GeekBrains

**Разработка учетной системы на 1С для магазина с возможностью выдачи продукции в аренду на примере книжного магазина с функцией библиотеки. Настройка локального интерфейса покупателя.**

IT-специалист:

Инженер - программист

Дрей Я. В.

Ростов-на-Дону

2024

**Дипломный проект “ Разработка учетной системы на 1С для магазина с возможностью выдачи продукции в аренду на примере книжного магазина с функцией библиотеки. Настройка локального интерфейса покупателя”**

Оглавление

[Введение 3](#_Toc167638815)

[Глава 1. Техническое задание на создание учетной системы 6](#_Toc167638816)

[Глава 2. Проект учетной системы 12](#_Toc167638817)

[Глава 3. Разработка учетной системы 22](#_Toc167638818)

[Глава 4. Настройка ролей и прав доступа 66](#_Toc167638819)

[Заключение 69](#_Toc167638820)

[Список использованной литературы и интернет ресурсов: 70](#_Toc167638821)

# 

# Введение

Компания 1С уже давно является одним из лидеров в области разработки программного обеспечения для автоматизации бизнес-процессов. Многие организации выбирают именно системы 1С в качестве своей учетной системы по следующим причинам:

1. Комплексность решения. 1С предлагает широкий спектр программных продуктов, которые позволяют автоматизировать практически все сферы деятельности предприятия: учет, управление производством, продажами, складским учетом, кадровым учетом и т.д. Благодаря этому организация может получить полноценную и интегрированную систему управления, которая удовлетворит все ее потребности.
2. Простота использования. Программные продукты 1С отличаются легкостью в освоении и использовании. Интерфейсы удобны, интуитивно понятны и имеют понятную структуру, что упрощает работу с системой даже малоквалифицированным пользователям.
3. Гибкость и настраиваемость. 1С позволяет настраивать и дорабатывать программное обеспечение под конкретные потребности и требования организации. Это позволяет создавать уникальные решения, которые полностью соответствуют бизнес-процессам компании.
4. Надежность и безопасность. Продукты 1С хорошо зарекомендовали себя как надежные и безопасные решения. Системы защищены от внешних угроз и обеспечивают сохранность данных, что особенно важно для компаний, работающих с конфиденциальной информацией.
5. Поддержка и обновления. Компания 1С предоставляет своим клиентам качественную техническую поддержку и регулярные обновления программного обеспечения, что позволяет быть уверенными в актуальности и работоспособности своей учетной системы.

В целом, выбор 1С в качестве учетной системы для организации является оправданным решением, которое позволяет повысить эффективность бизнес-процессов, улучшить контроль над деятельностью компании и обеспечить плавное развитие бизнеса.

Учитывая вышесказанное и изучив потребности рынка, я пришла к выводу о целесообразности создания учетной системы именно на основе платформы 1С, также мой выбор обусловлен тем, что в настоящее время я по роду своей основной деятельности являюсь аналитиком и разработчиком 1С.

В настоящее время услуга предоставления в аренду разного вида продукции является весьма востребованной и актуальной. Например, на базе магазинов открываются пункты аренды самокатов, велосипедов, танцевальных костюмов, книг, техники и т.д. Этим обусловлен выбор темы моей дипломной работы.

Работа реализована как отдельная информационная база, созданная на основе пустой конфигурации.

Но в зависимости от потребностей заказчика все или некоторые из модулей могут быть интегрированы в уже существующую информационную базу с помощью механизма расширений 1С.

Механизм расширения конфигурации предназначен для доработки типовой конфигурации без внесения в нее изменений. Простыми словами — это надстройка, которая вносит изменения в логику работы программы без изменения ее исходного кода. Расширение имеет более высокий приоритет, чем типовая конфигурация, поэтому первоначально выполняется код расширения. В результате клиент получает необходимый ему функционал, а конфигурация остается типовой и не снимается с поддержки.

В данном проекте я опробовала следующие роли:

* Бизнес – аналитик.

В качестве бизнес-аналитика мною проведен анализ рынка на предмет востребованности программных продуктов в различных организациях малого и среднего бизнеса, также анализ сфер бизнеса, нуждающихся в автоматизации учетной деятельности. Составлено техническое задание;

* Архитектор ПО.

В качестве архитектора ПО мною создан проект будущей системы, описана структура и схемы взаимодействия между модулями, обозначены цели и задачи внедрения учетной системы;

* Разработчик ПО.

Созданы объекты конфигурации, написан программный код, позволяющий реализовать проект учетной системы. Разработан интерфейс учетной системы;

* Администратор системы.

Настроены роли и права пользователей учетной системы;

* Тестировщик.

Добавлены данные в систему, проведен сквозной пример, позволяющий проверить корректность работы всех основных функций учетной системы.

# Глава 1. Техническое задание на создание учетной системы

1. Наименование проекта:

Создание учетной системы для книжного магазина и библиотеки на базе пустой конфигурации 1С.

1. Цель проекта:

Разработать учетную систему, которая:

- упростит процессы учета книг: приобретение, сдача в аренду, продажа;

- регламентирует процессы взаимодействия с поставщиками книг для библиотеки и магазина, читателями книг библиотеки и покупателями в магазине;

- позволит отслеживать возврат книг читателями в библиотеку, и в случае несвоевременного возврата начислять штраф;

- позволит клиенту библиотеки или магазина получать информацию о книге и ее наличии, а также оформлять резерв самостоятельно с локального терминала, расположенного в магазине или библиотеке;

- реализует бухгалтерский учет.

1. Требования к системе.
   1. Функциональные требования:
      1. По структуре система должна быть логично разбита на подсистемы, в дальнейшем это облегчит разграничение прав доступа.

Подсистемы:

- подсистема «Нормативно-справочная информация (НСИ)» включает в себя используемые справочники и отчеты, предназначена в той или иной степени для работы пользователей с различными ролями в зависимости от прав доступа;

- подсистема «Библиотека» предназначена для учета книг в библиотеке, работы пользователя с ролью «библиотекарь»;

- подсистема «Магазин» предназначена для учета книг в магазине, работы пользователя с ролью «менеджер магазина»;

- подсистема «Клиент» предназначена для посетителей библиотеки и клиентов магазина. В подсистеме должен быть реализован функционал, предоставляющий информацию о выбранной книге в библиотеке или магазине, а также для клиентов магазина должен быть реализован дополнительный функционал, позволяющий клиенту самостоятельно выбирать книги, «складывать» в виртуальную корзину и оформлять резерв, который автоматически будет попадать к менеджеру магазина;

- подсистема «Бухгалтерия» предназначена для ведения бухгалтерского учета, работы пользователя с ролью бухгалтер.

3.1.2. В системе для правильного и удобного ведения учета, а также для возможности проведения анализа бизнес показателей необходимо создать ряд справочников, документов, отчетов, обработок, регистров сведений и регистров накопления.

* Справочники:

- «Читатели»;

- «Номенклатура» (список книг);

- «Варианты номенклатуры»;

- «Контрагенты» (поставщики);

- «Договоры» (с поставщиками).

* Документы:

- «Приходная накладная» отражает факт взаимодействия с поставщиком, то есть поступление книг в магазин или библиотеку от поставщика согласно договору, в документе должно быть указано какие именно книги в каком количестве и по какой цене приобретены;

- «Расходная накладная» отражает факт взаимодействия с покупателем в магазине, то есть факт продажи книг. Расходная накладная должна создаваться автономно или на основании резерва покупателя;

- «Карточка книги» отражает факт взаимодействия с клиентом библиотеки, формируется автоматически раз в сутки по определенной книге на основании формуляра. На основании первого в течение суток формуляра должна формироваться новая «карточка книги», данные из последующих в течение суток формуляров будут заноситься в табличную часть карточки. В карточке по каждой книге должна отражаться информация о том, каким читателем, на какой срок взята данная книга, а также в случае несвоевременного возврата должен рассчитываться штраф за просрочку;

- «Формуляр» отражает факт выдачи книги конкретному читателю на определенный срок;

- «Резерв покупателя» отражает намерение клиента магазина приобрести выбранные им самостоятельно книги, «резерв покупателя» должен формироваться автоматически при нажатии кнопки клиентом из добавленных в «корзину» книг;

- документ «Снятие с резерва» отражает намерение покупателя отменить резерв, должен формироваться автоматически при нажатии кнопки покупателем и «отменять» конкретный «резерв покупателя».

* Отчеты:

- отчет «Номенклатура по группам» должен показывать количество наименований книг в каждой группе. Рекомендуется при организации учета объединять книги в группы по определенным признакам;

- отчет «Остатки в магазине» должен отражать движение номенклатуры в магазине (приход, расход, остаток) за выбранный период. Рекомендуется добавить возможность для пользователя задавать дополнительные критерии отбора;

- отчет «Продажи» должен показывать, сколько товара и на какую сумму было продано в магазине за выбранный период. Рекомендуется добавить возможность для пользователя задавать дополнительные критерии отбора;

- отчет «Резерв покупателя» должен показывать в разрезе номенклатуры сколько книг находится в резерве на определенную дату. Дополнительно необходимо реализовать функционал, который позволит менеджеру магазина из отчета при нажатии кнопки отменять невостребованный резерв;

- «Оборотно-сальдовая ведомость» - стандартный бухгалтерский отчет.

* Обработки:

- обработка «Выбор книги в библиотеке» должна предоставлять читателю информацию о характеристиках книги и о наличии ее в библиотеке в настоящий момент;

- в обработке «Выбор книги в магазине» помимо предоставления информации о характеристиках книги и наличии ее в магазине должен быть реализован функционал «корзины», то есть покупатель должен иметь возможность самостоятельно выбирать книги для покупки и оформлять резерв.

* Константы:

- «Штраф за сутки». Размер суточного штрафа за несвоевременный возврат книги рекомендуется реализовать в виде константы, с возможностью изменения значения на стороне пользователя.

* Регистры сведений:

- в регистре сведений «Значения свойств номенклатуры» будут указываться свойства номенклатуры со значениями, объединенные по признаку набор свойств;

- регистр сведений «Цены номенклатуры» рекомендуется сделать периодическим и вносить туда сведения о цене в зависимости от вида цены на дату.

* Регистры накопления:

- в регистр накопления «Остатки товаров в магазине» с видом регистра «остатки» должны записываться данные о движении номенклатуры в магазине согласно документам «Приходная накладная» и «Расходная накладная» с видом движения накопления «приход» или «расход» соответственно;

- регистр накопления «Продажи» должен иметь вид регистра «обороты», в него должны вноситься данные из «Расходных накладных»;

- регистр накопления «Резерв покупателя» должен иметь вид регистра «остатки», в него будут вноситься данные о движении номенклатуры из документов «Резерв покупателя» и «Снятие резерва» с видом движения накопления «приход» или «расход» соответственно.

* Регистры бухгалтерии:

- «Управленческий» - стандартный регистр бухгалтерского учета.

* Планы видов характеристик:

- план видов характеристик «Свойства номенклатуры» должен описывать дополнительные свойства номенклатуры для большей эффективности дальнейшей аналитики;

- «Виды субконто» - стандартный план видов характеристик бухгалтерского учета.

* Планы счетов:

- «Основной» - стандартный план счетов бухгалтерского учета.

* Перечисления:

Предполагается, что некоторые данные в системе будут иметь ограниченное количество значений, поэтому рекомендуется организовать их хранение с помощью перечислений.

- «Отделы нашей фирмы»;

- «Виды цен»;

- «Вид контрагента»;

- «Статус резерва».

* 1. Нефункциональные требования:
     1. Надежность и защищенность.

Система должна быть надежной и защищенной от несанкционированного доступа.

* + 1. Хранение информации.

Все данные должны храниться в базе данных с возможностью резервного копирования.

* + 1. Требования к интерфейсу.

Интерфейс системы должен быть интуитивно понятным и удобным для пользователя.

1. План работ:
   1. Анализ требований и составление проекта - 2 недели.
   2. Разработка и настройка справочников и документов - 2 недели.
   3. Разработка и настройка отчетов и обработок - 2 недели.
   4. Настройка ролей и выдача прав доступа – 1 неделя.
   5. Тестирование и отладка системы – 1 неделя.
2. Сроки реализации проекта: 2 месяца.

# Глава 2. Проект учетной системы

* 1. **Подсистема «Нормативно-справочной информации (НСИ)»**



Рис.1. Проект подсистемы «НСИ»

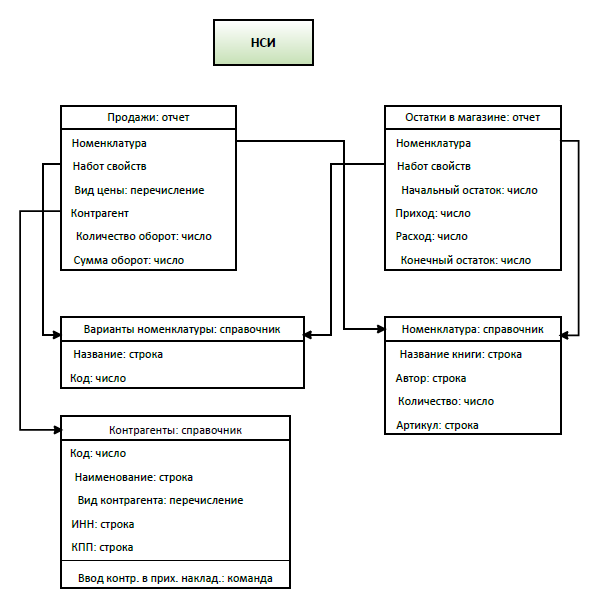


Рис.2. Проект подсистемы «НСИ»

* 1. **Подсистема «Библиотека»**

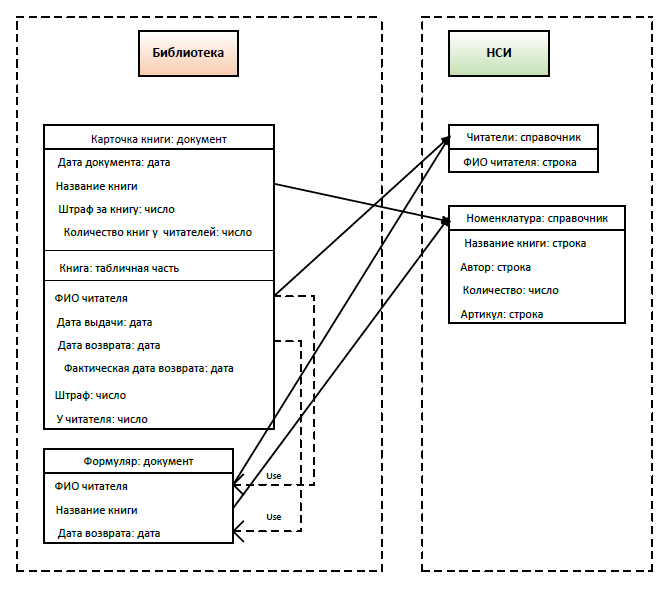


Рис.3. Проект подсистемы «Библиотека»

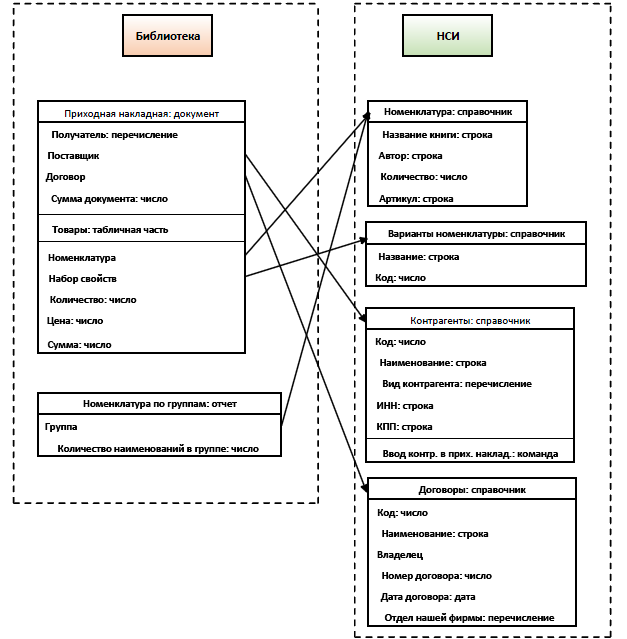


Рис.4. Проект подсистемы «Библиотека»

* 1. **Подсистема «Магазин»**

В документах «Резерв покупателя», «Снятие с резерва», в регистре сведений «Цены номенклатуры» и в регистрах накопления «Продажи», «Остатки в магазине», «Резерв покупателя» типы данных, являющиеся ссылками на справочники определяются аналогично документам «Приходная накладная» и «Расходная накладная», на схеме не указаны.

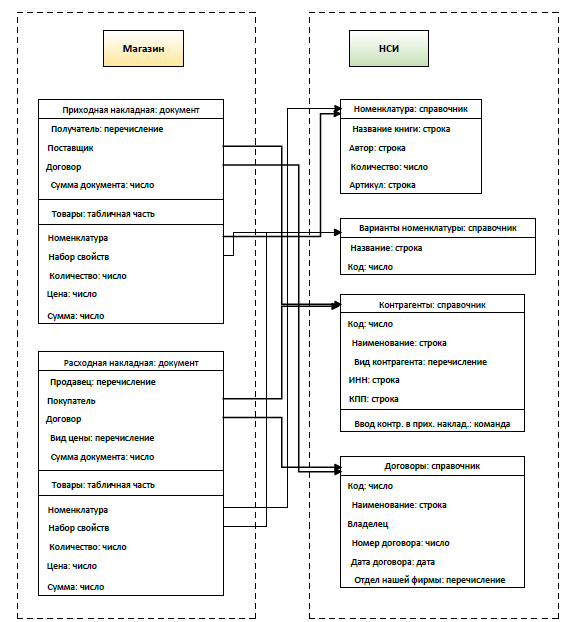


Рис.5. Проект подсистемы «Магазин»

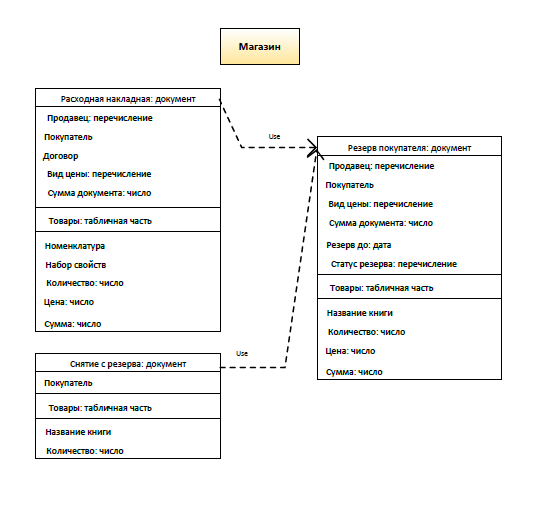


Рис.6. Проект подсистемы «Магазин»

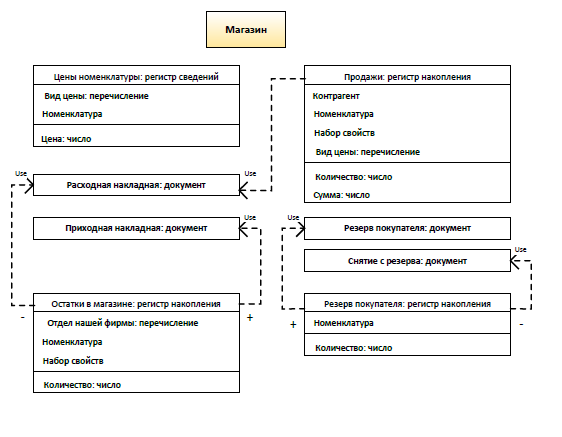


Рис.7. Проект подсистемы «Магазин»

* 1. **Подсистема «Клиент»**

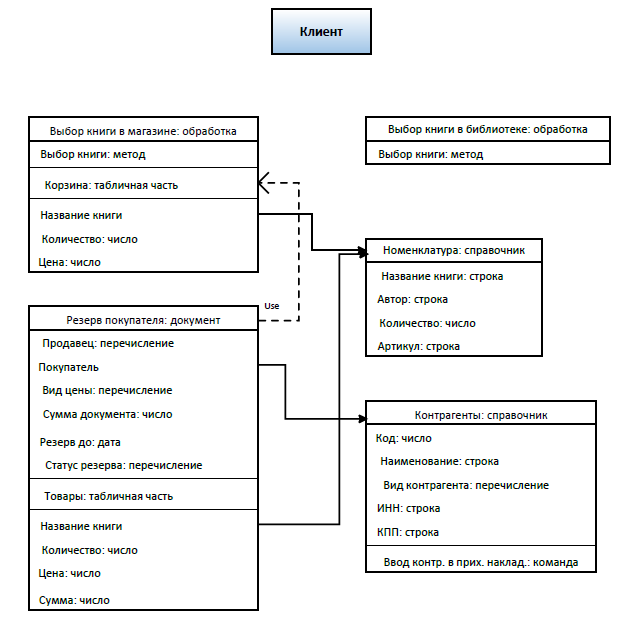


Рис.8. Проект подсистемы «Клиент»

* 1. **Подсистема «Бухгалтерия»**

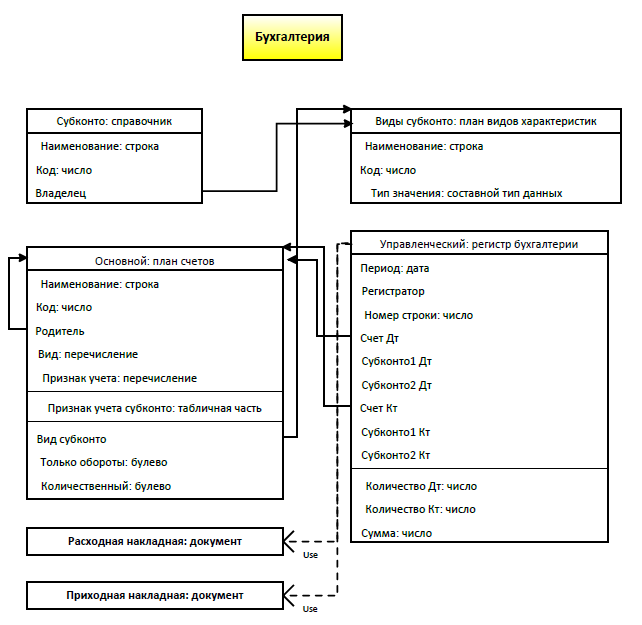


Рис.10. Проект подсистемы «Бухгалтерия»

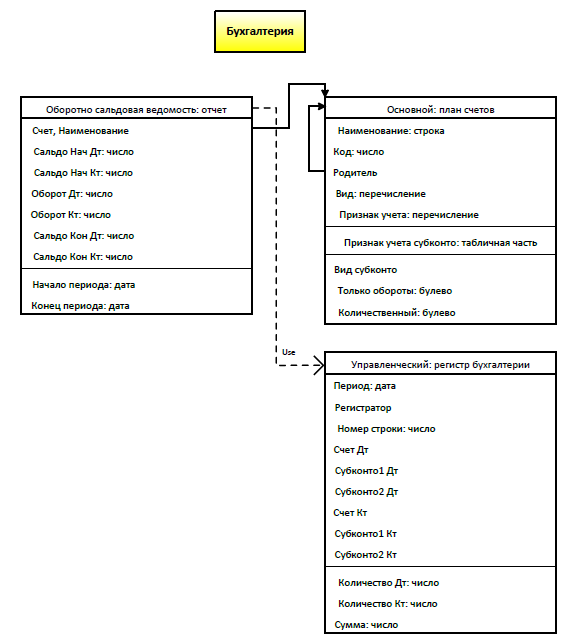
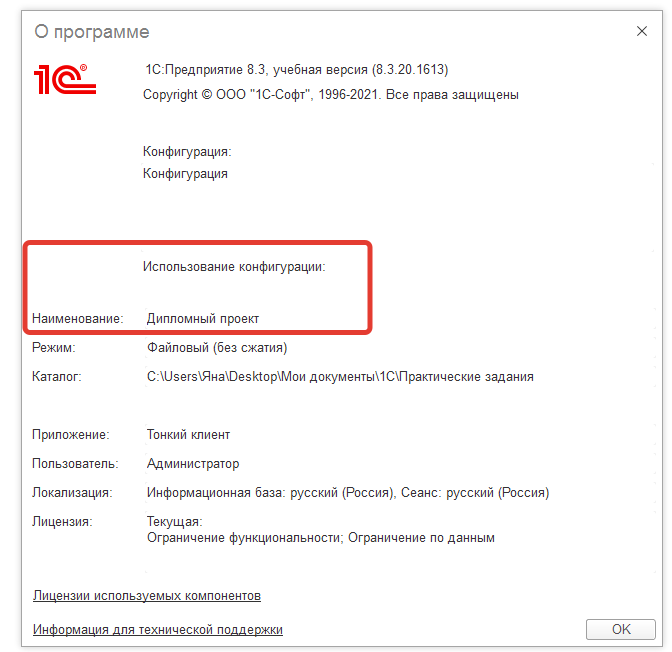


Рис.11. Проект подсистемы «Бухгалтерия»

# Глава 3. Разработка учетной системы

**3.1. Описание основных элементов конфигурации 1С**

Разработка учетной системы проводилась на основе пустой конфигурации 1С на платформе 1С:Предприятие 8.3, учебная версия.



Пустая конфигурация в программе 1С представляет собой базу данных или информационную систему, в которой нет ни одного объекта данных или настроек. Это означает, что пользователь должен создать все необходимые элементы конфигурации с нуля.

Элементы конфигурации в 1С - это различные объекты, которые могут быть созданы и настроены в конфигурации программного продукта 1С:Предприятие. Под элементами конфигурации понимаются справочники, документы, регистры сведений, регистры накопления, планы счетов, планы видов характеристик, отчеты, обработки и многие другие объекты.

При разработке учетной системы использовались следующие элементы конфигурации:

* Справочники в 1С — это прикладные объекты конфигурации, которые позволяют хранить в информационной базе данные, имеющие одинаковую структуру и списочный характер.

Примеры справочников: список сотрудников, перечень товаров, список поставщиков или покупателей.

Каждый элемент справочника характеризуется кодом и наименованием. Система поддерживает режим автоматической нумерации элементов и позволяет контролировать уникальность кодов.

Справочники могут поддерживать иерархическое расположение элементов и создавать предопределённые элементы, которые существуют в справочнике всегда вне зависимости от действий пользователя.

* **Документы** — это [прикладные объекты конфигурации](https://v8.1c.ru/platforma/obekty-konfiguracii/). Они позволяют хранить в прикладном решении информацию о совершенных хозяйственных операциях или о событиях, произошедших в «жизни» предприятия вообще. Это могут быть, например, приходные накладные, приказы о приеме на работу, счета, платежные поручения и т. д.

Каждый документ имеет свою структуру, которая включает:

- Реквизиты (номер, дата, время создания документа, данные о поставщике).

- Табличную часть (информация о товаре, сумме сделки, складе, на который приходуется продукция).

- Строку для комментария.

* Регистры сведений — это прикладные объекты конфигурации. Они позволяют хранить в прикладном решении произвольные данные в разрезе нескольких измерений. Например, в регистре сведений можно хранить курсы валют в разрезе валют, или цены предприятия в разрезе номенклатуры и типа цен. Структура. Информация в регистре сведений хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов.
* Регистры накопления в 1С — это основа учёта числовых значений. Они накапливают и содержат информацию о планировании, взаиморасчётах и складском учёте.

Регистры представляют собой таблицы, содержащие всю необходимую информацию об оборотах компании. Поступления, списания и остатки хранятся в одном месте, что упрощает процесс постройки отчёта.

Существует два вида регистров накопления:

-Обороты отвечают за поступление и реализацию объёма продаж.

-Остатки контролируют систему остаточного регистра. Остатки напрямую зависят от оборотов.

Регистры накопления являются программистскими объектами.

* Планы счетов определяют структуру учета финансовых операций предприятия.
* Планы видов характеристик — это прикладные объекты конфигурации. Они предназначены для хранения информации о характеристиках различных объектов.

С их помощью пользователь может создавать всевозможные характеристики, описывать тип этих характеристик и задавать их значения. Например, для того, чтобы описывать товары произвольным количеством произвольных характеристик (цвет, размер, запах и т. д.).

* Отчеты — это прикладные объекты конфигурации. Они предназначены для обработки накопленной информации и получения сводных данных в удобном для просмотра и анализа виде. Конфигуратор позволяет формировать набор различных отчетов, достаточных для удовлетворения потребности пользователей системы в достоверной и подробной выходной информации.
* Обработки в 1С — это специальный объект конфигурации, предназначенный для изменения информации в базе данных или создания нового функционала для администратора или пользователей.

Обработки можно условно разделить на следующие группы:

Отельные наборы функций и процедур, подобие объекта или класса. Программисты могут использовать их как библиотеку или пользоваться всеми прописанными в обработке кусками кода, если включат ее в конфигурацию.

Вспомогательные обработки. Выполняют автоматизацию небольшого участка механической работы или предоставляют пользователям данные в требуемом виде.

Дополнительные инструменты работы с данными. Это специализированные обработки, чаще всего позволяющие работать с любой конфигурацией и базой данных.

Таким образом, элементы конфигурации в 1С представляют собой различные объекты, которые используются для автоматизации деятельности предприятия и обеспечивают работу с данными и информацией.

Пустая конфигурация может использоваться для начала разработки нового проекта или для тестирования новых функций и возможностей программы 1С. Таким образом, для работы с пустой конфигурацией разработчику потребуется создать все необходимые настройки, параметры и объекты, чтобы система стала полноценно функционировать.

**3.2. Общие сведения о разработке в 1С**

Логика поведения объектов в 1С задается в общем модуле, модуле объекта, менеджера объекта или в модуле формы.

1. Модуль объекта предназначен для реализации поведения отдельного экземпляра объекта (СправочникОбъект, ДокументОбъект и т.п.). В модуле объекта размещаются процедуры и функции, которые работают с данными объекта (ЭтотОбъект и переменные модуля объекта), в том числе когда он еще не записан в информационную базу.

Например, в модуле объекта могут размещаться:

- обработчики событий объекта

- процедуры заполнения экземпляра объекта.

Следует иметь в виду, что для вызова экспортных процедур и функций модуля объекта из других модулей может потребоваться предварительно получить сам экземпляр объекта из информационной базы с помощью метода ПолучитьОбъект. При этом происходит загрузка объекта из базы целиком, вместе с его табличными частями, что достаточно ресурсоемко.

2. Модуль менеджера объекта предназначен для размещения "статической" функциональности, которая логически неразрывно связана с объектом метаданных, но не зависит от состояния конкретного экземпляра объекта данных. Это могут быть процедуры и функции:

- относящиеся не к одному, а сразу к некоторой совокупности объектов. Например, это функции для вывода на печать списка объектов; функции, возвращающие информацию, общую для всех экземпляров объекта метаданных; процедуры обновления данных информационной базы, которые связаны с объектом метаданных;

- которые работают с объектом, записанным в ИБ. В таких функциях входным параметром является ссылка на объект. Например, это функции для получения печатной формы по ссылке на объект, процедуры формирования движений по ссылке на объект и т.п.

Для выполнения функций модуля менеджера объекта не должен требоваться экземпляр объекта данных (СправочникОбъект, ДокументОбъект и т.п.).

1. Если функциональность невозможно однозначно отнести к тому или иному объекту метаданных, то она является логически общей для нескольких объектов. В этом случае ее следует размещать в общем модуле.
2. В модуле формы размещаются процедуры и функции, которые необходимы только для реализации логики работы этой формы и исполняются в контексте этой формы.

Также стоит сразу отметить, что в 1С существует четыре директивы компиляции, доступные в модулях формы: «&НаКлиенте», «&НаСервере», «&НаСервереБезКонтекста» и «&НаКлиентеНаСервереБезКонтекста».

Директивы, в имени которых упоминается «Клиент», устанавливают ограничение на обращение к базе данных.

Процедуры или функции, написанные под директивой «Без контекста», не имеют доступа к контексту (данным) формы.

Можно вызывать только те процедуры и функции, которые находятся под одноимённой (с родительским методом) директивой или под директивой, находящейся ниже (чем у родительского метода) согласно списку:

- &НаКлиенте

- &НаСервере

- &НаСервереБезКонтекста

- &НаКлиентеНаСервереБезКонтекста.

То есть из метода, описанного под директивой «&НаКлиенте», можно вызывать процедуры и функции, описанные под любой директивой. А вот «из-под» директивы «&НаСервереБезКонтекста» можно вызывать только то, что описано под директивой «&НаСервереБезКонтекста» или «&НаКлиентеНаСервереБезКонтекста».

Передача данных между клиентом и сервером возможна только посредством серверного вызова.

Серверный вызов – это передача какой-то информации с клиентской части «1С:Предприятие 8» на серверную часть с целью вернуть обратно некий набор данных.

**3.3. Разработка подсистемы «Магазин»**

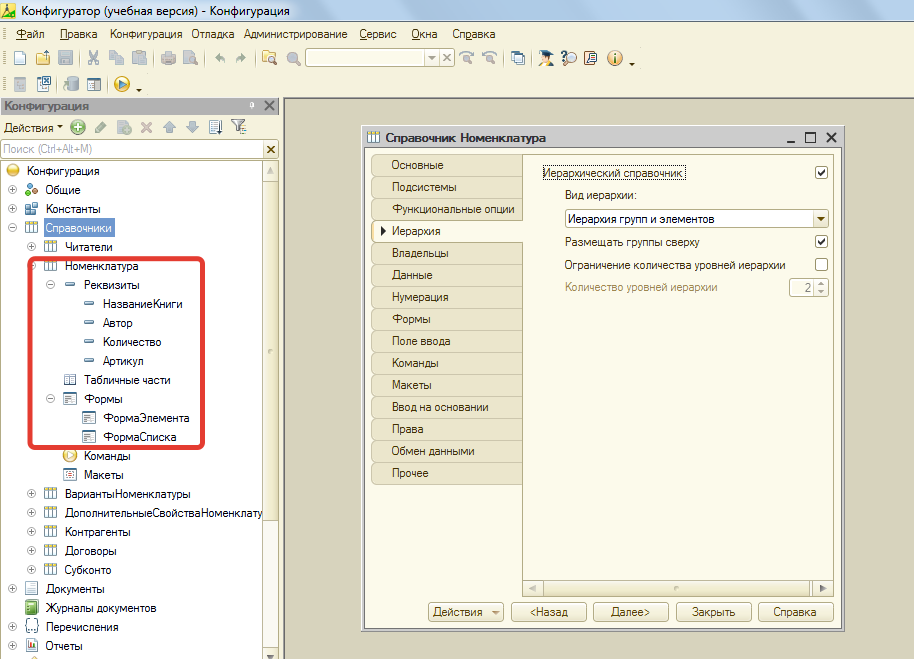
Самой большой и функционально-значимой подсистемой учетной системы является «Магазин».

Разработку любой подсистемы целесообразно начинать с создания справочников. Согласно проекту для полноценной работы подсистемы необходимо реализовать следующие справочники: Номенклатура, Варианты номенклатуры, Контрагенты, Договоры.

Справочник «Номенклатура» был реализован путем добавления нового стандартного объекта Справочник.

Для удобства восприятия информации пользователем системы было принято решение сделать справочник иерархическим. Для системы программы 1С существует две вариации иерархии – это иерархия поэлементная и иерархия групп и элементов. В нашем случае была выбрана иерархия групп и элементов. При иерархии групп или элементов существуют группы (каталоги), в которые входят или другие группы, или элементы. И группы, и элементы являются экземплярами объекта справочника.

Группа это тоже экземпляр объекта справочника, по которому не ведется какой-либо учёт (как правило), и который может быть родителем другого экземпляра, т.е. в группу могут входить другие группы и элементы справочников.



В справочник добавлены необходимые реквизиты. Табличные части не предусмотрены. Форма элемента справочника, в целом, осталась стандартной, было изменено расположение элементов формы.

Форма списка была реализована в виде динамического списка.

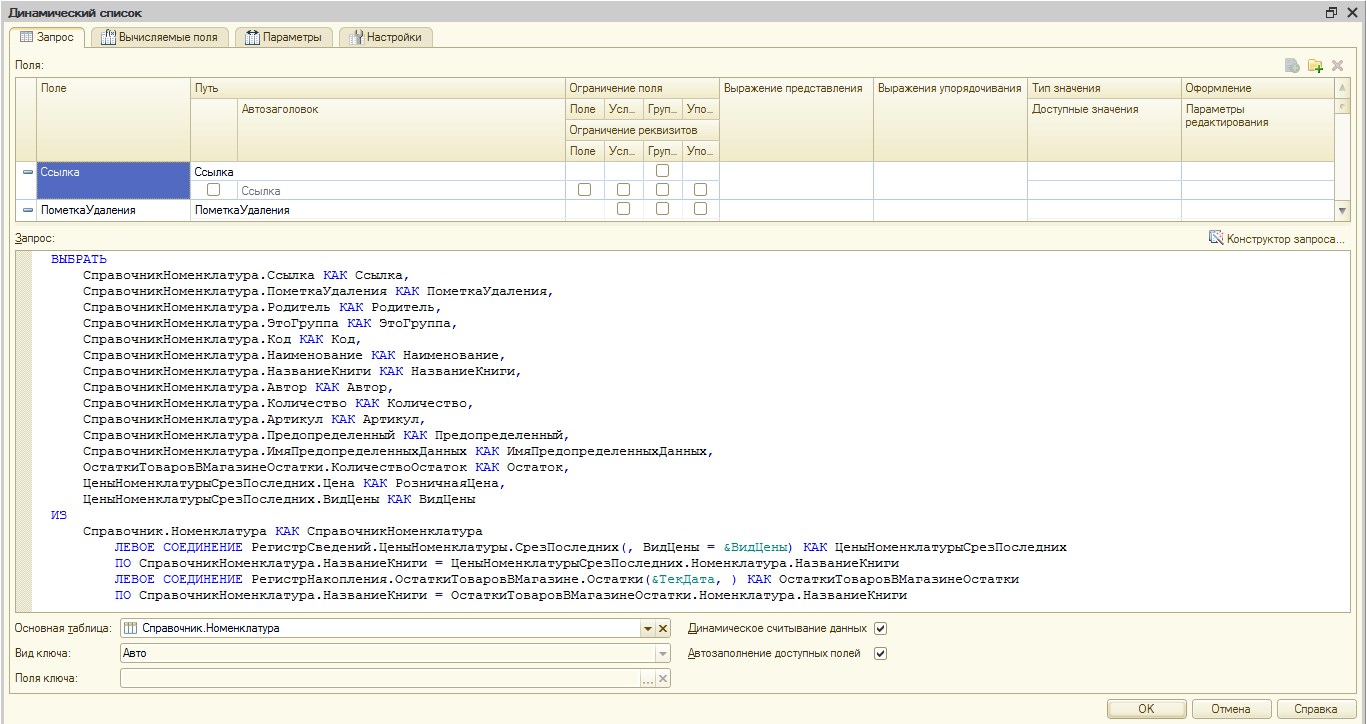
Динамический список — это интерфейсный объект встроенного языка, который используется для отображения различных списков объектов базы данных или необъектных данных — записей регистров.

Динамические списки в формах строятся на основе системы компоновки данных. Для динамического списка разработчик или указывает отображаемый объект конфигурации (фактически выбирает таблицу), или задает произвольный текст запроса, который будет использован для считывания данных.

Система автоматически выполняет считывание данных запроса порциями, по мере навигации пользователя по списку.

Для динамических списков с произвольными запросами существует возможность указать ключевые поля для запроса. Это позволяет спискам работать более эффективно и обеспечивать всю базовую функциональность, включая группировку записей.

Динамический список формы списка справочника «Номенклатура» реализован с помощью произвольного запроса. Такой способ позволяет пользователю из формы списка справочника, не открывая дополнительные источники, сразу получать информацию о количестве доступной номенклатуры (регистр накопления «Остатки товаров в магазине») и об актуальных ценах (регистр сведений «Цены номенклатуры»).



На рисунке показан внешний вид формы списка справочника «Номенклатура» с реализованным динамическим списком.

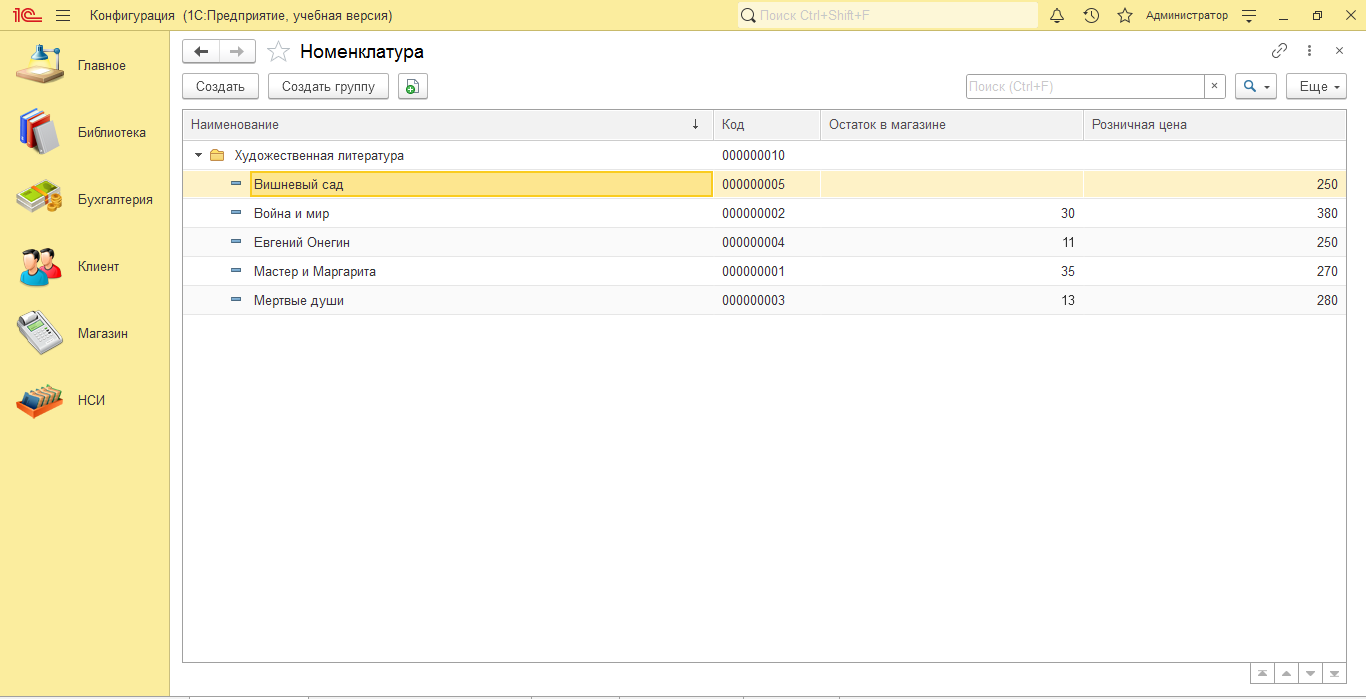


Рис. 12. Форма списка справочника «Номенклатура»

В справочник «Варианты номенклатуры» не добавлялись дополнительные реквизиты и табличные части. Формы списка и элемента остались стандартными. Содержат информацию о наименовании и коде элемента.

Справочник «Контрагенты» представляет собой список контрагентов, содержит информацию о наименовании, ИНН, КПП и виде контрагента. Реквизит «Вид контрагента» имеет тип перечисление.

Перечисления в 1С — это объект конфигурации, который используется для хранения в информационной базе фиксированного списка значений, не изменяющихся в процессе работы пользователей.

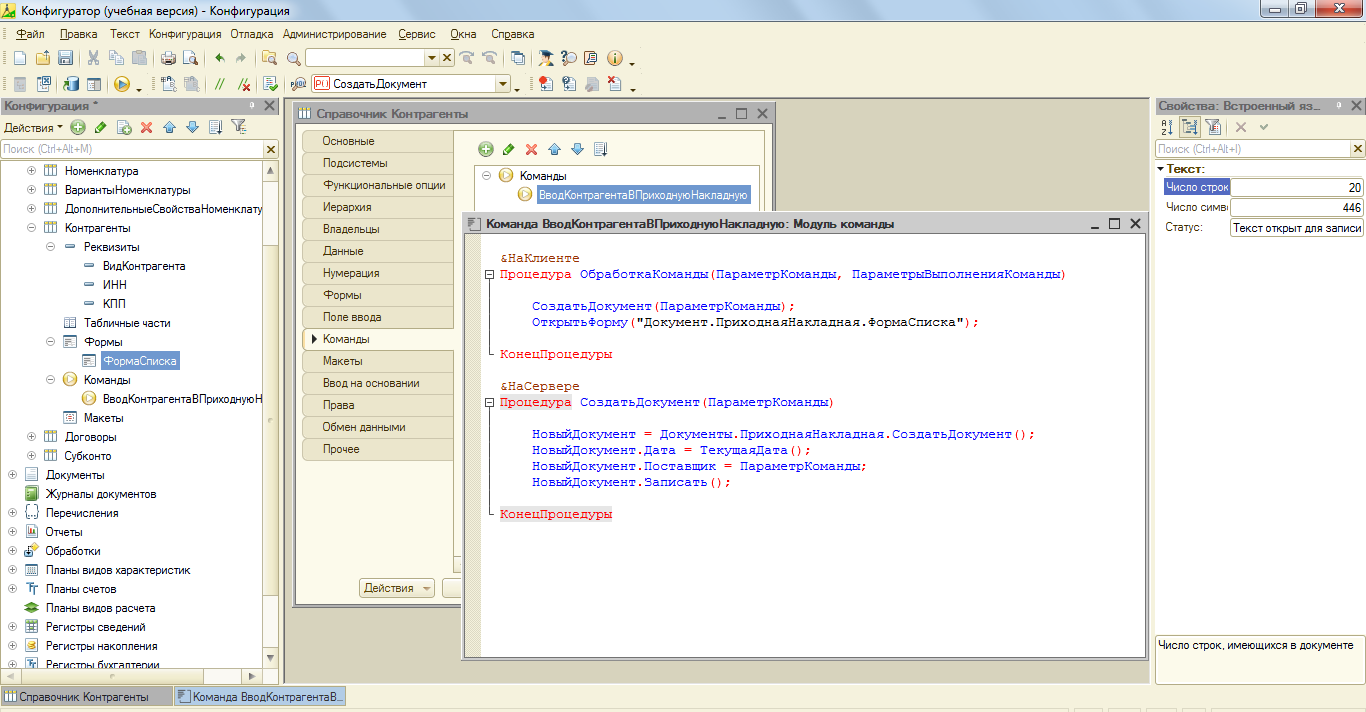
Форма элемента справочника «Контрагенты» не изменялась, осталась стандартной. Форма списка реализована в виде динамического списка, но в отличие от справочника «Номенклатура», для формирования динамического списка указан отображаемый объект конфигурации, а именно таблица справочника «Контрагенты».



В справочнике «Контрагенты» на форме списка и на форме элемента реализована параметризованная команда «Ввод контрагента в приходную накладную».

Использование параметризованных команд в формах объектов 1С позволяет при выполнении команды передать в обработчик команды какой-либо параметр, например, значение ссылочного реквизита.

В нашем случае при исполнении команды создается новый документ «Приходная накладная» с уже заполненным полем «Поставщик». Значение поставщика, т.е. ссылка на элемент справочника «Контрагенты» передается в качестве параметра команды.



Справочник «Договоры» содержит информацию о договорах с поставщиками. Данный справочник является подчиненным, его владелец – справочник «Контрагенты». Это означает, что каждый элемент справочника «Договоры» принадлежит определенному элементу справочника «Контрагенты». Отражает принадлежность реквизит «Владелец», данный реквизит обязателен к заполнению.

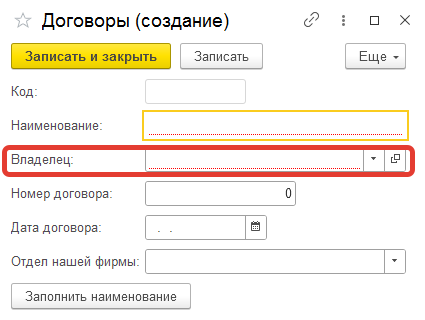
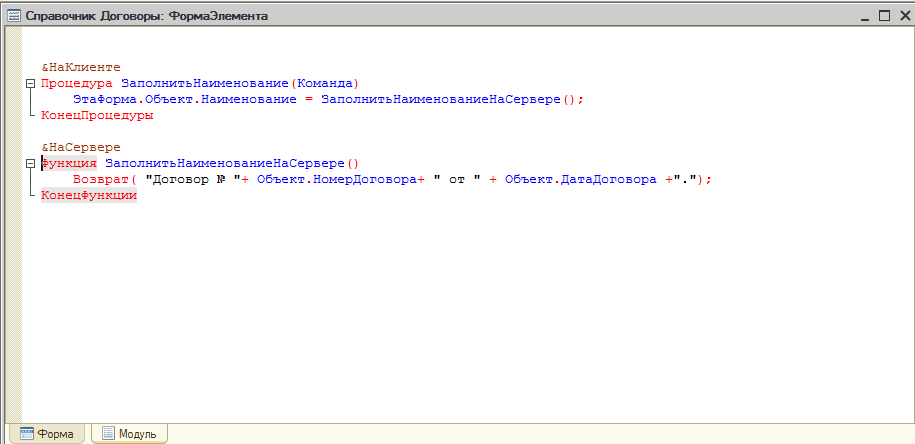


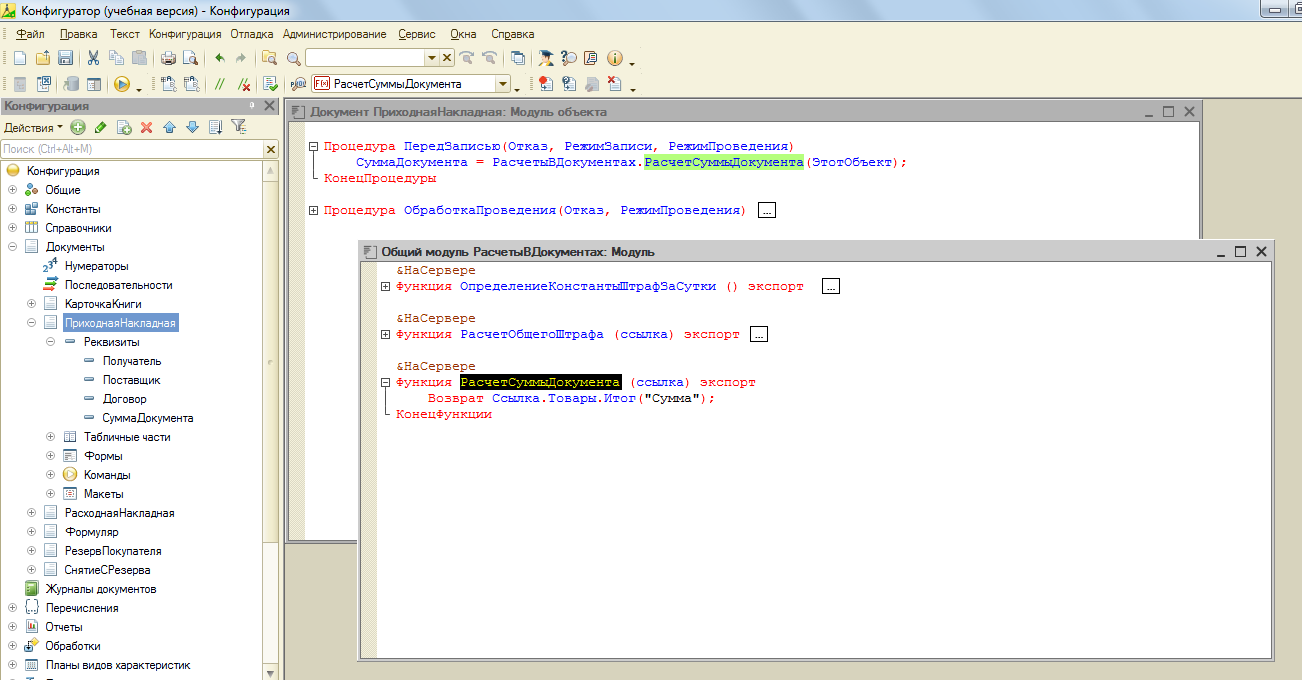
Рис. 13. Форма элемента справочника «Договоры»

В справочнике «Договоры» реализовано автозаполнение наименования. Автозаполнение происходит при нажатии кнопки «Заполнить наименование» на форме элемента.



Хозяйственными операциями, информацию о которых необходимо хранить и обрабатывать в подсистеме «Магазин», являются операции приходования и реализации товара. Эти операции в системе отражают документы «Приходная накладная» и «Расходная накладная». «Расходная накладная» может быть создана самостоятельно или на основании документа «Резерв покупателя», который также входит в подсистему «Магазин».

Документ «Приходная накладная» содержит стандартные реквизиты: «Номер документа», «Дата документа» и дополнительные реквизиты: «Получатель» (магазин или библиотека), «Поставщик», «Договор», «Сумма документа». Дополнительные реквизиты «Получатель», «Поставщик», «Договор» являются обязательными для заполнения. Реквизит «Сумма документа» является расчетным. Расчет производится в процедуре «ПередЗаписью», определенной в модуле объекта «Приходной накладной». В указанной процедуре вызывается функция общего модуля системы «РасчетСуммыДокумента».



Расчет ведется по табличной части документа «Товары», по полю сумма. Табличная часть содержит информацию о приходуемой номенклатуре, наборе свойств данной номенклатуры, цене, количестве и сумме.

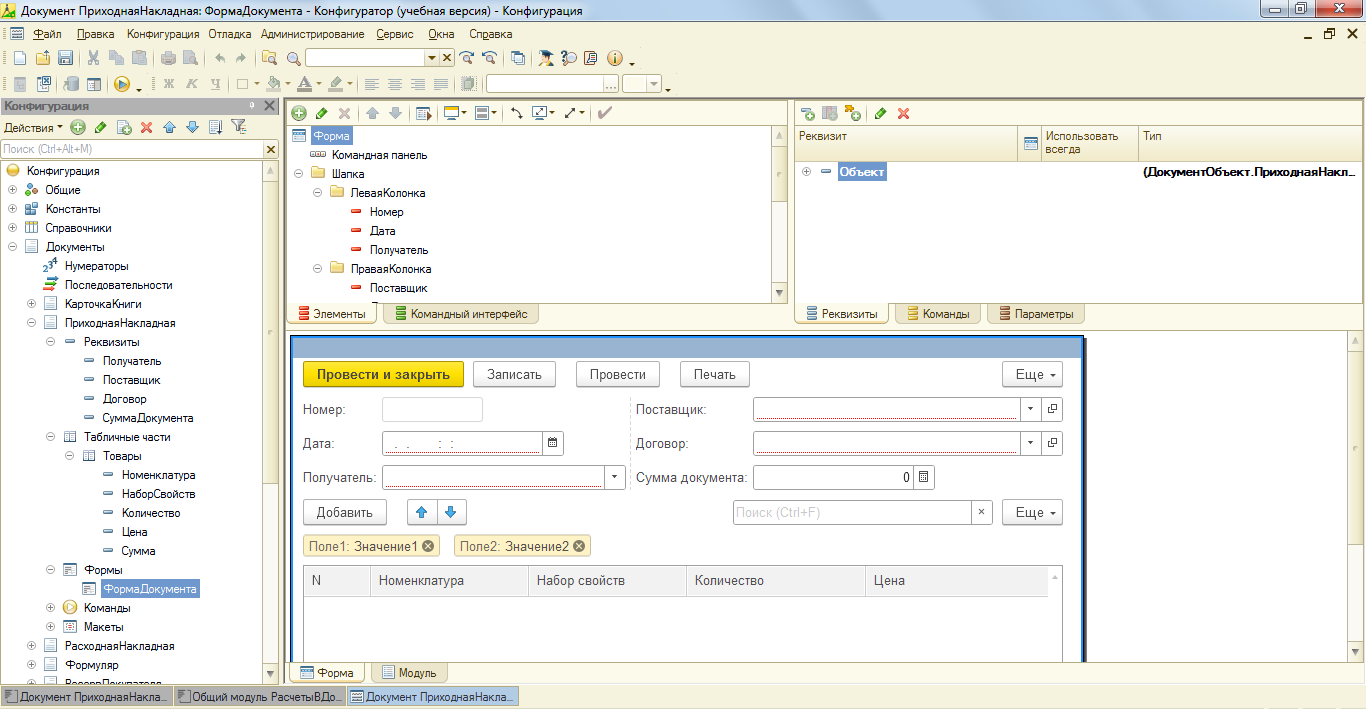
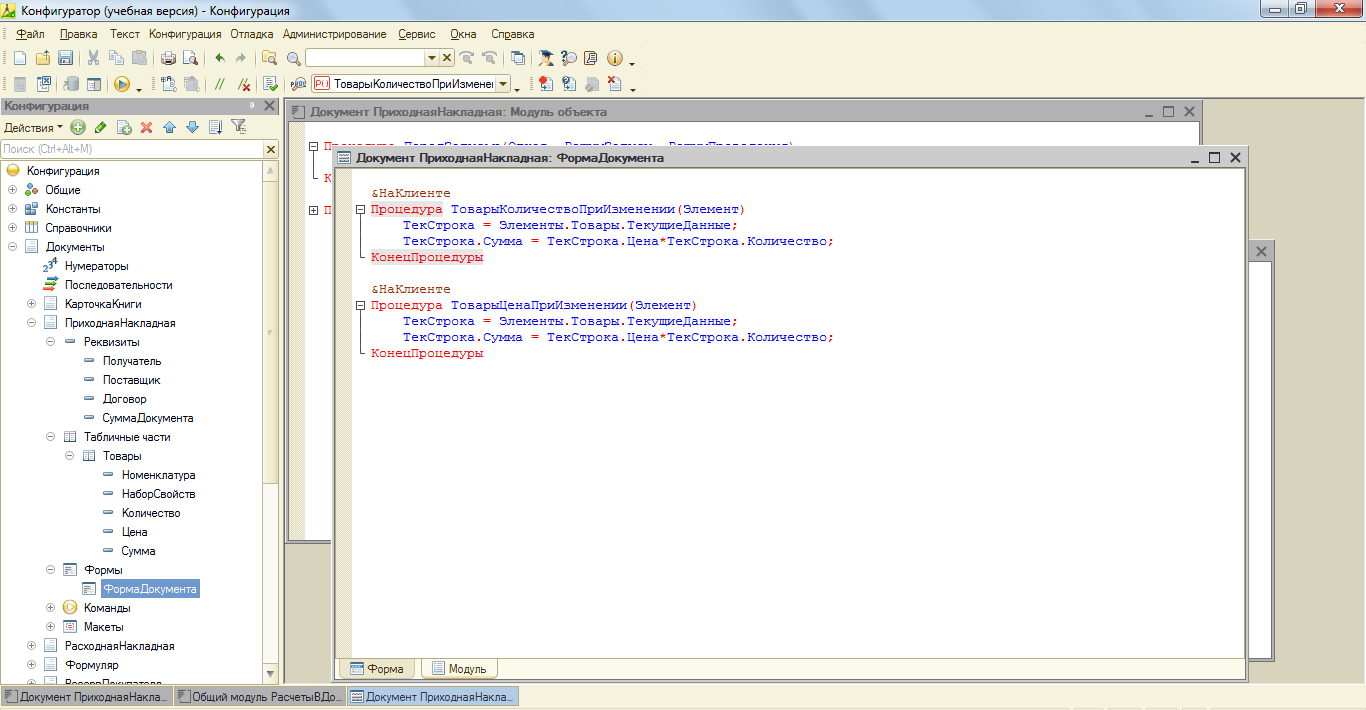
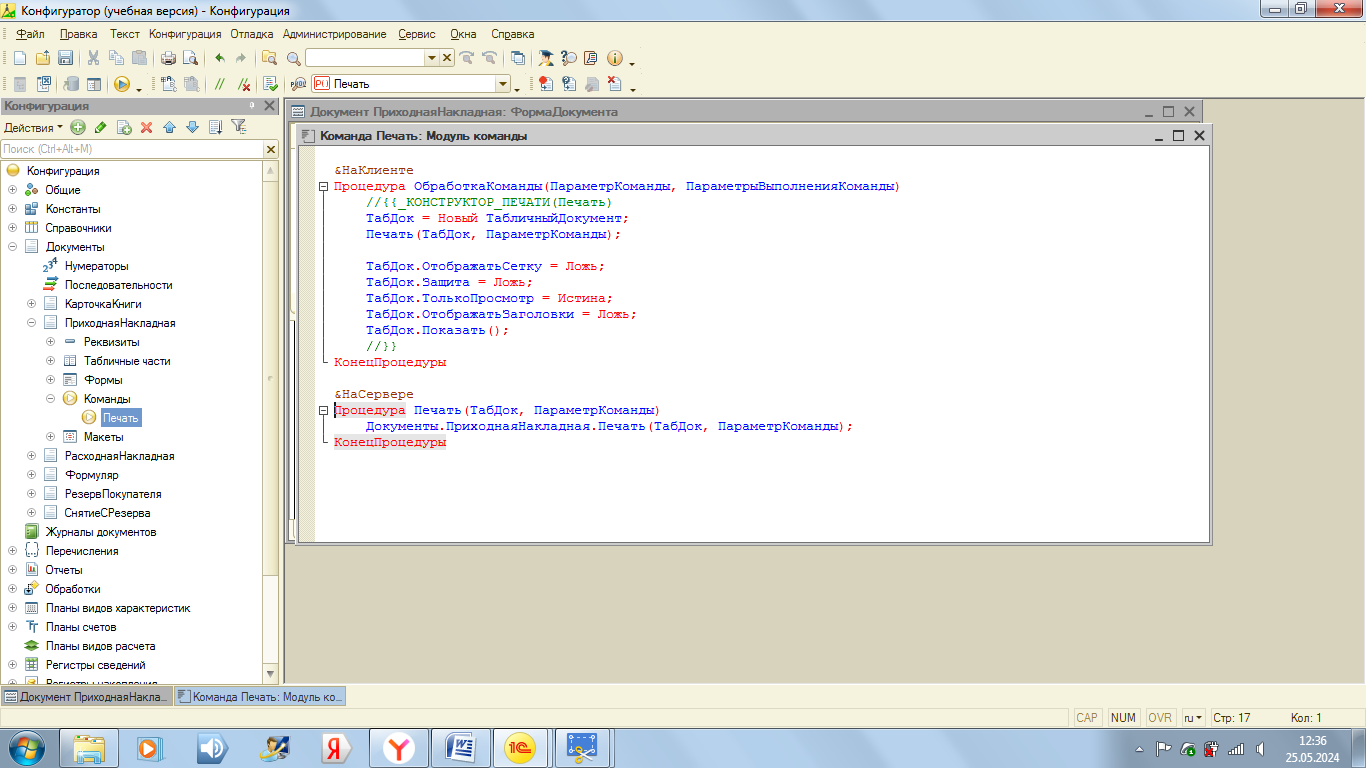


Рис. 14. Разработка формы документа «Приходная накладная»

В табличной части для полей «Количество» и «Цена» определены процедуры «ПриИзменении». Таким образом, при изменении цены или количества номенклатуры сумма рассчитывается автоматически.

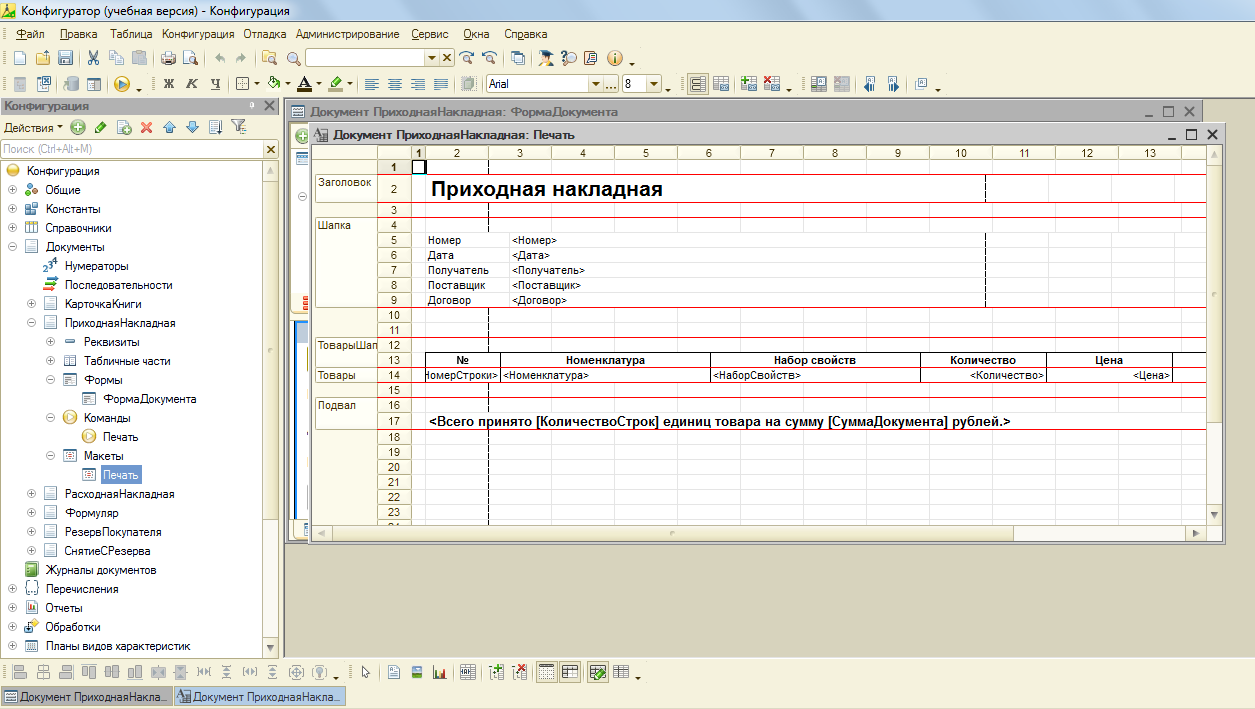


На форму элемента и на форму списка документа «Приходная накладная» добавлена кнопка «Печать». При нажатии на данную кнопку происходит вызов одноименной команды, при выполнении которой формируется печатная форма документа.

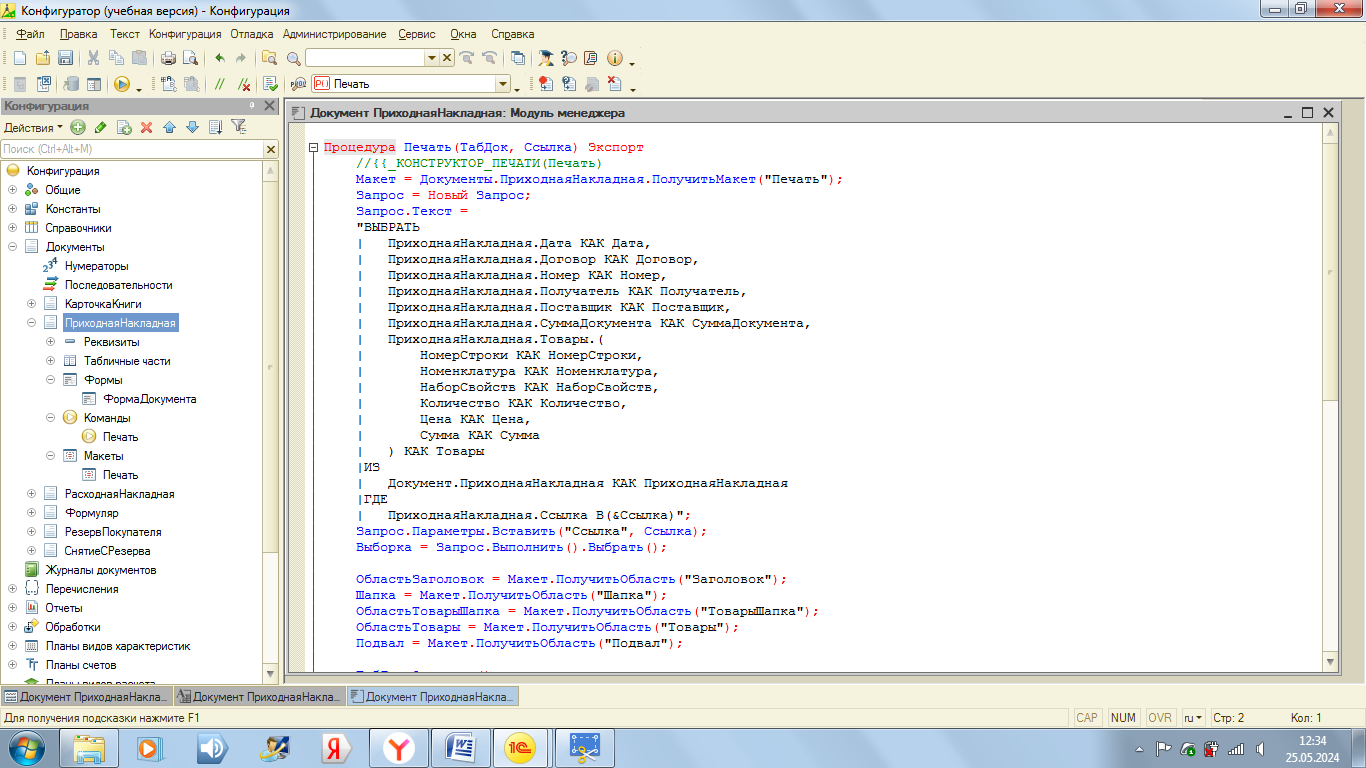


Внешний вид печатной формы задается в макете. Макет представляет собой шаблон для формирования печатной формы. Построение макета производится в конфигураторе в специальном табличном редакторе. Разработчик может создавать области, изменять высоту строк и ширину колонок, задавать форматирование ячеек и всего табличного документа, включать в макет рисунки, диаграммы и сводные таблицы.

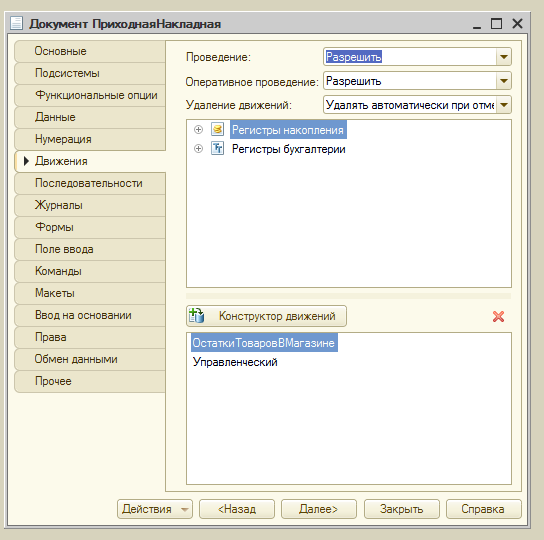
Области макета предназначены для однократного или многократного вывода.



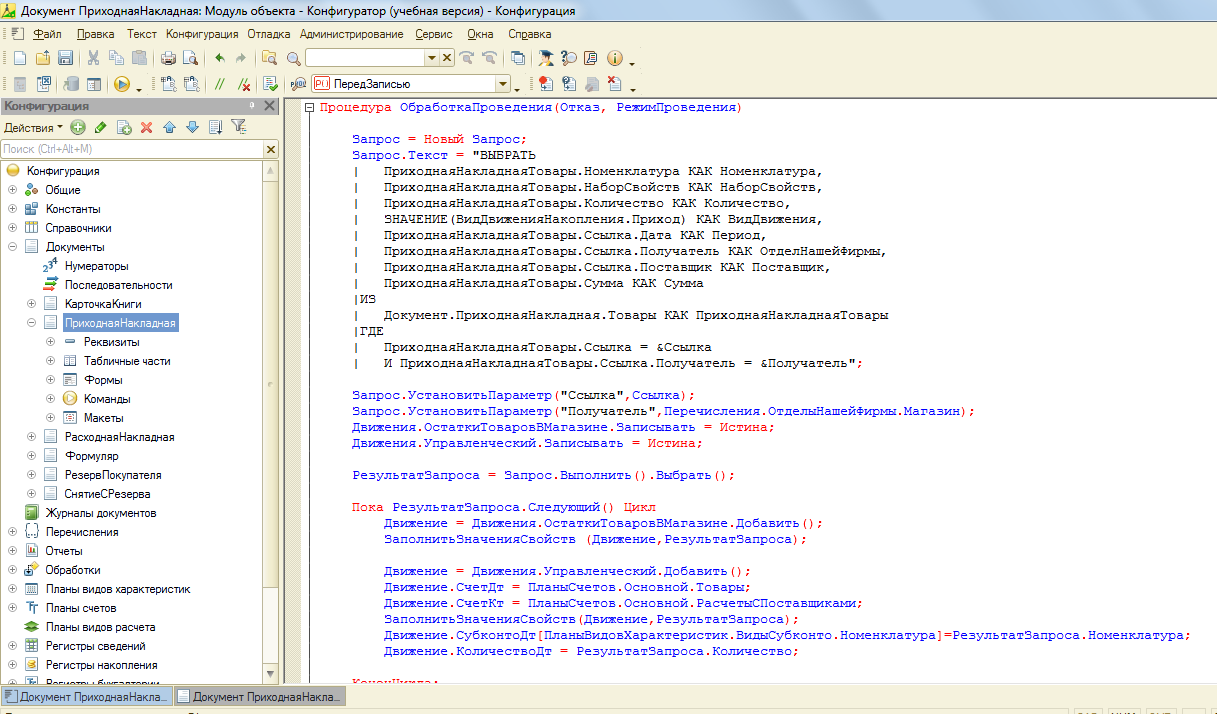
Некоторые ячейки содержат только текст, который должен быть выведен. Другие ячейки содержат имена параметров (в угловых скобках). Значения этих параметров указываются в программном модуле. Так в модуле менеджера документа «Приходная накладная» создана экспортируемая процедура «Печать», в которой с помощью запроса получены все необходимые данные для вывода в табличный документ и далее в программном коде задан процесс формирования печатной формы.



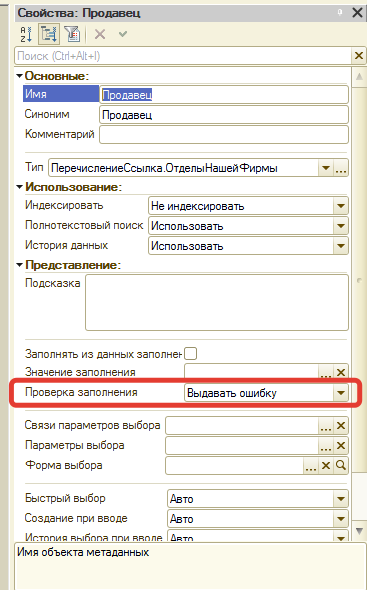
Документ «Приходная накладная» является регистратором для регистра накопления «Остатки товаров в магазине» и регистра бухгалтерии «Управленческий». Первоначальная настройка связей между документом и регистрами производится в окне свойств документа во вкладке «Движения». Там необходимо выбрать, для каких регистров данный документ будет являться регистратором.



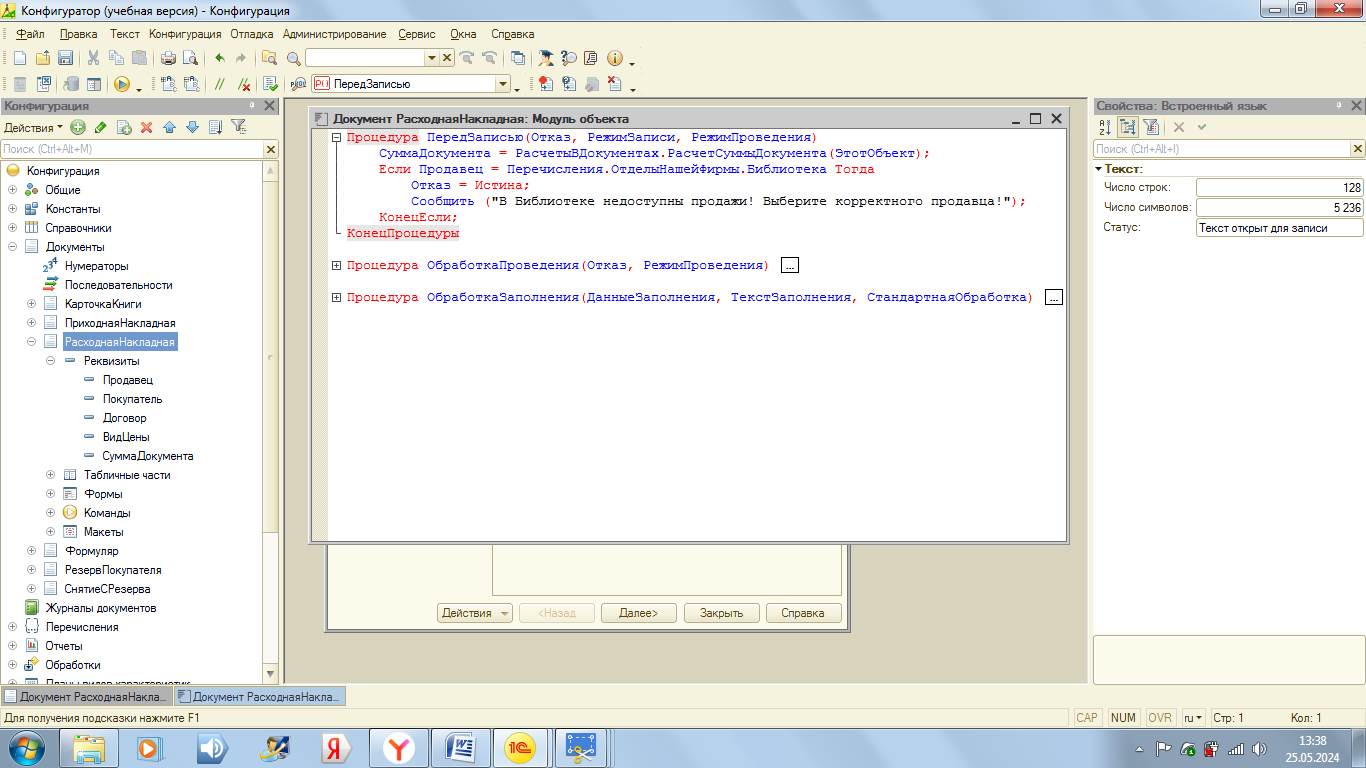
Логика записи движений в регистры описывается в процедуре модуля объекта «ОбработкаПроведения». Все необходимые данные получаются из документа с помощью запроса, а далее записываются в регистры. Использование запросов помогает существенно снизить ресурсоемкость программы, поскольку нет необходимости получать весь объект для получения значения необходимого реквизита.



Документ «Расходная накладная» содержит стандартные реквизиты: «Номер документа», «Дата документа» и дополнительные реквизиты: «Продавец», «Покупатель», «Договор», «Вид цены», «Сумма документа». Дополнительные реквизиты «Продавец», «Вид цены» являются обязательными для заполнения.



Для реквизита «Продавец» помимо проверки заполнения, настроена также проверка корректности значения. В случае некорректного заполнения, пользователь не сможет записать и провести документ и получит сообщение об ошибке.



Реквизиты «Покупатель» и «Договор» заполняются, в случае если происходит продажа по мелкооптовой цене, для розничной цены эти реквизиты могут оставаться пустыми. Реквизит «Сумма документа» является расчетным. Расчет производится аналогично документу «Приходная накладная».

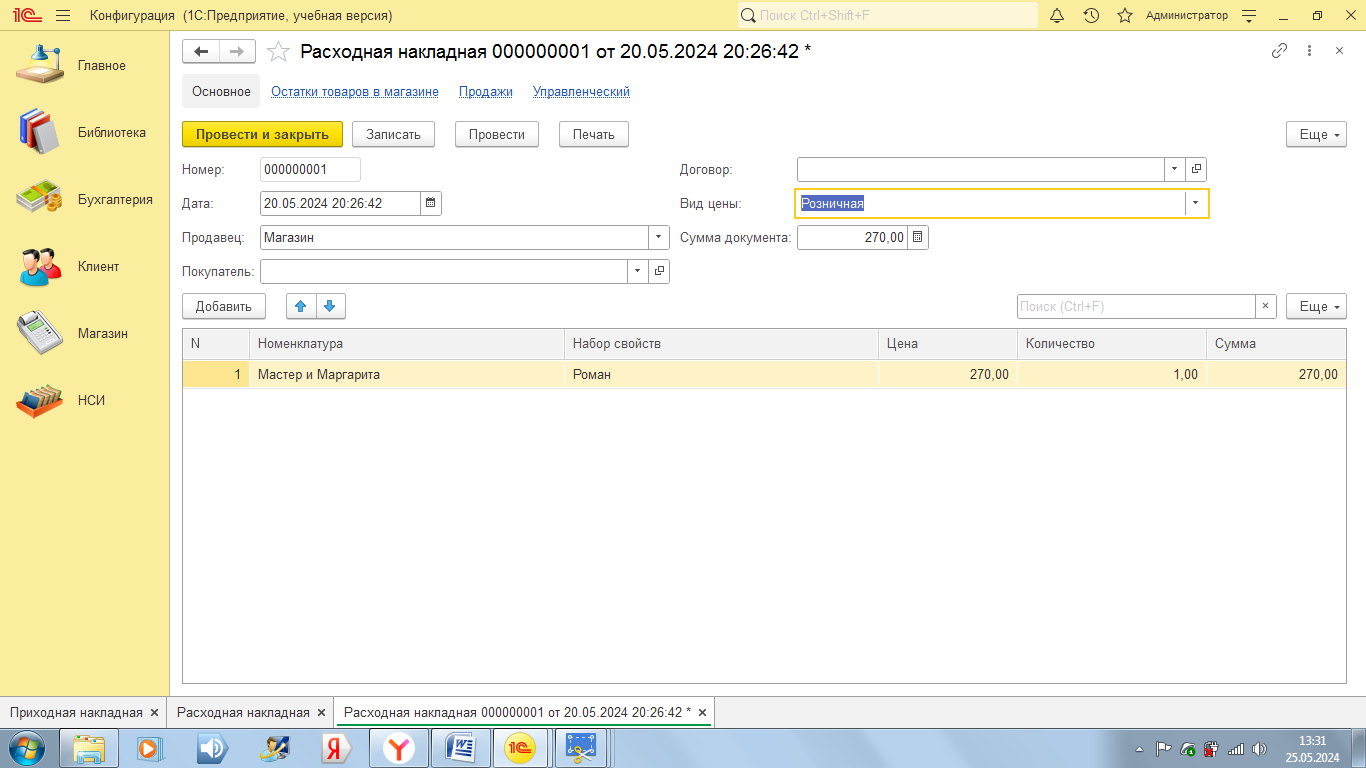
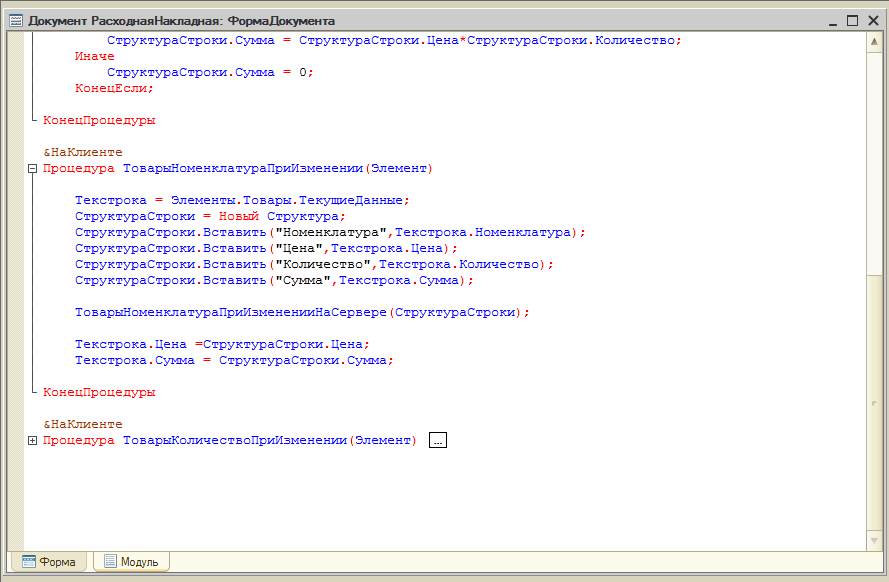


Рис.15. Внешний вид документа «Расходная накладная»

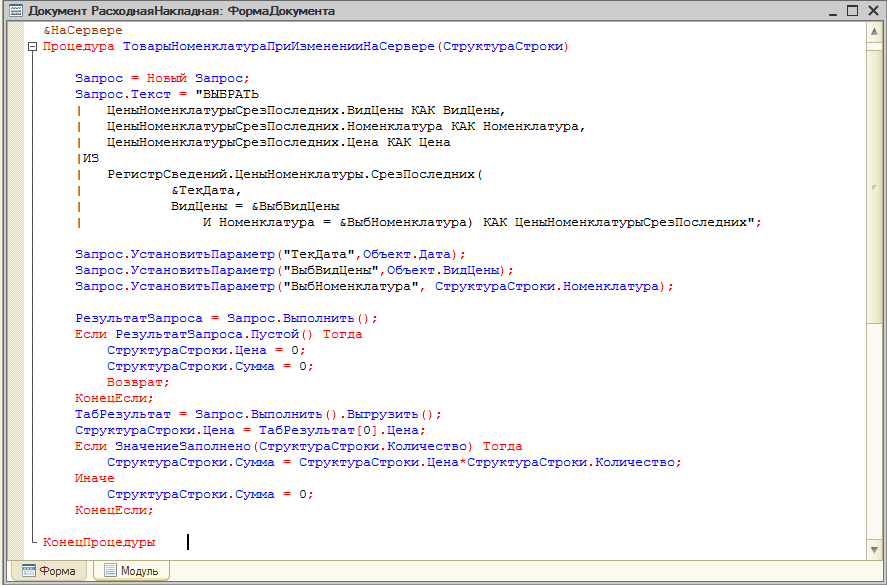
Табличная часть содержит информацию о продаваемой номенклатуре, наборе свойств данной номенклатуры, цене, количестве и сумме. Цена заполняется автоматически из регистра сведений «Цены номенклатуры» в зависимости от выбранного в документе вида цены и выбранной номенклатуры.

Логика автоматического заполнения цены прописана в модуле формы документа. В клиентской процедуре «ТоварыНоменклатураПриИзменении» из текущих данных строки, в которой изменилось значение номенклатуры, формируется структура строки, которая передается в серверную процедуру «ТоварыНоменклатураПриИзмененииНаСервере».



В серверной процедуре формируется запрос к регистру сведений «Цены номенклатуры», в качестве параметров устанавливается дата документа, выбранный вид цены и выбранная номенклатура. Дату документа необходимо учитывать поскольку из регистра сведений берется «Срез последних» на указанную дату.

Таким образом, в соответствии с заданными параметрами, мы получаем из регистра сведений значение цены. Далее это значение вставляется в структуру строки. В случае если «Количество» заполнено, сразу происходит расчет «Суммы» по строке.



Далее в клиентской процедуре обновленные данные из структуры строки записываются в текущую строку документа.

В документе «Расходная накладная» аналогично «Приходной накладной» реализована команда «Печать» для формирования печатной формы документа.

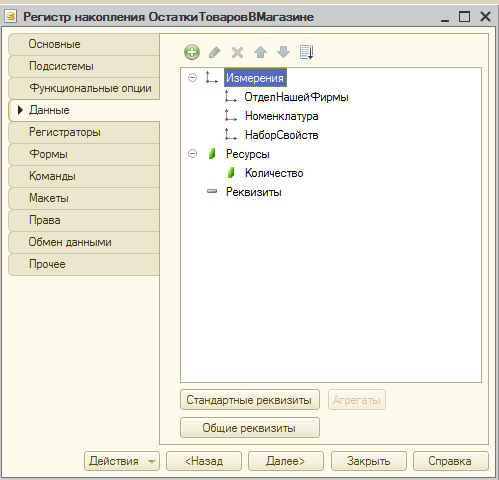
«Расходная накладная» является регистратором для регистров накопления «Остатки товаров в магазине», «Продажи» и регистра бухгалтерии «Управленческий». Запись движений в регистры настроена аналогично документу «Приходная накладная».

Процесс и логика создания «Расходной накладной» на основании документа «Резерв покупателя» будут описаны в разделе 3.3. Разработка подсистемы «Клиент».

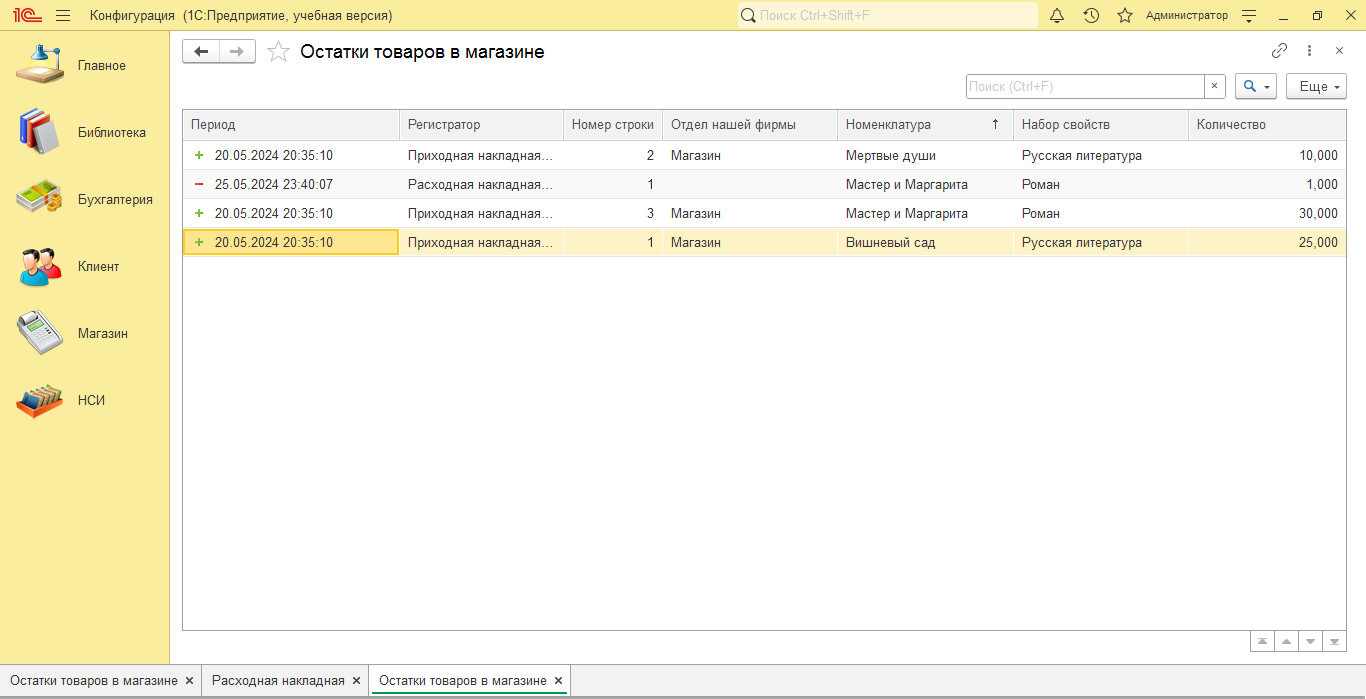
Со стороны клиента регистры накопления не доступны ни в одной из подсистем учетной системы, но формально все регистры накопления принадлежат подсистеме «Магазин».

Существует два вида регистров накопления: регистры накопления остатков и регистры накопления оборотов. Регистр накопления остатков позволяет хранить как итоговые значения ресурсов — остатки, так и изменения этих ресурсов — обороты.

Регистр накопления «Остатки товаров в магазине» имеет вид «Остатки» позволяет накапливать числовые данные о количестве единиц номенклатуры в разрезе нескольких измерений: номенклатура, набор свойств, отдел нашей фирмы.

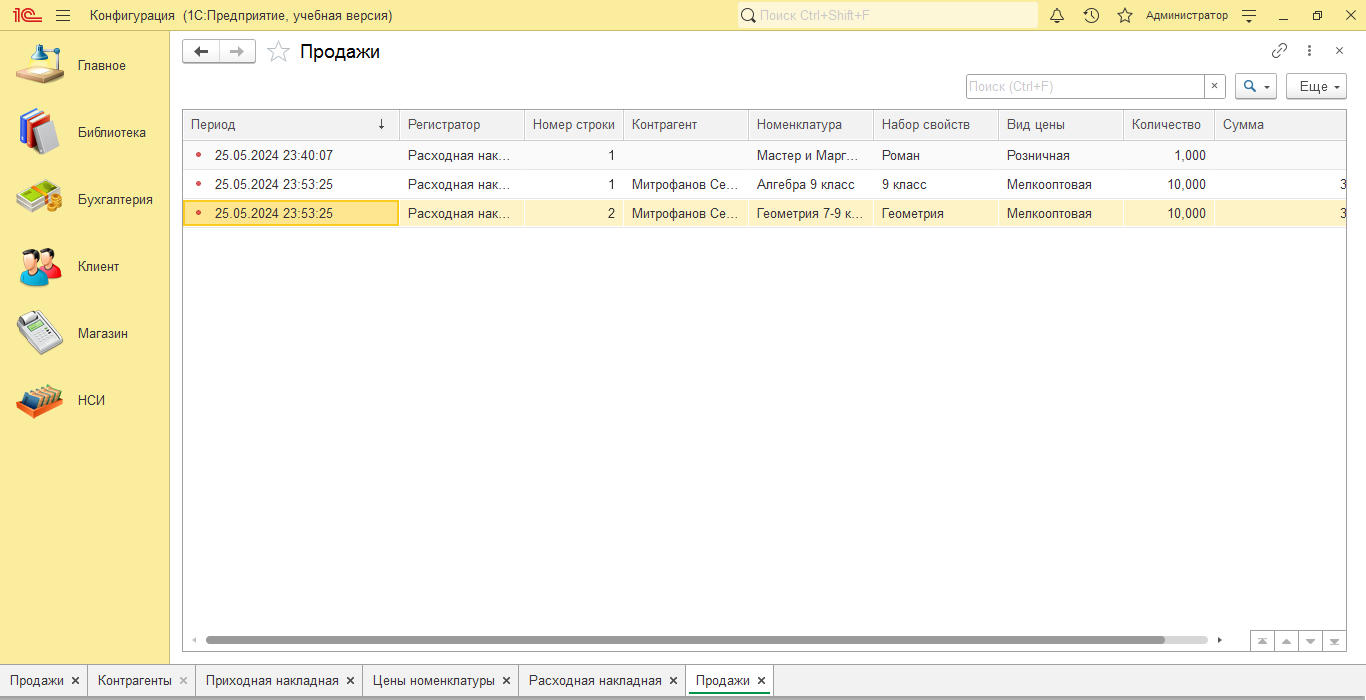


Данные с видом движения накопления «Приход», то есть со знаком «+» записываются в регистр при проведении документа «Приходная накладная». Данные с видом движения накопления «Расход», то есть со знаком «-» записываются в регистр при проведении документа «Расходная накладная».



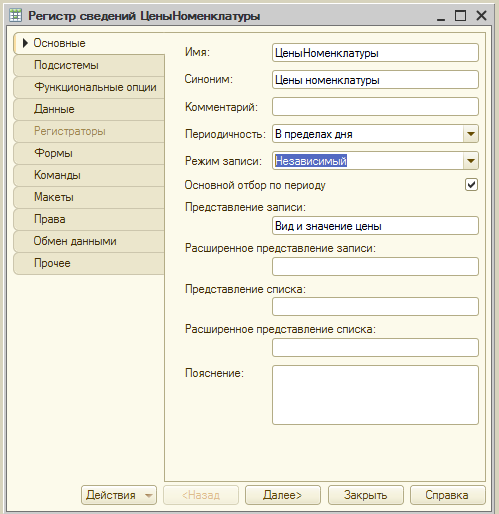
Регистр накопления «Продажи» имеет вид «Обороты» хранит данные о количестве номенклатуры и количестве денежных средств в разрезе измерений: контрагент, номенклатура, набор свойств, вид цены.

Поскольку регистр накопления оборотов не накапливает остатки ресурсов, для него не имеет смысла «направление» движения ресурсов (приход или расход); накапливается только величина изменения ресурсов. Поэтому все записи регистра накопления оборотов отмечены одинаковыми пиктограммами.



Регистр накопления «Резерв покупателя» формально относится к подсистеме «Магазин», но будет описан в разделе 3.3. Разработка подсистемы «Клиент».

Регистр сведений «Цены номенклатуры» хранит информацию о ценах номенклатуры в разрезе номенклатуры и вида цены. Регистр является периодическим в пределах дня, это означает, что к каждой записи регистра будет добавляться поле Период, хранящее дату, которой были внесены записи в регистр. Использование периодичности регистра сведений позволяет не просто хранить статические данные, но и отслеживать их изменение во времени.



Регистр сведений «Цены номенклатуры» является независимым, это означает, что записи формируются не документом-регистратором, а пользователем с необходимыми правами вручную.

Система обеспечивает контроль уникальности записей, хранящихся в регистре сведений. Таким образом, в регистре сведений не может находиться двух одинаковых записей. Одинаковыми считаются записи, у которых совпадает ключ записи. Ключ записи формируется системой автоматически, на основании значений, содержащихся в полях записи, и зависит от вида регистра сведений.

В нашем случае в формировании ключа записи будут участвовать значение периода и значения измерений. Таким образом, в нашем периодическом регистре сведений «Цены номенклатуры» с независимым режимом записи не может существовать двух записей о розничной цене одной и той же книги, внесенных одной и той же датой.

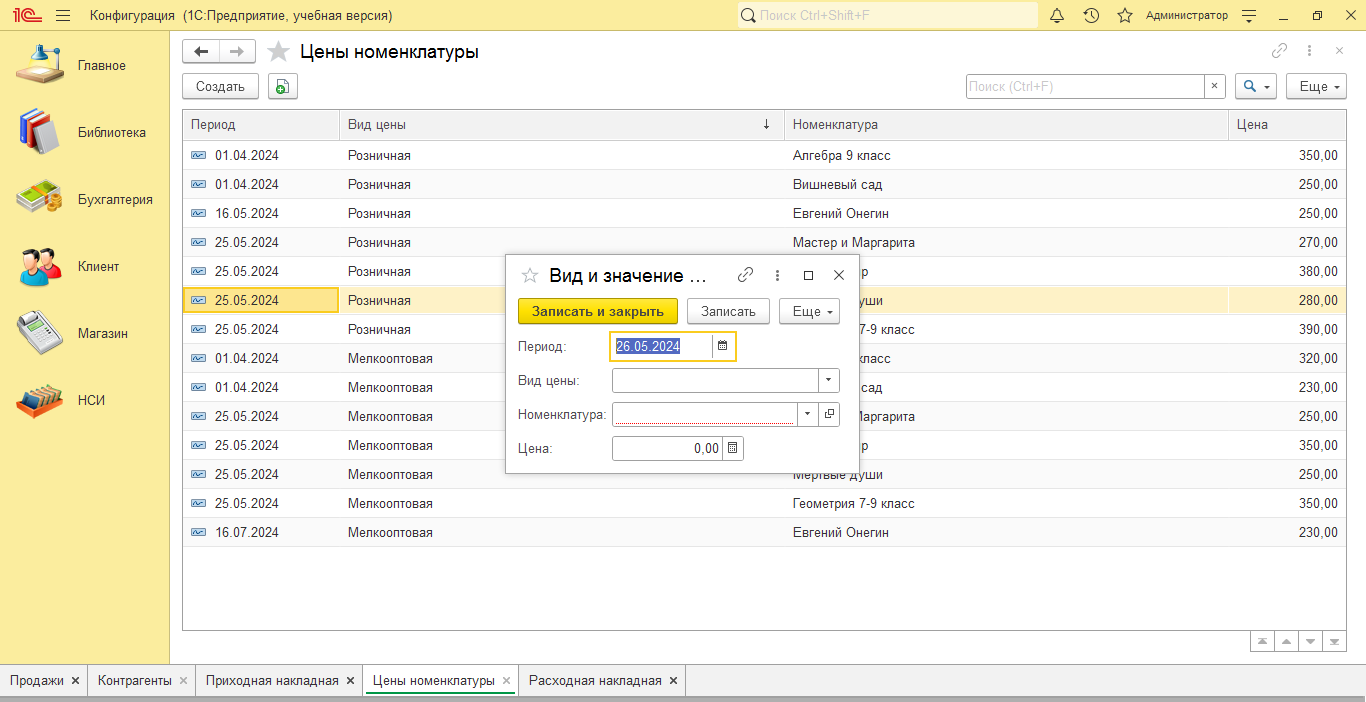
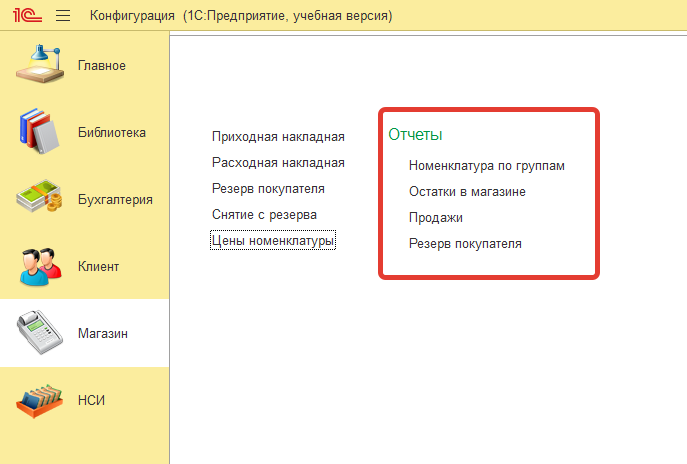


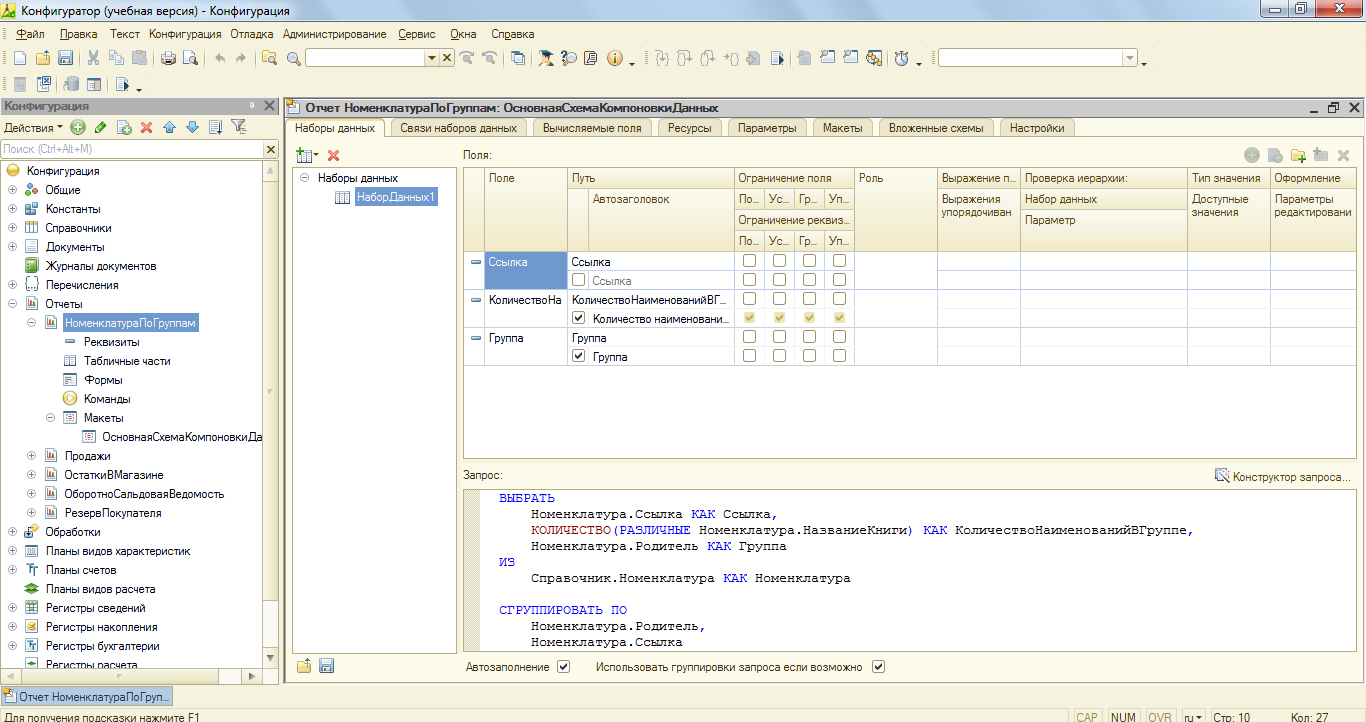
Рис. 16. Запись нового значения в регистр сведений «Цены номенклатуры»

В подсистеме «Магазин» для анализа хозяйственной деятельности и принятия управленческих решений реализован ряд отчетов.



Все отчеты в учетной системе созданы с помощью Системы компоновки данных (СКД).

Отчет «Номенклатура по группам» позволяет получить информацию о том, какое количество наименований книг в каждой группе номенклатуры.



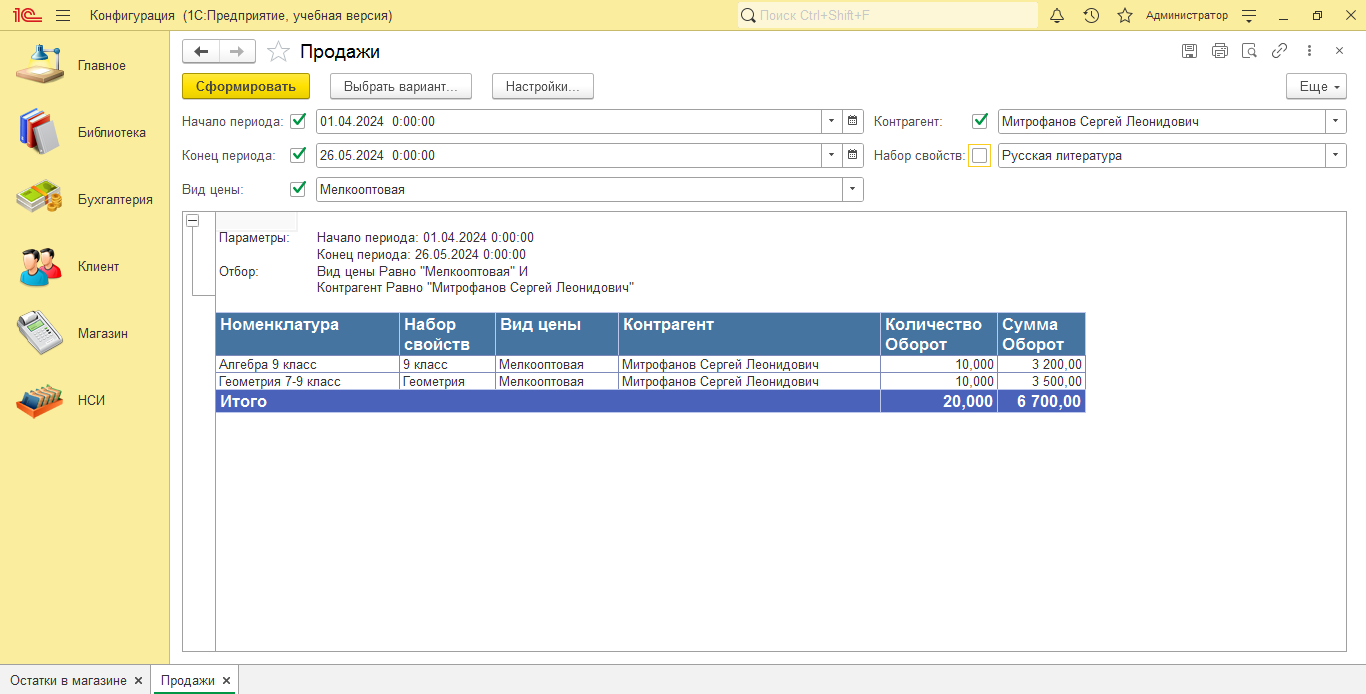
С помощью запроса получаются все необходимые данные, основная схема компоновки данных определяет внешний вид отчета.



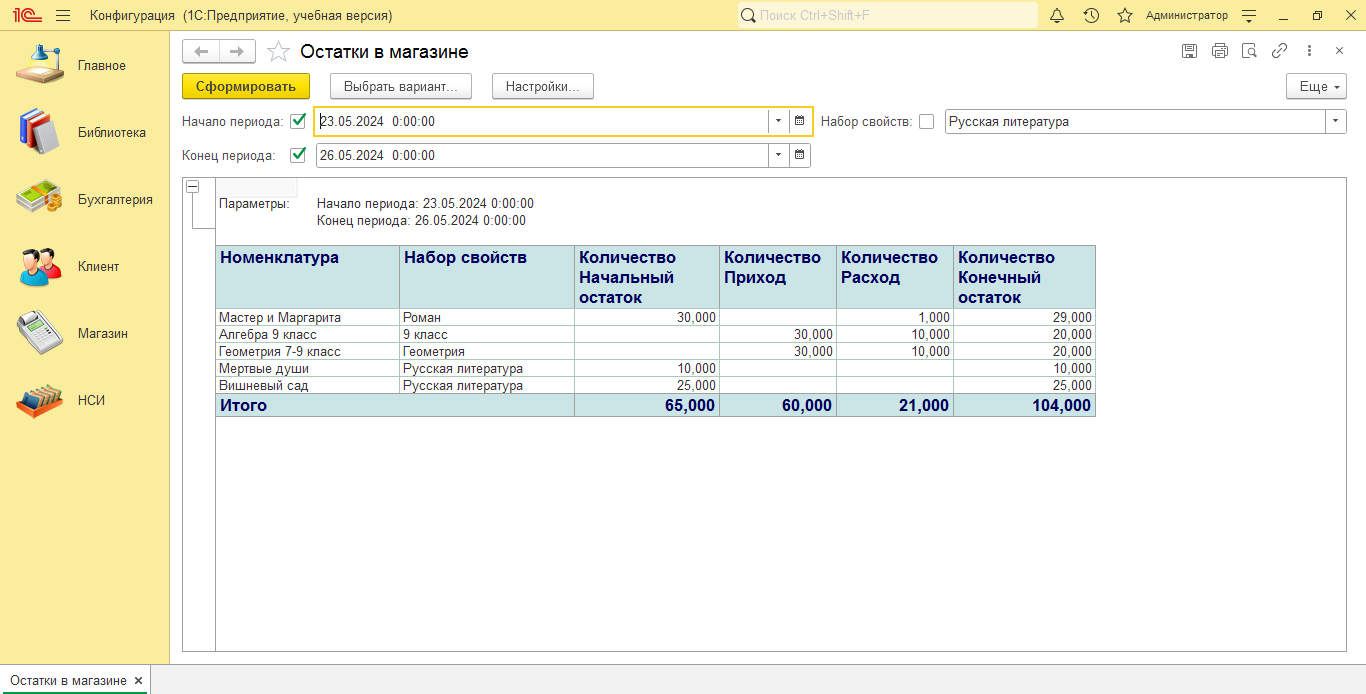
Отчет «Продажи» позволяет получить информацию о количестве проданной номенклатуры за определенный период. Также с помощью основной схемы компоновки данных были добавлены дополнительные поля параметров отбора для пользователей.



Таким образом, пользователь может посмотреть объем всех продаж за период, либо продаж только по определенному виду цены, определенному контрагенту или номенклатуры с определенным набором свойств.



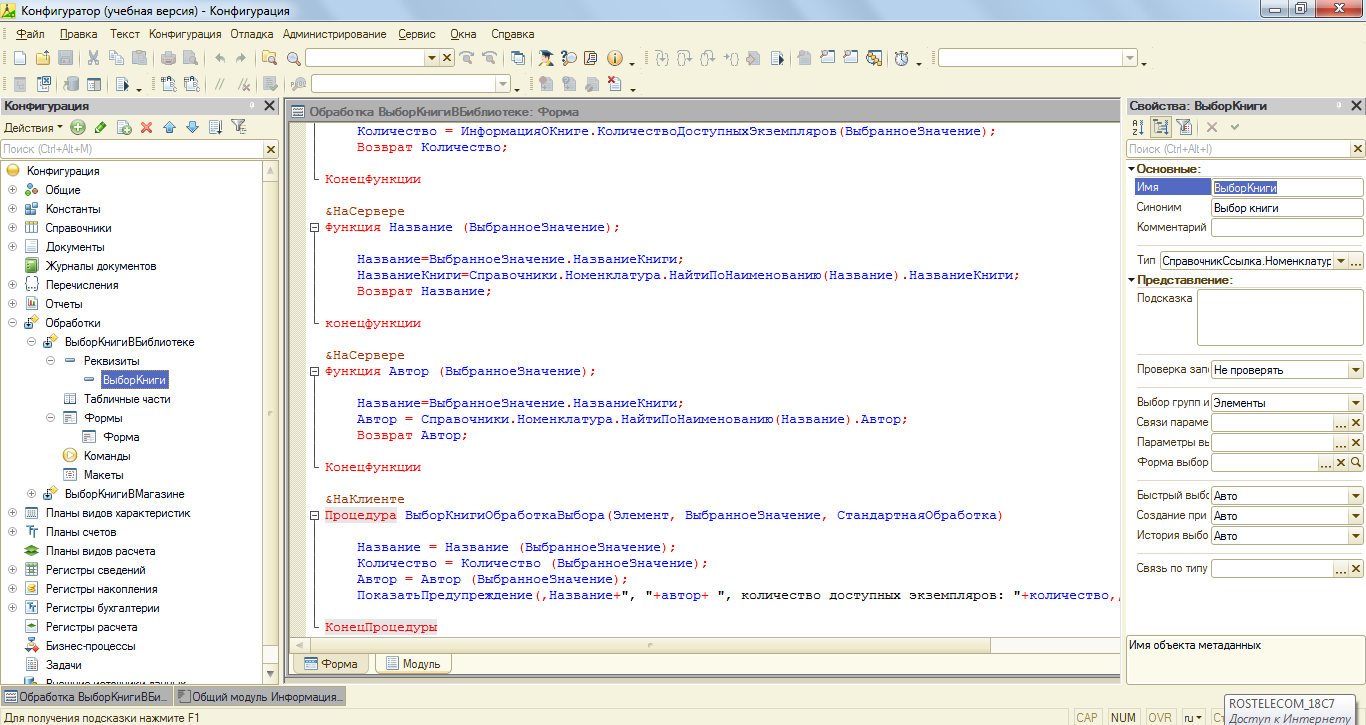
Отчет «Остатки в магазине» отражает информацию о движении и конечном остатке номенклатуры в магазине за определенный период. В качестве дополнительного параметра отбора был добавлен «Набор свойств».



**3.3. Разработка подсистемы «Клиент»**

Предполагается, что учетная система будет установлена на стационарных терминалах в магазине и библиотеке для использования клиентами. Для посетителей магазина и библиотеки будет доступна только подсистема «Клиент».

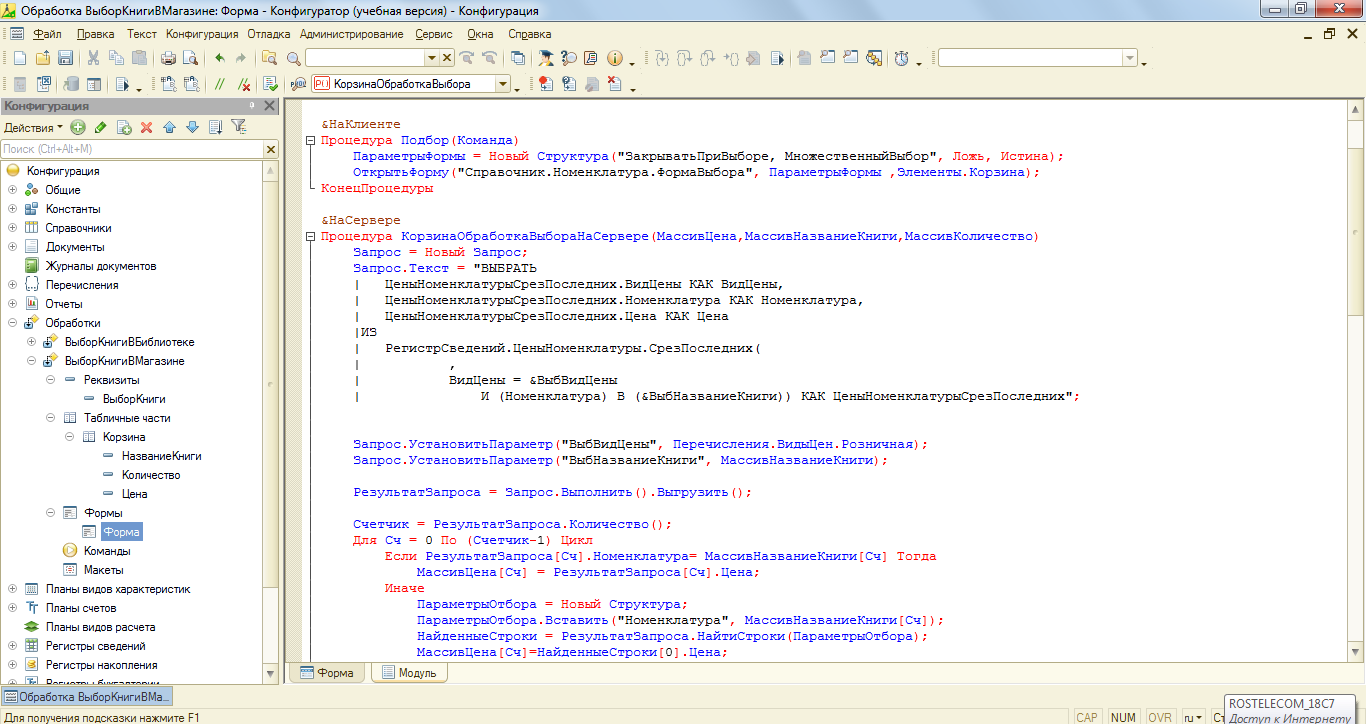
Для посетителей библиотеки подсистема выполняет только информационную функцию. Получить информацию о выбранной книге позволяет реализованная обработка «Выбор книги в библиотеке». Реквизит обработки «ВыборКниги» имеет тип Справочник.Ссылка.Номенклатура. Реквизиты выбранного элемента передаются в качестве параметров в серверные функции, которые, в свою очередь, обращаются к другим объектам системы и возвращают всю необходимую информацию о книге.

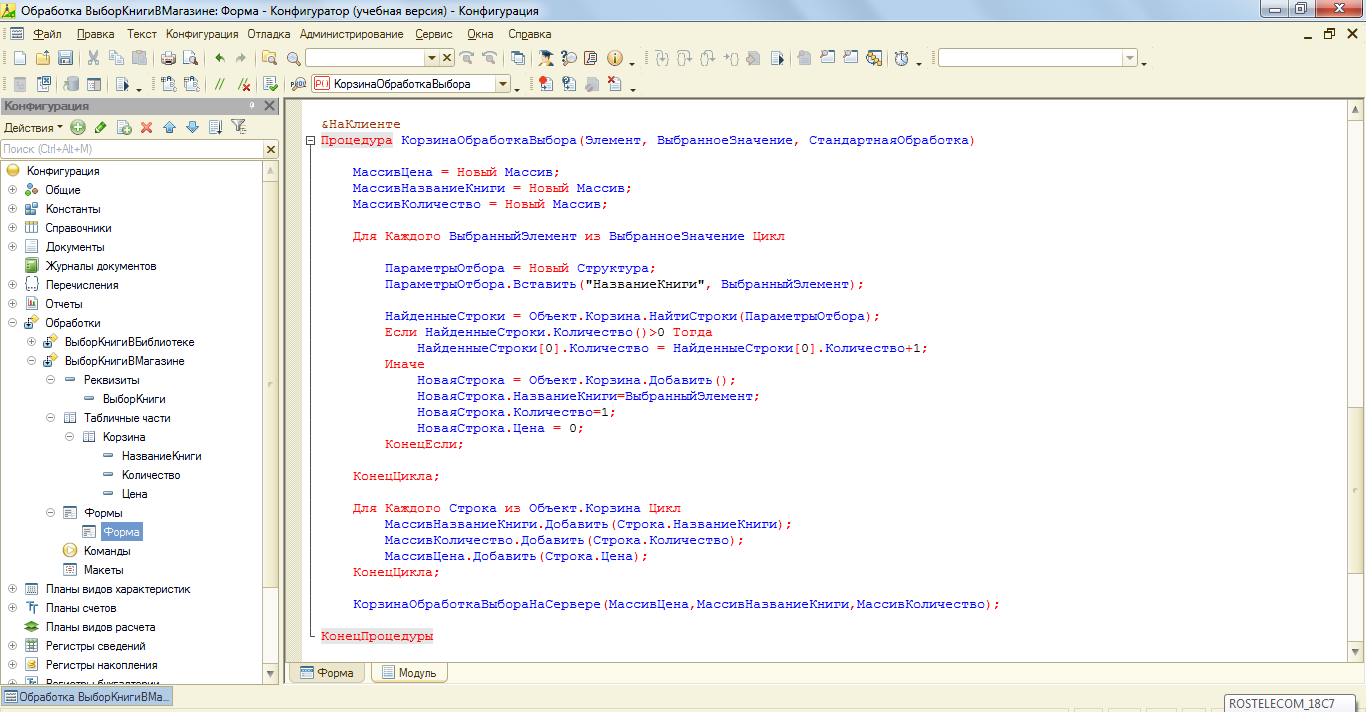


Таким образом, при выборе книги в окне выбора на экране появляется сообщение со всей необходимой информацией о данной книге.



Обработка «Выбор книги в магазине» имеет гораздо более широкий функционал. Кроме получения информации, для клиентов магазина реализована возможность самостоятельного резервирования товара. Для этого в обработку «Выбор книги в магазине» добавлена табличная часть «Корзина», которая заполняется при нажатии кнопки «Добавить в корзину». Для более удобного выбора товара реализован множественный выбор, также при многократном выборе одной позиции в корзину не добавляются новые строки, а изменяется количество. Корзина доступна для розничного покупателя, поэтому розничная цена подтягивается автоматически.





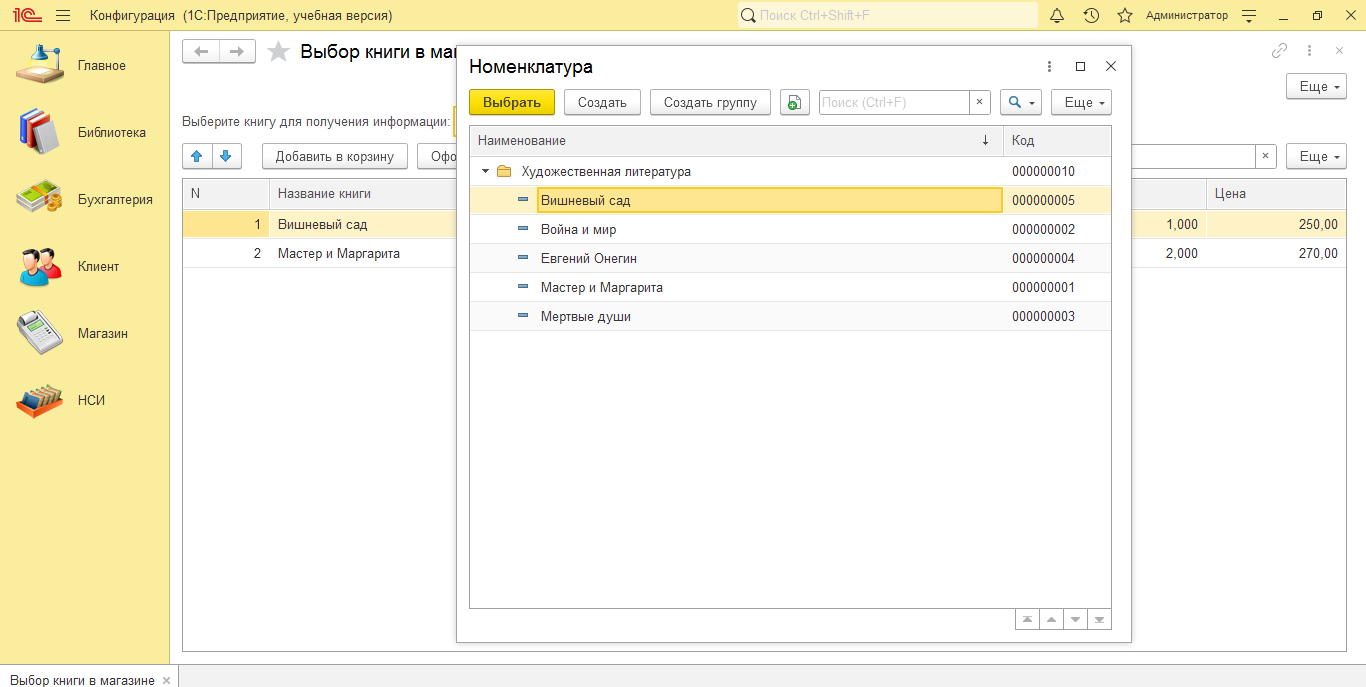
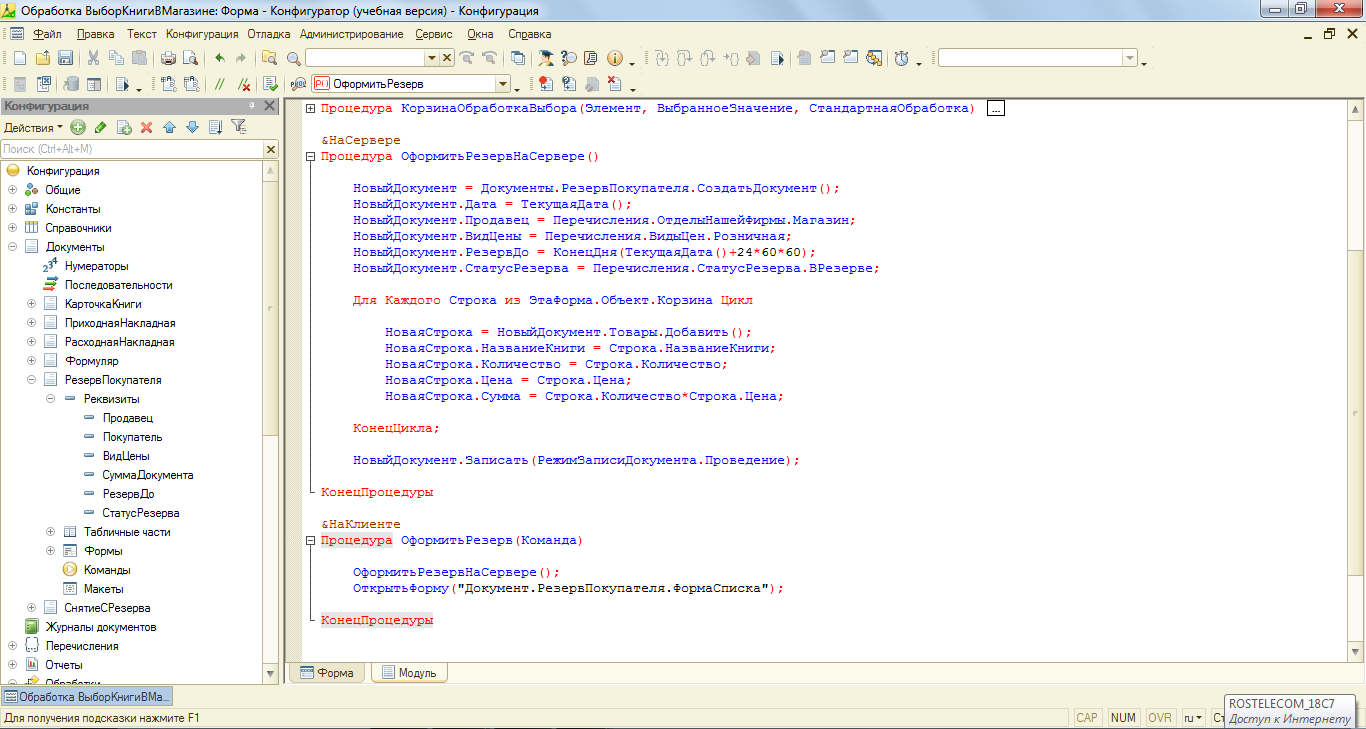
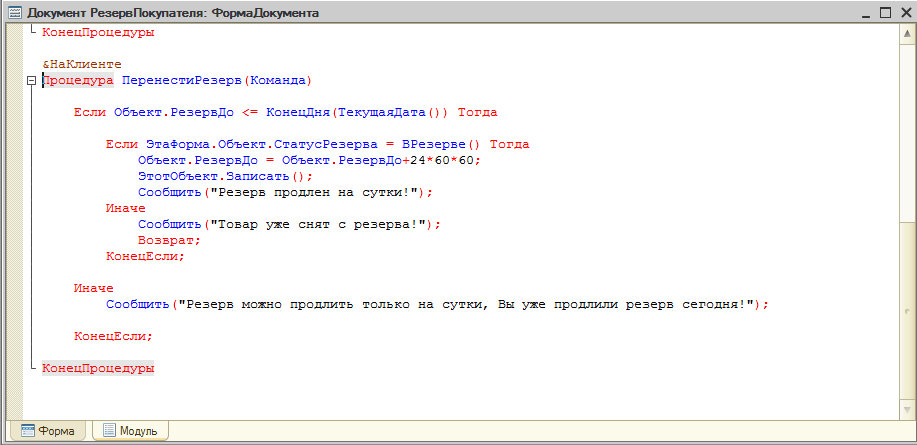


Рис.17. Добавление товаров в «Корзину»

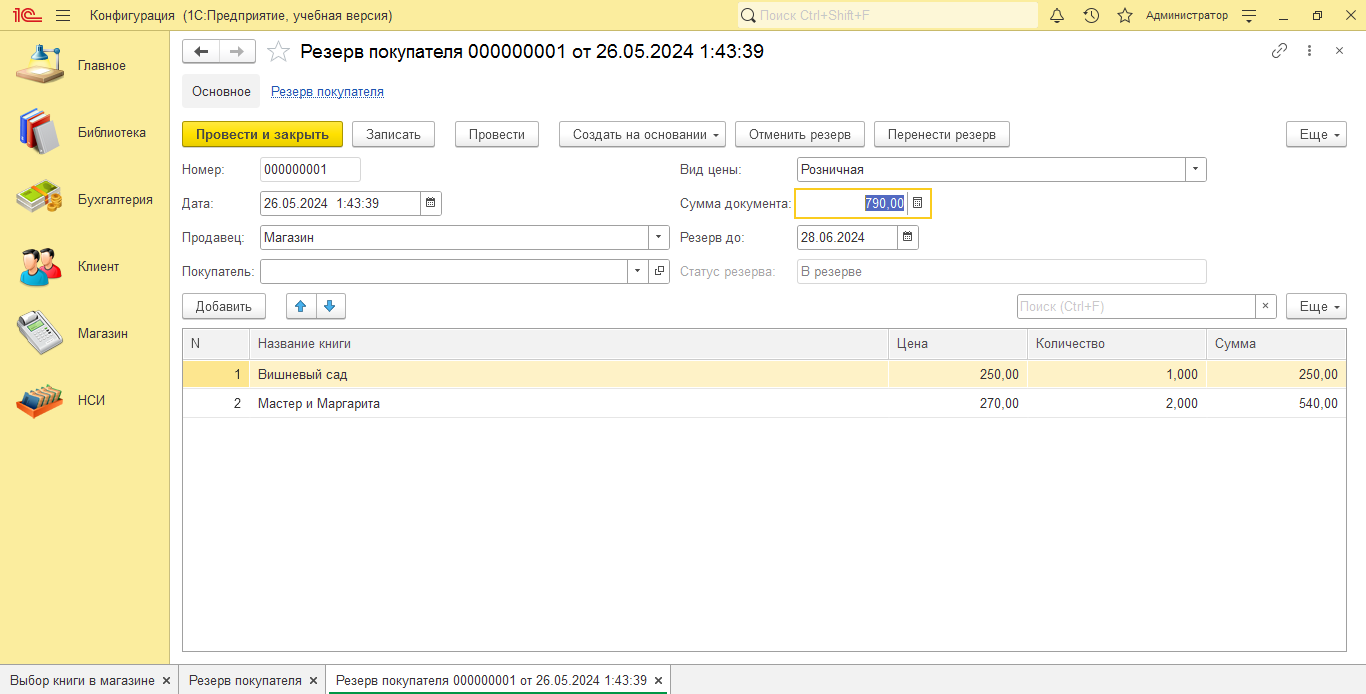
После того, как клиент добавил все необходимые товары в «Корзину», он может самостоятельно оформить резерв. Этот функционал реализован следующим образом: при нажатии на кнопку «Оформить резерв» создается новый документ «Резерв покупателя» и все товары из «Корзины» переносятся в этот документ.



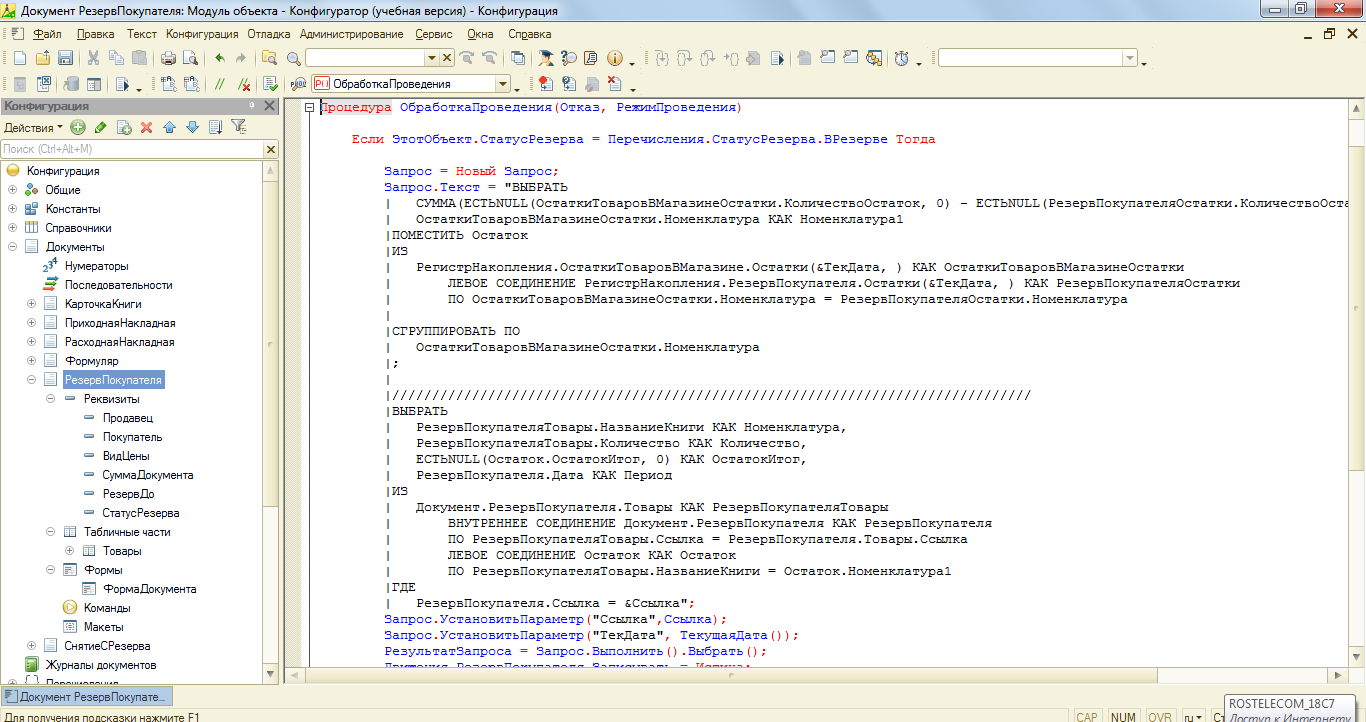
По умолчанию оформить резерв можно только на сутки от текущей даты, поэтому в значение реквизита «Резерв до» устанавливается дата, равная текущей дате + 24 часа. По кнопке перенести резерв можно изменить дату реквизита «Резерв до» на сутки вперед. Реализована проверка, которая не позволяет переносить резерв чаще, чем раз в сутки.



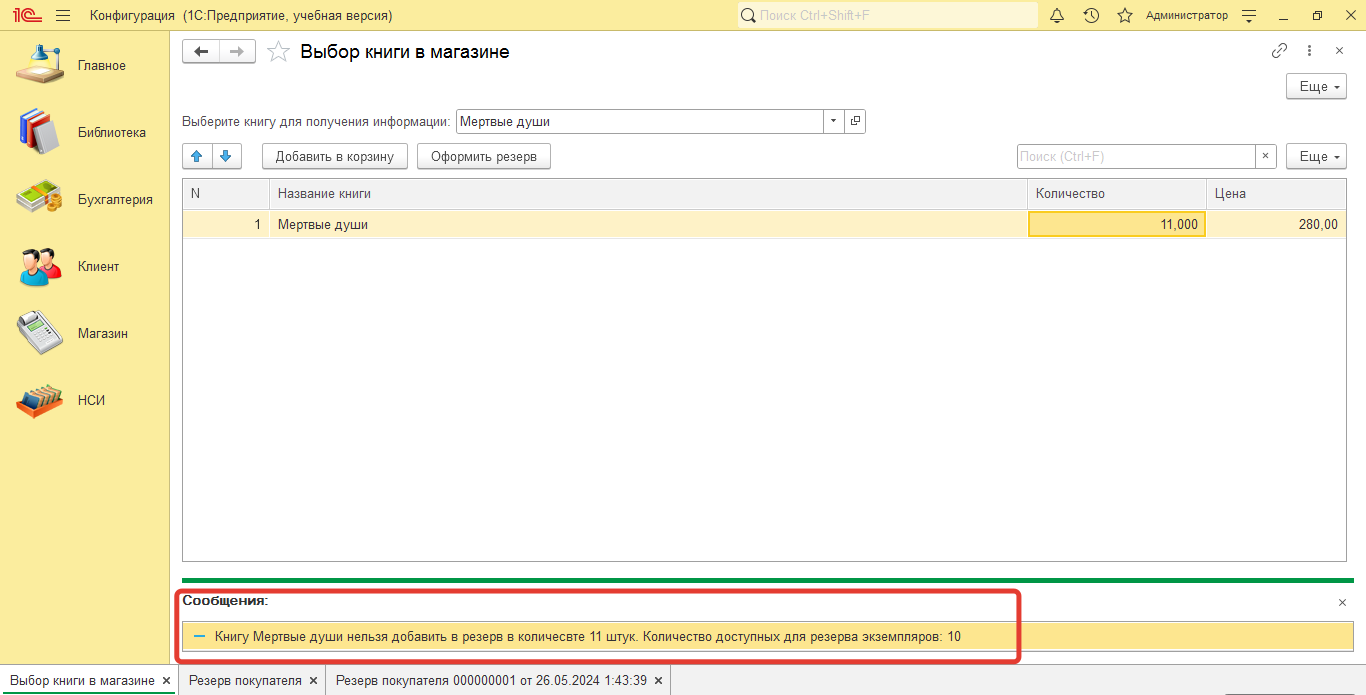
Статус документа «Резерв покупателя» автоматически заполняется значением «В резерве». Пользователь не имеет прав на изменение статуса. Документ «Резерв покупателя» проводится автоматически.



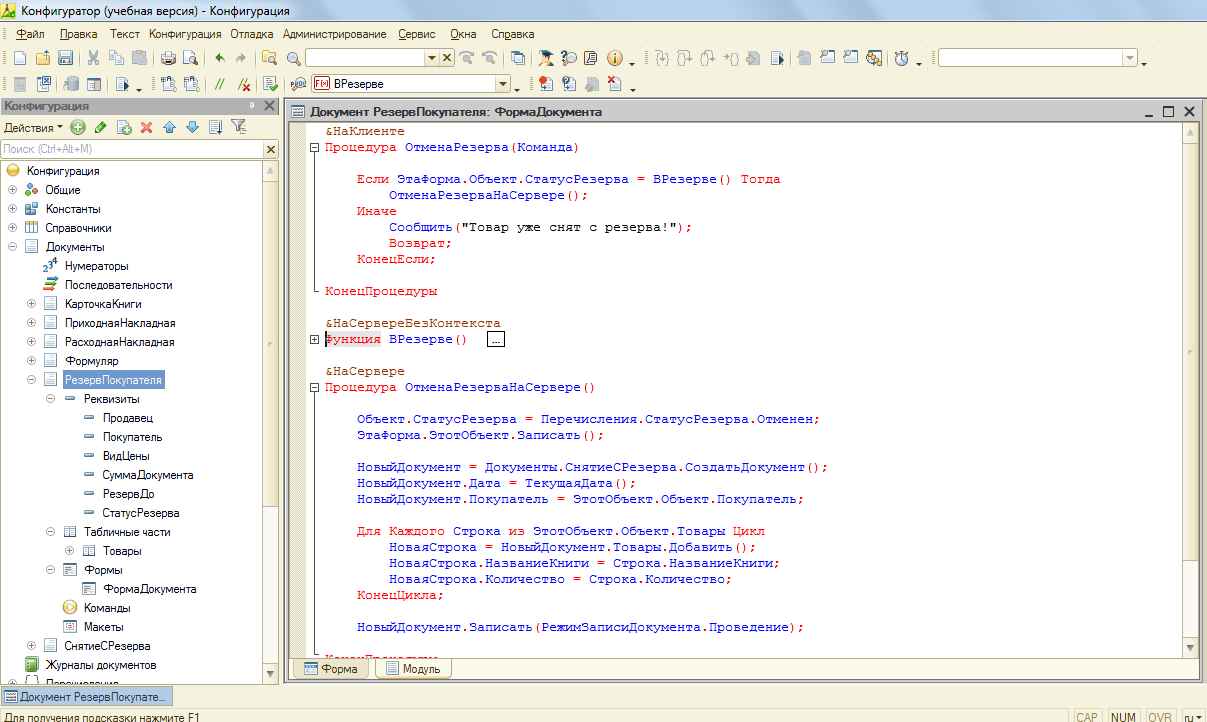
Документ не проведется, если в наличии меньше экземпляров книги, чем пользователь пытается зарезервировать. Проверка реализована в модуле объекта документа «Резерв покупателя» в процедуре «Обработка проведения».



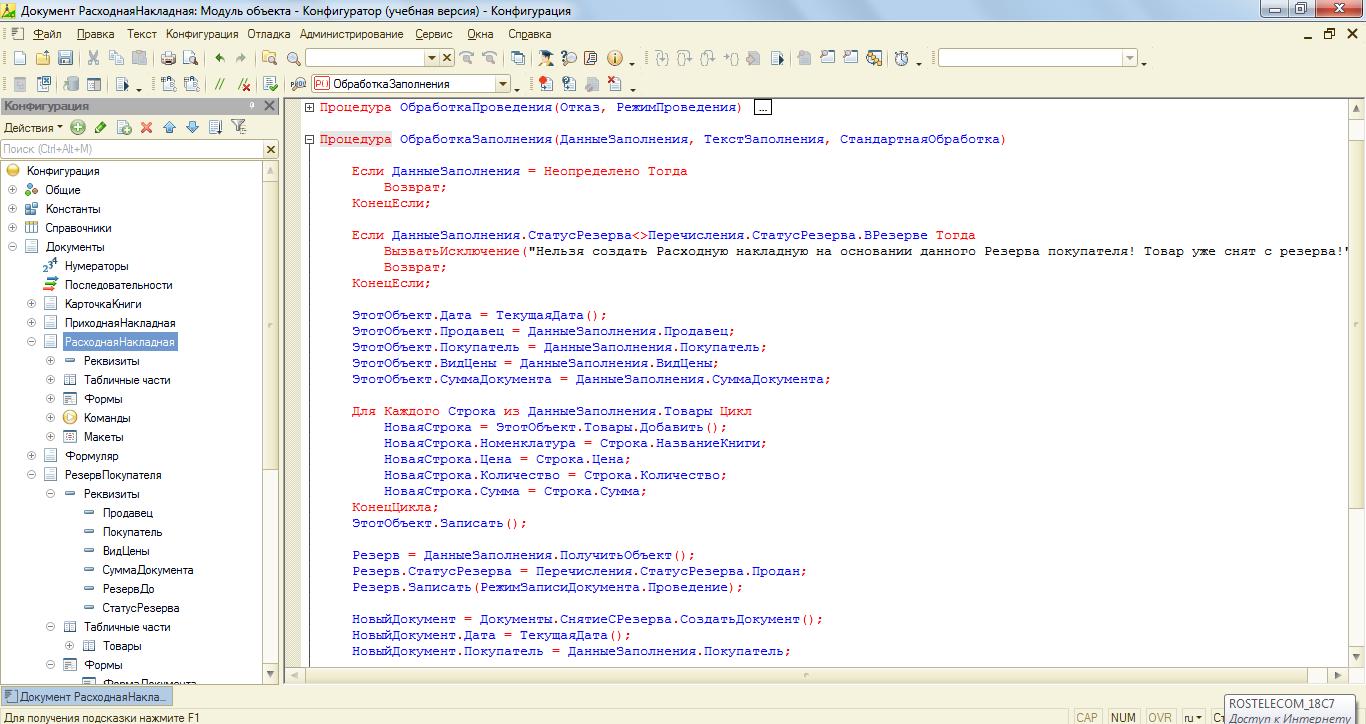
При попытке оформить такой резерв пользователь получит сообщение об ошибке.



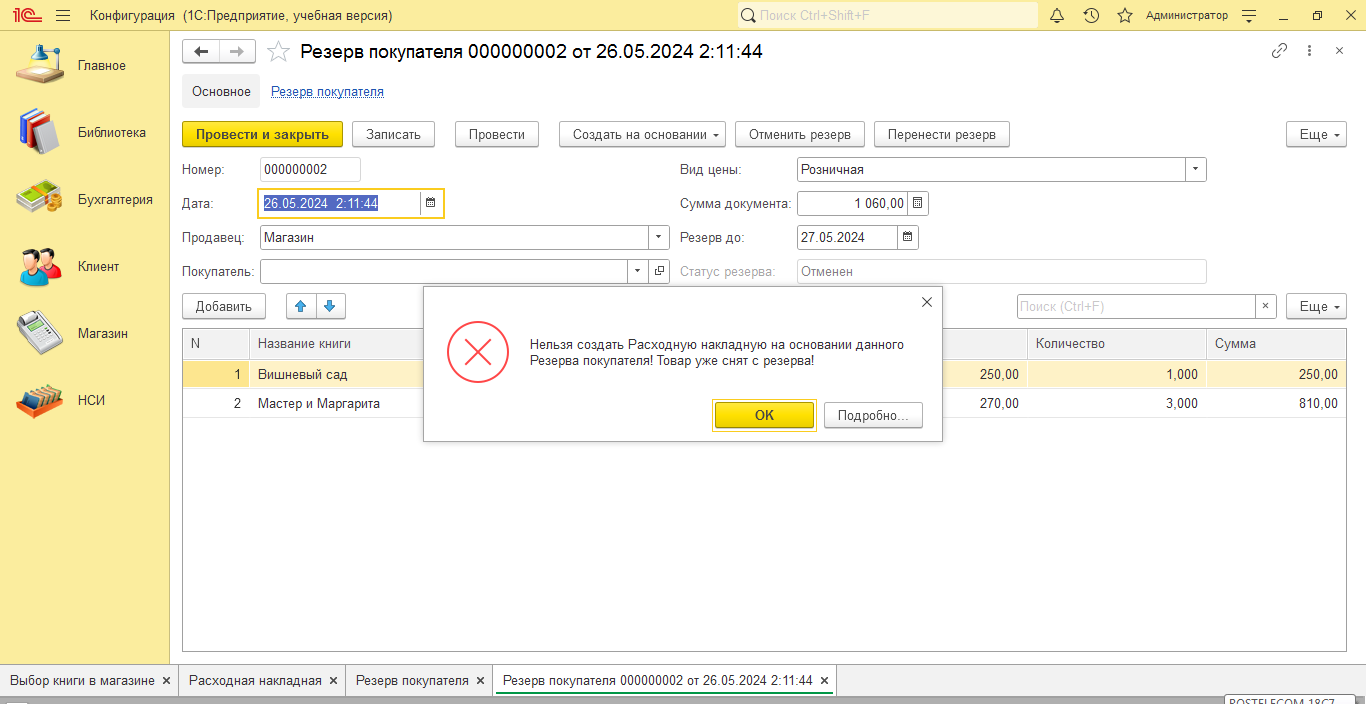
Пользователь может самостоятельно отменить резерв. Для этого ему необходимо нажать на кнопку «Отменить резерв». При этом в «Статус» резерва текущего документа «Резерв покупателя» установится значение «Отменен». И будет автоматически создан новый документ «Снятие с резерва». Этот документ невозможно создать вручную. Он создается либо при нажатии на кнопку «Отменить резерв», либо при проведении «Расходной накладной», созданной только на основании резерва.



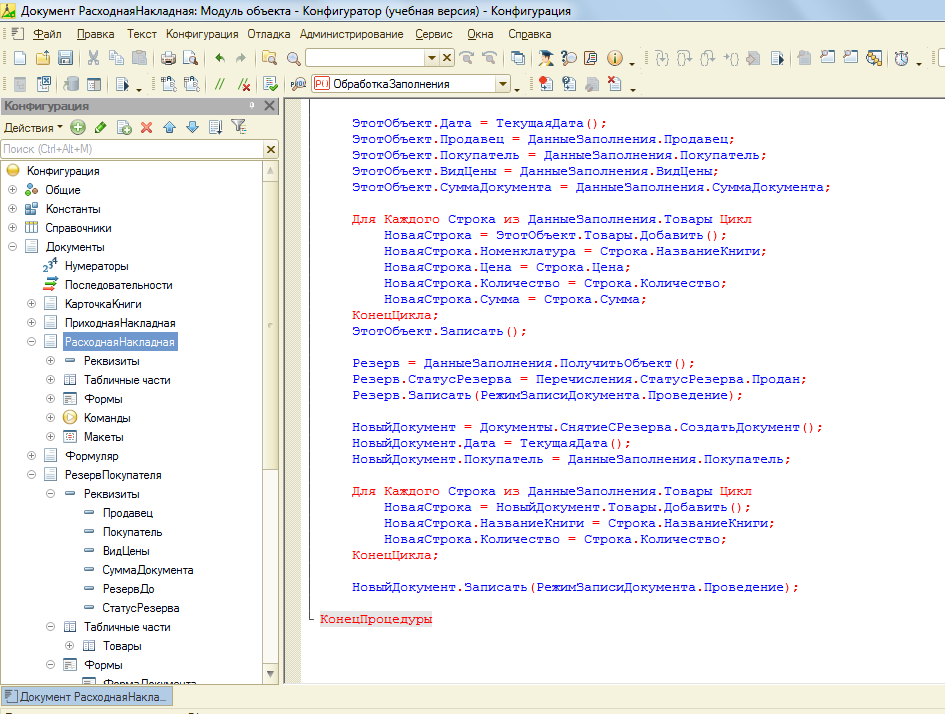
Оформленный документ «Резерв покупателя» доступен также менеджеру магазина. На основании «Резерва покупателя» при нажатии на кнопку «Создать на основании» менеджер может создать расходную накладную. Реализована проверка статуса документа «Резерв покупателя» при создании «Расходной накладной»



Таким образом, если «Статус» документа отличен от «В резерве» пользователь получит сообщение об ошибке.

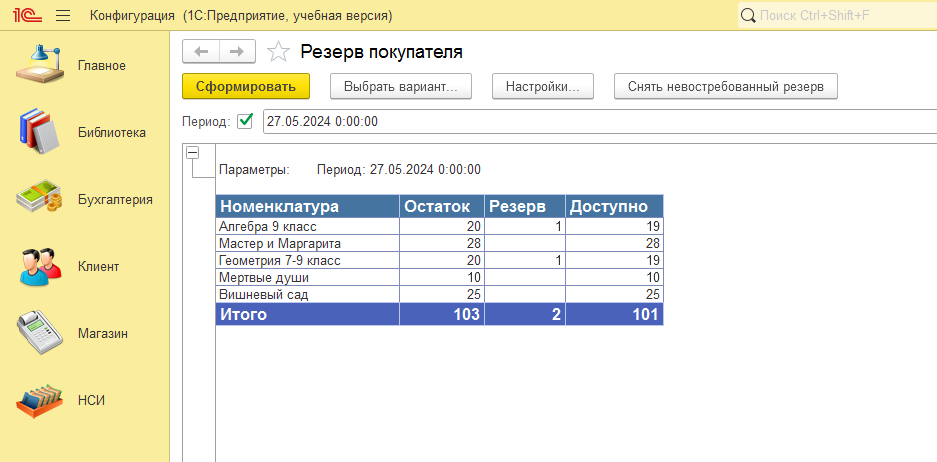


При успешном создании «Расходной накладной» на основании «Резерва покупателя» в статусе документа «Резерв покупателя» устанавливается значение «Продан», и создается документ «Снятие с резерва».

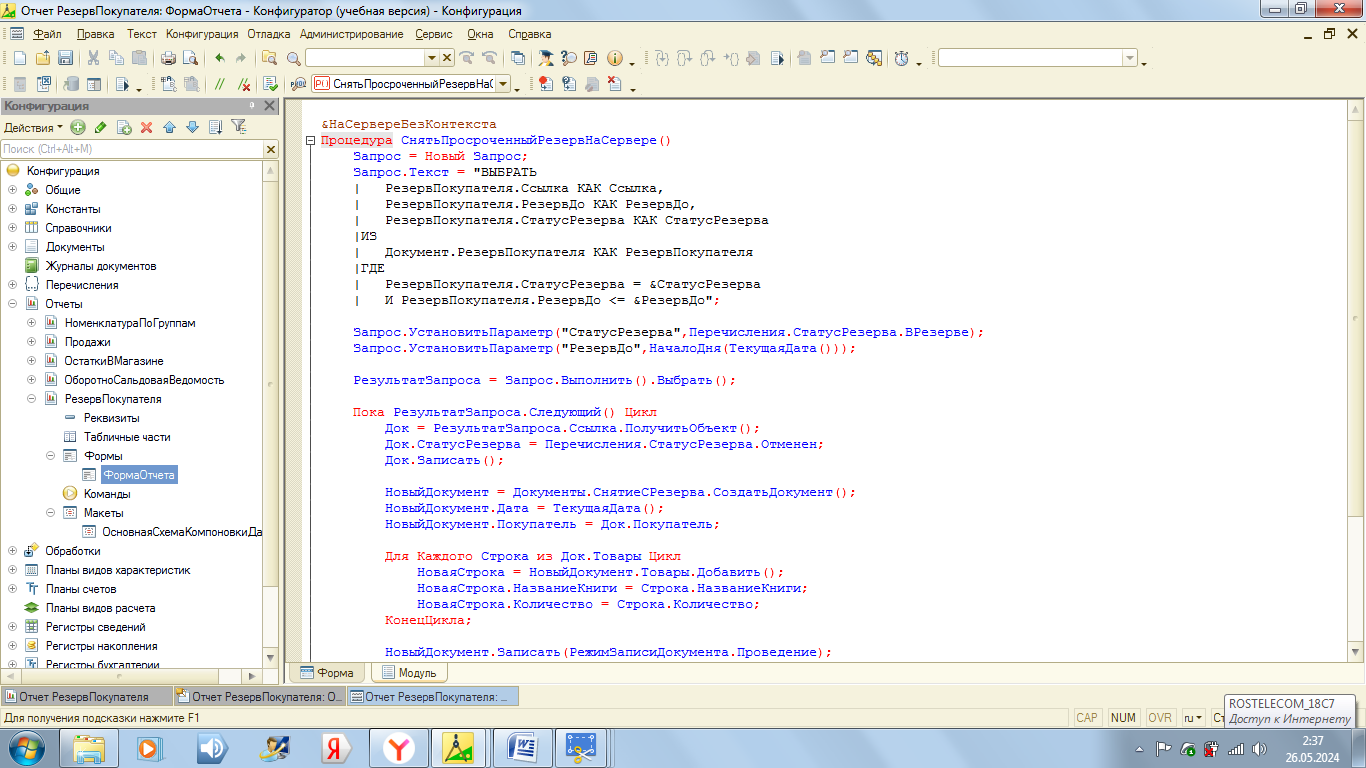


Документы «Резерв покупателя» и «Снятие с резерва» являются регистраторами для регистра накопления «Резерв покупателя».

В подсистеме «Магазин» реализован отчет «Резерв покупателя», который отражает актуальные данные о реальном остатке номенклатуры в магазине, о текущем резерве и о доступном остатке с учетом резерва на дату.



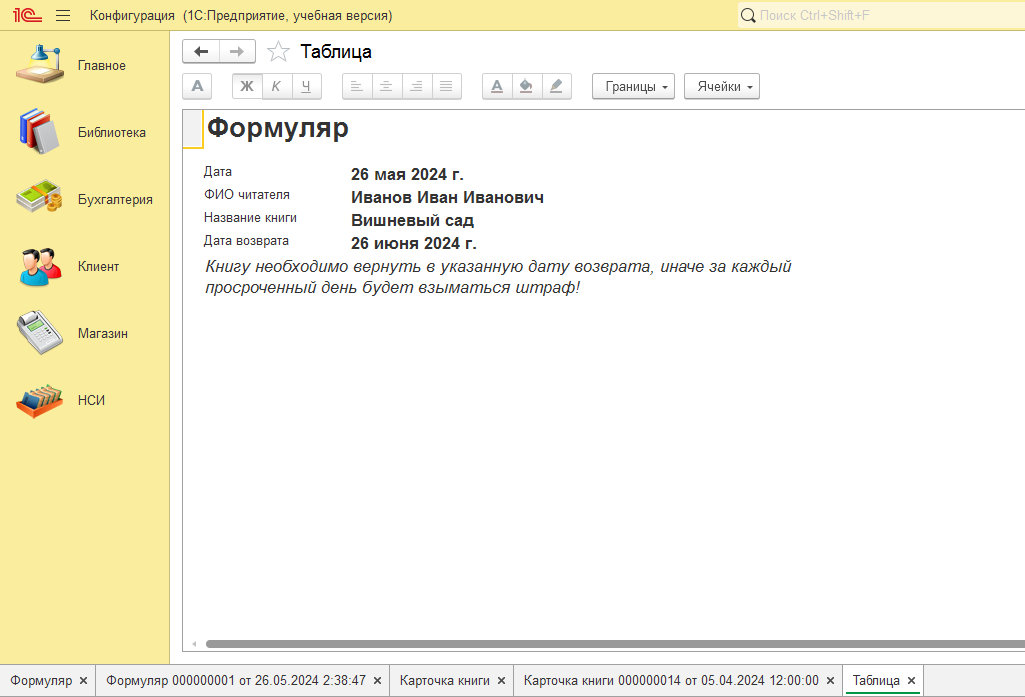
Кнопка «Снять невостребованный резерв» реализована для менеджера магазина и позволяет отменить все резервы, срок которых истек. При этом создаются документы «Снятие с резерва», а в документах «Резерв покупателя» устанавливается значение статуса «Отменен».



**3.4. Разработка подсистемы «Библиотека»**

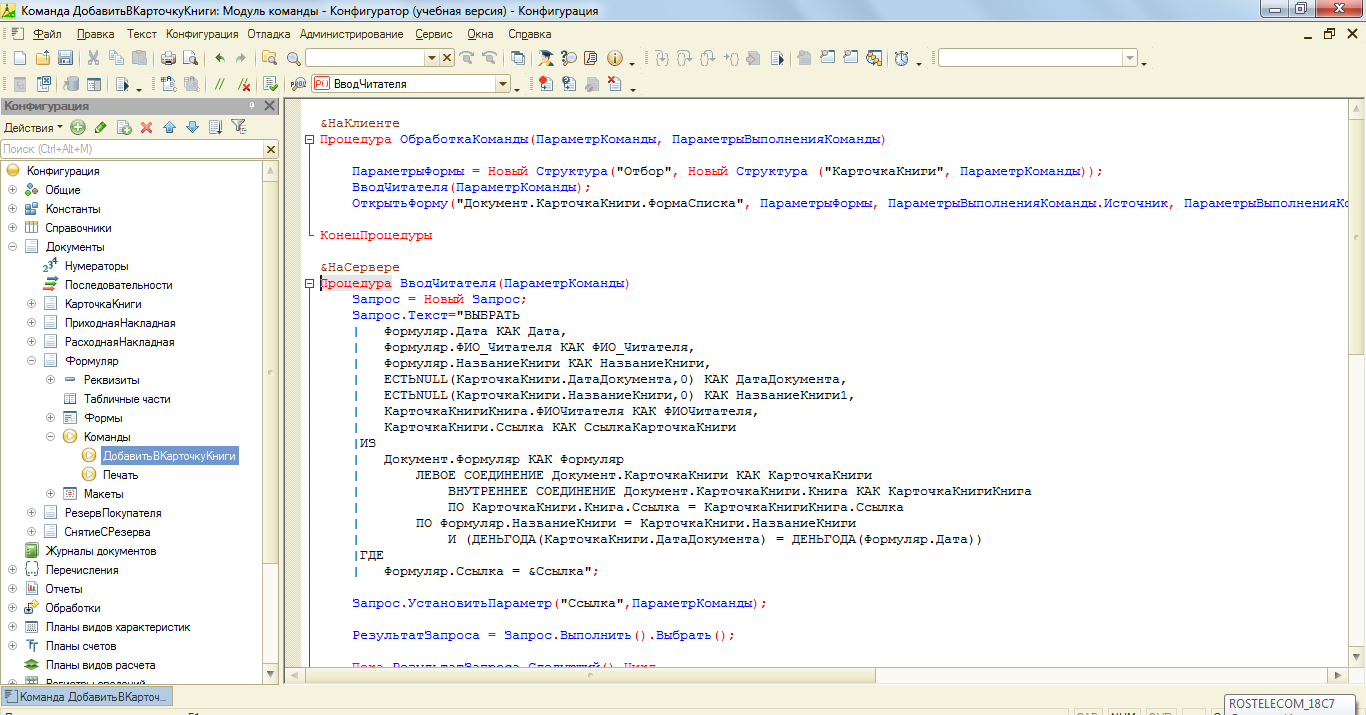
Основными учетными документами в подсистеме «Библиотека» являются «Формуляр» и «Карточка книги».

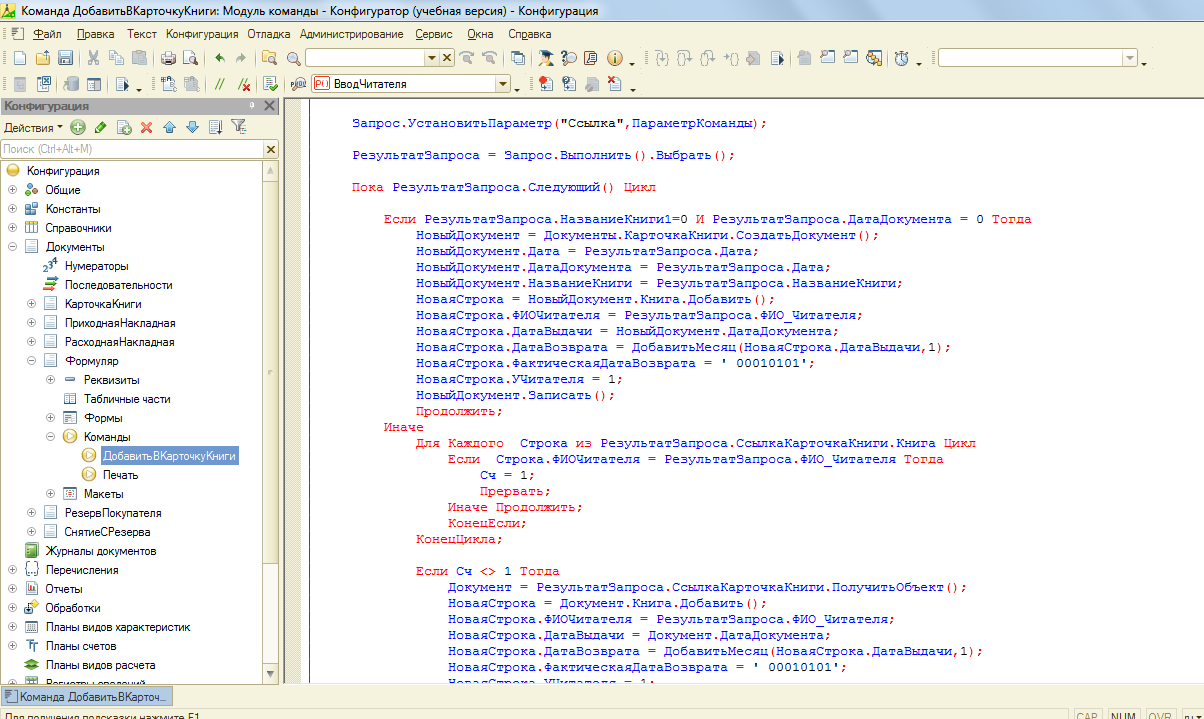
«Формуляр» оформляется на каждого читателя по каждой книге отдельно. В нем также отражается информация о дате возврата (+1 месяц от текущей даты). Печатная форма «Формуляра» выдается читателю.



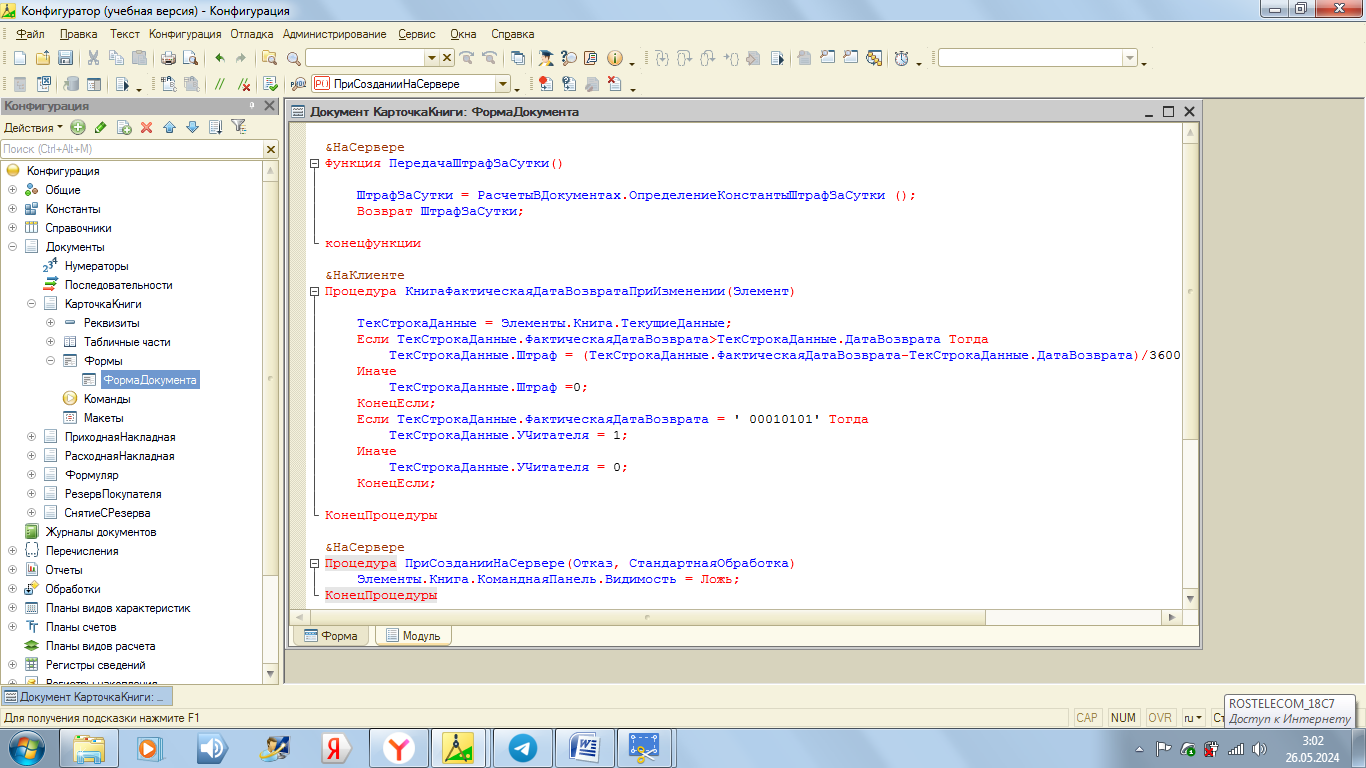
На основании «Формуляра» создается «Карточка книги». Если в текущий день эту книгу берут впервые, создается новый документ, если данную книгу уже брали, в ранее созданный документ добавляется строка с новым читателем.

Этот процесс реализован с помощью параметризованной команды «Добавить в карточку книги».

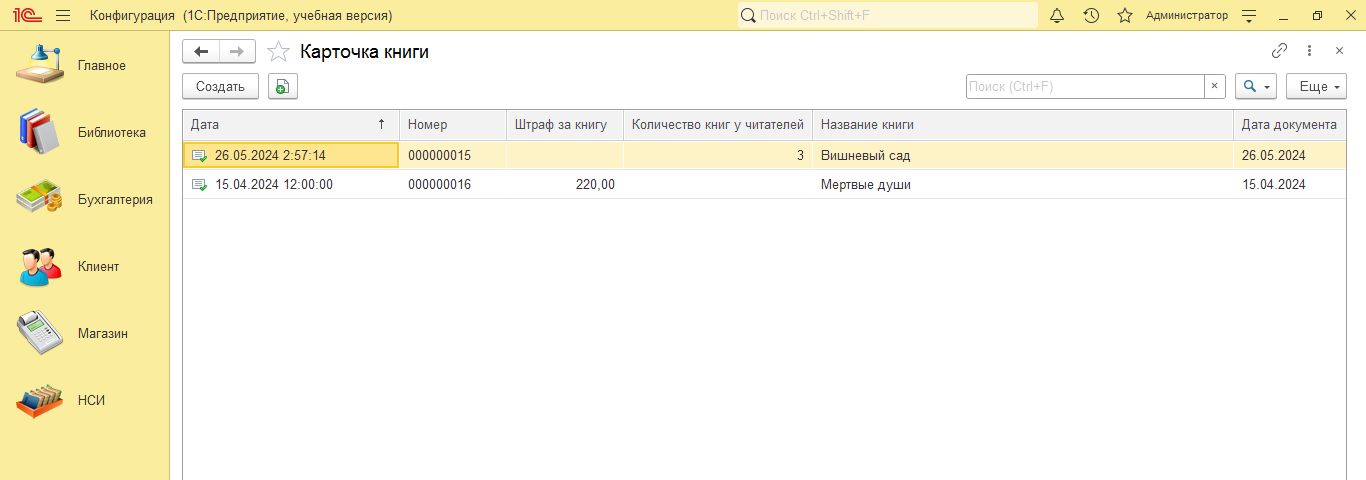




В документе «Карточка книги» по каждому читателю установлен признак «У читателя», значение «1» означает, что книгу еще не вернули. Также в табличной части есть поле «Фактическая дата возврата», если эта дата больше установленной даты возврата, читателю начисляется штраф. Размер штрафа за сутки реализован в виде константы. Данная константа доступна для изменения пользователем.



Очень удобно реализована форма списка «Карточки книги». Так библиотекарю нет необходимости заходить в каждый документ для того, чтобы узнать, есть ли по нему книги, которые еще не вернули. В форме списка показано общее количество экземпляров книг у читателя по данному документу.



**3.5. Разработка подсистемы «НСИ» и «Бухгалтерия»**

Все основные справочники и отчеты, входящие в подсистему «НСИ» были рассмотрены при описании подсистем «Магазин» и «Клиент».

Все документы, отчеты и регистры подсистемы «Бухгалтерия» были разработаны аналогично соответствующим стандартным документам, отчетам и регистрам типовых учетных систем 1С.

# Глава 4. Настройка ролей и прав доступа

Для реализации ограничения прав доступа в 1С предназначены роли.

Роль в конфигурации может соответствовать должностям или видам деятельности различных групп пользователей, для работы которых предназначена данная конфигурация.

Роль определяет, какие действия и над какими объектами метаданных может выполнять пользователь, выступающий в этой роли.

В процессе ведения списка пользователей прикладного решения каждому пользователю ставится в соответствие одна или несколько ролей. При попытке пользователя выполнить действие, на которое у него нет разрешения, действие выполнено не будет, а система выдаст окно предупреждения.

В учетной системе реализовано 6 ролей:

- администратор;

- директор;

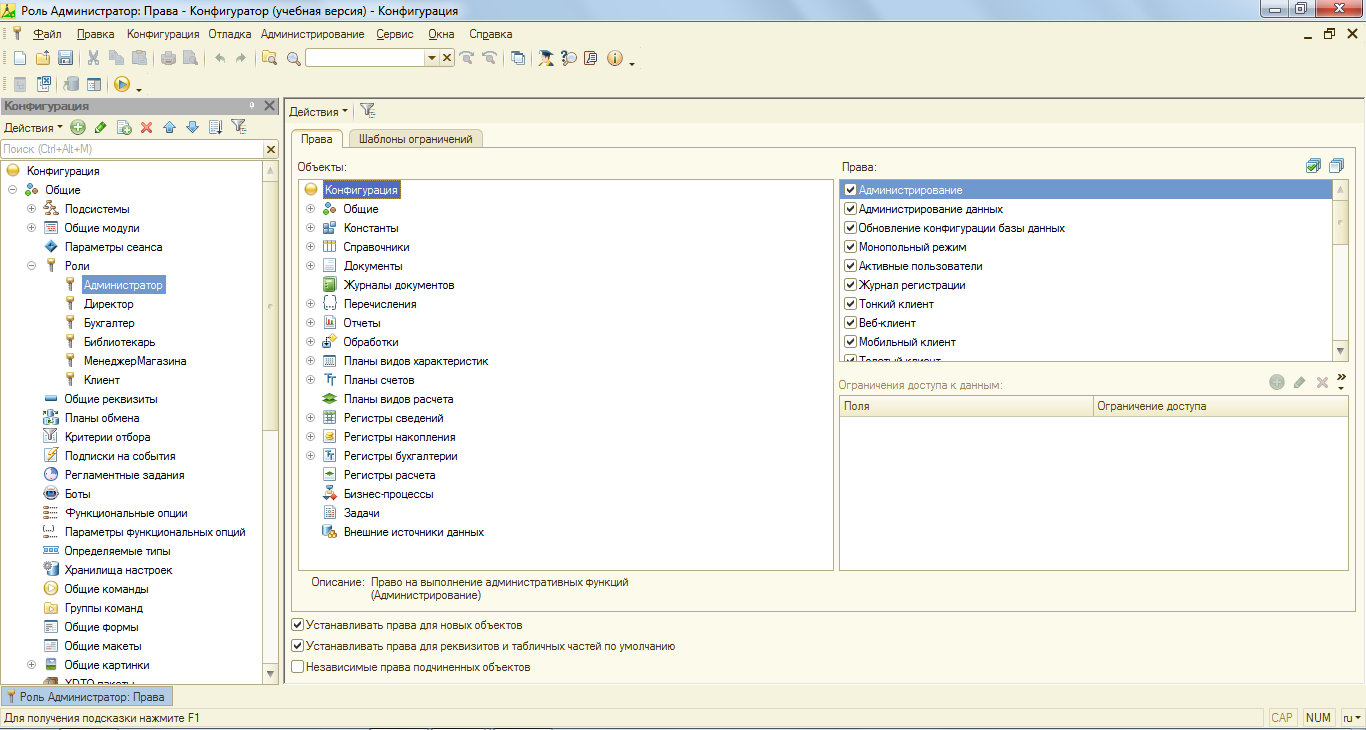
- бухгалтер;

- библиотекарь;

- менеджер магазина;

- клиент.

Администратор имеет полные права на все объекты конфигурации. Директор имеет права на просмотр всех объектов системы. Бухгалтер, библиотекарь, менеджер магазина и клиент имеют права на соответствующие подсистемы и на часть объектов подсистемы НСИ и смежных подсистем.



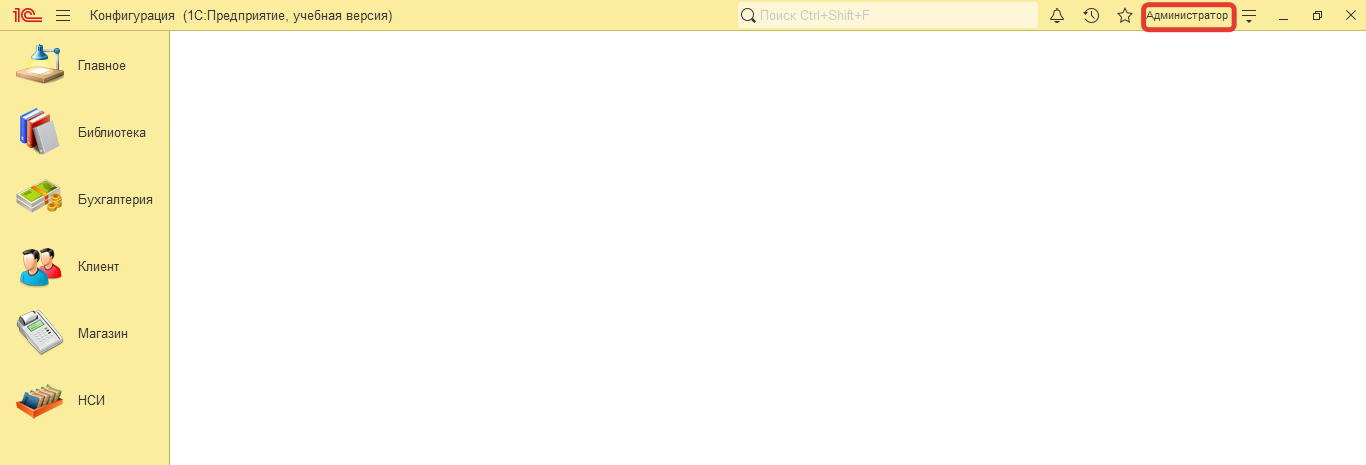


Рис.18. Внешний вид учетной системы для пользователя с ролью «Администратор»

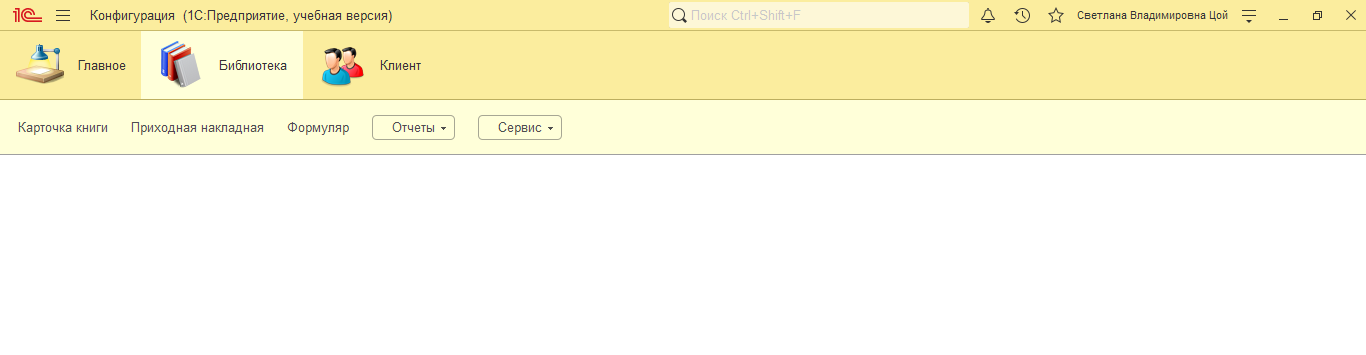


Рис.19. Внешний вид учетной системы для пользователя с ролью «Библиотекарь»

# Заключение

В результате работы была реализована учетная система, которая:

- упростит процессы учета книг: приобретение, сдача в аренду, продажа;

- регламентирует процессы взаимодействия с поставщиками книг для библиотеки и магазина, читателями книг библиотеки и покупателями в магазине;

- позволит отслеживать возврат книг читателями в библиотеку, и в случае несвоевременного возврата начислять штраф;

- позволит клиенту библиотеки или магазина получать информацию о книге и ее наличии, а также оформлять резерв самостоятельно с локального терминала, расположенного в магазине или библиотеке;

- реализует бухгалтерский учет.

# Список использованной литературы и интернет ресурсов:

* + - 1. <https://its.1c.ru>
      2. [https://42clouds.com](https://42clouds.com/ru-ru/articles/why-do-we-need-configuration-extensions)
      3. <https://v8.1c.ru/platforma>
      4. [https://wiseadvice-it.ru](https://wiseadvice-it.ru/o-kompanii/blog/articles/obrabotki-1s-8-3/)
      5. [https://www.koderline.kz](https://www.koderline.kz/expert/narabotki/article-ierarkhicheskiy-spravochnik-v-konfiguratsii-sistemy-1s-predpriyatie/)
      6. <https://habr.com/ru/articles/490006/>
      7. <https://gb.ru/blog/arhitektuta-sistemy/>
      8. <https://products.aspose.app/diagram/ru/uml>
      9. <https://www.microsoft.com/ru-ru/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling>
      10. 1C:Предприятие 8.3. Практическое пособие разработчика. Примеры и типовые приемы. М. Г. Радченко, Е. Ю. Хрусталева
      11. Разработка сложных отчётов в «1С:Предприятии 8». Система компоновки данных. Е. Ю. Хрусталева