужна проверка заполнения у реквизитов справочника.*
- *Что такое быстрый выбор и когда его использовать.*

- *Как отобразить справочник и определить его представление в различных разделах интерфейса приложения.*
- *Как отобразить команды создания нового элемента справочника в интерфейсе подсистем.*
- *Как редактировать командный интерфейс подсистем.*

Практическаяработа № 2. Документы.
ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Документ

Объект конфигурации Документ предназначен для описания информации о совершенных хозяйственных операциях или о событиях, произошедших в жизни организации вообще. Как правило, в работе любой фирмы используются такие документы, как приходные накладные, приказы о приеме на работу, платежные поручения, счета и т.д. Свойства и структура этих документов описываются в объектах конфигурации Документ, на основе которых платформа создает в базе данных таблицы для хранения информации из этих документов.

Логика работы документов отличается от логики работы других объектов конфигурации. Документ обладает способностью проведения. Факт проведения документа означает, что событие, которое он отражает, повлияло на состояние учета.

До тех пор, пока документ не проведен, состояние учета неизменно, и документ - не более чем черновик, заготовка. Как только документ будет проведен, изменения, вносимые документом в учет, вступят в силу и состояние учета будет изменено.

Поскольку документ вносит изменения в состояние учета, он всегда «привязан» к конкретному моменту времени. Это позволяет отражать в базе данных фактическую последовательность событий.

Каждый документ содержит информацию, которая подробнее описывает этот документ. Набор такой информации является одинаковым для всех документов одного вида, и для описания такого набора используются реквизиты объекта конфигурации Документ, являющиеся подчиненными объектами конфигурации. Большинство реквизитов разработчик создает самостоятельно, однако у каждого объекта Документ по умолчанию существуют стандартные реквизиты. Два наиболее важных из них - это Дата и Номер. Для описания конкретной информации служат табличные части объекта Документ. В этом случае в базе данных будут созданы дополнительные таблицы для хранения табличных частей, подчиненных конкретному документу.

Формы документа

Для «визуализации» документа существует несколько основных форм, которые имеют несколько вариантов названий (табл. 4.1).

В контекстном меню и в палитре свойств	В конструкторе форм	На закладке формы
Форма объекта	Форма документа	Документа
Форма списка	Форма списка документа	Списка
Форма для выбора	Форма выбора документа	Выбора

Таблица4.1. Основные формы документа.

Типы данных. Типообразующие объекты конфигурации

На предыдущем занятии, когда мы создавали реквизиты справочников или табличных частей, мы всегда указывали тип значения, которое может принимать этот реквизит. Это были примитивные типы данных: Число, Строка, Дата. Примитивные типы данных изначально определены в системе, и их набор ограничен.

Наряду с такими изначально определенными в любой конфигурации типами могут существовать типы данных, определяемые только конкретной конфигурацией.

Объекты конфигурации, которые могут образовывать новые типы данных, называются типообразующими. Эти типы данных не поддерживаются платформой изначально и существуют только в конкретном прикладном решении.

Это небольшое отступление было необходимо потому, что уже при создании первого

документа мы столкнемся с использованием типов данных СправочникСсылка.Склады и СправочникСсылка.Номенклатура, которые появились в нашей конфигурации в результате создания объектов конфигурации Справочник Склады и Номенклатура.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Создание Документа «Приходная накладная»

Для отражения событий, происходящих на предприятии, в базе данных мы создадим два документа: **Приходная накладная** и **Оказание услуги.**

Документ **Приходная накладная** будет фиксировать факт поступления в нашу организацию необходимых материалов, а документ **Оказание услуги** - фиксировать оказание услуг и расход материалов, которые используются при оказании этих услуг.

- **В режиме конфигуратор:**

Откроем конфигуратор и добавим новый объект конфигурации Документ.

- *На закладке Основные*: зададим имя документа – ПриходнаяНакладная. На основании имени платформа автоматически заполнит его синоним. Так же зададим Представление списка, как Приходные накладные.

- *На закладке Подсистемы:* список приходных накладных должен быть доступен в разделах Учет материалов и Бухгалтерия.

- *На закладке Данные:* создадим реквизит документа с именем Склад (нажать на «+»); далее выберем для реквизита ссылочный тип данных СправочникСсылка. Склады. Этот тип стал доступен в конфигурации после создания объекта конфигурации Справочник Склады. В качестве значения свойства **Значение заполнения** выберем предопределенный элемент справочника Склады - Основной.

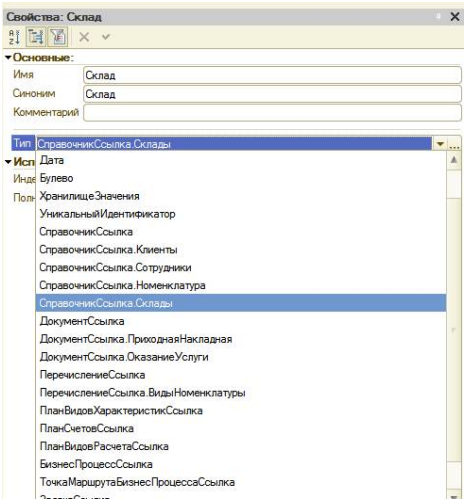


Рис. 4.1. Справочник Склады

После этого добавим в документ табличную часть с именем Материалы. Для этого нажмем кнопку Добавить табличную часть над списком табличных частей документа. Кроме имени табличной части установим свойство Проверка заполнения в значение Выдавать ошибку.

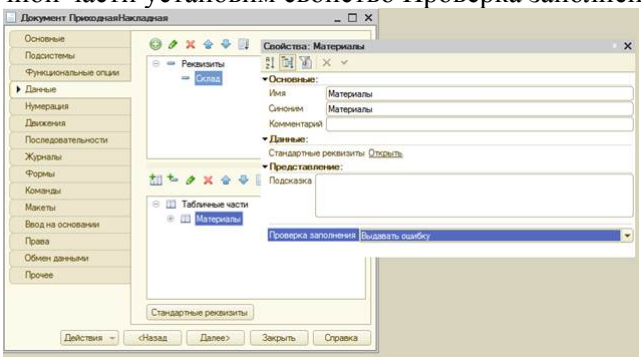


Рис. 4.2. Свойства: Материалы

Создадим реквизиты табличной части Материалы. Для этого нажмем кнопку Добавить реквизит в разделе описания табличных частей документа:

- Материал, тип СправочникСсылка. Номенклатура;
- Количество, тип Число, длина 15, точность 3, неотрицательное;
- Цена, тип Число, длина 15, точность 2, неотрицательное;
- Сумма, тип Число, длина 15, точность 2, неотрицательное.

Для каждого реквизита табличной части также установим свойство Проверка заполнения в значение Выдавать ошибку.

- *На закладке Нумерация:* убедимся, что свойство Авто-нумерация включено.

- *В дереве объектов конфигурации выделим ветвь* Подсистемы, вызовем ее контекстное меню и выберем пункт Все подсистемы. В открывшемся окне в списке Подсистемы выделим подсистему УчетМатериалов: включим видимость у команды Приходная накладная: создать, в группе Панельдействий.Создать.

- **В режиме 1С:Предприятие**

Запустим **1С:Предприятие** в режиме отладки и протестируем получившийся документ. В открывшемся окне **1С:Предприятия** мы видим, что в панели навигации разделов **Бухгалтерия и Учет материалов** появилась команда **Приходные накладные** для открытия списка приходных накладных.

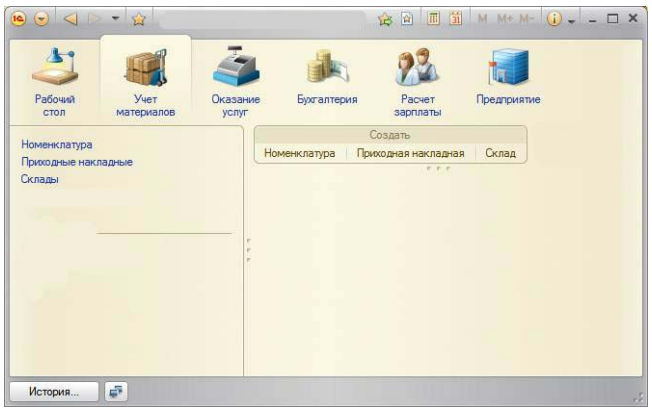


Рис. 4.3. В режиме 1С:Предприятие

Пока в БД нет ни одного документа **Приходная накладная**, поэтому выполним команду **Приходная накладная** в панели действий раздела **Учет материалов** и создадим новую приходную накладную.

Система автоматически подставит текущую дату создания документа и нулевое время, так как документ еще не проведен. В качестве времени документа при оперативном проведении ему присваивается оперативная отметка времени.

Поле **Номер** не заполнено, но система сама сгенерирует для нового документа уникальный номер, так как свойство **Автонумерация** для документа включено по умолчанию. Новый номер будет сохранен в момент записи документа. *Осталось только заполнить табличную часть приходной накладной материалами (добавьте3 любых записи в накладную). Так же система может вводить данные с клавиатуры и предлагать вам материалы по совпадающим буквам (создайте еще одну накладную таким образом)* После чего нажмем Провести и закрыть.

Документ будет сохранен и проведен, ему будет присвоен автоматически сгенерированный системой номер и текущее время проведения документа.

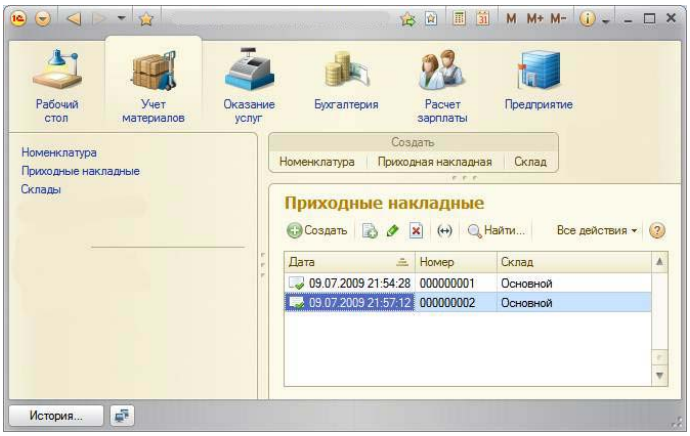


Рис. 4.4. Приходные накладные

Автоматический пересчет суммы в строках документа

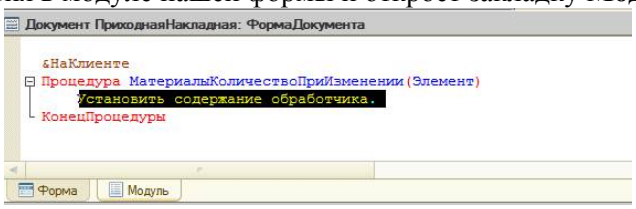
При заполнении документа приходится вводить сумму в каждой строке. Это неудобно, и возникает естественное желание автоматизировать работу документа так, чтобы сумма вычислялась автоматически каждый раз при изменении цены или количества материалов в строке.

Это совсем не сложно, и для этого нам потребуется сначала создать собственную форму документа, а затем воспользоваться возможностями встроенного языка.

- **В режиме конфигуратор:**

Откроем окно редактирования объекта конфигурации Документ**ПриходнаяНакладная,** далее пройдем на вкладку **Формы:** чтобы создать форму документа, нажмем кнопку открытия со значком лупы в поле ввода или кнопку Добавить над списком форм. Система вызовет полезный инструмент разработчика **-конструктор форм,** в нем выберем тип формы **Форма документа** и нажмем кнопку **Готово.** *Обращаем внимание, что в дереве объектов конфигурации у объекта конфигурации Документ ПриходнаяНакладная появилась форма ФормаДокумента, а на экране открылось окно редактора форм, содержащее эту форму.* Разработчик может через палитру свойств изменить свойства элемента, которые повлияют на его отображение в форме. Он может также изменить структуру элементов формы - создать новое поле, группу полей, добавить табличную часть, связав эти элементы с данными формы. Используя встроенный язык, разработчик может «вклиниться» в эти события и описать собственный алгоритм того, что должно происходить при наступлении этого события.

- Дважды щелкнем на элементе формы Материалы Количествоили правой кнопкой мыши откроем для него палитру свойств: прокрутив список до конца, мы увидим перечень событий, которые могут быть связаны с этим полем. Найдем в списке событий событие **ПриИзменении,** и нажмём на кнопку открытия (значок лупы), система создаст шаблон процедуры обработчика этого события в модуле нашей формы и откроет закладку **Модуль** редактора формы.



Листинг 4.1. Форма документа

Модуль - это «хранилище» для текста программы на встроенном языке. В модуль формы, в процедуру МатериалыКоличествоПриИзменении(), добавим следующий текст:

СтрокаТабличнойЧасти = Элементы.Материалы.ТекущиеДанные;

СтрокаТабличнойЧасти.Сумма = СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена;

Таким образом, в результате выполнения первой строки процедуры обработчика переменная СтрокаТабличнойЧасти будет содержать объект ДанныеФормыСтруктура. Этот объект содержит данные, находящиеся в текущей строке табличной части документа

(Элементы.Материалы.ТекущиеДанные). Получив этот объект, мы можем обратиться к данным конкретной колонки табличной части, указав имя колонки в качестве свойства объекта.

Одна процедура для обработки нескольких событий

Теперь хотелось бы и для поля Цена и Сумма сделать то же самое.

Поэтому лучше будет поместить расчет суммы в некоторое «общедоступное» место, чтобы разные документы, имеющие аналогичные реквизиты табличной части, могли использовать этот алгоритм. Для описания таких «общедоступных» мест служат объекты конфигурации Общий модуль, расположенные в ветке Общие>Общие модули. Процедуры и функции, содержащиеся в этих модулях, могут быть доступны для любых объектов конфигурации.

В режиме конфигуратор:

Добавим объект конфигурации Общий модуль. Для этого раскроем ветвь Общие в дереве объектов конфигурации, нажав на + слева от нее. Затем выделим ветвь **Общие модули** и нажмем кнопку**Добавить** в командной панели окна конфигурации:

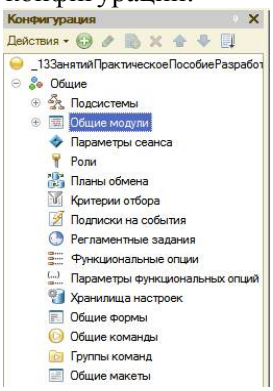


Рис. 4.5. Общие модули

Назовем его РаботаСДокументами и установим в его свойствах флажок Клиент (управляемое приложение), а флажок Сервер снимем. Это означает, что экземпляры этого модуля будут скомпилированы в контексте тонкого клиента и в контексте веб-клиента

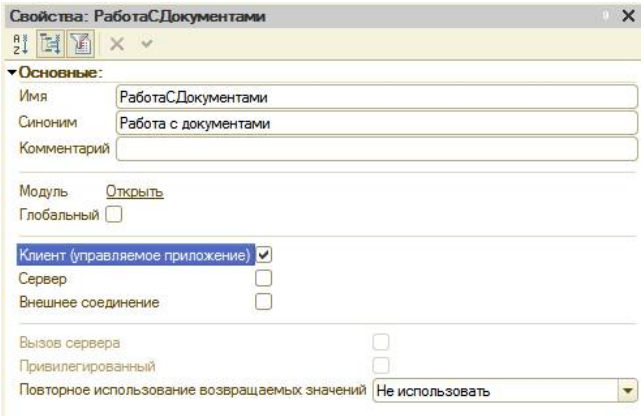


Рис. 4.6. Свойства: РаботаСДокументами

Внесем в общий модуль следующий текст:

Процедура РассчитатьСумму(строкаТабличнойЧасти) Экспорт
СтрокаТабличнойЧасти.Сумма = СтрокаТабличнойЧасти.Количество*СтрокаТабличнойЧасти.Цена;
КонецПроцедуры

Изменим текст обработчика. Для этого вернитесь в модуль формы документа, выберите поле КоличествоПриИзменении.

&НаКлиенте
Процедура МатериалыКоличествоПриИзменении (Элемент)
СтрокаТабличнойЧасти. = Элементы.Материалы.ТекущиеДанные;
РаботаСДокументами.РассчитатьСумму(СтрокаТабличнойЧасти);

КонецПроцедуры

Создадим обработчик события ПриИзменении для поля табличной части МатериалыЦена так же, как мы делали это для поля Материалы Количество, и повторим в нем вызов процедуры РассчитатьСумму из общего модуля.

- Листинг процедуры «МатериалыЦенаПриИзменении()»

&НаКлиенте
Процедура МатериалыЦенаПриИзменении (Элемент)
СтрокаТабличнойЧасти. = Элементы.Материалы.ТекущиеДанные;
РаботаСДокументами.РассчитатьСумму(СтрокаТабличнойЧасти);
КонецПроцедуры

- **В режиме 1С: Предприятие:**

Запустим **1С: Предприятие** в режиме отладки и убедимся, что теперь сумма в строках табличной части документов ПриходнаяНакладная пересчитывается как при изменении количества, так и при изменении цены.

Документ «Оказание услуги»

- **В режиме конфигуратор:**

Добавим новый объект конфигурации Документ и назовем его ОказаниеУслуги.

- *На закладке Основные*: Представление объекта задавать не будем, вместо него будет использоваться Синоним объекта. Представление списка зададим как Оказание услуг.

- *На закладке Подсистемы*: отметим, что документ будет доступен в подсистемах Оказание услуг и Бухгалтерия.

- *На закладке Данные:*

создадим реквизиты документа:

- Склад, тип СправочникСсылка.Склады. Выберем для свойства Значение заполнения предопределенный элемент Основной справочника Склады.
- Клиент, тип СправочникСсылка.Клиенты. Установим свойство Проверка заполнения в значение Выдавать ошибку.
- Мастер, тип СправочникСсылка.Сотрудники. Установим свойство Проверка заполнения в значение Выдавать ошибку.

Создадим табличную часть этого документа ПеречеьНоменклатуры с реквизитами:

- Номенклатура, тип СправочникСсылка.Номенклатура;
- Количество, тип Число, длина 15, точность 3, неотрицательное;
- Цена, тип Число, длина 15, точность 2, неотрицательное,
- Сумма, тип Число, длина 15, точность 2, неотрицательное;

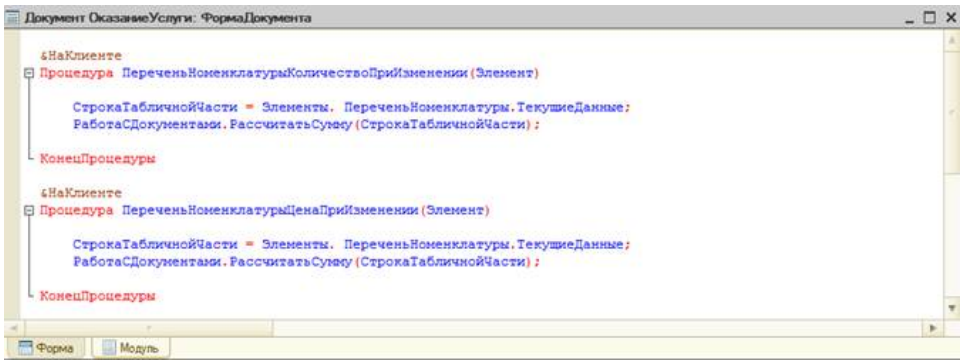
Установим для табличной части в целом и для каждого ее реквизита свойство Проверка заполнения в значение Выдавать ошибку.

- *На закладке Формы:* создадим основную форму документа:

Для поля ПеречеьНоменклатурыКоличествосоздадим обработчик события ПриИзменении, в котором будем вызывать процедуру РассчитатьСумму из общего модуля РаботаСДокументами.

При этом откроется модуль формы с шаблоном обработчика события ПеречеьНоменклатурыКоличествоПриИзменении, который мы пока заполнять не будем, а перейдем в окно элементов формы на закладку Форма и аналогичным образом создадим обработчик события ПеречеьНоменклатурыЦенаПриИзменении для поля ПеречеьНоменклатурыЦена.

Далее модуль формы документа ОказаниеУслуги нужно заполнить следующим образом:



Листинг 4.2. Модуль формы документа «ОказаниеУслуги»

В заключение отредактируем командный интерфейс, чтобы в подсистеме Оказание услуг была доступна команда создания новых документов.

Для разнообразия воспользуемся другим способом. Откроем окно редактирования объекта конфигурации Подсистема ОказаниеУслуг и нажмем кнопку Командный интерфейс.

В открывшемся окне отразятся все команды выбранной подсистемы.

В группе Панель действий.Создать включим видимость у команды Оказание услуги: создать

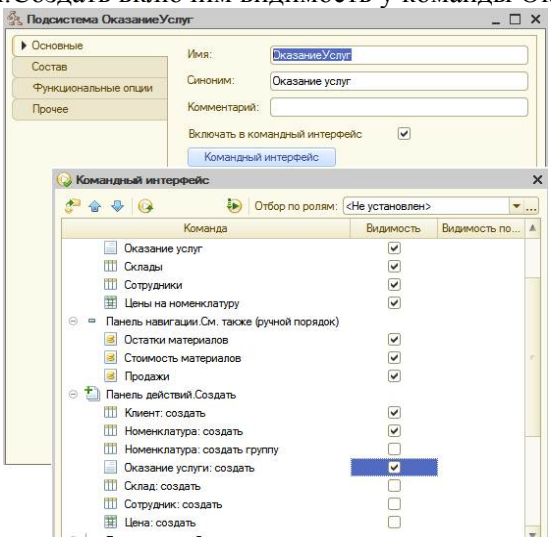


Рис. 4.7. Видимость у команды Оказание услуги: создать

В режиме 1С:Предприятие:

В панели действий раздела Оказание услуг вызовем команду создания документа Оказание услуги и заполним его следующим образом:

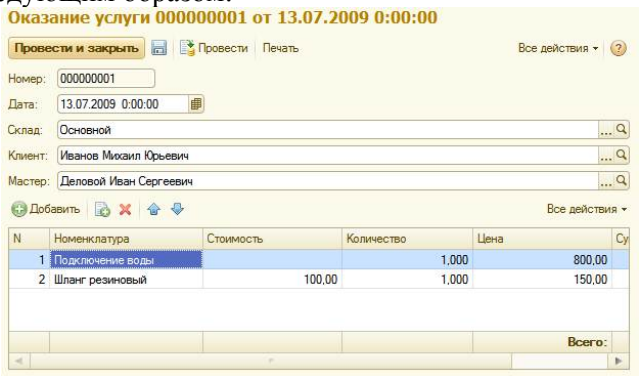


Рис. 4.8. Оказание услуги

Обратите внимание, что склад Основной подставляется по умолчанию, а для полей Мастер и Клиент выполняется проверка заполнения. А также при вводе цены и количества в табличную часть документа Оказание услуги сумма пересчитывается по нашему алгоритму.

Контрольные вопросы:

- *Какими характерными особенностями обладает документ.*
- *Для чего предназначены реквизиты и табличные части документа.*
- *Какие существуют основные формы документа.*
- *Что такое проведение документа.*
- *Как создать объект конфигурации Документ и описать его основную структуру.*
- *Как создать новый документ и заполнить его данными.*
- *Как создать собственную форму документа.*
- *Что такое конструктор форм.*
- *Что такое редактор форм.*
- *Что такое элементы формы.*
- *Что такое события и с чем они связаны.*
- *Что такое обработчик события и как его создать.*
- *Что такое модуль и для чего он нужен.*
- *Зачем нужны общие модули.*
- *Что такое типообразующие объекты.*

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Механизм основных форм

У всех прикладных объектов конфигурации существует некоторое количество основных форм. Они служат для отображения данных объекта в том или ином виде.

Если разработчик не назначит в качестве основных форм объекта свои собственные, система будет генерировать необходимые формы объекта самостоятельно, в те моменты, когда к ним происходит обращение.

Наличие такого механизма позволяет разработчику не тратить время на создание форм для тестирования своей разработки, а воспользоваться тем, что платформа создаст по умолчанию.

Создание этих форм происходит динамически, в процессе работы системы. Форма создается в тот момент, когда к ней происходит обращение. Причем не важно, интерактивное это обращение или программное.

Также примечательным фактом является то, что состав основных форм, определенных для объекта конфигурации, может не совпадать с перечнем тех форм, которые вообще можно создать для данного объекта, используя конструктор формы.

Например, для большинства регистров в конфигураторе можно задать основную форму списка. Однако если открыть конструктор форм для регистра, вы увидите, что кроме формы списка предлагается создать и форму набора записей регистра, которая отсутствует в перечне основных форм.

Обработчики событий

При работе с событиями в платформе 1С:Предприятие 8 следует различать два типа событий: события, связанные с формой и ее элементами, и все остальные.

Разница заключается в том, что обработчики событий, связанных с формой и ее элементами, - назначаемые, а обработчики всех остальных событий - фиксированные.

Фиксированный обработчик события должен иметь имя, совпадающие с именем события. Только в этом случае он будет вызываться при возникновении соответствующего события.

Назначаемый обработчик может иметь произвольное имя. Если имя процедуры совпадает с именем события формы или ее элемента, этого совсем недостаточно для вызова процедуры обработки события с таким именем. Требуется явное назначение процедуры обработчиком этого события в палитре свойств, в соответствующем событии.

Назначение обработчика может выполняться интерактивно, при работе с формой в конфигураторе, или программно, используя методы формы и ее элементов – УстановитьДействие().

Модули

Виды модулей:

- *Модуль управляемого приложения.* Модуль управляемого приложения выполняется при старте системы 1С:Предприятие в режимах тонкого клиента и веб-клиента.

В нем возможно объявление переменных, а также объявление и описание процедур и функций, которые будут доступны в любом модуле конфигурации (кроме модуля внешнего соединения). Их доступность также обеспечивается для неглобальных общих модулей с установленным свойством

Клиент (управляемое приложение). В контексте модуля управляемого приложения доступны экспортируемые процедуры и функции общих модулей. Чтобы открыть модуль управляемого приложения, нужно выделить корень дерева объектов конфигурации и вызвать из контекстного меню команду Открыть модуль управляемого приложения

- *Общие модули.* В общих модулях хранятся процедуры и функции, которые вызываются из других модулей системы. Сам по себе общий модуль не исполняется. Исполняются отдельные его процедуры/ функции в момент их вызова из других модулей. Чтобы открыть общий модуль, нужно раскрыть ветвь Общие в дереве объектов конфигурации, затем раскрыть ветвь Общие модули и дважды щелкнуть мышью на нужном модуле.
- *Модули объектов.* Модули объектов - это, например, модуль элемента справочника или модуль документа. Эти модули вызываются тогда, когда либо программно создается этот объект средствами встроенного языка, например, методами СоздатьЭлемент() у менеджеров справочников или СоздатьДокумент() менеджеров документов, либо когда пользователь создает новый элемент справочника или документ интерактивно. Чтобы открыть модуль объекта, нужно в окне редактирования объекта конфигурации перейти на закладку Прочее и нажать кнопку Модуль объекта. Или, выделив нужный объект в дереве объектов конфигурации, вызвать из контекстного меню команду Открыть модуль объекта.

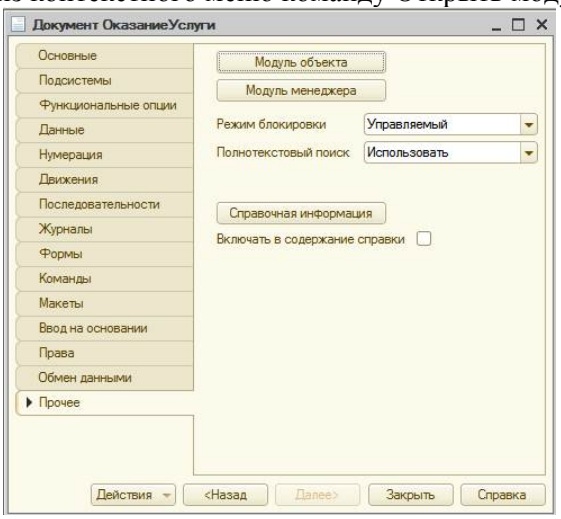
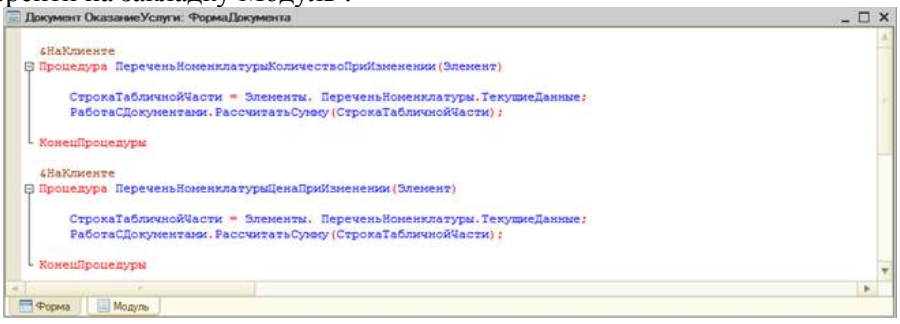


Рис.5.1. Открытие модуля объекта

- *Модули форм.* Каждая форма, определенная в конфигурации, имеет свой собственный модуль. Этот модуль исполняется при создании объекта УправляемаяФорма встроенного языка. Этот объект создается системой в режиме 1С предприятие в тот момент, когда мы программно (методами ПолучитьФорму() или ОткрытьФорму()). Чтобы открыть модуль формы, нужно открыть нужный объект конфигурации Форма объекта и в окне редактора форм перейти на закладку Модуль .



Листинг 5.1. Открытие модуля формы

- *Модуль сеанса.* Модулем сеанса называется модуль, который автоматически выполняется при старте системы 1С:Предприятие в момент загрузки конфигурации. Модуль сеанса предназначен для инициализации параметров сеанса и отработки действий, связанных с

сеансом работы. Модуль сеанса не содержит экспортируемых процедур и функций и может использовать процедуры из общих модулей конфигурации. Чтобы открыть модуль сеанса, нужно выделить корень дерева объектов конфигурации (строку Фамилия) и вызвать из контекстного меню команду Открыть модуль сеанса.

- *Модуль внешнего соединения* предназначен для размещения в нем текстов функций и процедур, которые могут вызываться в сессии внешнего соединения. Чтобы открыть модуль сеанса, нужно выделить корень дерева объектов конфигурации (строку ПособиеДляНачинающих) и вызвать из контекстного меню команду Открыть модуль внешнего соединения.
- *Модуль менеджеров.* Для каждого прикладного объекта существует менеджер, предназначенный для управления этим объектом как объектом конфигурации. С помощью менеджера можно создавать объекты, работать с формами и макетами. Модуль менеджера позволяет расширить функциональность менеджеров, предоставляемых системой, за счет написания процедур и функций на встроенном языке. Чтобы открыть модуль менеджера, нужно в окне редактирования объекта конфигурации перейти на закладку Прочее и нажать кнопку Модуль менеджера.
- *Модуль команды.* Как в самой конфигурации, так и у многих прикладных объектов могут существовать подчиненные объекты конфигурации - Команды. У каждой команды существует модуль команды, в котором можно написать предопределенную процедуру ОбработкаКоманды() для выполнения этой команды. Чтобы открыть модуль команды, подчиненной некоторому объекту конфигурации, нужно в окне редактирования объекта конфигурации перейти на закладку Команды и дважды щелкнуть мышью на нужной команде. Или, выделив нужную команду в дереве объектов конфигурации, вызвать из контекстного меню команду Открыть модуль команды.

Контекст модуля формы

Каждый модуль связан с остальной частью конфигурации. Эта связь называется контекстом модуля.

Контекст модуля определяет набор доступных во время выполнения модуля объектов, переменных, процедур и функций.

Контекст модуля формы образуется:

- локальным контекстом самого модуля формы;
- реквизитами формы, которой ;принадлежит; модуль;
- свойствами и методами объекта УправляемаяФорма встроенного языка;
- свойствами и методами расширения формы, определяемого типом того объекта, данные которого содержатся в основном реквизите формы;
- глобальным контекстом, в том числе неглобальными общими модулями и экспортируемыми функциями и процедурами глобальных общих модулей;
- экспортируемыми переменными, процедурами и функциями модуля управляемого приложения;

Форма как программный объект

Помимо того что форма;внутри; своего модуля предоставляет доступ к различным частям конфигурации, она также доступна из других частей конфигурации как программный объект.

При этом помимо стандартных свойств и методов объекта встроенного языка УправляемаяФорма, у неё могут существовать и другие свойства и методы, определенные разработчиком.

```
&НаКлиенте
Процедура МатериалыКоличествоПриИзменении(Элемент)

    СтрокаТабличнойЧасти = Элементы.Материалы.ТекущиеДанные;
    РаботаСДокументами.РассчитатьСумму(СтрокаТабличнойЧасти);

КонецПроцедуры
```

Листинг 5.2.

Например, если в модуле формы ФормаДокумента документа ПриходнаяНакладная описана

экспортируемая процедура, то может быть использован следующий вызов этой процедуры:

```
Форма=ПолучитФорму("Документ.ПриходнаяНакладная.Форма.ФормаДокумента");
Форма.МатериалыКоличествоПриИзменении();
```

Листинг 5.3.

Как понять работу кода на встроенном языке

Теперь мы покажем два способа, как самому разобраться с множеством незнакомых свойств и методов объектов конфигурации, чтобы в будущем самостоятельно изучать фрагменты кода или создавать свои собственные процедуры на встроенном языке.

Синтакс-помощник — инструмент, созданный для помощи разработчику, содержащий описание всех программных объектов, которые использует система, их методов, свойств, событий и пр.

Чтобы открыть Синтакс-помощник, нужно нажать соответствующую кнопку на панели инструментов конфигуратора или выполнить команду главного меню Справка >Синтакс-помощник, или с помощью горячей клавиши (Ctrl + F1).

Анализ кода с помощью Синтакс-помощника:

Пользоваться Синтакс-помощником удобно в тех случаях, когда нужно разобраться в уже написанном незнакомом коде.

Первый способ - найти нужный раздел в содержании и спускаться вниз «по дереву», раскрывая нужные подразделы, свойства, ссылки и т.п.

Второй способ - воспользоваться контекстной помощью Синтакс-помощника. Для этого нужно открыть программный модуль, установить курсор на интересующую вас конструкцию встроенного языка и нажать Ctrl+ F1.

Есть также еще одна полезная возможность использования Синтакс-помощника. Можно ограничить состав объектов, которые будут отображаться в нем. Так как мы находимся на клиенте, в форме, имеет смысл ограничиться только объектами встроенного языка, доступными в режимах Тонкий клиент и Веб-клиент. Для этого нужно выполнить команду главного меню Сервис > Параметры или нажать кнопку Открыть режим настройки параметров, находящуюся над окном описания объектов Синтакс-помощника. На закладке Справка окна Параметры можно снять или поставить отметку у нужных режимов исполнения.

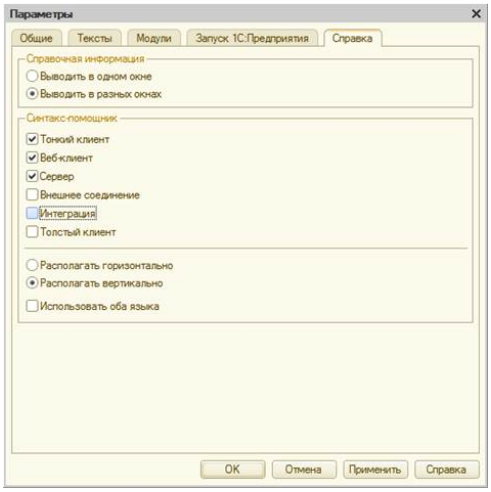


Рис. 5.2. Параметры

Анализ кода с помощью отладчика

Пользоваться отладчиком наиболее удобно в тех случаях, когда нужно написать какой-то собственный код. Можно просто остановиться в конкретном месте программы и посмотреть, какие же свойства здесь доступны или какие программные объекты здесь используются.

Отладчик - вспомогательный инструмент, облегчающий разработку и отладку программных модулей системы 1С:Предприятие. Отладчик предоставляет следующие возможности:

- пошаговое выполнение модуля,
- расстановка точек останова,

- прерывание и продолжение выполнения модуля,
- возможность отладки нескольких модулей одновременно,
- вычисление выражений для анализа состояния переменных,
- просмотр стека вызовов процедур и функций,
- возможность остановки по возникновению ошибки,
- возможность редактирования модуля в процессе отладки.

Объекты

Как правило, термин объект употребляется в одном из трех контекстов:

- конфигурация,
- база данных,
- встроенный язык.

Говоря о конфигурации, термином объект конфигурации мы обозначаем некоторую совокупность описания данных и алгоритмов работы с этими данными. Например, в конфигурации может существовать объект Справочник Сотрудники.

Когда мы говорим о базе данных, термином *объект* мы обозначаем всего лишь некий элемент такой информационной структуры. Характерной особенностью такого элемента является то, что на него (как на совокупность данных) существует ссылка, которая может являться значением какого-либо поля другой информационной структуры.

Если же мы начинаем говорить о встроенном языке и о том, каким образом средствами встроенного языка работать со справочниками, то термином *объект* мы обозначаем тип данных, позволяющий получить доступ к данным и обладающий набором свойств и методов.

Сервер и клиенты

Система 1С:Предприятие поддерживает два варианта работы системы: файловый и клиент-серверный.

Файловый вариант работы с информационной базой рассчитан на персональную работу одного пользователя или работу небольшого количества пользователей в локальной сети. В этом варианте все данные информационной базы (конфигурация, база данных, административная информация) располагаются в одном файле.



Рис. 5.3. Файловый вариант

Основное назначение файлового варианта - быстро и легко установить систему и работать с ней. Например, чтобы что-то посмотреть или доработать дома или на ноутбуке. В файловом варианте тоже можно вести реальную учетную работу, но при этом нужно понимать, что он не предоставляет абсолютно всех тех же возможностей по масштабируемости, защите данных, какие имеет клиент-серверный вариант.

Клиент-серверный вариант предназначен для использования в рабочих группах или в масштабе предприятия - это основной вариант для работы в многопользовательской среде с большим объемом данных. Он предоставляет абсолютно все возможности по масштабируемости, администрированию и защите данных. Однако он требует значительных усилий по установке и администрированию.

Система 1С:Предприятие изначально рассчитана на клиент-серверный вариант работы. Хотя сейчас вы разрабатываете свою учебную конфигурацию в файловом варианте работы, она будет работать и в клиент-серверном варианте без ваших дополнительных усилий.

Прикладные решения разрабатываются один раз и одинаково работают, что в одном, что в другом варианте. То есть переход с одного варианта на другой не требует переделки конфигурации.

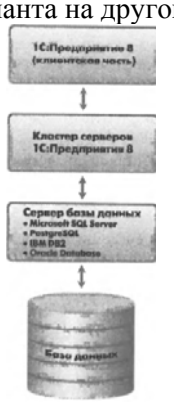


Рис. 5.4. Клиент-серверный вариант

Клиент-серверная архитектура разделяет всю работающую систему на три различные части, определенным образом взаимодействующие между собой, - Клиент, Сервер 1С: Предприятия и Сервер баз данных.

Клиентское приложение - это программа, часть системы 1С: Предприятие. Основное ее назначение - организация пользовательского интерфейса, отображение данных с возможностью их изменения. Кроме этого, клиентское приложение может исполнять код на встроенном языке.Такой подход позволяет клиентскому приложению быть очень «легким», не требовать много ресурсов, «путешествовать» по Интернету и работать даже в среде веб-браузеров.

Сервер баз данных - это тоже программа. Она уже не является частью системы 1С:Предприятие, это специализированная программа, поставляемая сторонними производителями. Ее основное назначение - это организация и ведение баз данных - структурированных организованных наборов данных, описывающих характеристики каких-либо физических или виртуальных систем.

В настоящее время система 1С:Предприятие может работать со следующими серверами баз данных:

- ***MS SQL Server;***
- ***Posteg SQL;***
- ***IBM DB2;***
- ***OracleDatabase.***

Практическаяработа № 3. Регистры накопления.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Зачем нужен регистр накопления

Казалось бы, все необходимое мы с вами уже создали: у нас есть что расходовать и приходовать (справочники), и у нас есть чем расходовать и приходовать (документы). Осталось только построить несколько отчетов, и автоматизация предприятия будет закончена. Однако это не так.

Во-первых, путем анализа документов можно, конечно, получить требуемые нам выходные данные. Но представьте, что завтра предприятие решит немного изменить свои бизнес-планы, и нам потребуется ввести в конфигурацию еще один документ.

Во-вторых, отчеты, анализирующие документы, будут работать довольно медленно, что будет вызывать раздражение пользователей и недовольство руководителей.

Поэтому в системе 1С Предприятие есть несколько объектов конфигурации, которые позволяют создавать в базе данных структуры, предназначенные для накопления информации в удобном для последующего анализа виде. Использование таких «хранилищ» данных позволяет нам, с одной стороны, накапливать в них данные, а с другой стороны, легко создавать нужные нам отчеты или использовать эти данные в алгоритмах работы конфигурации.

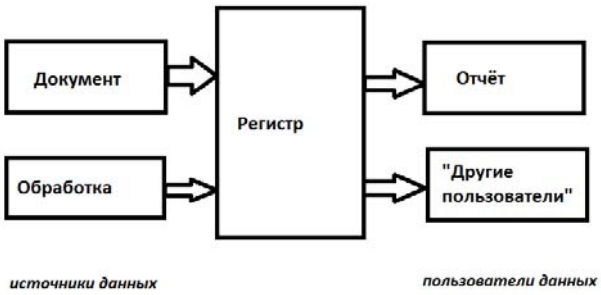


Рис. 6.1. Алгоритм работы конфигурации

В конфигурации существует несколько объектов, называемых регистрами, для описания подобных «хранилищ».

Что такое регистр накопления

Объект конфигурации Регистр накопления предназначен для описания структуры накопления данных. На основе объекта конфигурации Регистр накопления платформа создает в базе данных таблицы, в которых будут накапливаться данные, поставляемые различными объектами базы данных.

Эти данные будут храниться в таблицах в виде отдельных записей, каждая из которых имеет одинаковую, заданную в конфигураторе структуру.

Отличительной особенностью регистра накопления является то, что он не предназначен для интерактивного редактирования пользователем.

Основным назначением регистра накопления является накопление числовой информации в разрезе нескольких измерений, которые описываются разработчиком в соответствующем объекте конфигурации Регистр накопления и являются подчиненными объектами конфигурации.

Виды числовой информации, накапливаемой регистром накопления, называются ресурсами, также являются подчиненными объектами и описываются в конфигураторе.

Изменение состояния регистра накопления происходит, как правило, при проведении документа и заключается в том, что в регистр добавляется некоторое количество записей. Каждая запись содержит значения измерений, значения приращений ресурсов, ссылку на документ, который вызвал эти изменения (регистратор), и направление приращения (приход или расход). Такой набор записей называется движениями регистра накопления. Каждому движению регистра накопления всегда должен соответствовать регистратор -объект информационной базы (как правило, документ), который произвел эти движения.

Кроме этого, регистр накопления может хранить дополнительную информацию, описывающую каждое движение. Набор такой дополнительной информации задается разработчиком при помощи реквизитов объекта конфигурации Регистр накопления.

Движения документа

Движения документа - это записи в регистрах, которые создаются в процессе проведения документа и отражают изменения, производимые документом.

Способы работы с коллекцией

В процессе формирования движений документов, когда в цикле обходили табличные части документов ПриходнаяНакладная и ОказаниеУслуги, вы столкнетесь с одним из объектов встроенного языка, который является коллекцией.

Многие объекты встроенного языка являются коллекциями. Коллекция представляет собой совокупность объектов. Существуют общие принципы работы с любой коллекцией.

Во-первых, доступ к каждому объекту коллекции возможен путем перебора элементов коллекции в цикле. Для этого используется конструкция языка Для Каждого Из... Цикл ...

Листинг 6.1: Перебор элементов коллекции в цикле

```
Для Каждого СтрокаТабличноиЧасти из ТабличнаяЧасть Цикл
Сообщить(СтрокаТабличнойЧасти.Услуга);
КонецЦикла;
```

В этом примере ТабличнаяЧасть - это коллекция строк табличной части объекта конфигурации. При каждом проходе цикла в переменной СтрокаТабличнойЧасти будет содержаться очередная строка из этой коллекции.

Во-вторых, существует доступ напрямую к элементу коллекции, без перебора коллекции в цикле. Здесь возможны различные комбинации двух обращений.

1. Во встроенном языке бывают именованные коллекции. То есть коллекции, в которых каждый элемент имеет некоторое уникальное имя. В этом случае обращение к элементу коллекции возможно по этому имени.

Листинг 6.2: Обращение к элементу коллекции

Справочники.Сотрудники
Справочники[“Сотрудники”];

2. Если нет смысла в «персонификации» элементов коллекции (коллекция неименованная), тогда обращение к элементу коллекции возможно по индексу (индекс первого элемента коллекции - ноль).

Листинг 6.3: Обращение к элементу коллекции по индексу

ТабличнаяЧасть[0];

В этом примере ТабличнаяЧасть - это коллекция строк табличной части объекта конфигурации. И мы обращаемся к первому элементу этой коллекции, указывая его индекс - 0. Следует отметить, что существуют коллекции, сочетающие оба вида обращений. Например, к коллекции колонок таблицы значений можно обращаться как по именам колонок, так и по индексу.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление регистра накопления

В режиме конфигуратор

Откроем в конфигураторе нашу учебную конфигурацию и добавим новый объект конфигурации Регистр накопления. Для этого выделим в дереве объектов конфигурации ветвь Регистры накопления и нажмем кнопку Добавить в командной панели окна конфигурации.

- *На закладкеОсновные*: имя регистра – ОстаткиМатериалов. Расширенное представление списка как Движения по регистру Остатки материалов. Этот заголовок будет отображаться в окне списка записей регистра.

- *На закладкеПодсистемы: отметим в списке следующие подсистемы:*Учет материалов, Оказание услуг и Бухгалтерия.

- *На закладкеДанные:*нажмем на «+» и добавим следующие:

- Материал, тип СправочникСсылка. Номенклатура;
- Склад, тип СправочникСсылка.Склады.
- Ресурс**Количество**с длиной 15 и точностью 3.

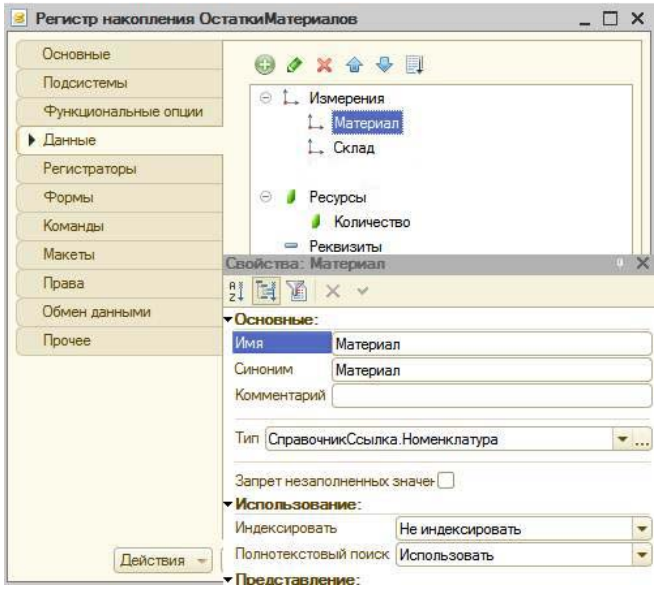


Рис. 6.2. Создание измерений регистра

В результате этих действий регистр**ОстаткиМатериалов**должен иметь следующий вид:

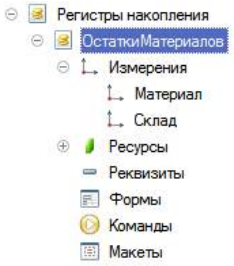


Рис. 6.3. ОстаткиМатериалов

Если вы сейчас попытаетесь запустить 1С:Предприятие в режиме отладки, то система выдаст сообщение об ошибке:**«РегистрНакопления.ОстаткиМатериалов: Ни один из документов не является регистратором для регистра».**Это сообщение ещё раз подтверждает тот факт, что назначение регистра накопления в том, чтобы аккумулировать данные, поставляемые различными документами.

Поэтому мы сформируем движения регистра накопления **ОстаткиМатериалов**в процессе проведения двух созданных нами документов **ПриходнаяНакладная**и **ОказаниеУслуг**.

Откроем окно редактирования объекта конфигурации **Документ ПриходнаяНакладная**. Перейдем на закладку **Движения**, раскроем список Регистры накопления и отметим регистр накопления **ОстаткиМатериалов**.

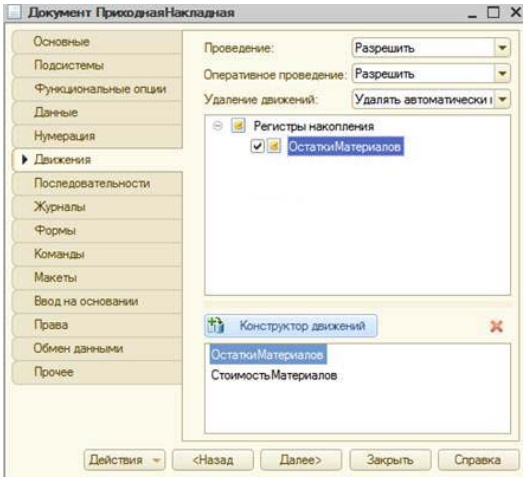


Рис. 6.4. Создание движения документа

После чего активизируется**Конструктор движений,**воспользуемся этим конструктором.

Конструктор устроен просто. В списке**Регистры**перечислены регистры, в которых документ может создавать движения. В нашем случае там пока один регистр**ОстаткиМатериалов**.

В списке**Реквизиты документа**должны находиться исходные данные для создания движений - реквизиты документа**ПриходнаяНакладная**. А в таблице**Поле – Выражение**должны быть заданы формулы, по которым будут вычисляться значения измерений и ресурсов регистра при записи движений.

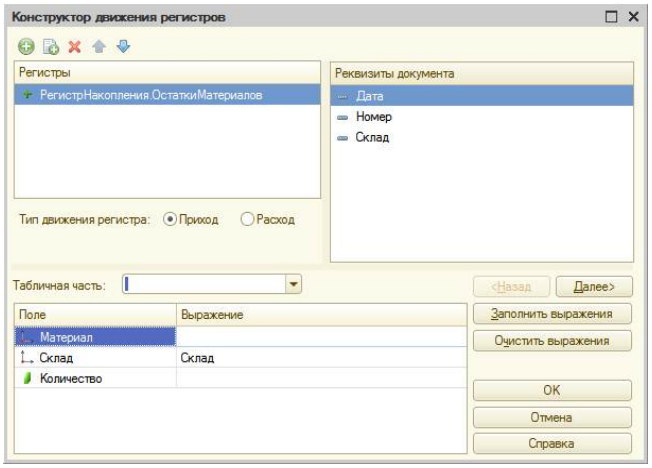


Рис. 6.5. Конструктор движения регистров

В поле выбора**Табличная часть**выберем табличную часть нашего документа -**Материалы**.

Список реквизитов документа, который уже заполнен реквизитами шапки документа, автоматически дополнится реквизитами нашей табличной части.

Теперь нажмем кнопку**Заполнить выражения.**

В нижнем окне сформируется соответствие полей (измерений и ресурсов) регистра и выражений для их расчета.

Нажмем кнопку ОК и посмотрим, какой текст сформировал конструктор в модуле документа**ПриходнаяНакладная**.

*Конструктор создал обработчик события**ОбработкаПроведения**объекта конфигурации Документ**ПриходнаяНакладная**,поместил его в модуль объекта и открыл текст модуля.*

В заключение отредактируем командный интерфейс, чтобы в подсистемах**Бухгалтерия, Оказание услуги****Учет материалов**была доступна ссылка для просмотра записей нашего регистра накопления.

*- Вдереве объектов конфигурации выделим ветвь**Подсистемы**,вызовем ее контекстное меню и выберем пункт**Все подсистемы:***

- В списке**Подсистемы**выделим подсистему**УчетМатериалов**.
- В группе**Панель навигации.Обычное**включим видимость у команды**Остатки материалов**и мышью перетащим её **См.также.**панели навигации

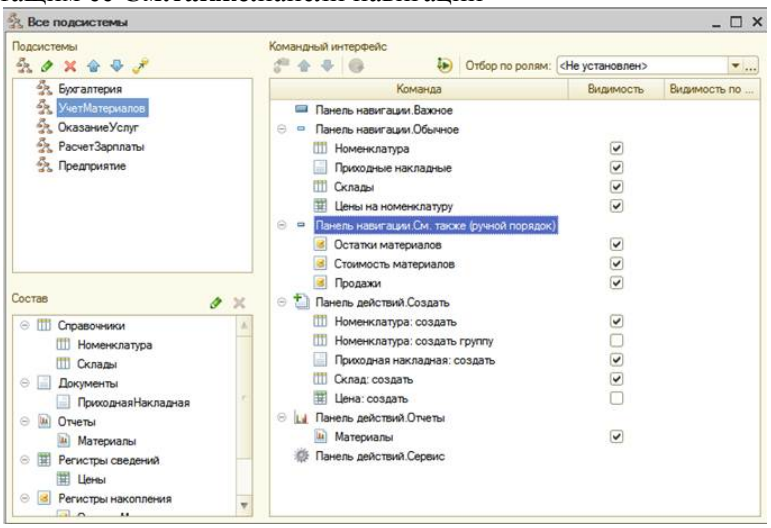


Рис. 6.6. Панель навигации См. также

- Аналогично, выделив подсистемы**ОказаниеУслуги****Бухгалтерия**, в панели навигации в группе**Обычное**включим видимость у команды**Остатки материалов**и перенесём ее в группу**См.также.**панели навигации

В режиме 1С:Предприятие

В открывшемся окне 1С предприятия мы видим, что в панели навигации в группе**См.также**разделов**Бухгалтерия**,**Оказаниеуслуги****Учёт**материалов появилась команда для открытия списка регистра**Остатки материалов**.

Чтобы проследить связь между проведением документа и накоплением информации в регистре, откроем список приходных накладных, выполнив команду**Приходные накладные**разделе**Бухгалтерия**.

Откроем**Приходную накладную №**1 и нажмём**Провести и закрыть**, то есть перепроведем ее. То же самое сделаем для**Приходной накладной № 2.**

Перепровести документы можно и не открывая документов. Для этого нужно выделить нужный документ в списке, нажать кнопку**Все действия**в командной панели формы списка и выбрать пункт**Провести**.

Теперь выполним команду**Остатки материалов**и откроем список нашего регистра накопления:

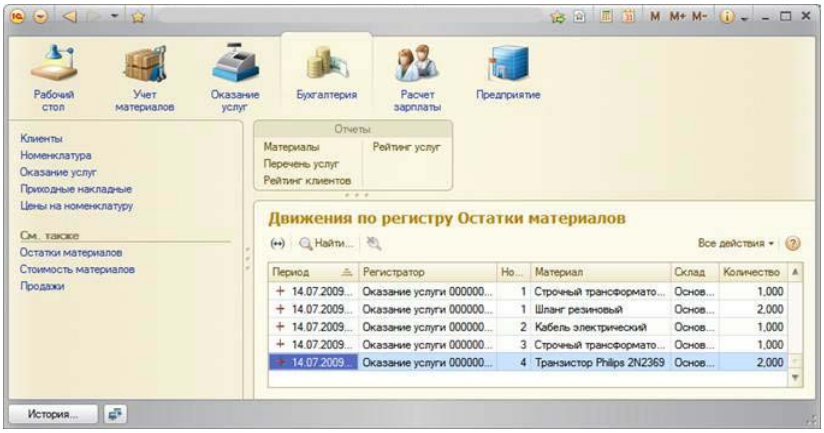


Рис. 6.7. Движения по регистру Остатки материалов

Команда перехода к движениям в форме документа

- **В режиме Конфигуратор**

При реальной работе записей в регистре**ОстаткиМатериалов**будет много, и будет трудно понять, какие записи относятся к определенному документу.

Поэтому наряду с общим списком регистра хотелось бы иметь возможность вызывать из формы документа список регистра, в котором показаны движения, произведенные только этим документом.

Чтобы реализовать такую возможность, вернемся в конфигуратор и откроем форму документа**ПриходнаяНакладная**.

В левом верхнем окне перейдём на закладку**Командныйинтерфейс**: в разделе Панель навигации раскроем группу перейти, установим свойство**Видимость**для этой команды

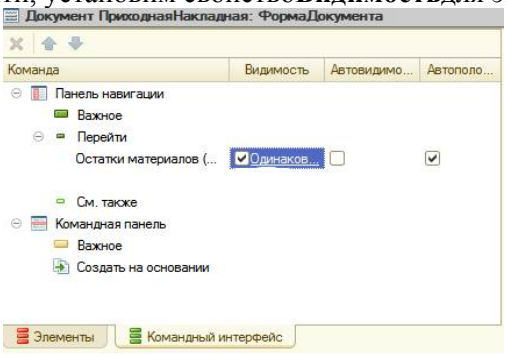


Рис. 6.8. Командный интерфейс

В режиме 1С:Предприятие

Запустим 1С: Предприятие в режиме отладки и откроем**Приходную накладную№2.**

В форме документа появилась панель навигации, в которой мы можем переходить к списку записей регистра**Остатки Материалов**, связанному с документом, и обратно к содержимому

документа.

Движения документа «Оказание услуги»

В режиме Конфигуратор

Откроем окно редактирования объекта конфигурации Документ**ОказаниеУслуги**.

Перейдем на закладку Движения и в списке регистров конфигурации отметим регистр накопления**ОстаткиМатериалов**.

- *Нажмём кнопку**Конструктор движений***:

1.Тип движения регистра-**Расход**

2. В поле выбора**Табличнаячасть**выберем табличную часть нашего документа – **ПеречеьНоменклатуры**.

3.Нажмем кнопку**Заполнить выражения**.

4. Выделим поле регистра**Материал**и в окне**Реквизитыдокумента**дважды щёлкнуть по строке**ТекСтрокаПеречеьНоменклатуры.Номенклатура**., нажмем**«ОК»**

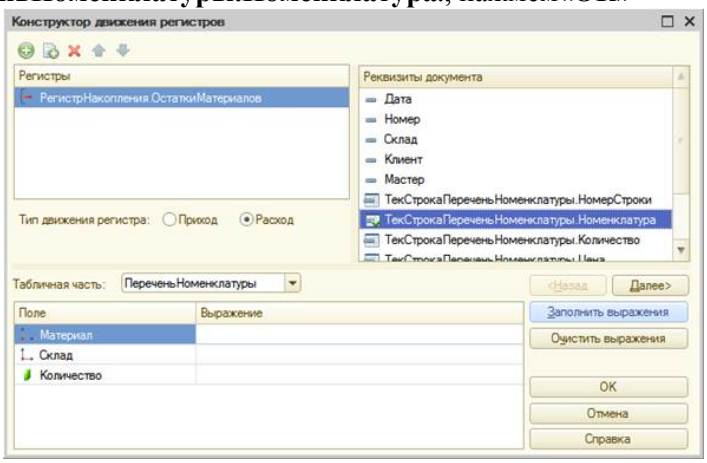


Рис. 6.9. Конструктор движения регистров

В заключение отредактируем командный интерфейс формы документа, чтобы в панели навигации формы иметь возможность переходить к списку записей регистра*Остатки Материалов,*связанному с документом.

Для этого откроем форму документа**ОказаниеУслуги**.

В левом верхнем окне перейдём на закладку**Командныйинтерфейс**.

В разделе**Панельнавигации**раскроем группу**Перейти**и установим видимость для команды открытия регистра накопления**Остаткиматериалов**.

В режиме 1С:Предприятие

Запустим 1С:Предприятие в режиме отладки и в разделе**Оказаниеуслуг**откроем документ**Оказание услуги № 1**и нажмём**Провести и закрыть**, то есть перепроведем его.

Теперь выполним команду**Остаткиматериалов**и откроем список нашего регистра накопления:

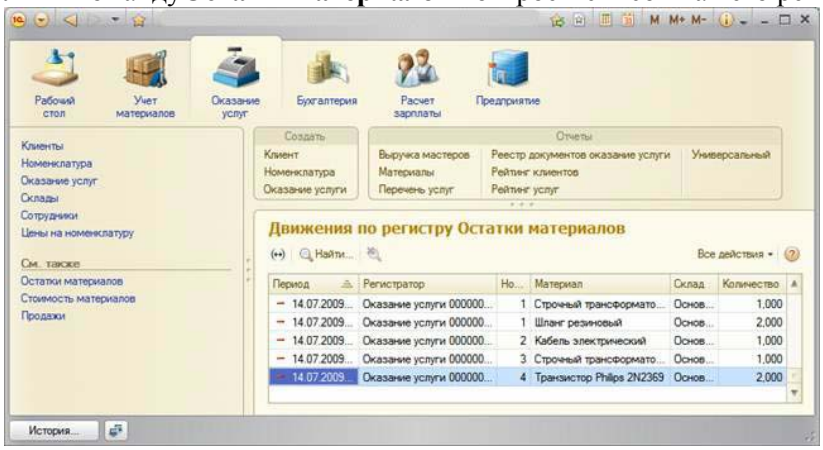


Рис. 6.10. Движения по регистру Остатки материалов

Мы видим, что в регистре накопления Остатки материалов появилась еще одна запись, что соответствует количеству строк в табличной части проведенного документа.

Все поля регистра заполнились данными документа так, как мы задали в обработчике проведения документа**Оказаниеуслуги**.

Пиктограмма со знаком «минус» слева от записи указывает на тип движения – Расход.

Сейчас мы видим весь список движений регистра. Открыв этот список из формы документа, мы можем отфильтровать движения по документу-регистратору.

Для этого откроем еще раз документ**Оказаниеуслуги№1.**

В форме документа появилась панель навигации, в которой мы можем переходить к списку записей регистра Остатки материалов, связанному с документом, и обратно к содержимому документа.

Сформированные таким образом движения этого документа будут не совсем правильны.

Дело в том, что в документе**Оказаниеуслуги**, в отличие от документа Приходная накладная, могут содержаться не только расходуемые материалы, но и услуги. Поэтому в регистр Остатки материалов будут попадать записи и о расходуемых услугах, что неправильно.

Пока мы ничего не будем делать с движениями, которые сформировал конструктор, но как только познакомимся с перечислениями, мы внесем вобработчик проведения необходимые изменения.

Контрольные вопросы

- *Длячегопредназначенобъектконфигурации**Регистр накопления.***
- *Почему следует использовать регистры, хотя необходимая информация содержится в других объектах.*
- *Для чего нужны измерения регистра, ресурсы и реквизиты.*
- *Что такое движения регистра и что такое регистратор.*
- *Как создать новый регистр накопления и описать его структуру.*
- *Как создать движения документа с помощью конструктора движений.*
- *Как средствами встроенного языка обойти табличную часть документа и обратиться к ее данным.*
- *Какпоказатькомандыоткрытияспискарегистра в интерфейсе конфигурации и в интерфейсе формы.*

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Что такое отчет

Объект конфигурации Отчет предназначен для описания алгоритмов, при помощи которых пользователь сможет получать необходимые ему выходные данные. Алгоритм формирования выходных данных описывается при помощи визуальных средств или с использованием встроенного языка. В реальной жизни объектам конфигурации Отчет соответствуют всевозможные таблицы выходных данных, сводных данных, диаграммы и пр.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление отчета

- **В режиме Конфигуратор**

Приступим к созданию отчета, который будет показывать нам приход, расход и остатки материалов.

Откроем в конфигураторе нашу учебную конфигурацию и добавим новый объект конфигурации**Отчет**.

Для этого выделим в дереве объектов конфигурации ветвь Отчеты и нажмем кнопку**Добавить**в командной панели окна конфигурации.

- *На закладке**Основные:***имя отчета -**Материалы**.Нажмем кнопку**Открыть схему компоновки данных**или кнопку открытия со значком лупы.

Макет

Так как у отчета, который мы создаем, еще не существует схемы компоновки данных, платформа предложит создать новую схему. Схема компоновки данных с точки зрения конфигурации является макетом, поэтому будет открыт конструктор макета, предлагающий выбрать единственный тип макета **-Схема компоновки данных**и нажмем кнопку**Готово**:



Рис. 7.1. Конструктор макета

Схема компоновки данных

Платформа создаст новый макет, содержащий схему компоновки данных, и сразу же откроет конструктор схемы компоновки данных.

- Добавим новый**набор данных - запрос**. Для этого нажмем кнопку**Добавить**и выберем соответствующий вариант.
- Добавимтекст запроса, запустим конструктор запроса нажмем кнопку**Конструктор запроса**.

Конструктор запроса - инструмент, созданный для помощи разработчику, позволяющий визуально конструировать запрос. Даже пользователь, не знакомый с языком запросов, может с помощью конструктора создать синтаксически правильный запрос.

В списке**Базаданных**представлены таблицы для создания запроса. На основе их данных мы имеем возможность построить отчет.

Если раскрыть ветку**РегистрыНакопления**, то мы увидим, что кроме таблицы регистра**ОстаткиМатериалов**в этой ветке присутствуют еще несколько виртуальных таблиц, которые формирует система.

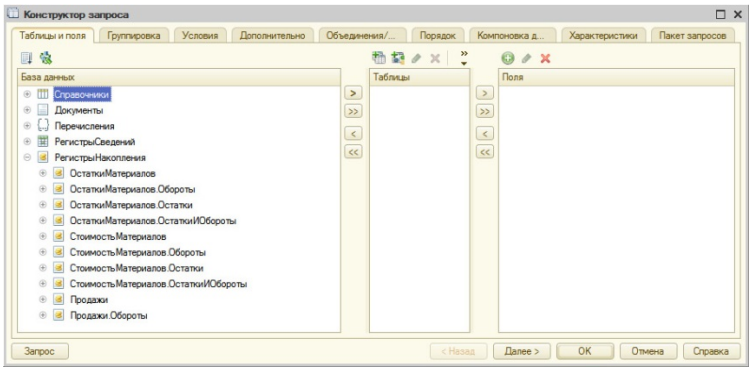


Рис. 7.2. Конструктор запроса

Поскольку мы хотим видеть как остатки материалов, так и информацию об их поступлении и расходовании, нас будет интересовать виртуальная таблица**ОстаткиМатериалов.ОстаткиИОбороты**. Раскроем ее.

Начнем выбирать поля таблицы в нужном нам порядке двойным щелчком мыши. Сначала выберем**Склади****Материал,**потом отметим**КоличествоНачальныйОстаток, КоличествоПриход, КоличествоРасход и КоличествоКонечныйОстаток.**

Нажмем ОК и вернемся в конструктор схемы компоновки данных.

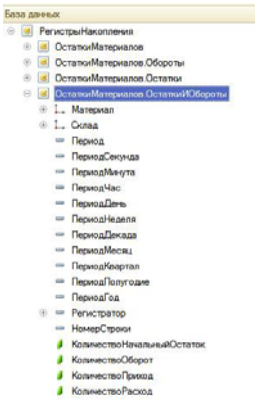


Рис. 7.3. Выбор баз данных

Текст запроса, который был создан с помощью конструктора, платформа поместит в поле**Запрос**.

Это поле представляет собой текстовый редактор, в котором можно вручную отредактировать существующий запрос. Кроме того, можно снова вызвать конструктор запроса и отредактировать запрос при помощи него.

Настройки отчета

- *На закладке**Настройки***:

- выделим в дереве структуры отчета корневой элемент**Отчет**и вызовем его контекстное меню. Можно также нажать кнопку**Добавить**в командной панели окна или нажать клавишу**Ins**.
- Добавим в отчет группировку –**Новаягруппировка**. При этом не станем указывать поле группировки, а просто нажмем ОК.

В структуре отчета появится группировка**Детальныезаписи**.

Теперь настроим поля, которые будут выводиться в результат отчета.

- Перейдем в нижнем окне настроек на закладку**Выбранныеполя**и перенесем мышью из списка доступных полей:
- Склад,
- Материал,
- КоличествоНачальныйОстаток,
- КоличествоПриход,
- КоличествоРасход,
- КоличествоКонечныйОстаток.

В результате окно настроек отчета должно иметь вид:

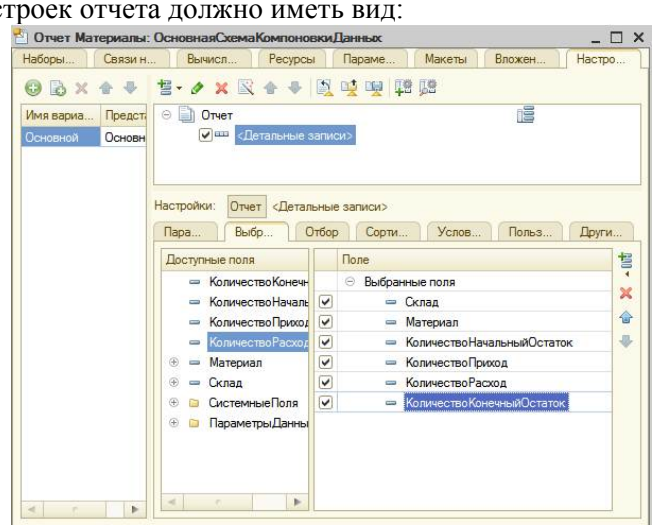


Рис. 7.4. Окно настроек отчета

- Перейдем на закладку**Параметры**и укажем, что параметры отчета**Дата начала**и**Дата окончания**будут включены в состав пользовательских настроек, и эти настройки будут

находиться непосредственно в форме отчета, то есть будут «быстрыми» настройками.

Сначала укажем, что оба эти параметра будут использоваться в отчете - установим флажки в первой колонке.

Затем выделим каждый из параметров, нажмем кнопку**Свойства элемента пользовательских настроек**и поставим флажок**Включать в пользовательские настройки**.

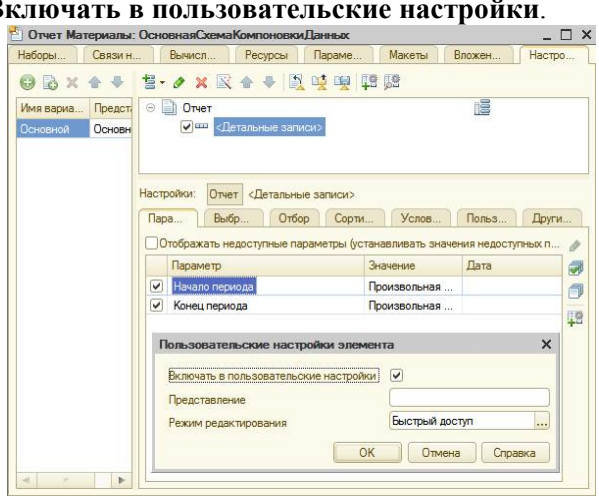


Рис. 7.5. Отчет Материалы

После проделанных действий перейдем по вкладке **Подсистемы**: отметим**Бухгалтерия, УчетМатериалов, ОказаниеУслуг.**

- **В режиме 1С:Предприятие**

Запустим 1С:Предприятие в режиме отладки и посмотрим, как работает отчет.

В открывшемся окне 1С предприятия мы видим, что в панели действий разделов**Бухгалтерия, Оказание услугиУчет материалов**появилась новая группа команд для выполнения отчетов и в ней команда для формирования отчета Материалы. Выполним ее.

Контрольные вопросы

- *Для чего предназначен объект конфигурации Отчет.*
- *Как создать отчет с помощью конструктора схемы компоновки данных.*
- *Как отобразить отчет в разделах прикладного решения.*

Практическаяработа № 4. Макеты.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Что такое макет

Объект конфигурации Макет предназначен для хранения различных форм представления данных, различных данных, которые могут потребоваться каким-либо объектам конфигурации или всему прикладному решению в целом.

Макет может содержать табличный или текстовый документ, двоичные данные, HTML-документ или ActiveDocument, графическую или географическую схему, схему компоновки данных или макет оформления схемы компоновки данных.

Макеты могут существовать как сами по себе, так и быть подчинены какому-либо объекту конфигурации.

Одно из предназначений макета, подчиненного объекту конфигурации и содержащего табличный документ, - создание печатной формы этого объекта. Создание печатной формы заключается в конструировании ее составных частей - именованных областей, из которых затем «собирается» готовая печатная форма.

Порядок заполнения областей данными и вывода их в итоговую форму описывается при помощи встроенного языка. Печатная форма может включать в себя различные графические объекты: картинки, OLE-объекты, диаграммы и т. д.

Помимо создания макета «вручную» конфигуратор предоставляет разработчику возможность воспользоваться специальным инструментом - конструктором печати, который берет на себя большинство рутинной работы по созданию макета.

Макет печатной формы

- **В режиме Конфигуратор**

Откроем в конфигураторе окно редактирования объекта конфигурации Документ **ОказаниеУслуги**.

- На закладке ***Макеты:*** запустим конструктор печати.

- В окне ***конструктора печати документа*** на первом шаге укажем: «**Создать новую команду**», имя команды: «**Печать**».

-Нажмем ***«Далее»-***

- На втором шаге нажатием кнопки >> определим, что все реквизиты нашего документа будут отображены в шапке печатной формы:

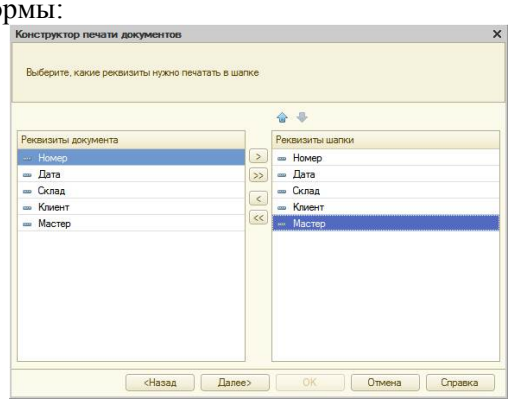


Рис. 8.1. Конструктор печати. Шаг 2

-Нажмем ***«Далее»-***

- На третьем шаге точно так же определим, что все реквизиты табличной части документа будут отображены в печатной форме:

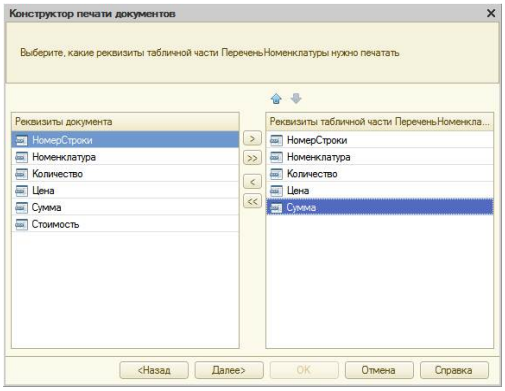


Рис. 8.2. Конструктор печати. Шаг 3

-Нажмем ***«Далее»-***

- Начетвертом шаге и пятом шагах указывать ничего не будем. Тем самым согласимся с тем, что команда для вызова процедуры формирования печатной формы будет помещена в командную панель формы, в раздел Важное.

*-Нажмем «****ОК****»-*

В конфигураторе откроется модуль команды Печать, модуль менеджера документа **ОказаниеУслуги** и макет этого документа:

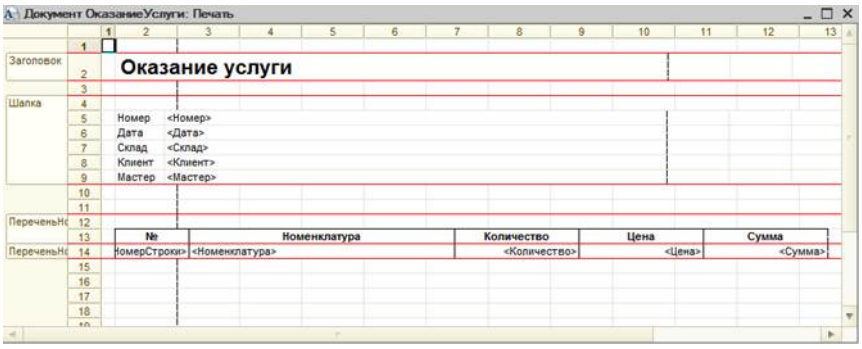


Рис. 8.3. Макет документа "Оказание услуги"

Заметим, что разработчик может создать макет печатной формы с нуля и для ее вывода создать соответствующую команду и кнопку в форме документа, но в данном случае всю работу сделал за нас конструктор печати.

В режиме 1С:Предприятие

Запустим 1С:Предприятие в режиме отладки и откроем документ **Оказание услуги №1.**

Обратите внимание, что в командной панели документа появилась новая кнопка **Печать**. Нажмем на нее и увидим печатную форму нашего документа

Как видите, конструктор сформировал вполне подходящую печатную форму для нашего документа. Единственное, чего не хватает в данной форме, - это итоговой суммы документа.

В следующем разделе, на примере добавления итоговой суммы документа, мы познакомимся с тем, как можно редактировать макеты и формы объектов конфигурации.

Редактирование макета

В режиме Конфигуратор

*- Добавим итоговую сумму в печатную форму документа **ОказаниеУслуги:***

1) Откроем конфигуратор, раскроем дерево документа **ОказаниеУслуги** и дважды щёлкнем на макете **Печать**.

2) Добавим новую область для вывода итоговой суммы документа. Выделим мышью две пустые строки под табличной частью документа и выполним **Таблица > Имена> Назначить имя...**

3) Назовем область «**Всего»**, нажмем «**ОК**».

Чтобы формат добавленных нами строк совпадал с имеющимся форматом заголовка и табличной части документа, изменим ширину колонок.

4) Аналогичные действия выполним и для колонок 3,4, 5 и 6.

5) В созданной области, в колонке **Цена**, напишем **ВСЕГО**:, а в колонке **Сумма** напишем **ВсегоПоДокументу**.

6) Вызвав палитру свойств для последней заполненной нами ячейки (контекстное меню - Свойства), в свойстве Заполнение укажем, что в этой ячейке будет находиться не текст, а параметр.

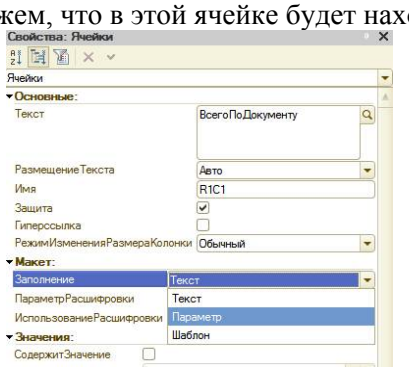


Рис. 8.4. Свойства ячейки "ВсегоПоДокументу"

Каждая ячейка редактируемого нами табличного документа может содержать либо текст, либо некоторый параметр, либо шаблон.

Текст, содержащийся в ячейке, будет показан на экране.

Параметр будет заменен некоторым значением, которое может быть присвоено ему средствами встроенного языка. Текст, содержащийся в ячейке, является именем этого параметра.

Шаблон представляет собой текстовую строку, в определенные места которой будут вставлены значения параметров.

Поэтому, указав для ячейки в качестве заполнения Параметр, мы определили параметр области с именем **ВсегоПоДокументу**, которому присвоим нужное нам значение при формировании печатной формы.

*- Откроем модуль менеджера документа **ОказаниеУслуги**:*

1) Для этого перейдем на закладку **Прочее** окна редактирования объекта конфигурации Документ **ОказаниеУслуги** и нажмем кнопку Модуль менеджера

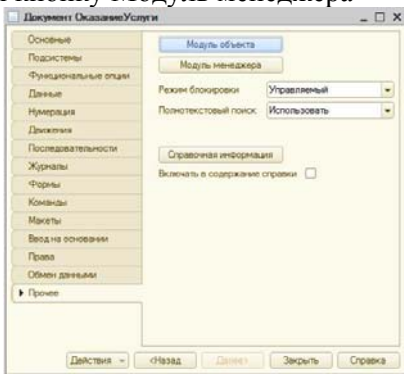


Рис.8.5. Документ ОказаниеУслуги

Найдем в нем процедуру **Печать** и отредактируем ее:

```
Запрос.Параметры.Вставить("Ссылка", Ссылка);
Выборка = Запрос.Выполнить().Выбрать();


ОбластьЗаголовок = Макет.ПолучитьОбласть("Заголовок");
Шапка = Макет.ПолучитьОбласть("Шапка");
ОбластьПереченьНоменклатурыШапка = Макет.ПолучитьОбласть("ПереченьНоменклатурыШапка");
ОбластьПереченьНоменклатуры = Макет.ПолучитьОбласть("ПереченьНоменклатуры");
ОбластьИтог = Макет.ПолучитьОбласть("Всего");
ТабДок.Очистить();

ВставлятьРазделительСтраниц = Ложь;
Пока Выборка.Следующий() Цикл
    Если ВставлятьРазделительСтраниц Тогда
        ТабДок.ВывестиГоризонтальныйРазделительСтраниц();
    КонецЕсли;

    ТабДок.Вывести(ОбластьЗаголовок);

    Шапка.Параметры.Заполнить(Выборка);
    ТабДок.Вывести(Шапка, Выборка.Уровень());

    ТабДок.Вывести(ОбластьПереченьНоменклатурыШапка);
    ВыборкаПереченьНоменклатуры = Выборка.ПереченьНоменклатуры.Выбрать();
    СуммаИтог = 0;
    Пока ВыборкаПереченьНоменклатуры.Следующий() Цикл
        ОбластьПереченьНоменклатуры.Параметры.Заполнить(ВыборкаПереченьНоменклатуры);
        ТабДок.Вывести(ОбластьПереченьНоменклатуры, ВыборкаПереченьНоменклатуры.Уровень());
        СуммаИтог = СуммаИтог + ВыборкаПереченьНоменклатуры.Сумма;
    КонецЦикла;

    ОбластьИтог.Параметры.ВсегоПоДокументу = СуммаИтог;
    ТабДок.Вывести(ОбластьИтог);

    ВставлятьРазделительСтраниц = Истина;
КонецЦикла;
```

Листинг 8.1. Редактируем процедуру Печать

Смысл добавленного фрагмента прост. Мы обращаемся к макету документа **ОказаниеУслуги** по его имени - **Макет**. Используя его метод **ПолучитьОбласть()**, получаем область**Всего** (ту, которую мы только что добавили к макету) и сохраняем её в переменной **ОбластьИтог**. В цикле обхода строк табличной части документа, полученных в результате выполнения запроса, мы

накапливаем в переменной **СуммаИтог** значение суммы табличной части документа по колонке **Сумма**.

Затем мы обращаемся к параметру **ВсегоПоДокументу**(ОбластьИтог.Параметры.ВсегоПо Документу), находящемуся в области **Всего**, и присваиваем ему значение переменной **СуммаИтог**. В заключение мы выводим итоговую область в табличный документ, который будет показан на экране и распечатан пользователем - **ТабДок.Вывести(ОбластьИтог)**.

Отображение табличного документа на экране выполняется в обработчике команды **Печать**, в модуле этой команды на клиенте, в то время как сама процедура печати, описанная в модуле менеджера документа, выполняется на сервере.

В режиме 1С:Предприятие

Запустим 1С:Предприятие в режиме отладки и проверим результат наших изменений.

Подобным образом, создавая именованные области и ячейки макета, используя их свойства и управляя порядком их вывода с помощью встроенного языка, разработчик имеет возможность создать печатную форму любого дизайна.

А теперь, для того чтобы наш документ **ОказаниеУслуги**, выглядел вполне законченным, добавим итоговую сумму по документу и на экранную форму, чтобы пользователь мог видеть ее в процессе заполнения табличной части документа.

Редактирование формы

В режиме Конфигуратор

После того как мы вывели итоговую сумму по табличной части в печатную форму документа, возникло естественное желание видеть такую же итоговую сумму и в форме документа. Чтобы в процессе его создания можно было оперативно, не печатая документ, знать итоговую сумму по документу.

Для этого мы внесем небольшие изменения в форму документа **ОказаниеУслуги**.

Для редактирования формы документа откроем конфигуратор, раскроем дерево документа **ОказаниеУслуги** и дважды щёлкнем на форме **ФормаДокумента**.

1) Дважды щёлкнем на элементе **ПереченьНоменклатуры** и установим свойство **Подвал**, которое определяет наличие подвала у таблицы формы.

2) Затем откроем свойства элемента формы **ПереченьНоменклатурыЦена** и установим:

- Текст подвала – **Всего**;
- Горизонтальное положение в подвале – **Право**;
- Свойство шрифт подвала изменим начертание на **Жирный** .

3) Откроем свойства элемента **ПереченьНоменклатурыСумма**, установим:

- Горизонтальное положение в подвале - Право.
- Свойство Шрифт подвала - Жирный.

Для того чтобы в подвале колонки Сумма отображался итог по ней, нажмем кнопку выбора в поле **ПутьКДаннымПодвала.** Раскроем дерево реквизитов объекта и выберем элемент **ИтогСумма**:

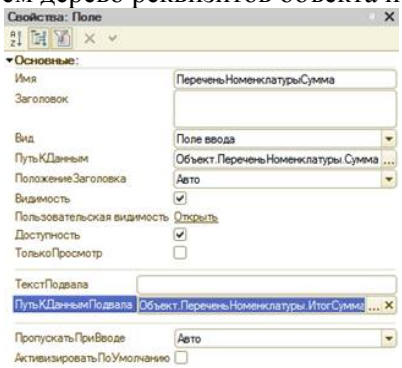


Рис. 8.6. Свойства Поле

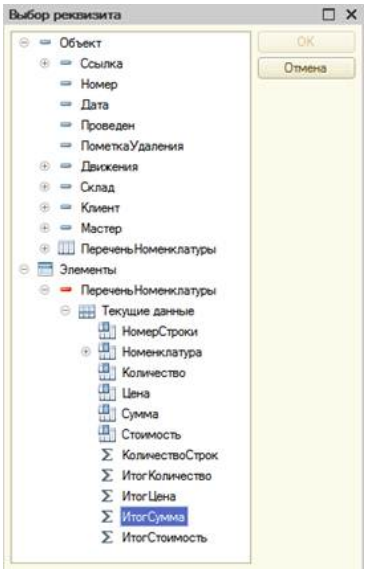


Рис. 8.7. Выбор объекта ИтогСумма

В режиме 1С:Предприятие

Запустим 1С:Предприятие в режиме отладки и посмотрим, как теперь выглядит форма документа Оказание услуги № 1.

Мы видим, что по колонке Сумма в табличной части документа, подсчитывается общий итог документа.

Контрольные вопросы

- *Для чего предназначен объект конфигурации Макет.*
- *Что такое конструктор печати.*
- *Как создать макет с помощью конструктора печати.*
- *Как изменить табличный документ.*
- *Какая разница в заполнении ячейки табличного документа текстом, параметром и шаблоном.*
- *Как с помощью встроенного языка вывести в табличный документ новую область.*
- *Как изменить внешний вид и поведение элемента формы.*
- *Как отобразить сумму по колонке таблицы.*

Практическаяработа № 5. Периодические регистры сведений.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Зачем нужен периодический регистр сведений

Начнем мы с того, что обратим, ваше внимание на документ **ОказаниеУслуги**. Как вы помните, в этом документе мы выбираем услугу, которая оказывается, и затем указываем цену.

Очевидно, что в компании существует перечень услуг, который определяет стоимость каждой услуги. Казалось бы, стоимость услуги является неотъемлемым свойством самой услуги, и поэтому ее следует добавить в качестве реквизита справочника **Номенклатура**.

Однако стоимость услуг имеет особенность меняться со временем. И может сложиться такая ситуация, когда нам потребуется внести изменения или уточнения в один из ранее проведенных документов **Оказаниеуслуги**. В этом случае мы не сможем получить правильную стоимость услуги, поскольку в реквизите справочника будет храниться последнее введенное значение.

Для хранения стоимости услуг мы используем новый объект - **Регистр сведений**.

Что такое регистр сведений

Объект конфигурации Регистр сведений предназначен для описания структуры хранения данных в разрезе нескольких измерений. На основе объекта конфигурации Регистр сведений платформа создает в базе данных таблицу, в которой может храниться произвольная информация, «привязанная» к набору измерений.

Принципиальное отличие регистра сведений от регистра накопления заключается в том, что

каждое движение регистра сведений устанавливает новое значение ресурса, в то время как движение регистра накопления изменяет существующее значение ресурса. По этой причине регистр сведений может хранить любые данные.

Следующей важной особенностью регистра сведений является его способность хранить данные с привязкой ко времени. Благодаря этому регистр сведений может хранить не только актуальные значения данных, но и историю их изменения во времени - периодический регистр сведений.

Периодичность регистра сведений можно определить одним из следующих значений:

- в пределах секунды,
- в пределах дня,
- в пределах месяца,
- в пределах квартала,
- в пределах года,
- в пределах регистратора .

Периодический регистр сведений всегда содержит служебное поле **Период**, добавляемое системой автоматически. Оно имеет тип **Дата** и служит для указания факта принадлежности записи к какому-либо периоду. При записи данных в регистр платформа всегда приводит значение этого поля к началу того периода, в который он попадает.

Например, если в регистр сведений с периодичностью в пределах месяца записать данные, в которых период указан как 08.04.2004, то регистр сохранит эти данные со значением периода, равным 01.04.2004.

Как и для других регистров, система контролирует уникальность записей для регистра сведений. Однако если для прочих регистров уникальным идентификатором записи является регистратор и номер строки, то для регистра сведений применяется другой принцип формирования ключевого значения.

Ключом записи, однозначно идентифицирующим запись, является в данном случае совокупность значений измерений регистра и периода (в случае если регистр сведений периодический). Регистр сведений не может содержать несколько записей с одинаковыми ключами.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление периодического регистра сведений

В режиме Конфигуратор

Откроем в конфигураторе конфигурацию и добавим новый объект конфигурации **Регистр сведений**.

Для этого выделим в дереве объектов конфигурации ветвь **Регистрысведений** и нажмем кнопку**Добавить**.

*- На закладке **Основные:*** Имя регистра - **Цены**. Периодичность регистра - **В пределах секунды.** Представление записи - **Цена**, а Представление списка –**Ценынаноменклатуру**. Режим записи – **Независимый**.

*Нажмем «**Далее**»*

*- На закладку **Подсистемы***: отметим в списке следующие разделы: **Учет материалов, Оказание услуг и Бухгалтерия.**

*Нажмем «**Далее**»*

*- На закладке **Данные**:* а) создадим измерение **Номенклатура** с типом **СправочникСсылка.Номенклатура**. Укажем, что это измерение будет **Ведущим**.

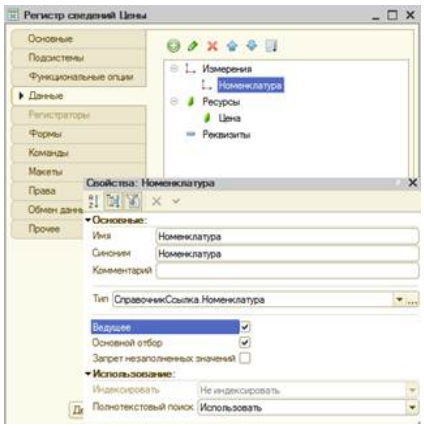


Рис. 9.1. Измерение Номенклатура будет Ведущим

Свойство Ведущее имеет смысл использовать лишь тогда, когда измерение имеет тип ссылки на объект базы данных. Установка свойства Ведущее будет говорить о том, что запись регистра сведений представляет интерес, пока существует тот объект, ссылка на который выбрана в качестве значения этого измерения в этой записи. При удалении объекта, все записи регистра сведений по этому объекту тоже будут автоматически удалены.

б) создадим ресурс **Цена**, тип **Число**, длина 15, точность 2, неотрицательное.

Для этого выделим ветвь **Ресурсы** к нажмем кнопку**Добавить** в командной панели окна:

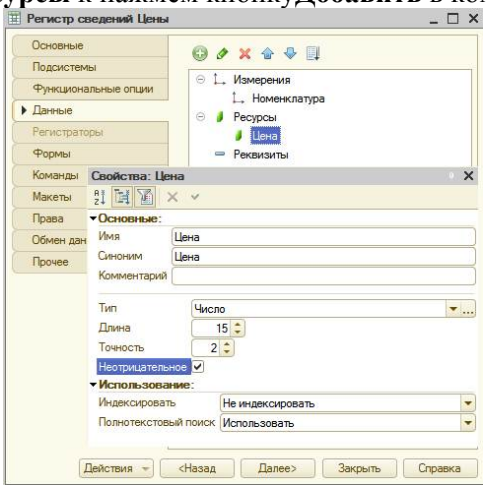


Рис. 9.2. Свойства Цена

В режиме 1С: Предприятие

Теперь запустим 1С: Предприятие в режиме отладки и посмотрим, как работает наш периодический регистр сведений **Цены**.

В открывшемся окне 1С: Предприятия мы видим, что в панели навигации разделов **Бухгалтерия, Оказание услуг и Учет материалов** появилась команда для открытия списка регистра **Цены на номенклатуру**.

Создание записей в регистре сведений

Чтобы добавить новую запись в регистр сведений, нажмем кнопку**Создать**.

Зададим стоимость услуг. При этом период зададим прошлым числом, так как он должен быть меньше или равен дате создания документа об оказании услуг.

Автоматическая подстановка цены в документ при выборе номенклатуры

Для этого нужно сделать две вещи:

- Написать некую функцию, которая будет возвращать актуальную цену номенклатуры.
- Вызвать эту функцию в тот момент, когда в документ добавляется номенклатура, и подставить в документ цену номенклатуры, которую вернет функция.

Необходимо данную функцию разместить в общедоступном, для других документов, месте.

В режиме Конфигуратор

Создадим функцию ***РозничнаяЦена(),****которая будет возвращать нам актуальную розничную цену номенклатуры, и поместим ее в общий модуль конфигурации:*

Откроем конфигуратор, в ветке **Общие >Общиемодули** добавим новый объект конфигурации **Модуль** и назовем его **РаботаСоСправочниками**.

Установим флажок Вызов сервера для того, чтобы экспортные процедуры и функции этого модуля можно было вызывать с клиента:

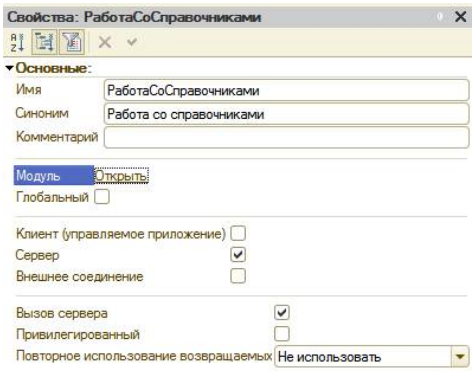


Рис. 9.3. Свойства РаботаСоСправочниками

Затем поместим в модуль следующий текст (см. Листинг 1):

```
Общий модуль РаботаСоСправочниками: Модуль
Функция РозничнаяЦена(АктуальнаяДата, ЭлементНоменклатуры) Экспорт

    // Создать вспомогательный объект Отбор
    Отбор = Новый Структура("Номенклатура", ЭлементНоменклатуры);

    // Получить актуальные значения ресурсов регистра
    ЗначенияРесурсов = РегистрыСведений.Цены.ПолучитьПоследнее(АктуальнаяДата, Отбор);
    Возврат ЗначенияРесурсов.Цена;

КонецФункции
```

Листинг 9.1. Редактируем Общий Модуль РаботаСоСправочниками

Для получения розничной цены мы будем передавать в функцию два параметра:

- **АктуальнаяДата** - параметр типа Дата, определяет точку на оси времени, в которой нас интересует значение розничной цены.
- **ЭлементНоменклатуры** - ссылка на элемент справочника Номенклатура, для которого мы хотим получить розничную цену.

Теперь эту функцию нужно вызвать в некоторый момент работы документа.

Вызов функции при выборе номенклатуры и заполнение цены в документе

1 - Найдем в конфигураторе документ **ОказаниеУслуги** и откроем его форму **ФормаДокумента**.

2 - Дважды щелкнем на элементе формы **ПереченьНоменклатурыНоменклатура**. Прокрутим список до конца, найдём событие **ПриИзменении**, которое возникает после изменения значения поля.

3 - Нажмем кнопку открытия со значком лупы в поле ввода. Система создаст шаблон процедуры обработчика этого события в модуле нашей формы и откроет закладку Модуль редактора формы. Внесем в него следующий текст (см. Листинг 2):

```
// Получить текущую строку табличной части
СтрокаТабличнойЧасти = Элементы.ПереченьНоменклатуры.ТекущиеДанные;

// Установить цену
СтрокаТабличнойЧасти.Цена = РаботаСоСправочниками.РозничнаяЦена(Объект.Дата, СтрокаТабличнойЧасти.Номенклатура);

// Пересчитать сумму строки
РаботаСДокументами.РассчитатьСумму(СтрокаТабличнойЧасти);
```

Листинг 9.2. Модуль редактора формы

В режиме 1С: Предприятие

Запустим 1С:Предприятие в режиме отладки и откроем регистр сведений **Цены**. Для элемента таблицы добавим другим числом новую цену.

Теперь откроем документ **Оказание услуги № 1**. Оставим дату документа без изменения и повторим выбор транзистора в колонке **Номенклатура** табличной части документа. Автоматически установится последнее значение цены товара.

Контрольные вопросы

- *ДлячегопредназначенобъектконфигурацииРегистр сведений.*
- *Какими особенностями обладаетобъект конфигурации Регистр сведений.*
- *В чем главные отличия регистра сведений от регистра накопления.*
- *Какиеполяопределяютключуникальности регистра накопления.*
- *Что такое периодический регистр сведений и что такое независимый регистр сведений.*
- *Как создать периодический регистр сведений.*
- *Что такое ведущее измерение регистра.*
- *Как получить значения ресурсов наиболее поздних записей регистра средствами встроенного языка.*

Перечисления.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Что такое перечисление

Объект конфигурации Перечисление предназначен для описания структуры хранения постоянных наборов значений, не изменяемых в процессе работы конфигурации. На основе объекта конфигурации Перечисление платформа создает в базе данных таблицу, в которой может храниться набор некоторых постоянных значений.

В реальной жизни этому объекту может соответствовать, например, перечисление вариантов указания цены («включая НДС», «без НДС»). Набор всех возможных значений, которые содержит перечисление, задается при конфигурировании системы, и пользователь не может изменять их, удалять или добавлять новые.

Особенностьперечисления*: значения перечисления не «обезличены» для конфигурации, на них могут опираться алгоритмы работы программы.*

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление перечисления

В режиме Конфигуратор

Откроем конфигуратор и создадим сначала новый объект конфигурации **Перечисление** с именем **ВидыНоменклатуры**.

- *На закладке **Данные***: нажмем на «+» и добавим два значения перечисления: **Материал** и **Услуга**.

Привязка номенклатуры к значениям перечисления «ВидНоменклатуры»

Чтобы привязать номенклатуру к значениям перечисления, мы сделаем следующее:

- В режиме Конфигуратор создадим у справочника **Номенклатура** реквизит, который будет хранить значение перечисления.
- В режиме 1С: Предприятие проставим нужные значения этого реквизита для всех элементов справочника **Номенклатура**.
- **В режиме Конфигуратор**

- *На закладке **Данные:*** добавим в справочник **Номенклатура** новый реквизит **ВидНоменклатуры** с типом **ПеречислениеСсылка.ВидыНоменклатуры**.

В режиме 1С:Предприятие

После этого запустим 1С: Предприятие в режиме отладки.

Платформа выдаст предупреждающее сообщение о том, что наше перечисление не включено ни в одну подсистему. Проигнорируем его и примем изменения конфигурации.

В режиме 1С: Предприятие зададим для каждого элемента справочника **Номенклатура** соответствующее значение реквизита **ВидНоменклатуры**:

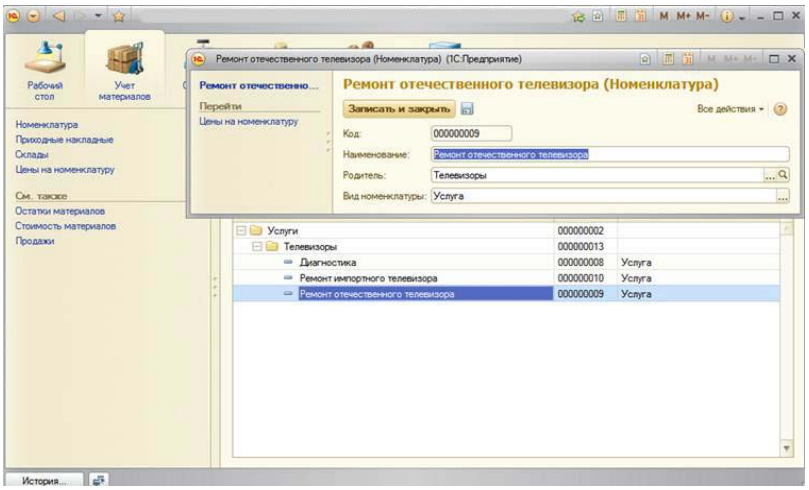


Рис. 10.1. Зададим значение реквизита ВидНоменклатуры

Регистрация расхода только той номенклатуры, которая является материалом

Сначала в режиме Конфигуратор изменим процедуру проведения документа так, чтобы в регистр попадали записи только о той номенклатуре, которая является материалом, а потом в режиме 1С:Предприятие заново проведем (перепроведем) все документы **Оказаниеуслуги**, чтобы данные в регистре изменились в соответствии с новым алгоритмом проведения документа.

Эта доработка будет не совсем эффективна с точки зрения производительности, зато позволит нам получить нужные данные в регистре **ОстаткиМатериалов**.

- **В режиме Конфигуратор**

Скорректируем движения документа, исключив из обработки те строки табличной части, в которых находятся услуги.

Для этого откроем в конфигураторе модуль документа **ОказаниеУслуги** (контекстное меню документа - Открыть модуль объекта) и добавим вобработчик события **ОбработкаПроведения** это условие.

Текст следует добавить в начало цикла обхода табличной части документа после строки**ДляКаждогоТекСтрокаПеречеьНоменклатурыИзПеречеьНоменклатурыЦикл.**

В результате процедура **ОбработкаПроведения** должна выглядеть следующим образом:

```
Процедура ОбработкаПроведения(Отказ, Режим)

    Движения.ОстаткиМатериалов.Записывать = Истина;
    Движения.СтоимостьМатериалов.Записывать = Истина;
    Движения.Продажи.Записывать = Истина;

    Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл
        Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры = Перечисления.ВидыНоменклатуры.Материал Тогда
            Движение = Движения.ОстаткиМатериалов.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
            Движение.Склад = Склад;
            Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;

            // регистр СтоимостьМатериалов Расход
            Движение = Движения.СтоимостьМатериалов.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
            Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Количество*ТекСтрокаПереченьНоменклатуры.Стоимость;
        КонецЕсли;

        // Регистр Продажи
        Движение = Движения.Продажи.Добавить();
        Движение.Период = Дата;
        Движение.Номенклатура = ТекСтрокаПереченьНоменклатуры.Номенклатура;
        Движение.Клиент = Клиент;
        Движение.Мастер = Мастер;
        Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;
        Движение.Выручка = ТекСтрокаПереченьНоменклатуры.Сумма;
        Движение.Стоимость = ТекСтрокаПереченьНоменклатуры.Стоимость * ТекСтрокаПереченьНоменклатуры.Количество;

    КонецЦикла;

КонецПроцедуры
```

Листинг 10.1. Скорректируем движения документа

Добавленный текст исключает выполнение операторов цикла для тех строк табличной части документа, в которых номенклатура не является материалом.

- **В режиме 1С:Предприятие**

Запустим 1С:Предприятие в режиме отладки и проверим работу процедуры проведения документа **Оказаниеуслуги**.

Откроем список документов, выполнив команду**Оказаниеуслуг** в панели навигации раздела **Оказаниеуслуг**.

Откроем документ **Оказание услуги № 1** и внесем в него изменения:

- удалим из табличной части строку какого-нибудь материала;
- добавим новую услугу;
- добавим материал для этой услуги.

Обратите внимание, что цены подставляются автоматически из регистра сведений **Цены**.

Нажмем кнопку**Провести** в командной панели формы документа.

Затем выполним команду **Остаткиматериалов** в панели навигации формы, чтобы перейти к записям регистра **Остаткиматериалов**, связанным с данным документом.

Как мы видим, в движения по регистру **Остаткиматериалов** включаются только строки, содержащие материалы. Запись про новую услугу в движения не попала.

Контрольные вопросы

- *Для чего предназначен объект конфигурации Перечисление.*
- *Как создать новое перечисление.*
- *Как с помощью перечисления задать принадлежность элементов справочника к той или иной смысловой группе.*
- *Как обратиться к значению перечисления средствами встроенного языка.*

Практическаяработа № 6. Проведение документа по нескольким регистрам.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Зачем нужно проведение документа по нескольким регистрам

В нашей базе учитывается только количественное движение материалов, очевидно, что необходимо также знать, какие денежные средства были затрачены на приобретение тех или иных материалов и каковы материальные запасы в денежном выражении.

На ряду с этим необходимо, чтобы весь суммовой учет материалов велся по средней стоимости.

То есть при закупке материалов они должны учитываться в ценах приобретения, а при расходе - по средней стоимости, которая рассчитывается исходя из общей суммы закупок данного материала и общего количества этого материала.

Поскольку подобная информация имеет совершенно другую структуру, нежели количественный учет, для хранения данных об общей стоимости тех или иных материалов будет использоваться еще один регистр накопления **СтоимостьМатериалов**.

Таким образом, документы **ПриходнаяНакладная** и **ОказаниеУслуги** должны будут создавать движения не только в регистре **ОстаткиМатериалов**, но одновременно и в регистре **СтоимостьМатериалов**, отражая изменения суммового учета.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление еще одного регистра накопления

- **В режиме Конфигуратор**

Создадим новый объект конфигурации Регистр накопления с именем **СтоимостьМатериалов**.

- *На закладке **Основные**:* зададимРасширенное представление списка - **Движения по регистру Стоимость материалов.**

- *На закладке **Подсистемы***: отметим, что этот регистр будет отображаться в подсистемах **Бухгалтерия**, **Учетматериалов** и **Оказаниеуслуг**.

- *На закладке **Данные***: создадим для регистра одно измерение -**Материал** с типом **СправочникСсылка.Номенклатура** и один ресурс -**Стоимость** с длиной 15 и точностью 2.

- *Отредактируем командный интерфейс*:

В дереве объектов конфигурации выделим ветвь **Подсистемы**, вызовем ее контекстное меню и выберем пункт**Всеподсистемы**. В открывшемся окне слева в списке Подсистемы выделим подсистему **Бухгалтерия**.

В группе **Панель навигации.Обычное** включим видимость у команды **Стоимость материалов** и мышью перетащим ее в группу **Панель навигации.См.также**.

Аналогично, выделив подсистемы **ОказаниеУслуг** и **УчетМатериалов,** в панели навигации в группе Обычное включим видимость у команды **Стоимость материалов** и перенесем ее в группу **См.также**.

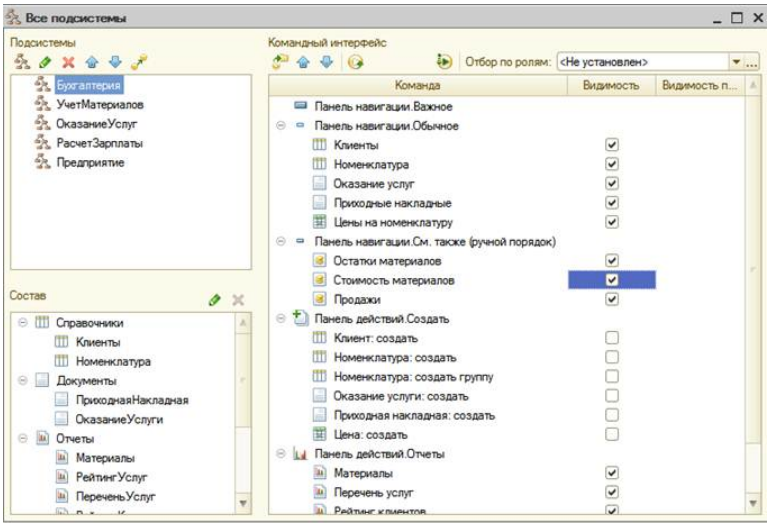


Рис. 11.1. Проведение приходной накладной по двум регистрам

- **В режиме Конфигуратор**

Откроем в конфигураторе окно редактирования объекта конфигурации Документ **ПриходнаяНакладная.**

- *На закладке **Движения***: В списке регистров отметим **СтоимостьМатериалов**.

- *На закладке **Прочее**:* нажмем кнопку **Модульобъекта**, откроем процедуру обработчика события **ОбработкаПроведения**. И изменим код следующим образом: (см.листинг):

Листинг 11.1

```
Процедура ОбработкаПроведения(Отказ, Режим)
Движения.ОстаткиМатериалов.Записывать = Истина;
Движения.СтоимостьМатериалов.Записывать = Истина;
 Для Каждого ТекСтрокаМатериалы Из Материалы Цикл
 //регистр ОстаткиМатериалов Приход
 Движение = Движения.ОстаткиМатериалов.Добавить();
Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаМатериалы.Материал;
Движение.Склад=Склад;
Движение.Количество = ТекСтрокаМатериалы.Количество:
 //регистр СтоимостьМатериалов Приход
 Движение = Движения.СтоимостьМатериалов.Добавить();
Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаМатериалы.Материал;
Движение.Стоимость=ТекСтрокаМатериалы.Сумма;
КонецЦикла;
КонецПроцедуры
```

Команда перехода к записям регистра

В заключение отредактируем командный интерфейс формы документа, чтобы в панели навигации формы иметь возможность переходить к списку записей регистра **СтоимостьМатериалов**, связанному с документом.

- *Откроем форму документа **ПриходнаяНакладная**:* перейдем на закладку **Командныйинтерфейс**. В разделе **Панельнавигации** раскроем группу**Перейти** и увидим команду для открытия регистра накопления **Стоимостьматериалов**. Установим свойство **Видимость** для этой команды:

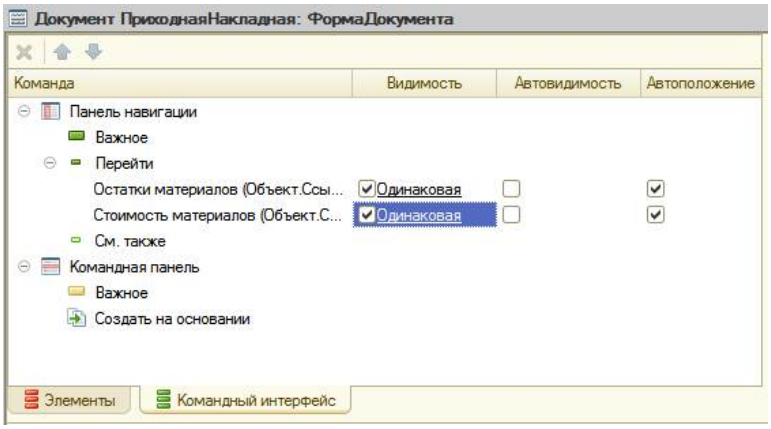


Рис. 11.2. Редактируем командный интерфейс формы документа

В режиме 1С:Предприятие

В режиме 1С:Предприятие необходимо перепровести все приходные накладные. Это необходимо для того, чтобы эти документы создали новые записи в регистрах, в соответствии с алгоритмом проведения.

Запустим 1С: Предприятие в режиме отладки. Откроем список документов, выполнив команду**Приходныенакладные** в панели навигации раздела **Учетматериалов**.

*Выделим одновременно, используя клавишу Ctrl,все приходные накладные и перепроведем их, выполнив команду**Вседействия**>**Провести**.*

Проведение документа «Оказание услуги» по двум регистрам

В заключение внесем изменения в процедуру обработки проведения документа **ОказаниеУслуги**.

Это необходимо для того, чтобы на первом этапе, при списании материалов, израсходованных в процессе оказания услуги, должна быть возможность указывать различную стоимость для одного и того же материала, которая рассчитана руководством исходя из текущих конъюнктурных соображений.

Поскольку в документе **ОказаниеУслуги** отражена только цена номенклатуры, необходимо сделать следующее:

- Добавить в табличную часть документа еще один реквизит, в котором будет указываться стоимость номенклатуры.
- После этого изменить процедуру проведения документа **ОказаниеУслуги**.
- В режиме 1С: Предприятие перепровести все эти документы.
- **В режиме Конфигуратор**

Откроем в конфигураторе окно редактирования объекта конфигурации Документ **ОказаниеУслуги**

- *На закладке **Данные**:* создадим новый реквизит табличной части документа с именем **Стоимость**, типом **Число**, длиной 15 и точностью 2, и отметим флажок **неотрицательное**:

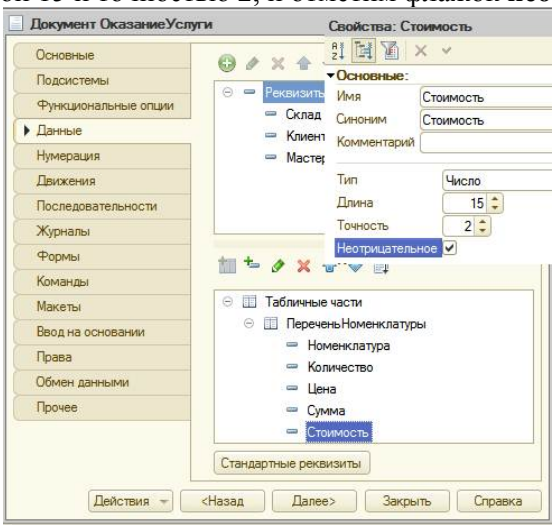


Рис. 11.3. Редактируем свойства Стоимость

Откроем форму **ФормаДокумента** документа **ОказаниеУслуг** и добавим табличную часть **ПеречеьНоменклатуры** поле, отображающее новый реквизит **Стоимость**:

- *На закладке Реквизиты:* раскроем реквизит формы **Объект**.

Найдем в табличной части реквизит **Стоимость** и с помощью мыши перетащим его в окно элементов формы, расположенное слева в верхней части редактора форм:

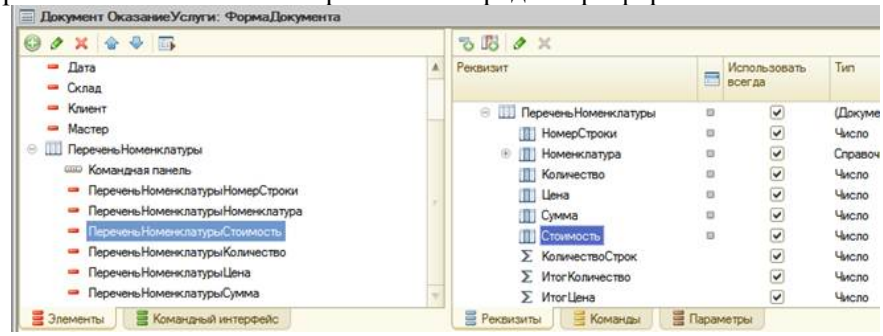


Рис. 11.4. Редактируем реквизит формы Объект

Новый элемент расположим в структуре элементов формы после поля Номенклатура. Оставим свойства элемента формы, предложенные по умолчанию.

Новый реквизит сразу же отобразится в форме документа, расположенной в левом нижнем окне редактора форм.

Изменение процедуры проведения

Создадим движения документа **ОказаниеУслуги** таким же образом, как мы делали это для документа **ПриходнаяНакладная**.

- *На закладке **Движения**: в* списке регистров отметим, что документ будет создавать теперь движения и по регистру **СтоимостьМатериалов**.

- *На закладке **Прочее**:* нажмем кнопку **Модульобъекта**. Откроем процедуру обработчика события **ОбработкаПроведения**. Отредактируем код следующим образом:

Листинг 11.2

```
Процедура ОбработкаПроведения(Отказ, Режим)
Движения.ОстаткиМатериалов.Записывать = Истина;
Движения.СтоимостьМатериалов.Записывать = Истина;
Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл
Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры =
Перечисления.ВидыНоменклатуры.Материал Тогда
//регистр ОстаткиМатериалов Расход
Движение = Движения.ОстаткиМатериалов.Добавить();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Склад=Склад;
Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;
//регистр СтоимостьМатериалов Расход
Движение = Движения.СтоимостьМатериалов.Добавить();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Стоимость=ТекСтрокаПереченьНоменклатуры.Количество
*ТекСтрокаПереченьНоменклатуры.Стоимость;
КонецЕсли;
КонецЦикла;
КонецПроцедуры
```

В заключение отредактируем командный интерфейс формы документа, чтобы в панели навигации формы иметь возможность переходить к списку записей регистра **СтоимостьМатериалов**, связанному с документом.

- *Откроем форму документа **ОказаниеУслуги**:* перейдём на закладку **Командныйинтерфейс**. В разделе **Панельнавигации** раскроем группу**Перейти** и увидим команду для открытия регистра накопления **Стоимостьматериалов**. *Установим свойство **Видимость**для этой команды.*

- **В режиме 1С: Предприятие**

В режиме 1С:Предприятие необходимо перепровести документ оказания услуги. Это необходимо для того, чтобы этот документ создал новые записи в регистрах. Запустим 1С: Предприятие в режиме отладки и откроем список документов, выполнив команду**Оказание услуг** в панели навигации раздела **Оказание услуг.**

Откроем документ **Оказание услуги № 1** и изменим в нем стоимость одного из материалов. Нажмем кнопку**Провести** и выполним команду перехода к регистру Стоимость материалов.

Создадим и проведем еще два документа **Оказаниеуслуги**.

Для этого в форме списка документов нажмем кнопку Создать. Эти документы понадобятся нам в дальнейшем.

Контрольные вопросы

- *Для чего может понадобиться проведение документа по нескольким регистрам.*
- *Как создать движения документа по нескольким регистрам в обработчике проведения документа.*
- *Как создать движения документа без использования конструктора движений.*
- *Как средствами встроенного языка сформировать и записать движения документа в*

регистр накопления.

- *Как добавить в форму документа новый реквизит.*

Оборотные регистры накопления.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Что такое оборотный регистр накопления

Регистры накопления могут быть регистрами остатков и регистрами оборотов.

Существующие в нашей конфигурации регистры **ОстаткиМатериалов** и **СтоимостьМатериалов** являются регистрами остатков.

Если вы помните, при создании отчета *Материалы* в конструкторе запроса мы видели, что для таких регистров система создает три виртуальные таблицы: таблица остатков, оборотов и совокупная таблица остатков и оборотов.

Оборотный регистр накапливает только обороты, остатки ему безразличны. Поэтому единственной виртуальной таблицей, которую будет создавать система для такого регистра, будет таблица оборотов. В остальном оборотный регистр ничем не отличается от регистра остатков.

Следует сказать об одной особенности конструирования регистров накопления, напрямую связанной с возможностью получения остатков. При создании оборотного регистра накопления нет особой сложности в определении того, какие именно данные должны являться измерениями регистра - мы можем назначить в качестве его измерений любые нужные нам данные.

Совсем иная ситуация в случае регистра накопления, поддерживающего накопление остатков. Для него выбор измерений должен выполняться исходя из того, что движения регистра могут быть осуществлены в две стороны: приход и расход. Таким образом, в качестве измерений нужно выбирать те данные, по которым движения точно будут осуществляться как в одну, так и в другую сторону.

Скорее всего, при поступлении материалов поставщик будет указан, а вот при расходовании материалов, с большой долей вероятности, поставщик указываться не будет. В большинстве случаев это совершенно лишняя информация.

Значит, поставщика следует добавить не как измерение, а как реквизит регистра накопления.

Если же при расходе материалов поставщик будет указываться наверняка, имеет смысл добавить поставщика в измерения регистра.

Иными словами, по каждому из измерений регистра накопления остатков ресурсы обязательно должны изменяться в обе стороны: приход и расход. Не должно существовать таких измерений, по которым осуществляется только приход или только расход.

Нарушение этого принципа построения регистров накопления будет вести к непроизводительному использованию ресурсов системы и как следствие к замедлению работы и падению производительности.

Для реквизитов же регистра этот принцип не важен. По реквизитам регистра ресурсы могут только приходоваться или только расходоваться.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление оборотного регистра накопления

- **В режиме Конфигуратор**

Откроем конфигуратор и создадим новый объект конфигурации Регистр накопления.

*- На закладке **Основные:*** имя **-** Продажи; вид регистра – **Обороты**; расширенное представление списка – **Движения документа по регистру Продажи.**

*- На закладке **Подсистемы:*** отметим, что этот регистр будет отображаться в подсистемах **Бухгалтерия**, **Учетматериалов** и **Оказаниеуслуг**.

*- На закладке **Данные:**создадим измерения регистра:*

- **Номенклатура**, тип **СправочникСсылка.Номенклатура**;
- **Клиент**, тип **СправочникСсылка.Клиенты**;
- **Мастер**, тип **СправочникСсылка.Сотрудники**.

Создадим три ресурса

- **Количество**, тип Число, длина 15, точность 3;
- **Выручка**, тип Число, длина 15, точность 2;
- **Стоимость**, тип Число, длина 15, точность 2.

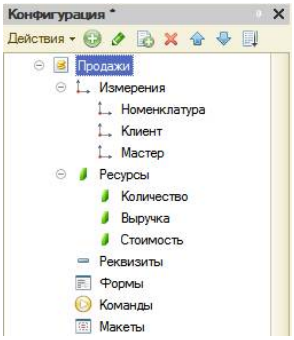


Рис. 12.1. Создание регистров

*- В дереве объектов конфигурации выделим ветвь **Подсистемы***, вызовем**Всеподсистемы**: В открывшемся окне выделим подсистему **Бухгалтерия; в** группе **Панель навигации.Обычное** включим видимость у команды **Продажи** и мышью перетащим ее в группу **Панель навигации.См.также**.

Аналогично, выделив подсистемы **ОказаниеУслуг** и **УчетМатериалов**, в панели навигации включим видимость у команды **Продажи** и перенесем ее в группу **См.также**.

Проведение документа «Оказание услуги» по трем регистрам

Изменим процедуру проведения документа **ОказаниеУслуги**, а затем в режиме **1С:Предприятие** перепроведем все эти документы.

В режиме Конфигуратор

Откроем окно редактирования объекта конфигурации Документ **ОказаниеУслуги** .

*- На закладке **Движения***: укажем, что этот документ будет создавать движения еще и по регистру **Продажи**.

*- На закладке **Прочее:***нажмем кнопку **Модульобъекта**, затем откроем процедуру обработчика события **ОбработкаПроведения** и изменим код следующим образом (см. Листинг 12.1):

Листинг 12.1

```
Процедура ОбработкаПроведения(Отказ, Режим)
Движения.ОстаткиМатериалов.Записывать = Истина;
Движения.СтоимостьМатериалов.Записывать = Истина;
Движения.Продажи.Записывать = Истина;
 Для Каждого ТекСтрокаПереченьНоменклатуры Из ПереченьНоменклатуры Цикл
Если ТекСтрокаПереченьНоменклатуры.Номенклатура.ВидНоменклатуры = Перечисления.ВидыНоменклатуры.Материал Тогда
 //регистр ОстаткиМатериалов Расход
 Движение = Движения.ОстаткиМатериалов.Добавить();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Склад=Склад;
Движение.Количество = ТекСтрокаПереченьНоменклатуры.Количество;
 //регистр СтоимостьМатериалов Расход
 Движение = Движения.СтоимостьМатериалов.Добавить();
Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
Движение.Период = Дата;
Движение.Материал = ТекСтрокаПереченьНоменклатуры.Номенклатура;
Движение.Стоимость=ТекСтрокаПереченьНоменклатуры.Количество
*ТекСтрокаПереченьНоменклатуры.Стоимость;
КонецЕсли;
 //РегистрПродажи
 Движение = Движения.Продажи.Добавить();
Движение.Период = Дата;
Движение.Номенклатура = ТекСтрокаПереченьНоменклатуры.Номенклатура;
```

Движение.Клиент = Клиент;
Движение.Мастер – Мастер;
Движение.Количество = ТекСтрокаПеречеьНоменклатуры.Количество;
Движение.Выручка = ТекСтрокаПеречеьНоменклатуры.Сумма;
Движение.Стоимость = ТекСтрокаПеречеьНоменклатуры.Стоимость
**ТекСтрокаПеречеьНоменклатуры.Количество;*
КонецЦикла;
КонецПроцедуры

Отредактируем командный интерфейс формы документа, чтобы в панели навигации формы иметь возможность переходить к списку записей регистра **Продажи**, связанному с документом:

- *Откроем форму документа **ОказаниеУслуги**:*перейдём на закладку **Командныйинтерфейс**, раскроем группу**Перейти** и увидим команду для открытия регистра накопления **Продажи**. *Установим свойство Видимость для этой команды.*

- **В режиме 1С: Предприятие**

В режиме 1С: Предприятие перепроведем все документы для этого запустим 1С: Предприятие в режиме отладки и откроем по очереди каждый документ **Оказаниеуслуги**.

Контрольные вопросы

- *Что такое оборотный регистр накопления.*
- *В чем отличие между регистром накопления остатков и оборотным регистром накопления.*
- *Как выбирать реквизиты и измерения при создании регистров накопления.*
- *Как создать оборотный регистр накопления*

Практическая работа № 7. Отчеты. Выбор данных из одной таблицы

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Способы доступа к данным

Система 1С:Предприятие 8 поддерживает два способа доступа к данным, хранящимся в базе данных:

- объектный (для чтения и записи),
- табличный (для чтения).

Объектный способ доступа к данным реализован посредством использования объектов встроенного языка.

Важной особенностью объектного способа доступа к данным является то, что, обращаясь к какому-либо объекту встроенного языка, мы обращаемся к некоторой совокупности данных, находящихся в базе данных, как к единому целому.

Объектная техника обеспечивает сохранение целостности объектов, кеширование объектов, вызов соответствующих обработчиков событий и т. д.

Табличный доступ к данным в 1С: Предприятии 8 реализован с помощью запросов к базе данных, которые составляются на языке запросов.

В этой технике разработчик получает возможность оперировать отдельными полями таблиц базы данных, в которых хранятся те или иные данные.

Табличная техника предназначена для получения информации из базы данных по некоторым условиям (отбор, группировка, сортировка, объединение нескольких выборок, расчет итогов и т.д.). Табличная техника оптимизирована для обработки больших объемов информации, расположенной в базе данных, и получения данных, отвечающих заданным критериям.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Выбор данных из одной таблицы

Создадим отчет *Реестр документовоказание услуги*, используя систему компоновки данных. Этот отчет будет выводить список существующих в базе данных документов ОказаниеУслуги в порядке их дат и номеров.

- **В режиме Конфигуратор**

Добавим в конфигураторе объект конфигурации Отчет.

На закладке Основные: имя отчёта – РеестрДокументовОказаниеУслуги; расширенное представление - Список оказанных услуг для представления отчета в интерфейсе программы.

Создадим схему компоновки данных для отчета, нажмем кнопку Открыть схему компоновки данных. В открывшемся диалоговом окне конструктора макета нажмем Готово. В конструкторе схемы компоновки данных создадим Набор данных – запрос.

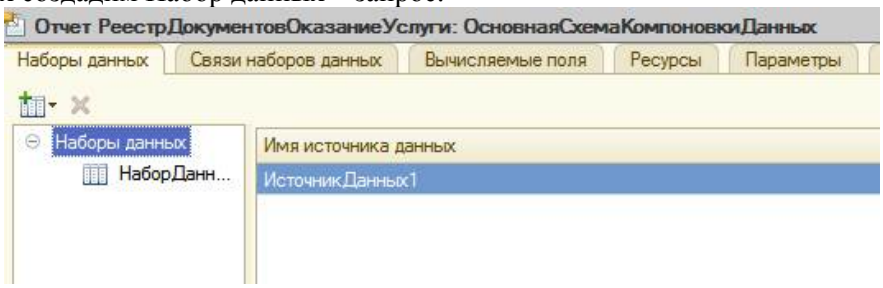


Рис. 13.1. Проведение приходной накладной по двум регистрам

После чего нажмем кнопку **Конструктор запроса**, в качестве источника данных для запроса выберем объектную таблицу документа ОказаниеУслуги. Из этой таблицы выберем следующие поля:

- Склад,
- Мастер,
- Клиент,
- Ссылка.

На закладке *Объединения/Псевдонимы* укажем, что поле *Ссылка* будет иметь псевдоним *Документ*. На закладке *Порядок* укажем, что результат запроса должен быть упорядочен по значению поля *Документ*.

На закладке *Настройки* создадим стандартные настройки, определяющие, как будет выводиться информация в отчет.

Иерархическая структура отчета может содержать в различных сочетаниях три основных элемента:

- Группировка - для вывода информации в виде обычного линейного отчета.
- Таблица - для вывода информации в виде таблицы.
- Диаграмма - для вывода информации в виде диаграммы.

Для добавления нового элемента, в нашем случае группировки, выделим в дереве структуры отчета корневой элемент Отчет и вызовем его контекстное меню. В окне выбора поля группировки просто нажмем ОК. В структуре отчета появится группировка Детальные записи. На закладке Выбранные поля перенесем мышью из списка доступных полей те поля, которые будут выводиться в отчет:

- Документ,
- Склад,
- Мастер,
- Клиент.

В результате окно настроек отчета должно иметь вид. На этом создание отчета закончено.

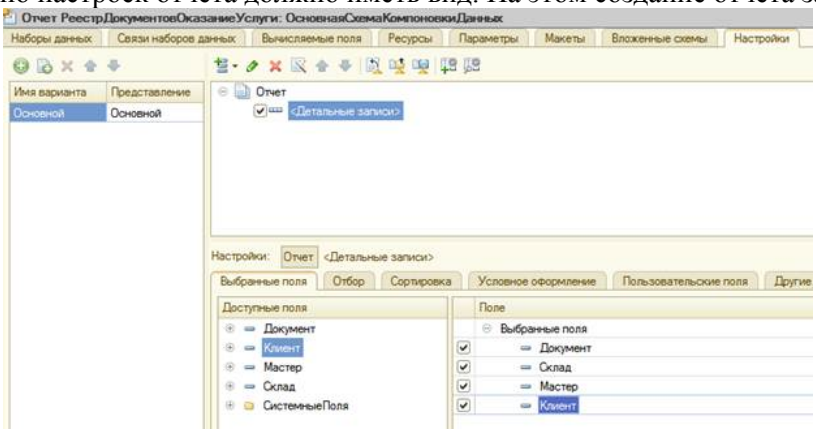


Рис. 13.2. Создание отчета

- В окне редактирования объекта конфигурации Отчет РеестрДокументовОказаниеУслуги

перейдем на закладку Подсистемы.

Отметим в списке подсистему Оказание услуг.

- **В режиме 1С: Предприятие**

Запустим 1С: Предприятие в режиме отладки.

В открывшемся окне 1С: Предприятия можно наблюдать, что в панели действий раздела Оказание услуг в группе команд для выполнения отчетов появилась команда для формирования отчета Реестр документов оказание услуги.

Причем если подвести к ней мышь, то появится всплывающая подсказка Список оказанных услуг, которая определяется свойством Расширенное представление.

- Выполним команду Реестр документов оказание услуги.

Перед нами откроется форма отчета, автоматически сформированная системой.

- Нажмем кнопку Сформировать.

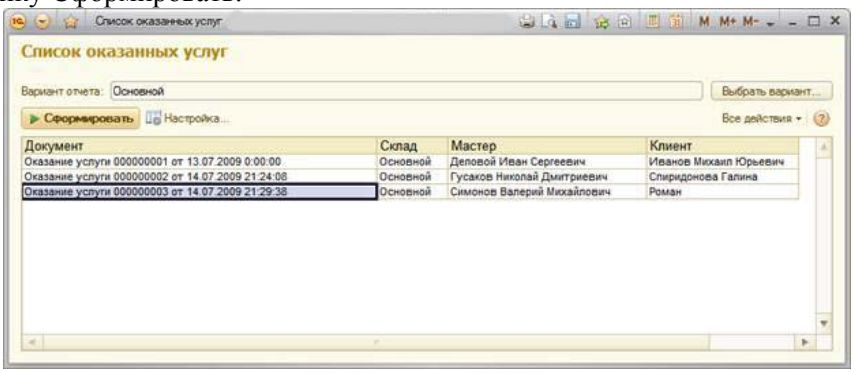


Рис. 13.3. Форма отчета, автоматически сформированная системой

Перед вам появится отчет содержащий реестр документов Оказание услуги.

Двойным щелчком мыши на поле Документ можно открыть исходный документ, а также выполнить другие действия «расшифровки», которые предоставляет система компоновки данных.

Контрольные вопросы

- *Для чего предназначен объект встроенного языка Запрос.*
- *Для чего предназначена система компоновки данных.*
- *Для чего предназначена схема компоновки данных.*
- *Для чего предназначены настройки компоновки данных.*
- *В чем отличие между реальными и виртуальными таблицами.*

Отчеты. Выбор данных из двух таблиц

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Работа с запросами

Для работы с запросами используется объект встроенного языка Запрос. Он позволяет получать информацию, хранящуюся в полях базы данных, в виде выборки, сформированной по заданным правилам.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Выбор данных из двух таблиц

Создадим Отчет - Рейтинг услуг, который будет содержать информацию о том, выполнение каких услуг принесло компании наибольшую прибыль в указанном промежутке времени.

- **В режиме Конфигуратор**
- Добавим в конфигураторе объект конфигурации Отчет.
- На закладке *Основные*: имя отчета – РейтингУслуг. Создадим схему компоновки данных для отчета, нажмем кнопку Открыть схему компоновки данных. В открывшемся диалоговом окне конструктора макета нажмем Готово. В конструкторе схемы компоновки данных создадим Набор данных – запрос. После чего нажмем кнопку Конструктор запроса, в качестве источника данных для запроса выберем объектную таблицу Номенклатура и виртуальную таблицу регистра накопления Продажи.Обороты.
- Чтобы исключить неоднозначность имен в запросе, необходимо переименовать таблицу

Номенклатура в *спрНоменклатура*: выделим *Номенклатуру* в списке *Таблицы*, щелкнем правой кнопкой мыши - *Переименовать таблицу*.

- На закладке *Связи*: снимем флажок Все у таблицы ПродажиОбороты и установить его у таблицы спрНоменклатура.

- На закладке *Условия*: раскроем таблицу спрНоменклатура, перетащим мышью поле ЭтоГруппа в список условий, установим флажок Произвольное и напишем в поле Условие следующий код:

Листинг 13.1. Условие запроса:

СпрНоменклатураЭтоГруппа=ЛОЖЬ

- На закладке *Объединения/Псевдонимы*: укажем псевдоним представления элемента справочника как Услуга, а псевдоним поля регистра как Выручка.

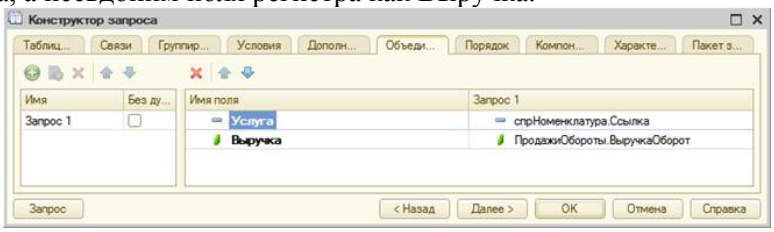


Рис. 13.4. Установка псевдонимов полей запроса

- На закладке *Порядок*: укажем сортировку значение поля Выручка по убыванию:

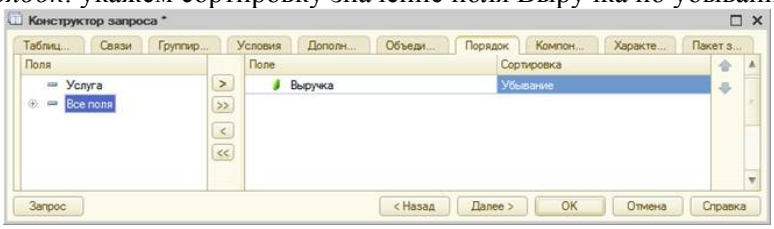


Рис. 13.5. Порядок записей запроса

После чего закончим создание запроса нажатием кнопки ОК.

Вернемся вконструктор схемы компоновки данных.

Ресурсы

Под ресурсами в системе компоновки данных подразумеваются поля, значения которых рассчитываются на основании детальных записей, входящих в группировку. По сути, ресурсы являются групповыми или общими итогами отчета.

- На закладке *Ресурсы*: нажмем кнопку », чтобы конструктор выбрал все доступные ресурсы, по которым можно вычислять итоги. В нашем случае это единственный ресурс Выручка. Платформа автоматически предложит рассчитывать сумму значений этого поля.

Параметры

Пользователя, как правило, интересуют данные о хозяйственной деятельности за определенный период. Поэтому практически в любом отчете используются параметры, задающие начало и конец отчетного периода.

- На закладке *Параметры*: на этой закладке увидим три параметра: НачалоПериода, КонецПериода и ВидНоменклатуры. Вы можете спросить: почему параметра три, хотя в запросе мы задавали всего один – ВидНоменклатуры?

- Дважды щелкнем в ячейке Тип, соответствующей параметру НачалоПериода. Затем нажмем кнопку выбора установим Состав даты в значение Дата.

- Нажмем *ОК*–

- Для параметра КонецПериода установим флажок Ограничение доступности. В ячейке Выражение напишем следующий текст:

КонецПериода(&ДатаОкончания,"День")

- Добавим новый параметр с именем ДатаОкончания. Зададим тип значения параметра - Дата, укажем состав даты - Дата.
- в ячейке Значение, соответствующей параметру ВидНоменклатуры. Воспользуемся кнопкой выбора и выберем значение Перечисление.ВидыНоменклатуры.Услуга.

Настройки

Перейдем к формированию структуры отчета.

- На закладке *Настройки:* добавим группировку и не укажем поле группировки. На закладке Выбранные поля укажем поля Услуга и Выручка.

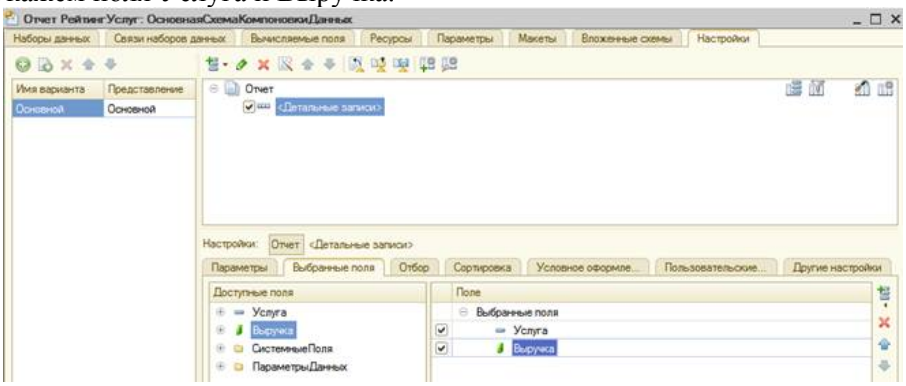


Рис. 13.6. Структура отчёта "РейтингУслуг"

- Затем перейдём на подзакладку **Другие настройки** и зададим заголовок отчета - Рейтинг услуг:

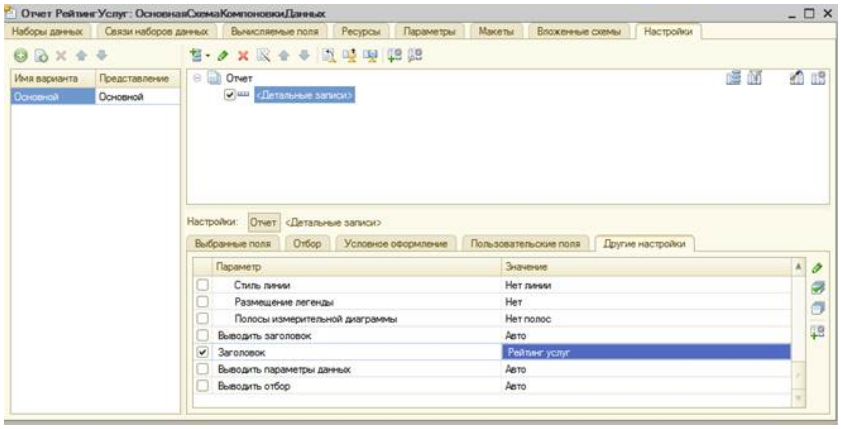


Рис. 13.7. Установка заголовка отчета

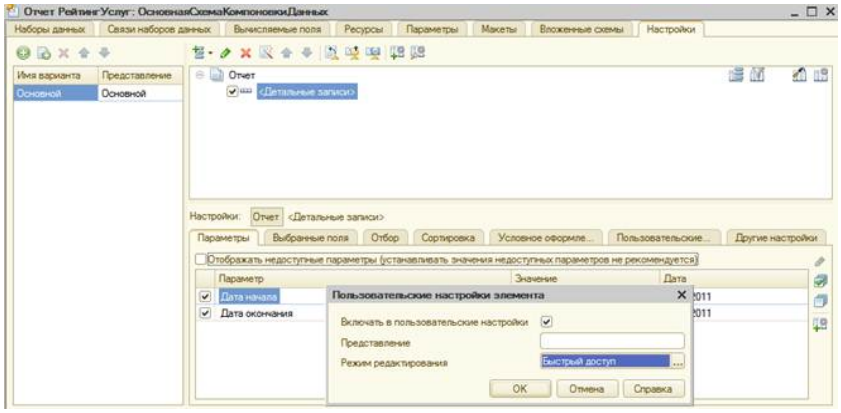


Рис. 13.8. Определение пользовательских настроек

- На подзакладке Параметры выделим по очереди каждый из параметров и нажмем кнопку Свойства элемента пользовательских настроек, расположенную в правом нижнем углу окна настроек. Установим флажок Включать в пользовательские настройки и оставим предложенное по умолчанию для свойства Режим редактирования значение Быстрый доступ.

- Закроем конструктор схемы компоновки данных и в окне редактирования объекта конфигурации ОтчётРейтингУслугперейдём на закладку Подсистемы. Отметим в списке подсистем: Оказание услуг и Бухгалтерия.

В режиме 1С:Предприятие

Запустим 1С: Предприятие в режиме отладки и посмотрим, как работает отчет.

В открывшемся окне 1С:Предприятия видим, что в панели действий разделов Оказание услуг и Бухгалтерия в группе команд для выполнения отчетов появилась команда для формирования отчета Рейтинг услуг (рис. 13.40).

Выполним эту команду. Откроется форма отчета, автоматически сформированная системой. В окне отчета мы видим параметры, определяющие отчетный период. Он по умолчанию задан - с начала месяца по сегодняшнее число. Но можно при желании изменить его, воспользовавшись кнопкой календаря. Нажмем кнопку Сформировать:

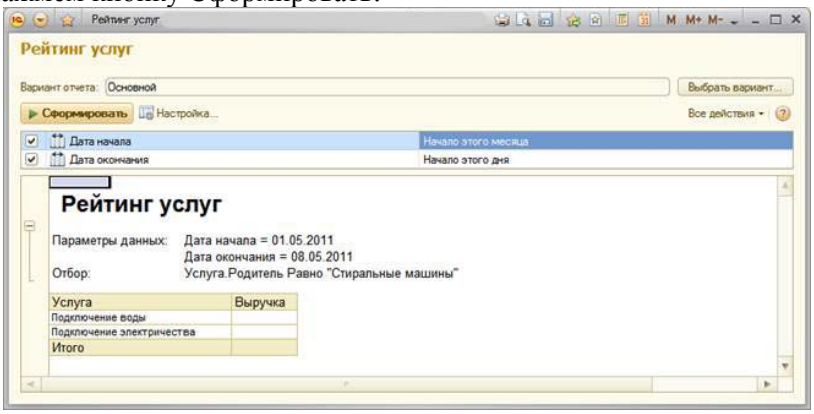


Рис. 13.9. Быстрые и обычные пользовательские настройки

Настройки в конфигураторе и в режиме 1С:Предприятие

На примере предыдущего отчета изучим другие настройки отчета - Условное оформление и Отбор.

Настройки можно задать как в режиме Конфигуратор, так и в режиме 1С:Предприятие по команде**Все действия > Изменить вариант**. Разница в том, что те настройки, которые будут изменяться в конфигураторе, называются стандартными настройками и будут сохранены в самой схеме компоновки данных, то есть любой пользователь конфигурации будет видеть отчет именно в таком виде, как вы его настроите в конфигураторе. А в режиме 1С:Предприятие, настройка уже не будет являться частью конфигурации и будет доступна только одному конкретному пользователю конкретной информационной базы.

Условное оформление

В таком отчете, как Рейтинг услуг, было бы удобно выделять цветом записи отчета, содержащие услуги с наименьшей или с наибольшей выручкой, или еще по какому-либо условию.

- **В режиме Конфигуратор**

Вернемся в конфигуратор и откроем схему компоновки данных отчета *Рейтинг услуг* на закладку *Настройки*: перейдем на закладку Условное оформление и нажмем кнопку Добавить. В пункте Оформление, нажмем кнопку выбора, после чего в поле установим нужный цвет текста.

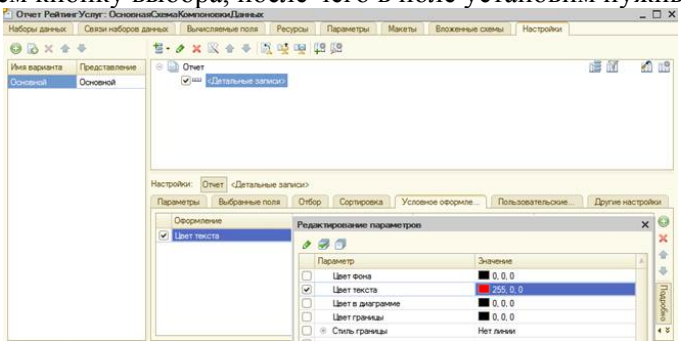


Рис. 13.10. Настройка условного оформления

- Нажмём ОК -

Затем укажем *Условие*: нажмем кнопку выбора в поле Условие и в появившемся окне нажмем кнопку Добавить и укажем в графе *Левоезначениеполе* - Выручка, в графе *Видсравнения* - Меньше, а в графе *Правоезначение* - 700.

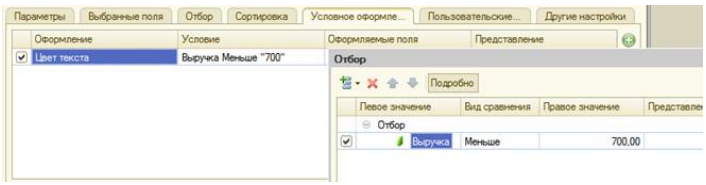


Рис. 13.11. Настройка условного оформления

- Нажмём ОК -

В поле *Оформляемыеполя*: и в появившемся окне, нажимая кнопку, Добавить, можно выбрать поля Услуга и Выручка.

- Нажмём ОК -

В поле *Представление*: установим как Непопулярная услуга (это то, что увидит пользователь в своих настройках).

Нажмем кнопку *Свойства* элемента пользовательских настроек, расположенную в правом нижнем углу окна настроек (см. рис. 13.12). Установим флажок Включать в пользовательские настройки и установим свойство *Режимредактирования* - Обычный.

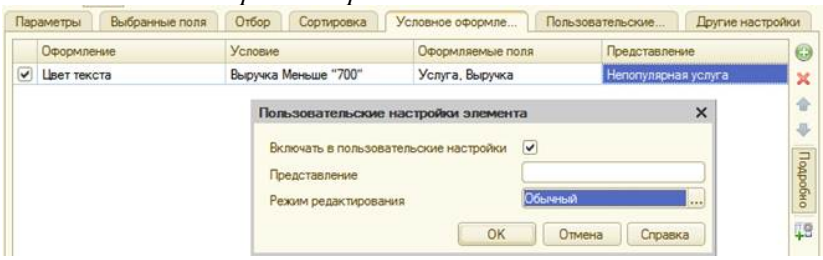


Рис. 13.12. Настройка условного оформления

В режиме 1С:Предприятие

- Вызовем отчет;
- Зададим Дату окончания отчетного периода как Начало этого дня и нажмем кнопку Сформировать;
- Нажмем кнопку Настройка - перед вами появится окно пользовательских настроек отчета, содержащее параметры отчетного периода и настройку условного оформления Непопулярная услуга. Можно снять флажок использования этой настройки, нажать кнопку Завершить редактирование и снова выполнить отчет.

Пользовательские настройки

В режиме Конфигуратор

На закладке Настройки схемы компоновки данных, отчета Рейтинг услуг, содержатся полные настройки отчета, которые задает разработчик. Часть из них может быть представлена пользователю для создания произвольного отбора, условного оформления отчета и пр.

- Нажмем кнопку *Свойства* элемента пользовательских настроек, расположенную вверху в командной панели окна настроек. Установим признак использования для настроек Отбор и Условное оформление и установим для них свойство Режим редактирования в значение Обычный.

Тем самым включились настройки отбора и условного оформления в состав пользовательских настроек и предоставили пользователю возможность задавать их в отдельном окне, вызываемом кнопкой Настройка.

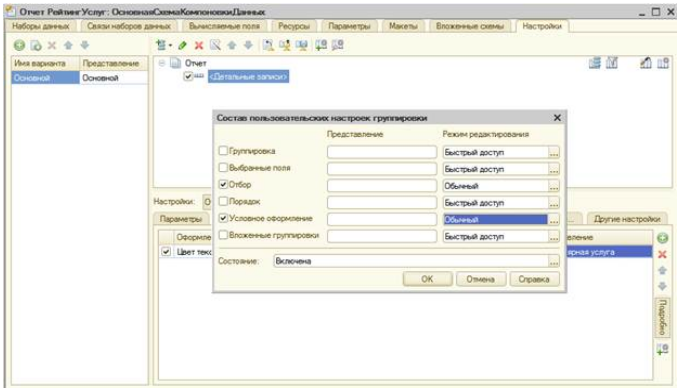


Рис. 13.13. Состав пользовательских настроек

Контрольные вопросы

- Из каких частей состоит текст запроса, какие из них являются обязательными.
- Каковы основные синтаксические конструкции языка запросов.
- Что является источником данных запроса.
- Что такое псевдонимы в языке запросов.
- Что такое параметры запроса.

Практическаяработа № 8. Отчеты. Вывод данных по дням в выбранном периоде

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Источники данных запросов

Исходную информацию запрос получает из набора таблиц. Эти таблицы представляют разработчику данные реальных таблиц базы данных в удобном для анализа виде.

Все таблицы, которыми оперирует язык запросов, можно разделить на две большие группы: реальные таблицы и виртуальные таблицы.

Отличительной особенностью реальных таблиц является то, что они содержат данные какой-либо одной реальной таблицы, хранящейся в базе данных.

Виртуальные таблицы формируются в основном из данных нескольких таблиц базы данных.

Общим для всех виртуальных таблиц является то, что им можно задать ряд параметров, определяющих, какие данные будут включены в эти виртуальные таблицы. Набор таких параметров может быть различным для разных виртуальных таблиц и определяется данными, хранящимися в исходных таблицах базы данных.

Реальные таблицы подразделяются на объектные (ссылочные) и необъектные (не ссылочные).

В объектных таблицах представлена информация ссылочных типов данных (справочники, документы, планы видов характеристик и т.д.). А в необъектных - всех остальных типов данных (константы, регистры и т. д.).

Отличительной особенностью объектных таблиц является то, что они включают в себя поле Ссылка, содержащее ссылку на текущую запись. Кроме этого, для таких таблиц возможно получение пользовательского представления объекта. Эти таблицы могут быть иерархическими, и поля таких таблиц могут содержать вложенные таблицы.

Диаграмма

Диаграмма предназначена для размещения в таблицах и формах диаграмм и графиков различного вида. Логически диаграмма является совокупностью точек, серий и значений серий в точке.

Как правило, в качестве точек используются моменты или объекты, для которых мы получаем значения характеристик, а в качестве серий - характеристики, значения которых нас интересуют. На пересечении серии и точки находится значение диаграммы.

Например, диаграмма продаж видов номенклатуры по месяцам будет состоять из точек - месяцев, серий - видов номенклатуры и значений - оборотов продаж.

Диаграмма как объект встроенного языка имеет три области, которые позволяют управлять оформлением диаграммы: область построения, область заголовка и область легенды.

Диаграмма может быть вставлена в структуру отчета как отдельный элемент. В следующем варианте настроек отчётаВыручкаМастеров мы будем использовать диаграмму в структуре настроек схемы компоновки данных.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Вывод данных по всем дням в выбранном периоде

Следующий отчет, который мы добавим, будет называться Выручка мастеров. Он будет содержать информацию о том, какая выручка была получена ООО «Фамилия курсанта» благодаря работе каждого из мастеров, с детализацией по всем дням в выбранном периоде и разворотом по клиентам, обслуженным в каждый из дней.

В режиме Конфигуратор

Добавим новый объект конфигурации Отчет. Назовем его ВыручкаМастеров и запустим конструктор схемы компоновки данных. Добавим новый **Набор данных - запрос** и вызовем конструктор запроса. В качестве источника данных для запроса выберем виртуальную таблицу регистра накопления **Продажи.Обороты.**

Запрос для набора данных. Параметры виртуальной таблицы

Зададим один из параметров этой виртуальной таблицы - Периодичность. Для этого перейдем в поле **Таблицы**, выделим таблицу и нажмем кнопку **Параметры виртуальной таблицы** (рис. 15.1). В открывшемся окне параметров зададим значение параметра **Периодичность - День** (рис. 15.2)

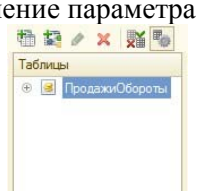


Рис. 15.1. Изменение параметров виртуальной таблицы

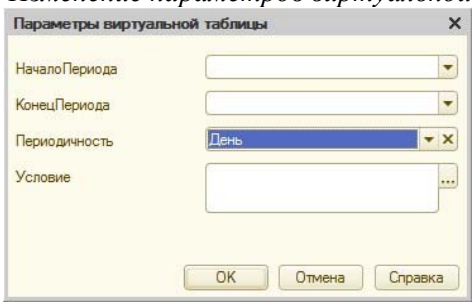


Рис. 15.2. Параметры виртуальной таблицы

Нажмем ОК.

После этого выберем из таблицы следующие поля

- ПродажиОбороты.Мастер,
- ПродажиОбороты.Период,
- ПродажиОбороты.Клиент,
- ПродажиОбороты.ВыручкаОборот.

Теперь перейдем на закладку **Объединения/Псевдонимы** и зададим псевдоним **Выручка** для поля **ПродажиОбороты.ВыручкаОборот** (рис. 15.3)

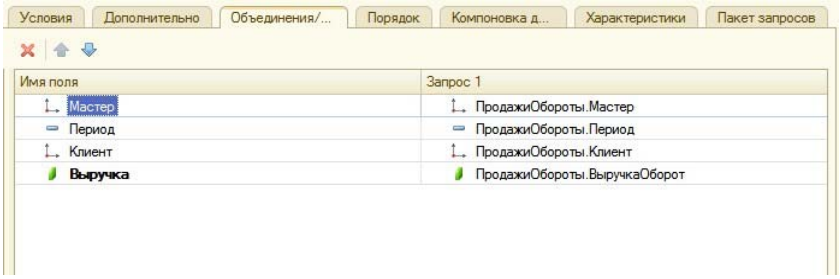


Рис. 15.3. Объединения/Псевдонимы

Ресурсы

Теперь перейдем к редактированию схемы компоновки данных. На закладке **Ресурсы** нажмем

кнопку >> и убедимся, что конструктор выбрал единственный имеющийся у нас ресурс - **Выручка**.

Параметры

На закладке **Параметры** выполним те же действия, что и при создании предыдущего отчета. Для параметра **НачалоПериода** зададим заголовок **Дата начала**. В поле Тип зададим состав даты - **Дата**. Затем добавим еще один параметр –**ДатаОкончания**, установим его тип как **Дата**, состав даты - **Дата**. Для параметра **КонецПериода** зададим выражение (листинг 15.1) и в поле **Ограничение доступности** установим флажок ограничения доступности.

Листинг 15.1. Выражение для расчета значения параметра «КонецПериода»

КонецПериода(&ДатаОкончания,"День")

В результате перечисленных действий параметры компоновки данных будут иметь следующий вид (рис. 15.4).

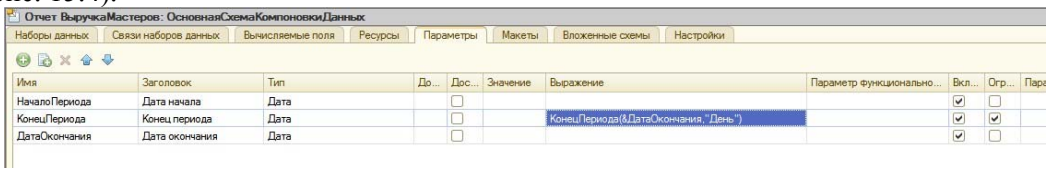


Рис. 15.4. Параметры компоновки данных

Настройки

Теперь создадим структуру отчета. На закладке **Настройки** последовательно создадим две вложенные группировки:

• верхнего уровня - по полю Мастер;

• вложенная в нее - по полю Период.

Для этого сначала выделим корневой элемент **Отчет** в структуре отчета, нажмем кнопку**Добавить** в командной панели окна настроек, добавим новую группировку и укажем поле группировки **Мастер** (рис. 15.5).

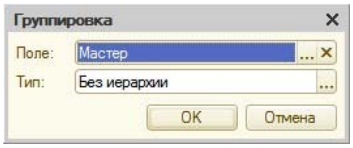


Рис. 15.5. Поле группировки

> Затем добавим в группировку **Мастер** вложенную группировку по полю **Период**.

Для этого выделим группировку **Мастер**, нажмем кнопку**Добавить**, добавим новую группировку и укажем поле группировки **Период**. Затем добавим еще одну группировку, вложенную в группировку по полю **Период**, - **Детальные записи** (без указания группировочного поля).

Для этого выделим группировку **Период**, нажмем кнопку**Добавить** и добавим новую группировку без указания группировочного поля. После этого перейдем на закладку **Выбранные поля** и добавим в список выбранных полей поля **Клиент** и **Выручка**.

Поля **Мастер** и **Период** мы не задаем, так как по этим полям производится группировка данных и их значение будет выведено автоматически. В результате структура отчета будет иметь вид (рис. 15.6).

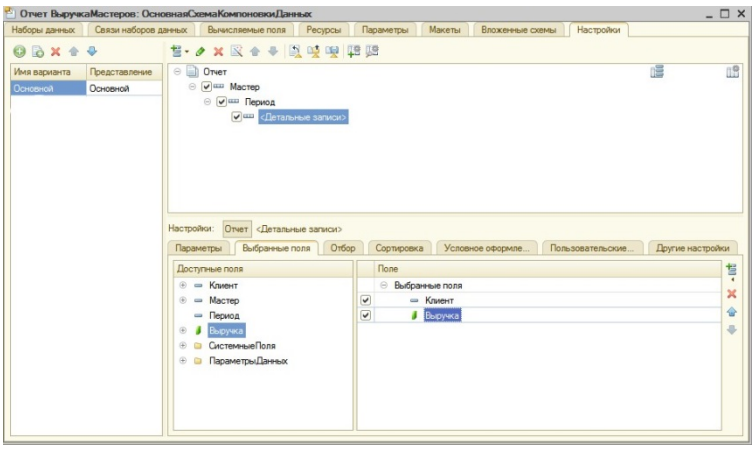


Рис. 15.6. Структура и поля отчета

В заключение перейдем на закладку **Другие настройки** и изменим следующие параметры. Для параметра **Расположение полей группировок** установим значение **Отдельно и только в итогах**.

По умолчанию поля группировок в отчете располагаются вертикально друг под другом. Установка этого свойства в значение **Отдельно и только в итогах** означает, что каждая группировка будет располагаться в отдельной области отчета слева направо и ее наименование будет выводиться только в данной группировке.

Для параметра **Расположение общих итогов по вертикали** зададим значение **Начало**. По умолчанию итоги по вертикали располагаются в конце . Установка этого свойства означает, что общие итоги будут отображаться в начале перед строками группировки. В результате другие настройки отчета примут вид (рис. 15.7).

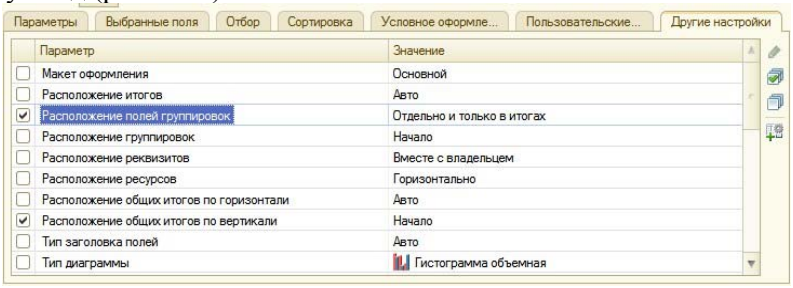


Рис. 15.7. Параметры настроек вывода отчета

Здесь же для параметра **Заголовок** зададим значение **Выручка мастеров**. Затем укажем, что параметры **Дата начала** и **Дата окончания** будут включены в состав пользовательских настроек, и эти настройки будут находиться непосредственно в отчетной форме, то есть будут «быстрыми» настройками. Таким образом, перед формированием отчета пользователь сможет задать отчетный период (рис. 15.8).

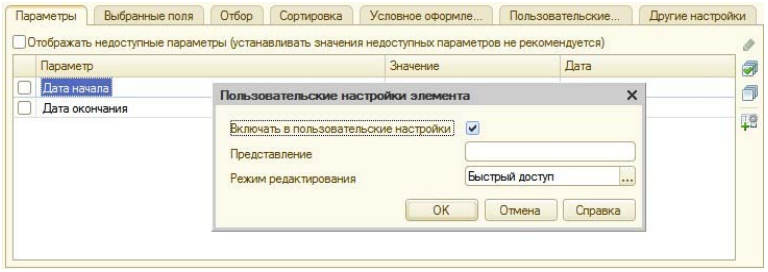


Рис. 15.8. Создание быстрых настроек отчетного периода

В заключение определим, в каких подсистемах будет отображаться наш отчет. Закроем конструктор схемы компоновки данных и в окне редактирования объекта конфигурации Отчёт**ВыручкаМастеров**перейдём на закладку **Подсистемы**. Отметим в списке подсистем конфигурации подсистемы **Оказание услуг** и **Расчет зарплаты**. Таким образом, ссылка на наш отчет автоматически попадет в панель действий этих подсистем.

В режиме 1С Предприятие

Запустим 1С:Предприятие в режиме отладки и посмотрим, как работает отчет.

В открывшемся окне 1С:Предприятия мы видим, что в панели действий разделов **Оказание услуг** и **Расчет зарплаты** в группе команд для выполнения отчетов появилась команда для формирования отчета **Выручка мастеров**.

Выполним эту команду. Зададим отчетный период с 01.07.2009 по 15.07.2009 и сформируем отчет (рис. 15.9).

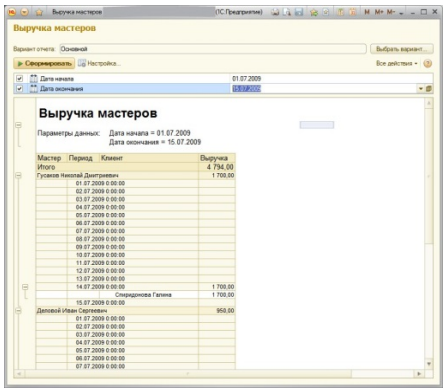


Рис. 15.9. Результат выполнения отчета

Вывод всех дат в выбранном периоде

Если вы помните, в начале раздела мы говорили, что этот отчет должен показывать данные с детализацией по всем дням в выбранном периоде.

У нас же отображаются только те дни, для которых существуют ненулевые записи в таблице регистра накопления **Продажи**.

Для детализации данных в отчете система компоновки данных позволяет указывать для группировок дополнение периодов с заданной периодичностью в указанном интервале.

Поэтому сейчас мы изменим настройки отчета таким образом, чтобы в отчет попадала каждая дата из периода, за который сформирован отчет.

В режиме Конфигуратор

Вернемся в режим Конфигуратор и выполним более тонкую настройку структуры отчета. Откроем схему компоновки данных на закладке **Настройки**. До сих пор все настройки структуры, которые мы выполняли, относились ко всему отчету в целом. Но система компоновки данных позволяет настраивать также и каждый элемент структуры в отдельности.

При установке настроек отчета в средней части окна, под деревом структуры отчета, должна быть выделена кнопка, соответствующая режиму настроек. Кнопка **Отчет** - для настройки отчета в целом или кнопка с именем группировки, например **Детальные записи**, если настройки относятся только к ней.

В нашем случае потребуется изменить настройку группировки **Период**.

Для того чтобы перейти к настройкам именно этой группировки, в поле структуры отчета установим курсор на эту группировку, а затем нажмем кнопку **Период** в командной панели окна. В нижней части окна будут отображены настройки, доступные для данной группировки.

Перейдем на закладку **Поля группировки**. Для поля **Период** установим **Тип дополнения - День** (рис. 15.10).

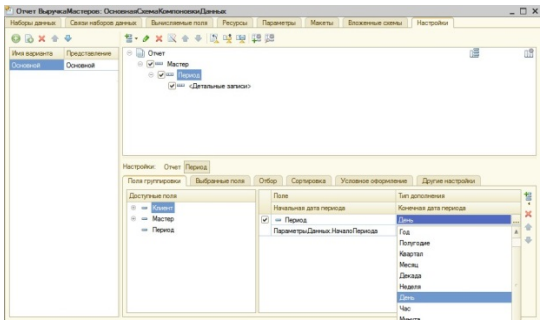


Рис. 15.10. Установка типа дополнения периода

Тем самым мы укажем, что для этой группировки существующие записи с ненулевым значением ресурса будут дополняться записями для каждого из дней.

После этого следует указать, в каком именно периоде будет выполняться такое дополнение.

В поля, расположенные строчкой ниже, можно ввести даты начала и окончания этого периода. Но указание дат в явном виде нас не устраивает, так как пользователь может сформировать отчет за произвольный период. И нам нужно, чтобы дополнение дат выполнялось не в некотором

фиксированном периоде, а именно в том периоде, который выбрал пользователь для всего отчета.

Для того чтобы обеспечить именно такую работу отчета, войдем в режим редактирования поля **Начальная дата периода**, дважды кликнув на нем, и нажмем кнопку очистки *.

После этого, нажав кнопку выбора типа данных, мы сможем выбрать тип данных, отображаемых в этом поле.

Выберем **Поле компоновки данных** (рис. 15.11).

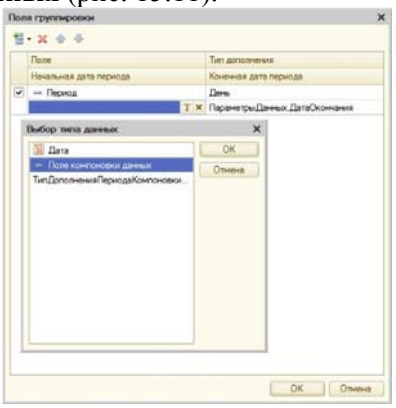


Рис. 15.11. Выбор типа данных

Нажмем ОК.

Теперь нажмем в поле ввода кнопку выбора и в открывшемся окне выбора поля отметим параметр **НачалоПериода**(рис. 15.12). Нажмем ОК.

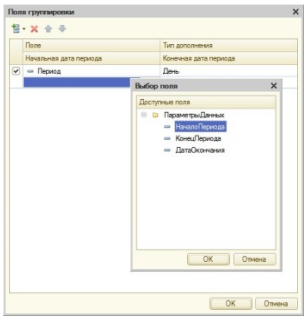


Рис. 15.12. Выбор поля

Для второго поля ввода аналогичным образом укажем, что дата окончания периода будет получена из параметра **ДатаОкончания** (рис. 15.13).

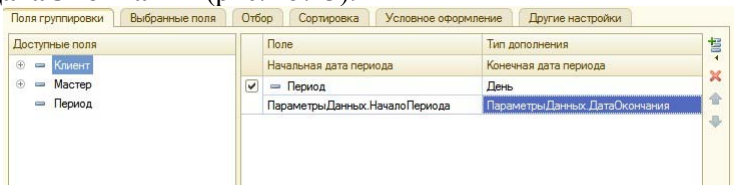


Рис. 15.13. Настройки группировки "Период"

В режиме 1С Предприятие

Запустим 1С:Предприятие в режиме отладки и выполним отчет **Выручка мастеров** за период с 10.07.2009 по 15.07.2009 (рис. 15.14).

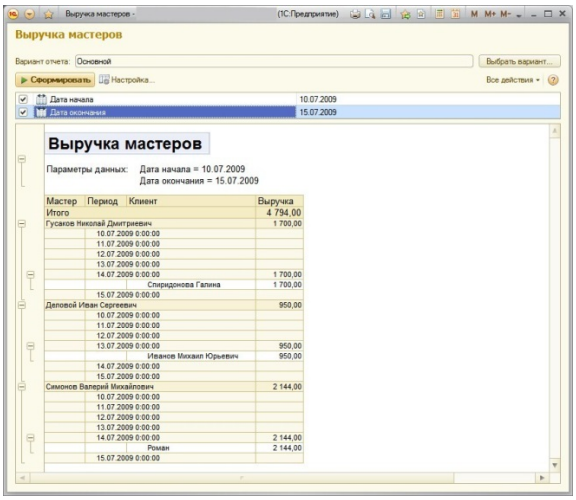


Рис. 15.14. Результат выполнения отчета

Новый вариант отчёта

Для анализа работы мастеров за определенный период может понадобиться представить ту же информацию в другом, более наглядном виде.

Например, директору при начислении зарплаты, чтобы понять, какой из мастеров лучше работает, вполне может понадобиться увидеть диаграмму, отражающую вклад каждого мастера в общую выручку предприятия за период.

Поэтому мы создадим другой вариант отчёта**ВыручкаМастеров**, представляющий данные в виде диаграммы.

В режиме Конфигуратор

Вернемся в конфигуратор и откроем схему компоновки данных на закладке **Настройки**.

В левой части окна находится список вариантов отчета.

При создании настроек отчета в первый раз система компоновки данных по умолчанию создает **Основной** вариант настроек. И мы видим его в списке вариантов нашего отчета.

Чтобы добавить новый вариант, нажмем кнопку**Добавить** над этим списком. Зададим имя варианта –**ОбъемВыручки** (рис. 15.15).

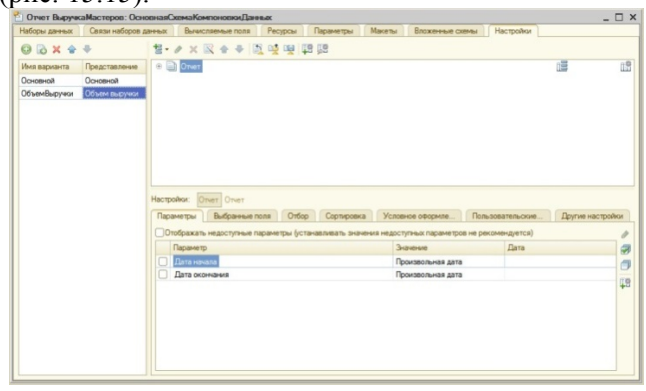


Рис. 15.15. Добавление нового варианта настроек

Мы видим, что структура отчета и все его настройки очистились.

Но они не пропали, а стали невидимы, так как относятся к **Основному** варианту настроек.

Если у отчета есть несколько вариантов, то мы видим и можем изменять настройки того варианта, который выделен в данный момент. Причем вся остальная информация в схеме компоновке данных (ресурсы, параметры, наборы данных) осталась без изменений. Данные для отчета будут получены с помощью того же запроса к базе данных. Изменятся лишь настройки, которые определят, как будет представлен отчет.

Добавим в структуру отчета диаграмму. Для этого выделим корневой элемент **Отчет**, вызовем его контекстное меню и добавим диаграмму. Затем выделим ветку **Точки** и добавим в нее группировку по полю **Мастер**.

Серии диаграммы оставим без изменений.

Для демонстрации вклада мастеров в общий объем выручки хорошо подойдет измерительная диаграмма, которую мы хотим показать. Для этого вида диаграммы достаточно задать только точки, поэтому серии мы не задаем.

В значения диаграммы всегда выводится один из ресурсов отчета. У нас всего один ресурс - **Выручка** (поле ресурса помечено соответствующей пиктограммой и отличается от обычных полей).

Поэтому перейдем на закладку **Выбранные поля**, перейдем на уровень настроек отчета в целом (нажав кнопку **Отчет**) и выберем поле **Выручка** для вывода в отчет.

Структура отчета должна принять следующий вид (рис. 15.16).

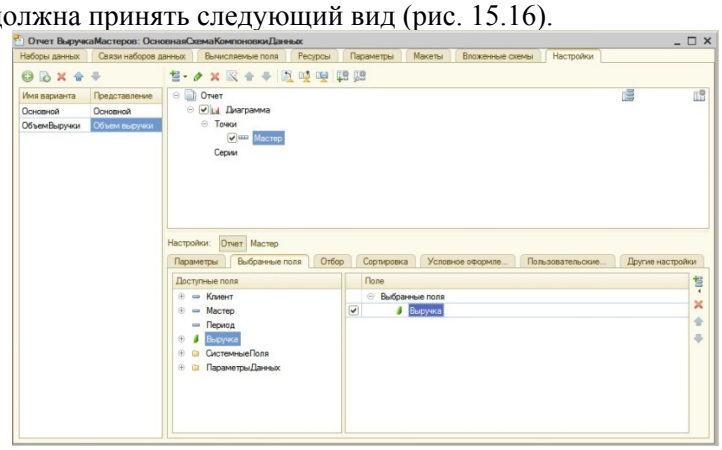


Рис. 15.16. Структура отчета и настройки диаграммы

На закладке **Другие настройки** выберем тип диаграммы - **Измерительная** (рис. 15.17).

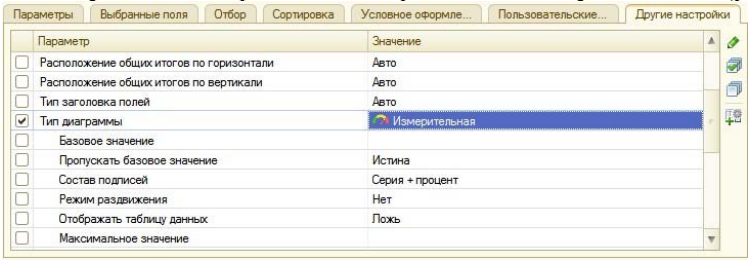


Рис. 15.17. Настройка типа диаграммы

Прокрутив вниз список свойств измерительной диаграммы, зададим ее полосы - **Плохо**, **Хорошо** и **Отлично** (рис. 15.18).

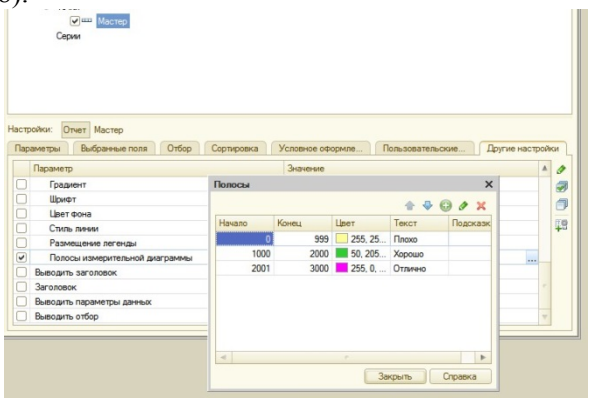


Рис. 15.18. Настройка полос измерительной диаграммы

В заключение включим параметры **Дата начала** и **Дата окончания** в состав пользовательских настроек и установим для них **Режим редактирования - Быстрый доступ**.

В режиме 1С Предприятие

Запустим 1С:Предприятие в режиме отладки и выполним команду **Выручка мастеров** в панели действий раздела **Расчет зарплаты**. В открывшемся окне отчета нажмем кнопку**Выбрать вариант** (рис. 15.19).

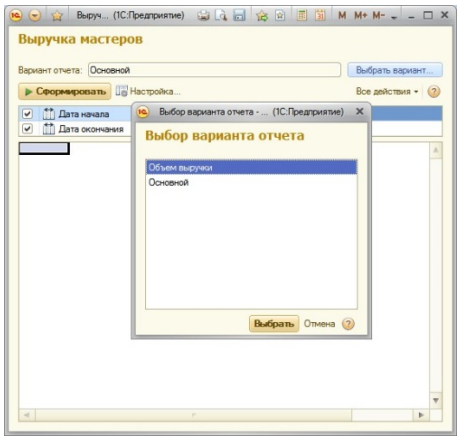


Рис. 15.19. Выбор варианта отчета

В окне вариантов отчета мы видим теперь два варианта - **Основной** и только что созданный нами вариант **Объем выручки**. Выделим его и нажмем кнопку**Выбрать**. Зададим отчетный период с 01.07.2009 по 15.07.2009 и сформируем отчет (рис. 15.20).

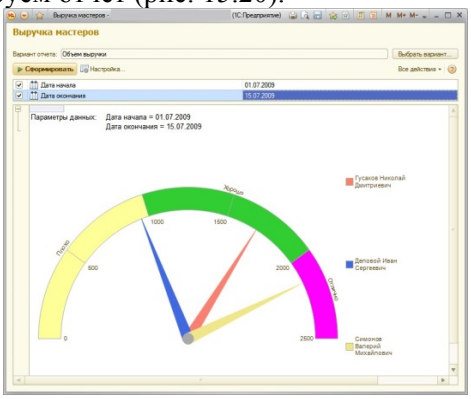


Рис. 15.20. Результат выполнения отчета

В результате мы видим те же данные, что и в основном варианте отчета, представленные в виде измерительной диаграммы. На диаграмме хорошо видна доля каждого мастера в общем объеме выручки. Обратите внимание, что при наведении курсора на сектор диаграммы появляется подсказка.

Если же понадобится просмотреть данные о работе какого-либо мастера с разбивкой по дням и клиентам, достаточно выбрать Основной вариант отчета и переформировать отчет.

Таким образом, на примере отчета Выручка мастеров мы показали создание и использование различных вариантов отчета в целях наилучшего представления информации о работе мастеров.

Контрольные вопросы

- Что такое параметры виртуальной таблицы.
- Что такое левое соединение.
- Как использовать конструктор запроса.
- Как выбрать данные в некотором периоде для отчета.
- Как упорядочить данные в отчете.

Отчёты. Получение актуальных сведений из периодического регистра сведений

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Язык запросов

Алгоритм, по которому данные будут выбраны из исходных таблиц запроса, описывается на специальном языке - языке запросов.

Текст запроса может состоять из нескольких частей:

- описание запроса,
- объединение запросов,
- упорядочивание результатов,

- автоупорядочивание,
- описание итогов.

Обязательной частью запроса является параметр «описание запроса».Описание запроса определяет источники данных, поля выборки, группировки и т. д. Объединение запросов определяет, как будут объединены результаты выполнения нескольких запросов. Упорядочивание результатов определяет условия упорядочивания строк результата запроса. Автоупорядочивание позволяет включить режим автоматического упорядочивания строк результата запроса. Описание итогов определяет, какие итоги необходимо рассчитывать в запросе и каким образом группировать результат.

Следует заметить, что в случае, когда язык запросов используется для описания источников данных в системе компоновки данных, секция описания итогов языка запросов не используется. Это связано с тем, что система компоновки данных самостоятельно рассчитывает итоги на основании тех настроек, которые сделаны разработчиком или пользователем.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Отбор

В режиме Конфигуратор

Создадим настройку отбора в отчете. Для этого в нижней части окна настроек перейдем на закладку Отбор.

Слева мы видим список доступных полей отчета.

Раскроем поле Услуга и двойным щелчком мыши на поле Родитель перенесем его в список условий отбора в правой части окна.

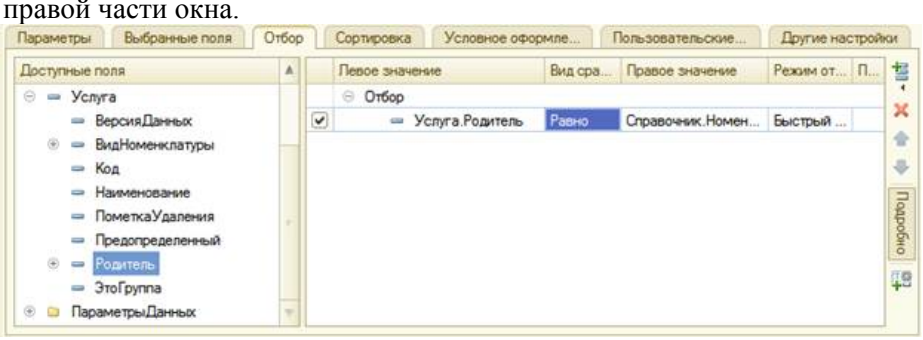


Рис. 13.14. Настройка отбора

Тем самым была создана возможность отбора по группам услуг, которые пользователь может задать в режиме 1С:Предприятие.

- **В режиме 1С:Предприятие**

Откроем отчет в режиме 1С: Предприятие и нажмем кнопку Настройка.

В окне пользовательских настроек отчета появились настройки Отбор и Условное оформление, которые мы только что отметили.

Настройку Непопулярная услуга мы заранее создали в конфигураторе. Добавив настройку условного оформления, мы предоставили пользователю возможность создавать любое количество собственных условий для условного оформления.

Пользователь сможет, при наличии определенной квалификации, задавать многие настройки по своему желанию.

Если же такого желания или соответствующих знаний у него нет, лучше задавать эти на стройки жестко, а пользователю останется только включать или выключать их использование. Да собственно часто достаточно только отчетного периода или еще какой-то жизненно важной настройки, и такие настройки, конечно, нужно размещать непосредственно в отчетной форме. Получение актуальных значений из периодического регистра сведений.

Изучим возможность получения последних значений из периодического регистра сведений и с возможность вывода иерархических справочников.

- **В режиме Конфигуратор**

Добавим новый объект конфигурации Отчет.

Назовём его ПереченьУслуг и запустим конструктор схемы компоновки данных. Добавим

новый Набор данных - запрос и вызовем конструктор запроса.

Запрос для набора данных

В качестве источника данных для запроса выберем объектную таблицу справочника Номенклатура и виртуальную таблицу регистра сведений ЦеныСрезПоследних.

Для того чтобы исключить неоднозначность имен в запросе, переименуем таблицу Номенклатура в СпрНоменклатура. Для этого выделим ее в списке Таблицы, вызовем её контекстное меню ивыберем пункт Переименовать таблицу.

Параметры виртуальной таблицы

Вызовем диалог ввода параметров виртуальной таблицы ЦеныСрезПоследних и укажем, что период будет передан в параметре ДатаОтчета. Для этого выделим эту таблицу в списке Таблицы и нажмем кнопку Параметры виртуальной таблицы.

Затем выберем из таблиц следующие поля:

- СпрНоменклатура. Родитель,
- СпрНоменклатура.Ссылка,
- ЦеныСрезПоследних.Цена.

Левое соединение таблиц

- На закладке *Связи*: в поле Условие связи, что значение измерения Номенклатура регистра сведений должно быть равно ссылке на элемент справочника Номенклатура. А также снимем флажок Все у таблицы регистра и установим его у таблицы справочника, тем самым установив вид связи как левое соединение для таблицы справочника:

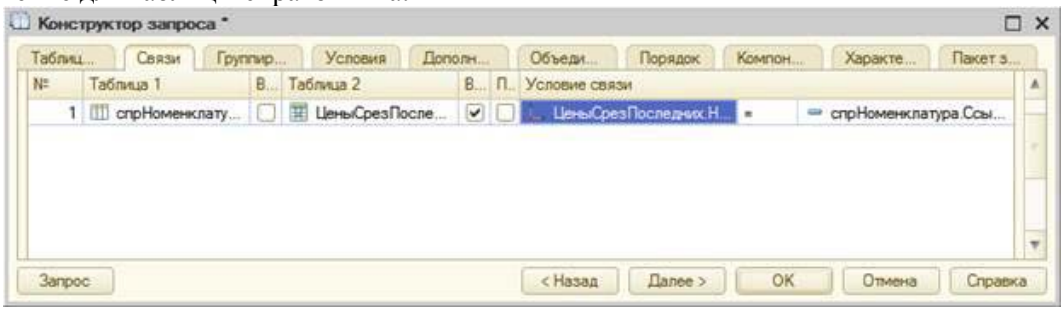


Рис. 13.15. Связь таблиц в запросе

- На закладке *Условия* зададим условие выбора элементов справочника Номенклатура - выбираемые элементы должны соответствовать виду номенклатуры, переданному в параметре запроса Вид Номенклатуры:

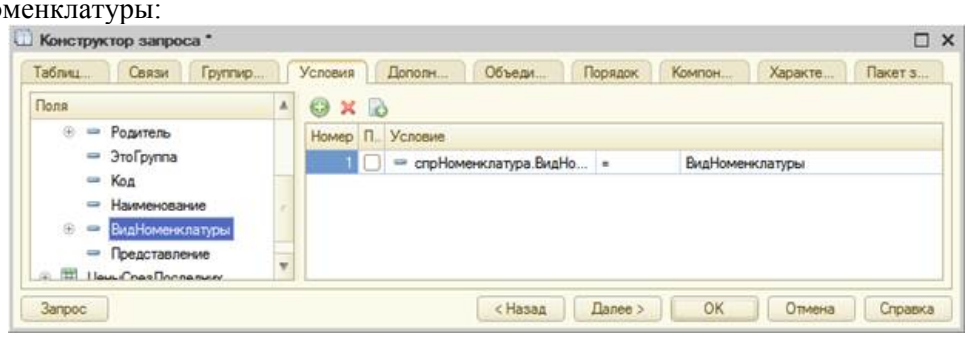


Рис. 13.16. Условия выбора элементов

- На закладке *Объединения/Псевдонимы:* указать псевдоним поля Родитель = ГруппаУслуг, а поля Ссылка = Услуга.

- *НажмемОК–*

После этого, необходимо отредактировать схему компоновки данных, для этого на закладке **Ресурсы**, нажмем на кнопку **добавить** и выберем ресурс - **Цена**

- На закладке *Параметры* зададим значение параметра ВидНоменклатуры - Перечисление.ВидыНоменклатуры.Услуга.

Кроме этого, снимем ограничение доступности для параметра ДатаОтчета. В поле Тип этого параметра зададим состав даты - Дата. Для параметра Период, наоборот, установим ограничение доступности:

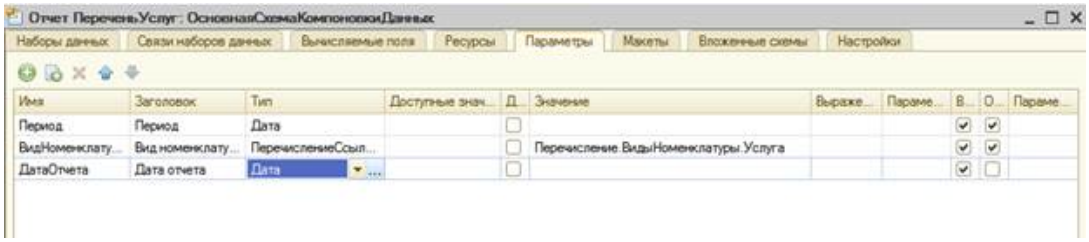


Рис. 13.17. Параметры схемы компоновки

Настройки

- Перейдем на закладку *Настройки:* создадим группировку по полю ГруппаУслуг, указав тип группировки Иерархия.

Существуют следующие типы иерархии для группировок отчета:

• Без иерархии - в группировке выводятся только неиерархические записи.

• Иерархия - в группировке выводятся как неиерархические, так и иерархические записи.

• Только иерархия - в группировке выводятся только иерархические (родительские) записи.

Внутри этой группировки создадим еще одну, без указания группового поля. На подзакладке *Выбранные поля:* укажем поля для вывода Услуга и Цена:

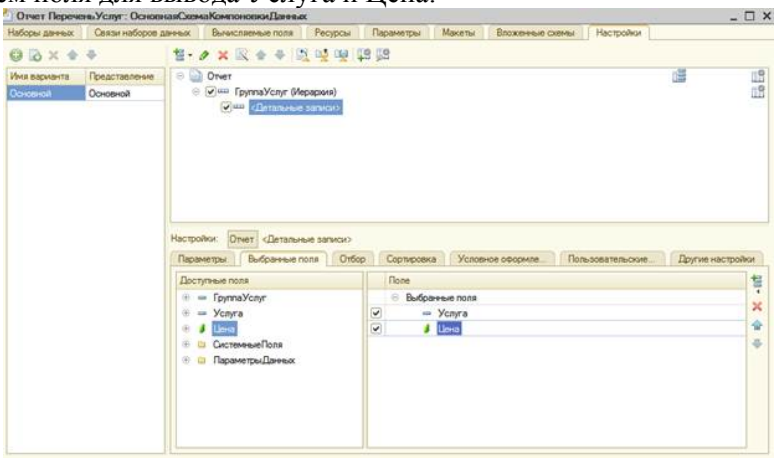


Рис. 13.18. Структура и поля отчета

- На подзакладке *Другие* настройки осуществим следующие действия:

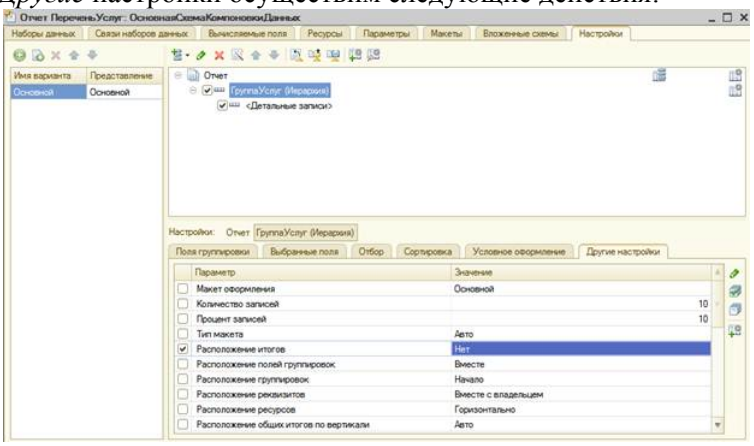


Рис. 13.19. Настройки вывода общих итогов для группировки "Группа Услуг"

- затем:

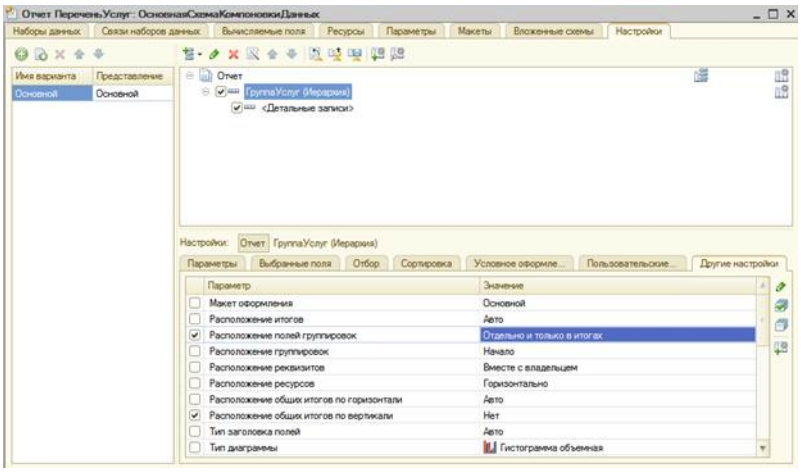


Рис. 13.20. Настройка и вывода итогов для глобального отчета

В заключение включим параметр Дата отчета в состав пользовательских настроек и установим для него Режим редактирования -Быстрый доступ.

Закроем конструктор схемы компоновки данных и в окне редактирования объекта ПереченьУслугперейдём на закладку Подсистемы.

Отметим в списке подсистем конфигурации подсистемы Оказание услуг и Бухгалтерия.

В режиме 1С: Предприятие

Запустим 1С:Предприятие в режиме отладки и прежде всего откроем периодический регистр Цены. *После чего протестируем отчет.*

На примере этого отчета было изучено, как система компоновки данных получает последние значения из периодического регистра сведений и как выводятся группировки по иерархии справочника.

Контрольные вопросы

- Как использовать в отчете данные нескольких таблиц.
- Как использовать группировки в структуре отчета.
- Как получить последние значения регистра сведений.
- Как вывести в отчет иерархические данные.
- Как управлять выводом итогов по группировкам и общих итогов.

Отчеты. Использование вычисляемого поля в отчете

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Система компоновки данных

Система компоновки данных предназначена для создания произвольных отчетов в системе 1С: Предприятие и состоит из нескольких основных частей.

Исходные данные для компоновки отчета содержит в себе схема компоновки данных.

Разработчик создает схему компоновки данных, в которой описывает текст запроса, наборы данных, связи между ними, доступные поля, параметры получения данных, и задает первоначальные настройки компоновки - структуру отчета, макет оформления данных и др.

Отчет системы компоновки имеет сложную иерархическую структуру и может состоять из различных элементов, таких как группировки, таблицы и диаграммы.

При этом пользователь может изменить существующую структуру отчета или вообще создать совершенно новую структуру отчета. Может настроить необходимый ему отбор, оформление элементов структуры отчета, получить расшифровку по каждому элементу и т. д.

Например, может быть задана такая структура отчета, состоящая из одной таблицы и одной диаграммы.

В этом случае сформированный отчет будет иметь следующий вид.

В представленном отчете таблица будет состоять из записей регистра накопления ПродажиОбороты о клиентах и оказанных им услугах. Эти записи сгруппированы по мастерам, которые выполняли заказы. А в группировке будет выведен список услуг, оказанных данным

мастером, и затраченных на это материалов.

Как мы уже говорили в начале раздела, система компоновки данных представляет собой совокупность нескольких объектов. При формировании и исполнении отчета происходит последовательная передача данных от одного объекта системы компоновки данных к другому, до получения конечного результата - документа, показанного пользователю.

Алгоритм взаимодействия этих объектов выглядит следующим образом:

Разработчик создает схему компоновки данных и настройки по умолчанию. В общем случае на основе одной схемы компоновки данных может быть создано большое количество различных отчетов. Настройки компоновки данных, создаваемые разработчиком или изменяемые пользователем, определяют, какой именно отчет будет получен в конкретном случае.

- На основе схемы компоновки и имеющихся настроек компоновщик макета создает макет. Это этап подготовки к исполнению отчета. Макет компоновки данных является уже готовым заданием для выполнения процессором компоновки. Он содержит необходимые запросы, макеты областей отчета и др.
- Процессор компоновки данных выбирает данные из информационной базы согласно макету компоновки, агрегирует и оформляет эти данные.

Результат компоновки обрабатывается процессором вывода, и в итоге пользователь получает результирующий табличный документ.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Использование вычисляемого поля в отчете

Следующий отчет - Рейтинг клиентов - будет показывать в графическом виде, каков доход от оказания услуг каждому из клиентов за все время работы предприятия.

- **В режиме Конфигуратор**

Добавим новый объект конфигурации Отчет. Назовем его РейтингКлиентов и запустим конструктор схемы компоновки данных.

Создадим новый Набор данных - запрос и вызовем конструктор запроса.

Запрос для набора данных

В качестве источника данных для запроса выберем виртуальную таблицу регистра накопления Продажи.Обороты.

Затем выберем из нее следующие поля:

- ПродажиОбороты. Клиент,
- ПродажиОбороты.ВыручкаОборот,
- ПродажиОбороты.СтоимостьОборот.

- На закладке *Объединения/Псевдонимы* укажем, что поле ВыручкаОборот будет иметь псевдоним Выручка, а поле СтоимостьОборот-Стоимость.

- *НажмемОК–*

Перейдем к редактированию схемы компоновки данных.

Вычисляемые поля

На этом этапе мы столкнулись с необходимостью отразить в отчете поле, которого нет в наборе данных. Раньше мы использовали в отчете те поля, которые описывались в наборе данных. Теперь, чтобы отобразить доход от оказания услуг в разрезе клиентов, нам необходимо дополнительное поле, рассчитанное как разница между выручкой и стоимостью оказания услуг.

Для этого в системе компоновки данных есть возможность определения вычисляемого поля.

Вычисляемые поля: дополнительные поля схемы компоновки данных, значения которых будут вычисляться по некоторой формуле.

- На закладке *Вычисляемыеполя* добавим вычисляемое поле, нажав кнопку Добавить. Дадим ему имя - Доход, в колонку Выражение введем следующий код:

Листинг 13.2. Выражение для расчета вычисляемого поля «Доход»

Выручка – Стоимость

Ресурсы

- На закладке *Ресурсы* нажатием кнопки » выберем все доступные ресурсы отчета:

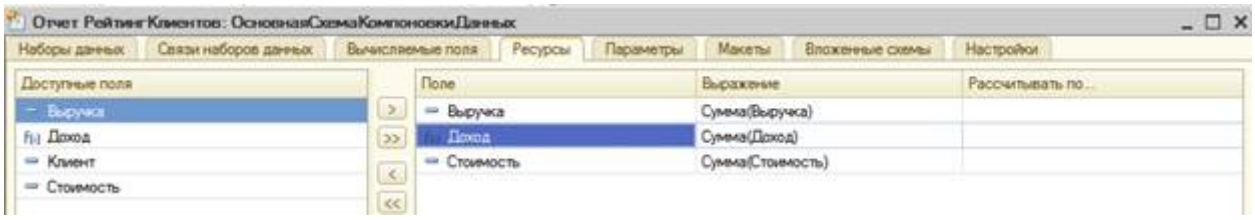


Рис. 13.21. Ресурсы схемы компоновки данных

Настройки

- На закладке *Настройки* добавим в структуру отчета диаграмму. Для этого нажмем кнопку Добавить в командной панели окна настроек и добавим диаграмму:

Затем выделим ветку Точки и добавим в нее группировку по полю Клиент. Серии диаграммы оставим без изменений.

Дело в том, что для демонстрации рейтинга клиентов хорошо подойдет круговая диаграмма, которую мы хотим показать. Для этого вида диаграммы достаточно задать только точки, поэтому серии мы не задаем.

- На закладке *Выбранные поля* выберем поле Доход для вывода в отчет.

Структура отчета должна принять следующий вид:

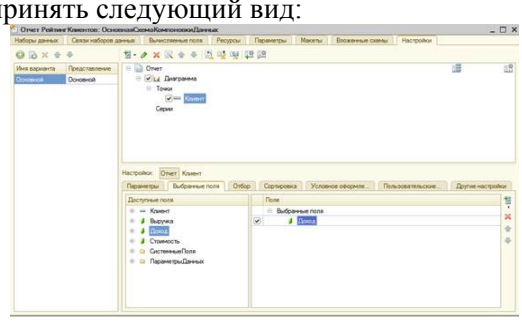


Рис. 13.22. Структура отчета и настройки диаграммы

- На закладке *Другие настройки* выберем тип диаграммы *Круговая объемная* и зададим заголовок отчета - Рейтинг клиентов.

После этого в окне редактирования объекта РейтингКлиентов перейдем на закладку Подсистемы. Отметим в списке подсистем конфигурации подсистемы Оказание услуг и Бухгалтерия.

- **В режиме 1С: Предприятие**

Запустим 1С:Предприятие в режиме отладки и выполним команду Рейтинг клиентов в панели действий раздела Бухгалтерия. Нажмем Сформировать.

Мы видим данные о доходе от оказания услуг по каждому из клиентов, представленные в виде круговой диаграммы. Если вернуться в конфигуратор и изменить тип диаграммы наГистограмма объемная, то наша диаграмма поменяет вид.

Контрольные вопросы

– Как создать отчет, содержащий диаграмму.
– Как использовать параметры в системе компоновки данных.
– Что такое ресурсы в системе компоновки данных.
– Что такое вычисляемые поля в системе компоновки данных.
– Как дополнить данные отчета всеми датами в группировке по периоду.

ДОКЛАД

Задание:подготовить докладпо предложенной тематике.

Примерная тематика докладов:

1. Типы данных в системе 1С:Предприятие 8

2. Типообразующие объекты 1С:Предприятие 8

3. Программные модули в системе 1С:Предприятие 8
4. Переменные во встроенном языке системы 1С:Предприятие 8
5. Операторы. Инструкции препроцессора системы 1С:Предприятие 8
6. Системные перечисления во встроенном языке системы 1С:Предприятие 8
7. Циклы во встроенном языке системы 1С:Предприятие 8
8. Условия во встроенном языке системы 1С:Предприятие 8
9. Работа с датой во встроенном языке системы 1С:Предприятие 8
10. Работа со строками во встроенном языке системы 1С:Предприятие 8
11. Глобальный контекст языка системы 1С:Предприятие 8
12. Встроенные функции системы 1С:Предприятие 8
13. Процедуры-обработчики во встроенном языке системы 1С:Предприятие 8
14. Объекты встроенного языка системы 1С:Предприятие 8, их свойства и методы
15. Работа с табличными документами. Общие методы
16. Работа с табличными документами. Сводные таблицы
17. Работа с табличными документами. Диаграммы

Критерии оценки

Оценки в баллах	Критерии
3	Самостоятельность в подготовке материала
4	Аргументированность и полнота материала
3	Использование современной и актуальной литературы
4	Качество презентации
14	**Итого**