

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агабекян Раиса Левоновна

Должность: ректор

Дата подписания: 25.01.2024 13:53:40

Уникальный программный адрес:

4237c7ccb9b9e111bbaf1f4fcda9201d015c4dbaa123ff774747307b9b9ffcb8

Негосударственное аккредитованное некоммерческое
частное образовательное учреждение высшего образования
«Академия маркетинга и социально-информационных технологий –
ИМСИТ»

(г. Краснодар)

Институт информационных технологий и инноваций

Кафедра математики и вычислительной техники

Рассмотрено и одобрено на
заседании кафедры математики и
вычислительной техники Академии
ИМСИТ, протокол № 5 от 11 декабря
2023 года, зав. кафедрой МиВТ, к.э.н.,
доцент Н.П. Исикова

УТВЕРЖДАЮ
Проректор по учебной работе,
доцент Н.И. Севрюгина
25 декабря 2023 г.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КУРСОВОГО ПРОЕКТА

по дисциплине «Базы данных»

для обучающихся направления подготовки
09.03.04 Программная инженерия
Направленность (профиль) программы:
«Разработка компьютерных игр и приложений
расширенной реальности»

Квалификация выпускника
«Бакалавр»

Краснодар
2023

Методические указания по выполнению курсового проекта по дисциплине «Базы данных» для обучающихся всех форм обучения направления подготовки 09.03.04 Программная инженерия направленность (профиль) образовательной программы «Разработка компьютерных игр и приложений расширенной реальности»

– Краснодар: Академия маркетинга и социально-информационных технологий – ИМСИТ (г. Краснодар), 2023.

Методические указания по выполнению курсового проекта содержат требования к составу и содержанию, рекомендации по выполнению и защите курсового проекта по дисциплине «Базы данных».

Методические указания составлены в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки 09.03.04 Программная инженерия, утвержденного приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 920, направленность (профиль) образовательной программы «Разработка информационно-вычислительных систем».

Методические указания составлены с учетом рекомендаций проекта примерной основной образовательной программы высшего образования по направлению подготовки 09.03.04 Программная инженерия, включенного в реестр примерных основных образовательных программ высшего образования 5 февраля 2018 года.

Составитель, канд. физ.-мат. наук, доцент В.В. Бужан

Методические указания рассмотрены и рекомендованы на заседании кафедры математики и вычислительной техники от 11.12.2023 г., протокол № 5

Методические указания утверждены на заседании Научно-методического совета Академии ИМСИТ протокол № 4 от 25 декабря 2023 г.

Председатель Научно-методического Совета Академии ИМСИТ, профессор Н.Н. Павелко

Согласовано:

Проректор по учебной работе, доцент Н.И. Севрюгина

Проректор по качеству образования, доцент К.В. Писаренко

Рецензенты:

Видовский Л.А., д.т.н., профессор, профессор кафедры информационных систем и программирования КубГТУ

Глебов О.В., директор АО «ЮГ-СИСТЕМА ПЛЮС»

Содержание

Введение	
1 Нормативные ссылки	4
2 Общие требования к выполнению курсового проекта	4
2.1 Состав материалов, предоставляемых на защиту курсового проекта	5
2.2 Структура пояснительной записки	5
3 Процесс проектирования	7
3.1 Основные этапы выполнения курсового проекта	7
3.2 Методические рекомендации к выполнению этапов	8
4 Информационный анализ предметной области	9
4.1 Планирование разработки базы данных	9
4.2 Цели автоматизации. Определение требований к системе	9
4.3 Сбор и анализ требований пользователей	10
4.3.1 Автоматизируемые задачи	10
4.3.2 Участники автоматизируемых процессов	11
4.3.3 Используемые документы	11
5 Проектирование базы данных	12
5.1 Концептуальная модель	12
5.1.1 Спецификации объектов	12
5.1.2 Концептуальная модель бизнес-процесса «Приход товара»	12
5.1.3 Концептуальная модель бизнес-процесса «Отпуск товара»	13
5.2 Логическое проектирование	13
5.2.1 Упрощение концептуальной модели	13
5.2.2 Этап нормализации	14
5.3 Физическое проектирование	15
6 Разработка приложения	23
6.1 Основные моменты	23
6.2 Формы программы	23
6.3 Отчеты программы	24
7 Реализация	25
7.1 Приложение администрирования пользователей	25
7.2 Основное приложение	26
8 Загрузка данных	32
9 Тестирование	32
10 Задания на курсовое проектирование	33
Список литературы	36
Приложение А (справочное). Требования к оформлению пояснительной записки	37
Приложение Б (обязательное). Форма титульного листа курсового проекта .	39
Приложение В (обязательное). Задание на курсовое проектирование	40

Введение

Прогресс, достигнутый за последние годы во всех аспектах вычислительной техники, включая теорию, технологию и приложения, привел к значительному расширению области применения компьютеров и росту числа их пользователей. Перевод существующей информации на компьютерную основу с использованием современных многопользовательских систем управления базами данных (СУБД), клиент/серверных технологий позволяет сберечь значительные средства и время для получения необходимой информации, а главное, значительно повысить эффективность таких важнейших процессов, как хранение, извлечение и управление большими объемами данных.

Дисциплина «Базы данных» относится к базовой части профессионального цикла дисциплин, определенных федеральным государственным образовательным стандартом высшего профессионального образования в учебном плане подготовки бакалавров по направлению 09.03.04-Программная инженерия.

1 Нормативные ссылки

В настоящих методических указаниях используются ссылки на следующие нормативные документы:

ГОСТ Р 1.5-2004 Стандарты национальные РФ. Правила построения, изложения, оформления и обозначения

ГОСТ Р 7.0.5-2008 СИБИД. Библиографическая ссылка. Общие требования и правила составления

ГОСТ 2.105-95 ЕСКД. Общие требования к текстовым документам

ГОСТ 2.301- 68 ЕСКД. Форматы ГОСТ 7.9-95 СИБИД. Реферат и аннотация. Общие требования

ГОСТ 7.82-2001 СИБИД. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления.

2 Общие требования к выполнению курсового проекта

Цель курсового проектирования – систематизация, обобщение, закрепление и углубление навыков, приобретенных в процессе изучения теоретического курса и прохождения лабораторного практикума дисциплины «Базы данных» и практическая их реализация при разработке информационной системы отдельно взятого фрагмента предметной области (ПрО).

Все этапы курсового проекта студент выполняет самостоятельно, периодически демонстрируя преподавателю выполняемую работу, разрешая при этом сложные для себя моменты проектирования.

В процессе выполнения проекта студенту предоставляется возможность пройти путь рядового разработчика: от предварительного планиро-

вания базы данных (БД) до ее физического воплощения и создания на языке высокого уровня клиентского приложения для работы с БД.

Тематика заданий на курсовую работу, приведенных в данных методических указаниях, может быть дополнена и расширена в связи с решением актуальных научно-исследовательских задач, выполняемых на кафедре.

2.1 Состав материалов, предоставляемых на защиту курсового проекта

На защиту курсового проекта студент представляет: пояснительную записку, к которой прилагаются электронные копии материалов в соответствии с таблицей 2.1.

Таблица 2.1 – Пакет результатов курсового проектирования

Директория	Содержимое
\	Корневая
\DataBases\	ER-диаграммы БД разработанного программного обеспечения, файлы с текстами запросов создания БД и её тестового наполнения, файлы БД в формате выбранной СУБД.
\Distributives\	«Собранные», работоспособные дистрибутивы разработанного программного обеспечения.
\Documentation\	Пояснительная записка курсового проекта в формате Microsoft Word 97-2003.
\Sources\	Полный комплект исходных кодов разработанного программного обеспечения (в виде проекта или решения, включая все используемые библиотеки, не входящие в стандартную поставку платформы разработки или средств разработки), пригодный для компиляции.

2.2 Структура пояснительной записки

В пояснительную записку должны входить:

- титульный лист (приложение Б);
- задание на курсовое проектирование (приложение В) с подписью студента и датой выдачи;
- реферат (объем ПЗ, количество таблиц, рисунков, схем, программ, приложений, краткая характеристика и результаты работы); по ГОСТ 7.9;
- содержание;
- нормативные ссылки;
- введение;
- основная часть, включающая:

1 Анализ предметной области и требований

1.1 Цель автоматизации

1.2 Перечень задач / процессов, подлежащих автоматизации в рамках курсового проектирования

- 1.3 Участники автоматизируемых процессов
- 1.4 Последовательности действий участников автоматизируемых процессов в контексте задачи курсового проектирования
- 1.5 Перечень используемых документов (входящие, исходящие)
- 1.6 Концептуальная модель ПрО с описанием

2 Разработка базы данных и программных приложений к ней

- 2.1 Логическая модель ПрО с описанием
 - 2.2 ER - диаграмма с описанием
 - 2.3 Перечень функций программы
 - 2.4 Перечень форм программы и пояснения к ним
 - 2.5 Перечень отчетов программы и пояснения к ним
- руководство пользователя;
 - заключение (результаты работы, предложения по их реализации);
 - список использованных источников;
 - приложения (при необходимости).

Освещение вопросов стандартизации предполагает краткое изложение автором курсового проекта требований, предъявляемых к документированию процесса разработки информационной системы со стороны нормативных документов.

В *заключении* кратко излагаются основные результаты курсового проекта, даются предложения по их реализации и усовершенствованию.

В *списке используемых источников* приводятся сведения об источниках, используемых автором в процессе выполнения курсового проекта, (по ГОСТ Р 7.0.5, ГОСТ 7.82).

Приложения к пояснительной записке содержат все полезные и ценные для уяснения сути курсового проекта сведения, такие как иллюстрации, расчеты, обоснования, листинги программ, если их присутствие в пояснительной записке может загромождать основной текст.

Пояснительная записка иллюстрируется схемами, копиями экрана, выходными документами, листингом программного кода приложений.

В приложении желательно предусмотреть возможность анализа данных результатов запроса в виде графиков или диаграмм на основе хранимых данных. Правила оформления текстовых документов даны в приложении А.

2.3 Коллективные работы

В коллективных работах, в которых принимают участие два (и бо-лее) студента, четко определяется объем и характер работы каждого из них.

Если программа является коллективным продуктом, то пояснительная записка оформляется каждым студентом с четким определением, какая часть работы была выполнена именно им.

3 Процесс проектирования

Ограниченный объем методических указаний предполагает рассмотрение лишь главных моментов проектирования информационной системы (ИС). Чтобы сконцентрировать на них внимание студента, составители по возможности старались обходиться без привлечения дополнительных программных продуктов, в настоящее время широко используемых в процессе эксплуатации ИС (например, генераторов отчетов). Однако усилия студента освоить такие программные продукты и использовать их в процессе проектирования приветствуются.

3.1 Основные этапы выполнения курсового проекта

Курсовое проектирование подразумевает строгое выполнение некоторой последовательности этапов в контексте результатов предыдущих и с учётом требований последующих при соблюдении степени проработки каждого этапа.

Процесс создания ИС базируется на использовании структурного подхода к разработке ее программного обеспечения, поскольку именно он обеспечивает необходимые уровни производительности труда разработчиков и эффективности функционирования системы. Фундаментом его является концепция жизненного цикла БД, определяющая последовательность этапов процесса создания БД и в целом ИС (табл. 3.1).

Таблица 3.1 – Этапы выполнения курсового проекта

Номер этапа	Наименование этапа	Количество часов
		6 семестр
1	Информационный анализ предметной области: – планирование разработки базы данных – цели автоматизации, определение требований к системе – сбор и анализ требований пользователей	4
2	Проектирование базы данных: – концептуальное проектирование базы данных – логическое проектирование базы данных – физическое проектирование базы данных	6
3	Разработка приложений	8
4	Реализация	4
5	Загрузка данных	2
6	Тестирование	4
7	Оформление пояснительной записки	6
	Итого:	34

Естественно, что конкретное наполнение каждого этапа в значительной степени зависит от сложности разрабатываемого продукта.

3.2 Методические рекомендации к выполнению этапов

Ниже приводятся краткие методические рекомендации к выполнению каждого из вышеперечисленных этапов проектирования БД выбранной ПрО и разработки программных приложений, необходимых для использования различными пользователями хранимой в ней информации. Для иллюстрации наполняемости этапов выполнения курсового проекта в качестве примера рассматривается процесс разработки ИС для ПрО «Торговая компания».

Следует подчеркнуть, что в современном понимании указанная ПрО включает в себя множество взаимосвязанных объектов, процессов и событий. При разработке ИС для достижения требуемых уровней ее эффективности и производительности в каждом конкретном случае создается абстрактная модель, включающая только важные именно для разрабатываемой системы сущности, связи, свойства, а второстепенные игнорируются.

В методических указаниях в качестве иллюстрирующего примера, вследствие строгих ограничений на их объем, выбрана упрощенная модель ПрО с ограниченным кругом решаемых задач: задач автоматизации процессов поставок и продаж компании с целью ускорить ход этих процессов и улучшить сервис по работе с клиентами и поставщиками в целом. Но учитывая, что процесс решения каждой задачи включает главные моменты создания любой ИС, есть уверенность, что студент, разобравшись в них, приобретет навыки создания ИС разнообразного назначения. В последующих разделах рассмотрим подробно каждый этап.

4 Информационный анализ предметной области

Анализ ПрО предполагается осуществлять посредством анализа требований, суть которого состоит в изучении ПрО с последующим выделением объекта (совокупности задач / процессов) автоматизации и дальнейшей его формализацией.

4.1 Планирование разработки базы данных

Содержание данного этапа – разработка стратегического плана, в процессе которого осуществляется предварительное планирование ИС, включающее оценку трех основных компонентов: объема работ, ресурсов и стоимости проекта. Планирование разработки должно быть связано с общей стратегией построения ИС организации.

Важной частью разработки стратегического плана является проверка трех аспектов осуществимости проекта.

Проверка *технологической осуществимости*: существует ли оборудование и ПО, удовлетворяющее информационным потребностям компании. Такая проверка для ПрО «Торговая компания» показала, что информационный отдел в фирме отсутствует; все бизнес-процессы протекают без использования ЭВМ; необходима закупка ЭВМ, на которую будет установлен сервер БД; компания не имеет филиалов и будет использован один локальный сервер.

Проверка *операционной осуществимости* для той же компании показала, что в организации имеются специалисты, способные выступать в роли экспертов; квалифицированный персонал для работы с БД отсутствует и необходимо обучение своих сотрудников или привлечение новых.

При проверке *экономической целесообразности* осуществления проекта было выявлено, что возможности компании позволят ей справиться с необходимыми затратами, а срок окупаемости ИС равен одному году. К тому же внедренная в будущем система окажет положительное влияние на реализацию долгосрочных планов организации.

4.2 Цели автоматизации. Определение требований к системе

В рамках проекта выделены следующие цели автоматизации:

- сокращение времени обработки основных торговых операций;
- уменьшение влияния «человеческого фактора»;
- увеличение достоверности и скорости получения отчетности.

Достижение поставленных целей приведет к повышению эффективности работы организации и качества обслуживания ее клиентов, а значит, уменьшению издержек компании и увеличению ее прибыли.

Анализ операций по работе с клиентами и поставщиками показал:

- данная система призвана автоматизировать бизнес-процесс «покупатель/поставщик»;
- необходимость в распределенной системе отсутствует;

– требуется закупка как минимум одной ЭВМ, на базе семейства операционных систем Windows, т.к. разработка клиентской программы будет происходить в среде Visual Studio C#.

4.3 Сбор и анализ требований пользователей

Информационный анализ ПрО предваряет процесс проектирования БД. Информационные потребности выясняются с помощью анкет, опросов работников компании, с помощью наблюдений за деятельностью предприятия, а также отчетов и форм, которые используются в текущий момент.

На данном этапе необходимо создать для себя модель движения важных материальных объектов и уяснить процесс документооборота. По каждому документу необходимо установить периодичность использования, определить данные, необходимые для выполнения выделенных функций (анализируя существующую и планируемую документацию, выяснить, как получается каждый элемент данных, кем получается, где в дальнейшем используется, кем контролируется).

Самое пристальное внимание должно быть уделено дублированию информации, возможности появления ложной информации и причинам, которые ведут к их появлению. Также на этом этапе желательно представить общие параметры создаваемой базы.

Последовательность вопросов, решаемых в процессе анализа ПрО:

- определение автоматизируемых задач;
- определение участников данных процессов;
- определение объектов и их взаимосвязей (раздел 5);
- перечень атрибутов для каждого объекта (раздел 5);
- перечень документов, обрабатываемых в рамках создаваемого программного продукта и структура данных этих документов.

В итоге собранная информация о каждой важной области применения приложения и пользовательской группе должна включать следующие компоненты: исходную и генерируемую документацию, а также список требований с указанием их приоритетов. На основании всей этой информации будут составлены спецификации требований пользователей, описывающих деятельность предприятия с разных точек зрения.

Произведенный опрос потенциальных пользователей будущей системы позволил выделить ряд сценариев при работе с будущей системой.

4.3.1 Автоматизируемые задачи

Разрабатываемое ПО должно решать следующие задачи:

- хранение справочной информации (товары, клиенты и т. д.);
- ввод информации о поступлениях товара, заказах покупателей, отгрузках товара по заказам покупателей, взаиморасчетах с контрагентами;

- получение отчетной информации об объемах продаж за определенный период времени, исполнении заказов покупателей, текущих складских запасов, состоянии взаиморасчетов с покупателями и поставщиками.

4.3.2 Участники автоматизируемых процессов

Товаровед/менеджер по закупкам:

- поддерживает в актуальном состоянии справочную информацию о товарах на складе компании и поставщиках;
- вводит информацию о поступлении товаров от поставщиков;
- контролирует состояние текущих запасов компании, определяет необходимость их пополнения, предотвращает появление «отрицательных остатков».

Менеджер по продажам:

- поддерживает в актуальном состоянии справочную информацию о клиентах компании;
- принимает и фиксирует заказы от покупателей;
- принимает решение и фиксирует информацию об отгрузке товаров по заказам покупателей;
- контролирует полноту исполнения заказов покупателей, взаиморасчеты с покупателями и их своевременность.

Администратор:

- поддерживает в актуальном состоянии справочную информацию о менеджерах по продажам и о менеджерах по закупкам;
- поддерживает в актуальном состоянии информацию всей БД.

4.3.3 Используемые документы

Сформулируем перечень используемых ИС документов:

- заказ покупателя – отражает намерение клиента приобрести определенный перечень товаров;
- расходная накладная – отражает факт исполнения заказа клиента и переход указанного перечня товаров в собственность покупателя;
- приходная накладная – отражает факт поступления товаров от поставщика на склад компании;
- оплата покупки – отражает факт поступления денежных средств от покупателя за отгруженные товары;
- оплата поставки – отражает факт перечисления денежных средств поставщику компании за полученные товары.

Выходные данные ИС:

- расходный кассовый ордер;
- приходный кассовый ордер;
- накладная расходная/приходная;
- отчеты.

5 Проектирование базы данных

Первая фаза процесса проектирования БД заключается в создании для анализируемой ПрО *концептуальной модели данных*. Построение осуществляется в определенном порядке: создаются подробные модели пользовательских представлений данных; затем они интегрируются в единую концептуальную модель данных. Концептуальное проектирование приводит к созданию *концептуальной схемы* БД.

5.1 Концептуальная модель

Напомним, что в ПрО «Торговая компания» для автоматизации в рамках данного курсового проекта выбраны две подзадачи, где сущность «покупатель/поставщик» является стержневой: приход товара и отпуск товара.

5.1.1 Спецификации объектов

Основой проектируемой системы является договор, который заключается с поставщиком/покупателем и на основании которого ведется вся остальная деятельность предприятия по поставкам и продажам.

Анализ выделенных задач ИС позволил включить в нее следующие объекты: Контрагент, Сотрудник, Товар, Накладная.

Спецификации выделенных объектов будущей системы:

- Контрагент: ФИО, адрес, реквизиты;
- Сотрудник: ФИО, адрес, паспорт, телефон;
- Товар: цена, название, количество;
- Накладная: дата, цена, реквизиты сторон.

5.1.2 Концептуальная модель бизнес-процесса «Приход товара»

Концептуальная модель бизнес-процесса первой подзадачи, описывающая приход товара, представлена на рисунке 5.1.

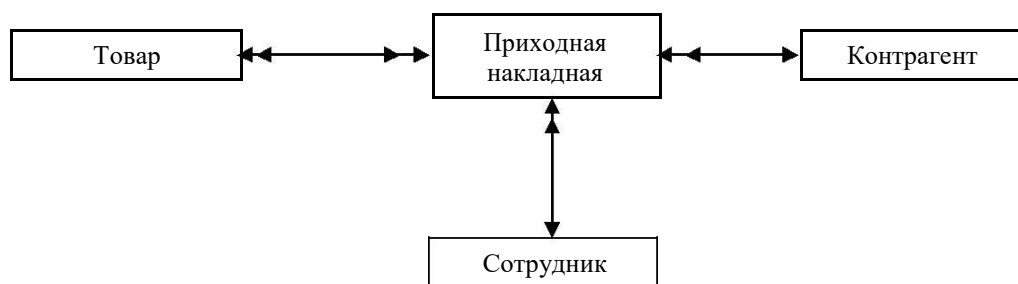


Рисунок 5.1 – Подзадача «Приход товара»

Спецификации бинарных связей:

- Товар_Приходная накладная: связь типа «N:M»;
- Контрагент_Приходная накладная: связь типа «1:N»;
- Сотрудник_Приходная накладная: связь типа «1:N».

5.1.3 Концептуальная модель бизнес-процесса «Отпуск товара»

Концептуальная модель бизнес-процесса по отпуску товара представлена на рисунке 5.2.

Спецификации бинарных связей:

- Товар_Расходная накладная: связь типа «N:M»;
- Контрагент_Расходная накладная: связь типа «1:N»;
- Сотрудник_Приходная накладная: связь типа «1:N».

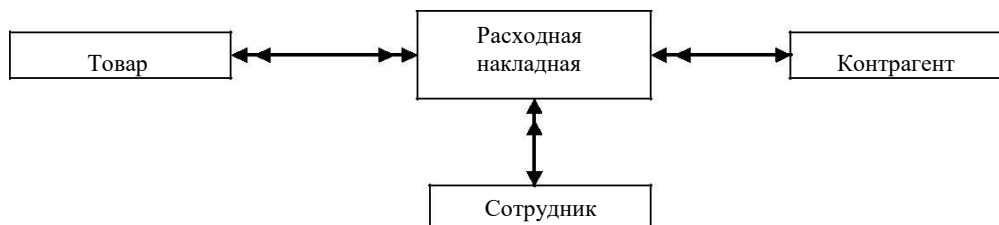


Рисунок 5.2– Подзадача «Отпуск товара»

Соединив в единую логическую единицу две подзадачи, получим концептуальную модель БД следующего вида (рис. 5.3).

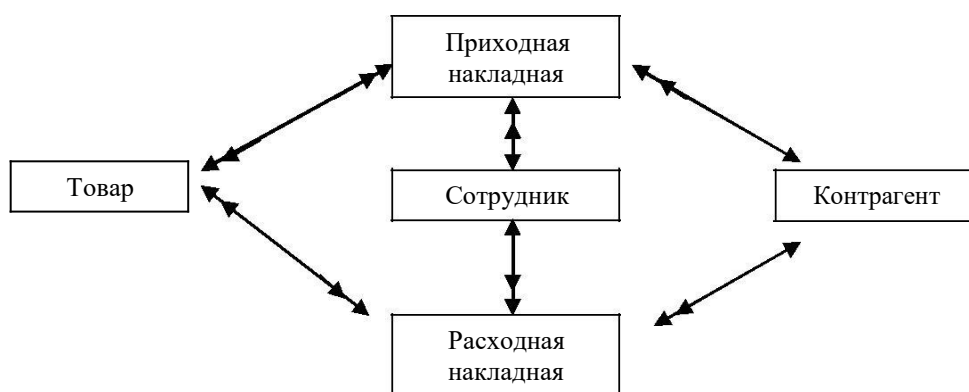


Рисунок 5.3 – Концептуальная модель БД

5.2 Логическое проектирование

На данном этапе производится упрощение концептуальной модели, а затем ее преобразование в реляционную модель.

5.2.1 Упрощение концептуальной модели

В реляционной модели данных (РБД) связь типа «многие ко многим» не может быть представлена напрямую. Она убирается путем добавления новой таблицы. Дополнительный объект «Поставка» используется для замены одной бинарной связи «многие ко многим» между двумя объектами – «Товар» и «Приходная накладная» двумя бинарными связями типа «один ко многим». Причем в обоих случаях «много» направлено в сторону дополнительного объекта «Поставка».

После дальнейших аналогичных упрощений связей получена следующая концептуальная модель (рис. 5.4).

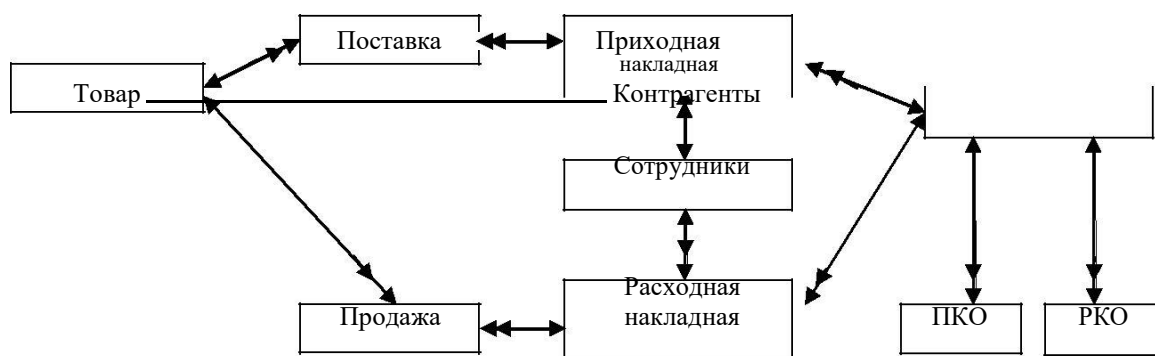


Рисунок 5.4 – Упрощенная концептуальная модель

5.2.2 Этап нормализации

Процесс нормализации – это формальный метод анализа отношений на основе их первичных или потенциальных ключей и существующих функциональных зависимостей, являющийся одним из наиболее строгих способов улучшения характеристик БД [1].

Для РБД необходимо, чтобы все ее отношения обязательно находились в 1НФ. Нормальные формы более высокого порядка могут использоваться разработчиками по своему усмотрению. Однако грамотный специалист стремится довести уровень нормализации БД хотя бы до 3НФ, тем самым исключив из базы данных избыточность данных и аномалии обновления.

В процессе нормализации на каждой его фазе производится анализ первоначальных отношений на наличие нежелательных зависимостей между атрибутами каждого отношения. При обнаружении таких зависимостей в структуру БД будут внесены изменения, позволяющие от них избавиться.

В данной БД атрибуты всех отношений имеют атомарные значения, а значит, все отношения находятся в 1НФ.

Все отношения в данной базе данных имеют простой первичный ключ, следовательно, они уже находятся во 2НФ.

В данной БД нет отношений, в которых какой-либо атрибут транзитивно зависел от первичного ключа, поэтому отношения находятся в 3НФ.

Детерминанты всех ее функциональных зависимостей являются потенциальными ключами, поэтому отношения находятся в НФБК.

Многозначные зависимости и зависимости соединений в отношениях рассматриваемой БД отсутствуют, поэтому 4НФ и 5НФ не рассматриваются.

5.3 Физическое проектирование

Физическое проектирование представляет собой создание на сервере реляционной БД. Рассмотрим процесс создания таблицы «Товар» (табл. 5.1).

Таблица 5.1 – Формирование реляционной таблицы

Объект	Атрибут	Первичный ключ
Товар	id Модель Фирма Страна Количество Цена	Id

Для соединения с требуемым сервером выберем его имя, введем уже имеющийся на нем логин администратора «sa» и его пароль, который был задан при установке сервера (рис. 5.5).

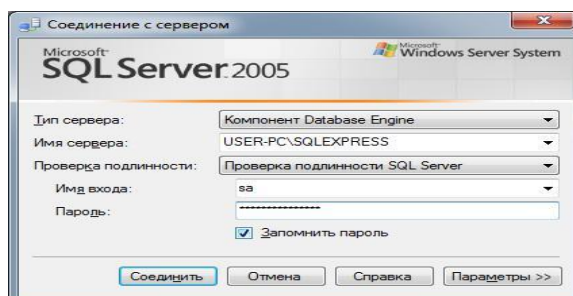


Рисунок 5.5 – Соединение с сервером

Успешное соединение с сервером позволяет создать новую БД. Рассмотрим два варианта этого процесса.

Создание БД визуальными средствами: вызвать вкладку «Создание базы данных...» (рис. 5.6).

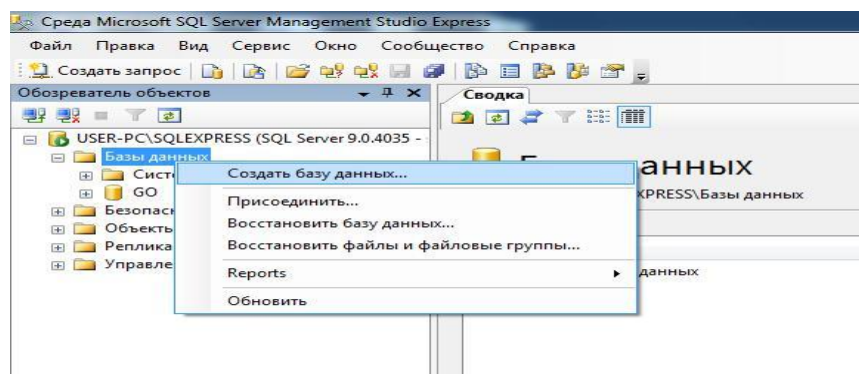


Рисунок 5.6 – Активация менеджера создания БД

В открывшемся диалоговом окне (рис. 5.7) требуется указать имя БД и путь к папке, в которой будут храниться файлы новой БД.

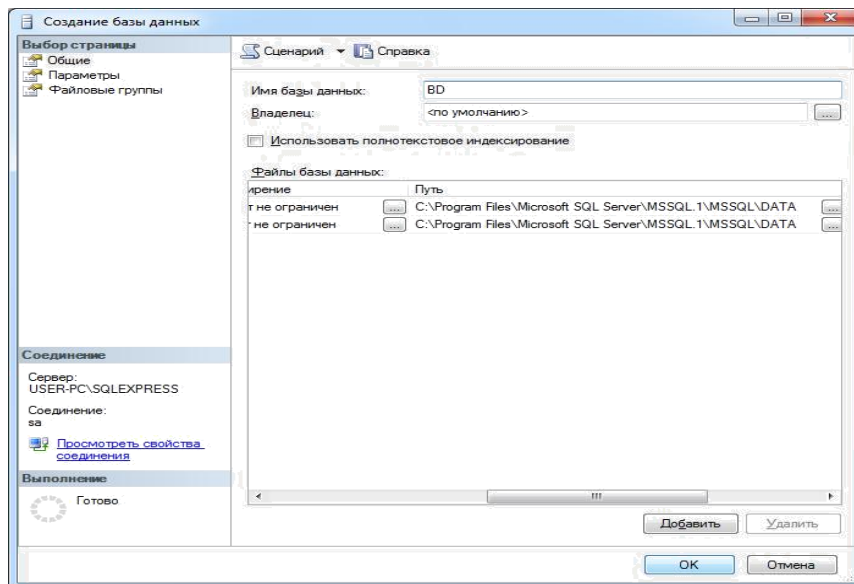


Рисунок 5.7 – Окно «Создание базы данных»

Создание БД с помощью инструмента «Создать запрос», вводя код на языке Transact-SQL, показано на листинге 1.

Листинг 1 – Создание БД

```
USE [master]
GO
CREATE DATABASE [BD] ON PRIMARY
( NAME = N'BD', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL.1\MSSQL\DATA\BD.mdf' , SIZE = 3072KB , MAXSIZE =
UNLIMITED, FILEGROWTH = 1024KB )
LOG ON
( NAME = N'BD_log', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL.1\MSSQL\DATA\BD_log.ldf' , SIZE = 1024KB , MAXSIZE = 2048GB
, FILEGROWTH = 10%)
```

Создание таблицы осуществляется вызовом в обозревателе объектов соответствующего меню (рис. 5.8).

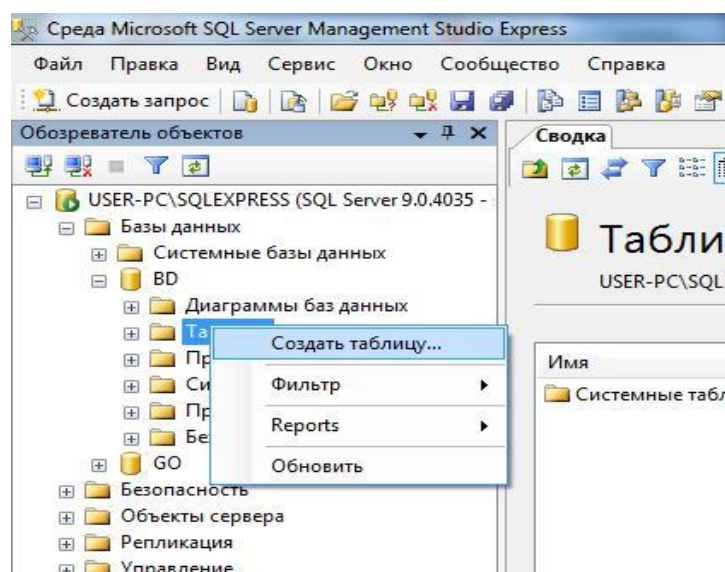


Рисунок 5.8 – Вызов меню «Создать таблицу»

Откроется визуальный компонент, запрашивающий имена и типы данных атрибутов будущей таблицы. Для таблицы определяется первичный ключ (рис. 5.9).

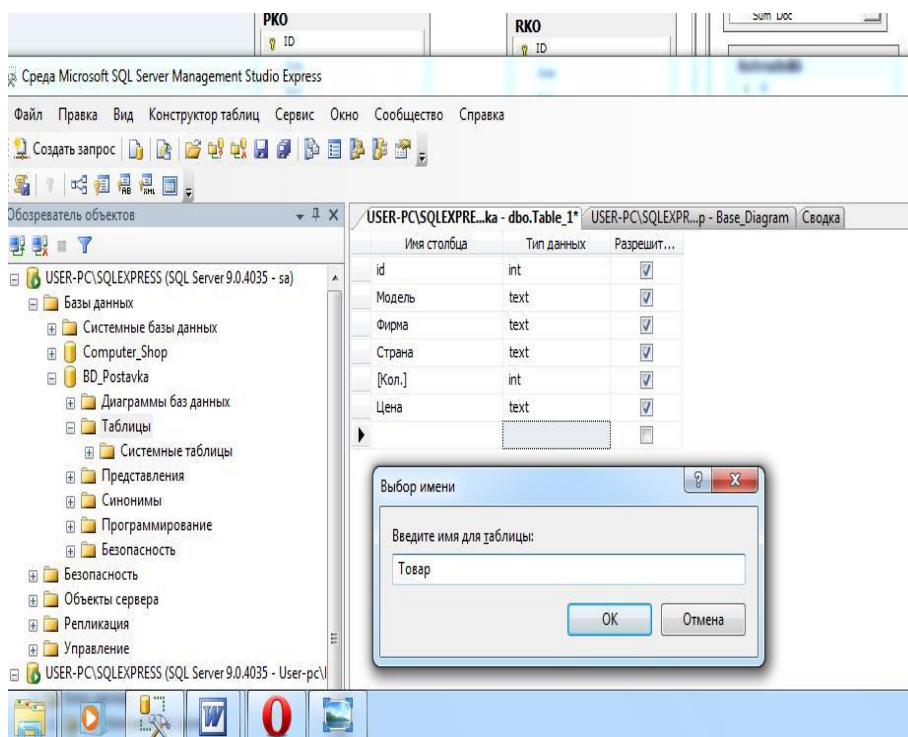


Рисунок 5.9 – Создание таблицы

Для заполнения таблицы данными вызывается меню «Открыть таблицу» (рис. 5.10). В открывшемся окне появляется созданная таблица, заполним ее.

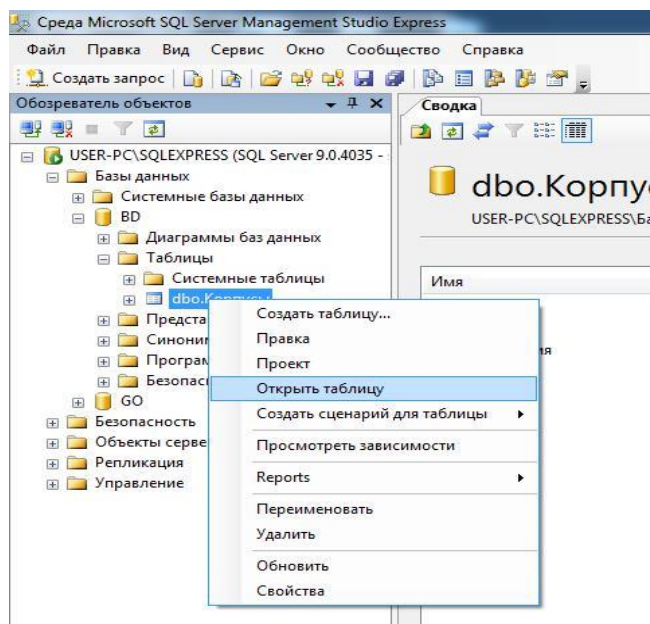


Рисунок 5.10 – Вызов окна заполнения таблицы данными

Создать и заполнить таблицы данными можно и при помощи операторов языка Transact – SQL, как показано на листингах 2-3.

Листинг 2 – Создание таблицы

```

USE [bd]
GO
/*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Товар] (
    [id] [int] NOT NULL,
    [Модель] [text] NULL,
    [Фирма] [text] NULL,
    [Страна] [text] NULL,
    [Кол.] [int] NULL,
    [Цена] [text] NULL)

```

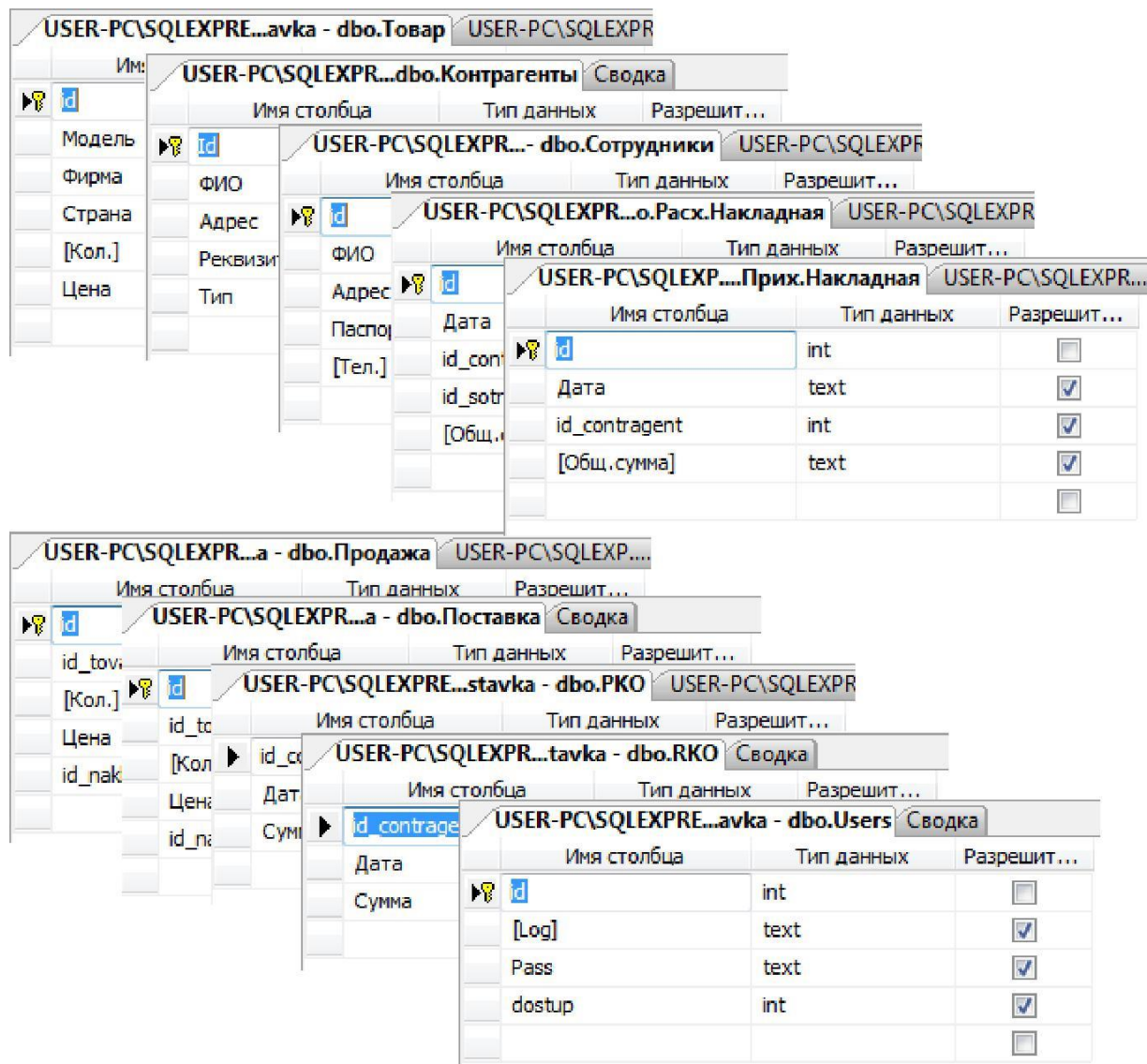


Рисунок 5.11 – Созданные таблицы БД

Листинг 3 – Заполнение таблицы информацией

```
INSERT INTO [BD_Postavka].[dbo].[Товар]
([Модель]
,[Фирма]
,[Страна]
,[Кол.]
,[Цена])
VALUES
(<x200, text,>
,<Nvidia, text,>
,<Китай, text,>
,<10, int,>
,<2456, text,>)
```

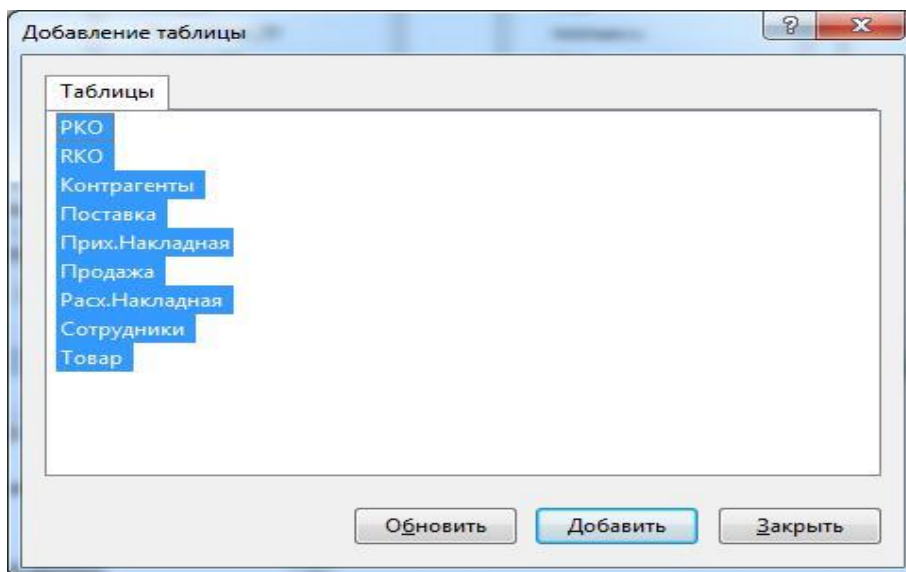


Рисунок 5.12 – Добавление таблиц в конструктор диаграмм

Заполнение данными всех остальных таблиц БД производится аналогично любым из рассмотренных способов (рис. 5.11).

Переход в развернутом меню к позиции «Диаграммы базы данных» позволяет создать с помощью конструктора физическую модель БД, добавив в нее необходимые таблицы и связи между ними (рис. 5.12-5.14).

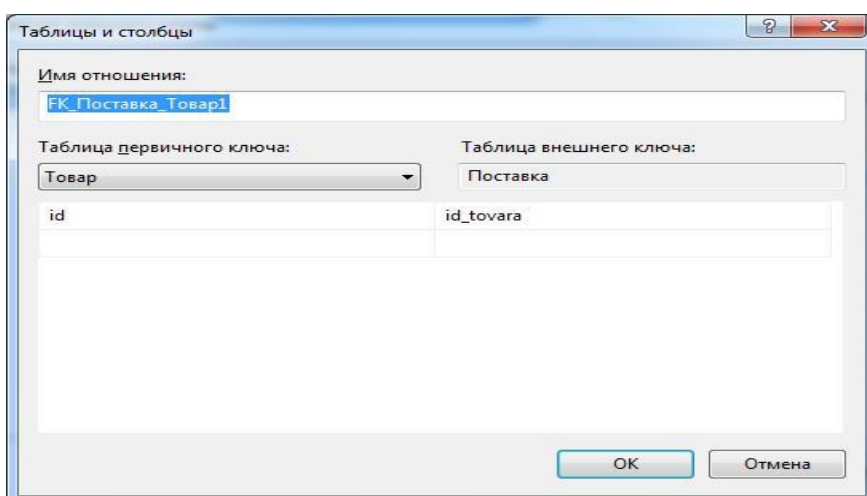


Рисунок 5.13 – Настройка свойств в отношении

Доступ пользователя к БД осуществляется в соответствии с его правами. Рассмотрим процесс наделения пользователя правами, позволяющими ему изменять и просматривать данные в таблицах БД, но запрещающими внесение изменений в структуру БД: создание таблиц, изменение их структуры и связей.

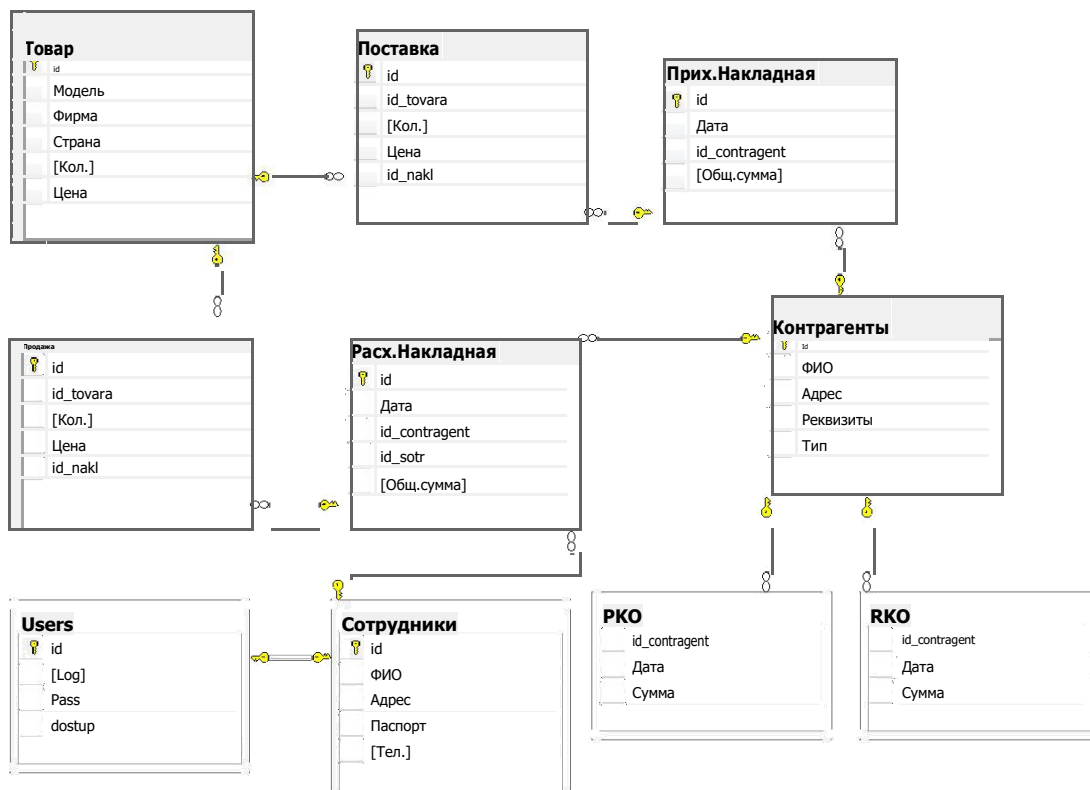


Рисунок 5.14 – ER - диаграмма БД

Роль такого пользователя ограничивается только безопасным соединением клиентского приложения с БД (рис. 5.15).

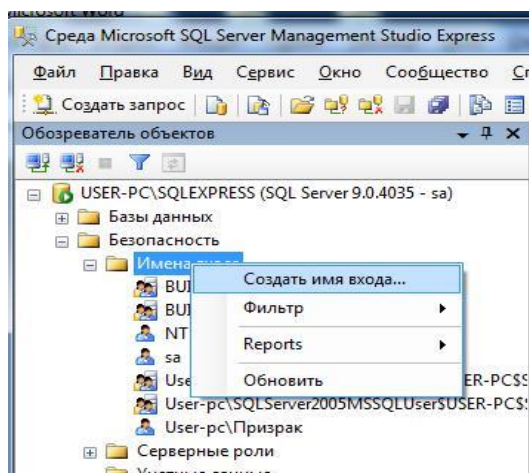


Рисунок 5.15 – Вызов диалога создания пользователя на сервере

Для создаваемого пользователя назначается имя входа и выбирается один из вариантов: будет ли это пользователь SQL Server или Windows,

осуществляется привязка к БД и устанавливаются ограничения на сервере во вкладке «Серверные роли» (рис. 5.16).

Созданного на сервере с ограниченными правами пользователя можно привязать к БД путем установки ограничений, но уже для объектов БД.

При привязке пользователя к БД выбирается имя входа на сервере, которое было создано на предыдущем этапе, вводится имя пользователя БД и устанавливаются для него ограничения (рис. 5.17-5.18).

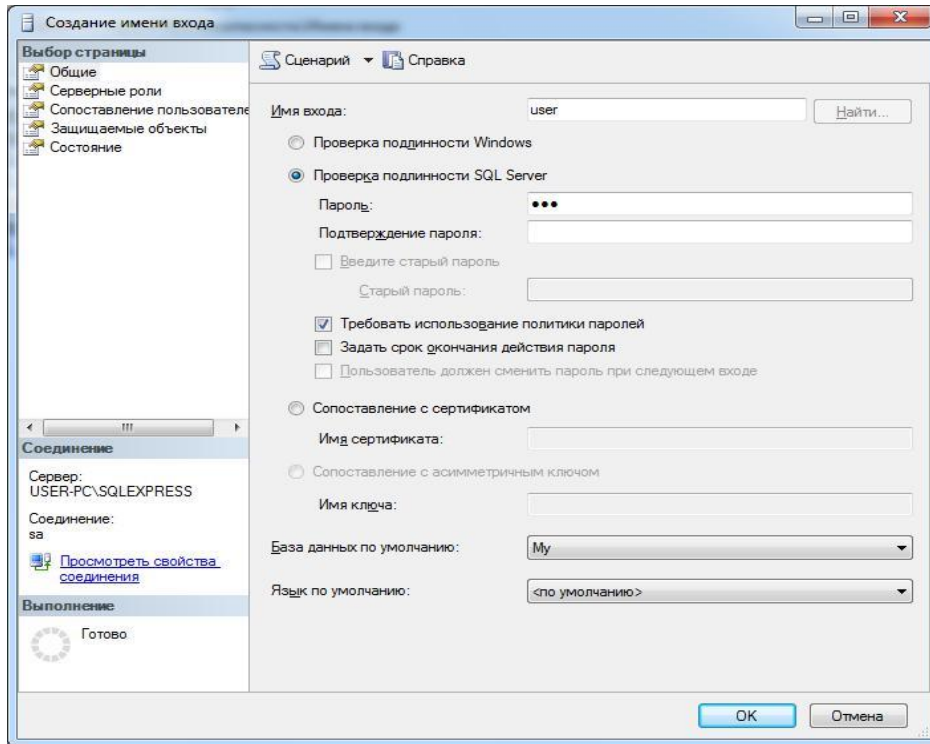


Рисунок 5.16 – Настройка параметров пользователя

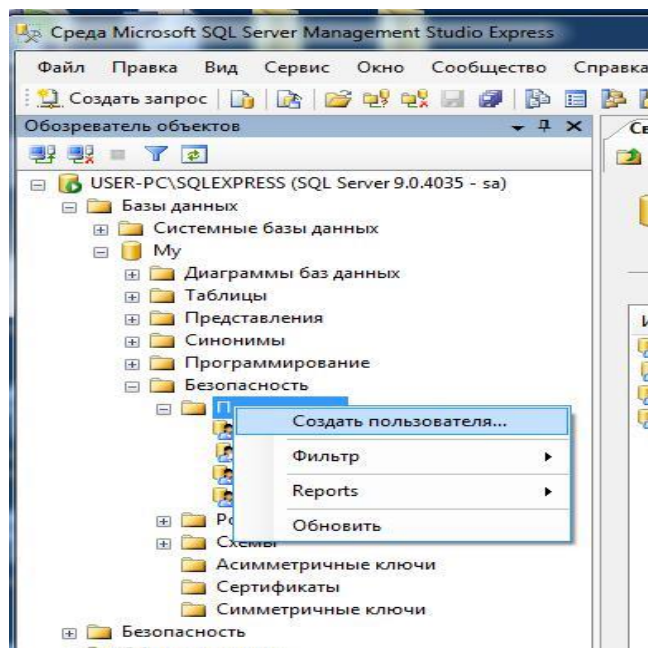


Рисунок 5.17 – Привязка созданного пользователя к БД

Серверные роли и роли БД в SQL Server:

- db_owner могут выполнять все действия по настройке и обслуживанию базы данных, а также удалять базу данных;
- db_securityadmin могут изменять членство в роли и управлять разрешениями. Добавление участников к этой роли может привести к непреднамеренному повышению прав доступа;
- db_accessadmin могут добавлять или удалять права удаленного доступа к БД для имен входа и групп Windows, а также имен входа SQL Server;
- db_backupoperator могут создавать резервные копии базы данных;
- db_ddladmin могут выполнять любые команды языка определения данных (DDL) в базе данных;
- db_datawriter могут добавлять, удалять или изменять данные во всех пользовательских таблицах;
- db_datareader могут считывать все данные из всех пользовательских таблиц;
- db_denypdatawriter не могут добавлять, изменять или удалять данные в пользовательских таблицах базы данных;
- db_denypdatareader не могут считывать данные из пользовательских таблиц базы данных.

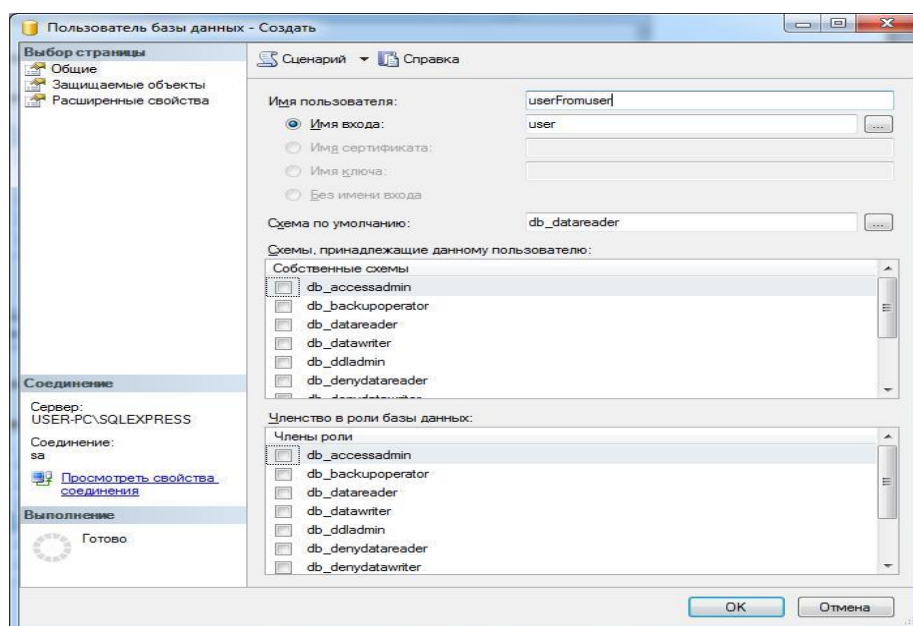


Рисунок 5.18 – Установка ограничений пользователя к выбранной БД

6 Разработка приложения

Основные вопросы, рассматриваемые на данном этапе:

1. Функции программы (в контексте перечня задач / процессов, подлежащих автоматизации).
2. Формы программы, их назначение и состав полей.
3. Отчёты программы с описаниями.

6.1 Основные моменты

Необходимый набор для работы с БД основных компонентов, используемых при разработке приложения:

- `SqlConnection` – устанавливает соединение с сервером;
- `SqlCommand` – осуществляет обмен запросов с сервером и получение результатов;
- `DataGridView` – таблица для вывода результатов.

К приложению будут подключены пространства имен для вызова дополнительных классов, например, таких как: `DataSet` и `DataAdapter`.

Программа должна выполнять следующие функции:

- ввод информации о товарах, реализуемых компанией;
- ввод информации о поставщиках и покупателях компании;
- ввод информации о менеджерах по продажам и заказам; – ввод и редактирование заказов покупателей; – ввод и редактирование приходных накладных; – ввод и редактирование расходных накладных;
- ввод и редактирование документов об оплате покупателей; – ввод и редактирование документов об оплате поставщикам; – формирование отчетов по выполнению заказов покупателей; – формирование отчетов по складским запасам товаров; – формирование отчетов по взаиморасчетам с поставщиками; – формирование отчетов по взаиморасчетам с покупателями.

6.2 Формы программы

В состав форм программы должны входить:

- главная форма, позволяющая перемещаться между функционалом приложения;
- формы редактирования справочников товаров, контрагентов, менеджеров; редактирование должно осуществляться «в списке»;
- формы журналов документов – заказов, приходных и расходных накладных; редактирование «в списке» не должно быть доступным; редактирование или создание нового документа должно выполняться на отдельной форме; должны присутствовать кнопки для создания нового документа и обновления списка документов;

– формы редактирования/создания отдельных документов (заказ, приходная накладная, расходная накладная, оплата покупателем, оплата поставщику); на форме должны размещаться элементы редактирования всех полей, являющихся частью соответствующей документу таблицы в БД; поле «номер» должно заполняться автоматически и не может изменяться пользователем, тем не менее, он должен видеть содержимое этого поля; должны присутствовать элементы редактирования «табличных частей» документов; необходимо существование удобного механизма подбора товаров в табличную часть документа; запись документа должна осуществляться по нажатию соответствующей кнопки.

6.3 Отчеты программы

Программа должна генерировать следующие типы отчетов:

– отчет по складским остаткам на определенную дату; должен принимать один входной параметр – дату, на которую будет получена информация; выходная форма отчета – таблица, содержащая колонки «Товар», «Остаток»;

– отчет по контрагентам должен принимать информацию о том, какие контрагенты нужны, и выводить информацию по выбранной группе контрагентов либо информацию о выбранном контрагенте; выходная форма отчета – таблица, содержащая колонки информации о контрагентах;

– отчет по взаиморасчетам с покупателями на определенную дату должен принимать один входной параметр – дату, на которую будет получена информация; выходная форма отчета – таблица, содержащая колонки «Покупатель», «Заказ»;

– отчет по взаиморасчетам с поставщиками на определенную дату должен принимать один входной параметр – дату, на которую будет получена информация; выходная форма отчета – таблица, содержащая колонки «Поставщик», «Получено».

7 Реализация

На данном этапе осуществляется физическая реализация базы данных и разработанных приложений, позволяющих пользователю формулировать требуемые запросы к БД и манипулировать данными в БД.

БД описывается на языке определения данных выбранной СУБД. В результате компиляции и выполнения его команд создаются схемы и пустые файлы базы данных. На этом же этапе определяются и все специфические пользовательские представления.

Прикладные программы реализуются с помощью языков третьего или четвертого поколений. Некоторые элементы этих прикладных программ будут представлять собой транзакции обработки данных, записываемые на языке манипулирования данными СУБД и вызываемые из программ на базовом языке программирования. На этом этапе создаются другие компоненты проекта – например, экраны меню, формы ввода данных и отчеты. Следует учитывать, что многие существующие СУБД имеют свои собственные непроцедурные инструменты четвертого поколения, позволяющие быстро создавать приложения. Это, прежде всего, разнообразные генераторы отчетов, форм, графических изображений и приложений.

На этом этапе реализуются также используемые приложением средства защиты БД и поддержки ее целостности. Одни из них описываются с помощью языка выбранной СУБД, а другие могут представлять собой утилиты СУБД или прикладные программы, реализующие требуемые функции.

7.1 Приложение администрирования пользователей

Для регистрации, удаления и разграничения прав доступа разных групп пользователей к программе были созданы дополнительные объект и приложение (рис. 7.1, листинги 4-5).

	Id	Ф.и.о.	Адрес	Паспорт	Телефон
	1	Иванов Иван И...	Красная 1, кв 8	0901 848493	+7903343244
	2	Путин Иван Ива...	Березанская 10...	0101 844493	2749991
	3	Смирнов Викто...	Мирная дом 3	0505 123143	--

Рисунок 7.1 – Окно администрирования пользователей

Удаление пользователя осуществляется посредством запроса из приложения, после чего срабатывает триггер, проверяющий связи таблиц.

Листинг 4 – Удаление пользователя из системы

```
sqlConnection1.Open();
sqlCommand1.CommandText = "delete from [bd].[dbo].[Сотрудники] WHERE
id =" +
dataGridView1.Rows[dataGridView1.CurrentRow.Index].Cells[0].Value.ToString(
) + """;
sqlCommand1.ExecuteNonQuery();
sqlConnection1.Close();
```

Листинг 5 - Триггер

```
create trigger [dbo].[del]
on [dbo].[Сотрудники]
for delete
as
delete Log from Users, deleted
where Log.id = deleted.id
```

7.2 Основное приложение

Запуск основного приложения начинается с запуска окна авторизации пользователя (рис. 7.2, листинг 6).

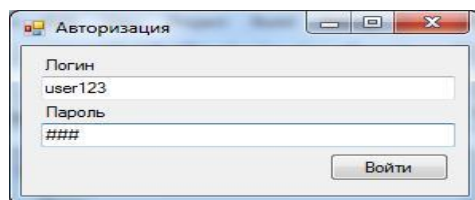


Рисунок 7.2 – Авторизация

Листинг 6 – Процедура авторизации

```
private void button1_Click(object sender, EventArgs e)
{
    SqlDataReader myReader;
    sqlCommand1.CommandText = "SELECT id, log, pass,dostup FROM
Users";
    sqlCommand2.CommandText = "SELECT id, log, pass, dostup
FROM logins WHERE (login like @login) AND (pass like @password)";
    try
    {
        sqlConnection1.Open();
        myReader =
sqlCommand1.ExecuteReader(); if
(!myReader.Read()) {
            sqlConnection1.Close();
            myReader.Close();
        }
        else
        {
            myReader.Close();
            sqlCommand2.Parameters["@login"].Value = textBox1.Text;
            sqlCommand2.Parameters["@password"].Value =
textBox2.Text;

            myReader =
sqlCommand2.ExecuteReader(); if
(!myReader.Read()) {
                MessageBox.Show("Доступ запрещен", "Ошибка");
                myReader.Close();
                sqlConnection1.Close();
            }
        }
    }
}
```

```

        return;
    }
    if (textBox1.Text == myReader.GetString(1) &&
        textBox2.Text == myReader.GetString(2) )
    {
        if (myReader.GetInt32(3) ==0)
        {
            Form2 f2 = new Form2 ();
            f2.Show ();
        }
        else
        {
            Form3 f3 = new Form3 ();
            f3.Show ();
        }
        this.Hide ();
        myReader.Close ();
    }
    else
    {
        MessageBox.Show ("Доступ запрещен", "Ошибка");
    }
    myReader.Close ();
    sqlConnection1.Close ();
}
}
catch (Exception ex)
{
    MessageBox.Show (ex.Message, "Ошибка");
}
}
private void Form1_Load(object sender, EventArgs e)
{
    sqlCommand2.Parameters.Add("@login",
System.Data.SqlDbType.Text, 0, "login");
    sqlCommand2.Parameters.Add("@password",
System.Data.SqlDbType.Text, 0, "login");
}
}

```

После проверки на подлинность пользователя запускается форма, соответствующая его задачам. Каждый пользователь работает со своим набором данных и отношений. Так пользователь, работающий с клиентом, может его добавить в базу, если он в ней отсутствует (рис. 7.3, листинг 7).

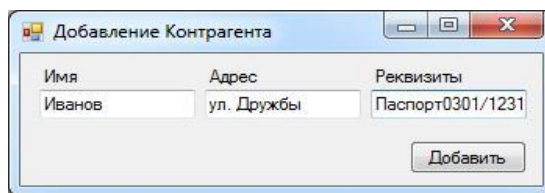


Рисунок 7.3 – Добавление клиента в БД

Листинг 7 – Запрос на добавление клиента

```

sqlConnection1.Open ();
    sqlCommand1.CommandText="INSERT INTO [bd].[dbo].[ Контрагенты]
([ФИО] , [Адрес], [Реквизиты]) VALUES
'" + textBox1.Text + "', '" + textBox2.Text + "', '" + textBox3.Text + "'";
    sqlCommand1.ExecuteNonQuery ();
    sqlConnection1.Close ();

```

Поддержка приложением поиска по ключевым параметрам записей в таблице БД является важнейшей его функцией. Рассмотрим быстрый поиск требуемых товаров. Прежде всего, перечень товаров выводится в таблицу:

```
sqlCommand1.CommandText = "select * FROM [bd].[dbo].[Товар]";
SqlDataAdapter da = new SqlDataAdapter(); DataSet ds = new
DataSet();
da.SelectCommand = sqlCommand1;
da.Fill(ds, "goods");
dataGridView1.DataSource = ds.Tables["goods"];
```

В результате проделанной операции будет получен набор данных DataSet(ds). Манипулируя этим объектом (рис. 7.4), можно менять содержание таблицы через DataSource следующим образом:

```
ds.Tables["goods"].DefaultView.RowFilter = "id = " + textBox1.Text
+ " or Наименование like \' " + textBox1.Text + "%' ";
```

Приведенный код, помещенный в обработчик события на изменение объекта TextBox, позволяет манипулировать выборкой и осуществлять фильтрацию.

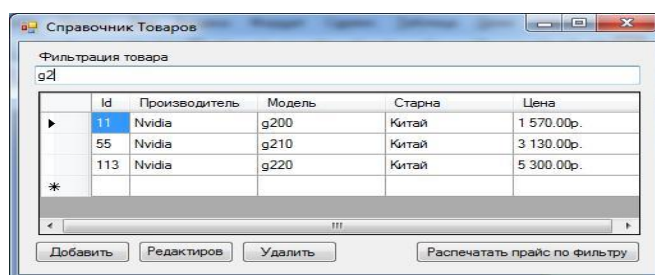


Рисунок 7.4 – Справочник товаров

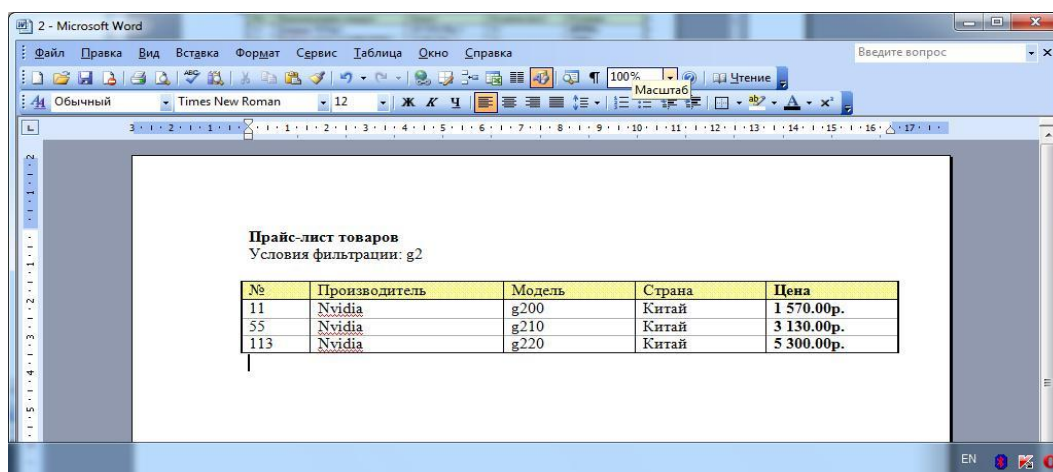


Рисунок 7.5 – Сформированный прайс-лист товаров

По информации в справочнике товаров можно формировать прайс-лист (в зависимости от условий фильтрации) либо по всем категориям товаров, либо по той группе товаров, которая выведена в таблицу. Вывод на печать прайс-листа представлен на рисунке 7.5.

Форма создания приходной накладной представлена на рис. 7.6, а код экспорта накладной в MS WORD в листинге 8.

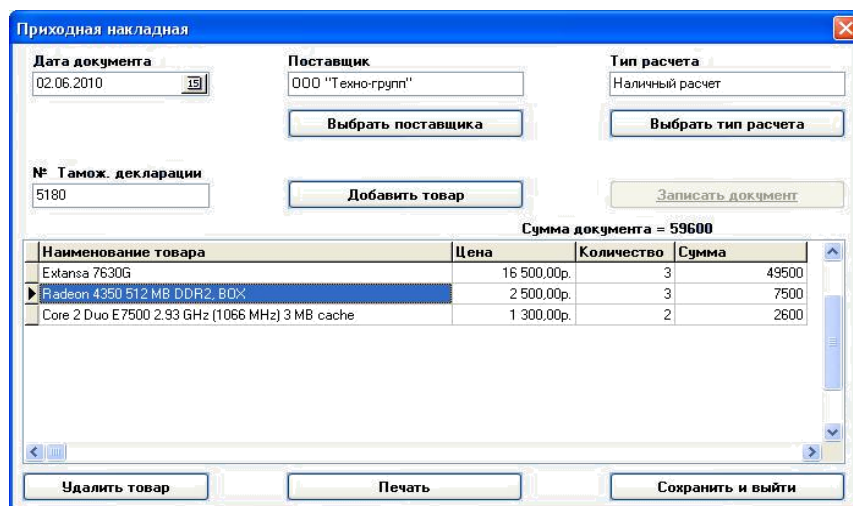


Рисунок 7.6 – Приходная накладная

Листинг 8 – Экспорт накладной в MS WORD

```

wordapp = new Word.Application();
wordapp.Visible = true;
Object filename = @"C:\1.doc";
Object confirmConversions = true;
Object readOnly = false;
Object addToRecentFiles = true;
Object passwordDocument = Type.Missing;
Object passwordTemplate = Type.Missing;
Object revert = false;
Object writePasswordDocument = Type.Missing;
Object writePasswordTemplate = Type.Missing;
Object format = Type.Missing;
Object encoding = Type.Missing;
Object oVisible = Type.Missing;
Object openConflictDocument = Type.Missing;
Object openAndRepair = Type.Missing;
Object documentDirection = Type.Missing;
Object noEncodingDialog = false;
Object xmlTransform = Type.Missing;
worddocument = wordapp.Documents.Open(ref filename,
    ref confirmConversions, ref readOnly, ref addToRecentFiles,
    ref passwordDocument, ref passwordTemplate, ref revert,
    ref writePasswordDocument, ref writePasswordTemplate,
    ref format, ref encoding, ref oVisible,
    ref openAndRepair, ref documentDirection, ref
noEncodingDialog, ref xmlTransform);
    object oMissing = System.Reflection.Missing.Value;
worddocument.Paragraphs.Add(ref oMissing);
for (int i = 0; i <= dataGridView1.Rows.Count+15; i++)
{ worddocument.Paragraphs.Add(ref oMissing); } wordparagraph =
worddocument.Paragraphs[1]; wordparagraph.Range.Text =
"Поставка №"+textBox10.Text+" от
"+dateTimePicker1.Text;
wordparagraph = worddocument.Paragraphs[3];
wordparagraph.Range.Text = "Поставщик";
wordparagraph = worddocument.Paragraphs[4];
wordparagraph = worddocument.Paragraphs[9];
Word.Range wordrange = wordparagraph.Range;
Object defaultTableBehavior = 65
Word.WdDefaultTableBehavior.wdWord8TableBehavior;

```

```

    Object autoFitBehavior =
Word.WdAutoFitBehavior.wdAutoFitContent;
    Word.Table wordtable1 =
worddocument.Tables.Add(wordrange, dataGridView1.Rows.Count, 5,
    ref defaultTableBehavior, ref
autoFitBehavior); Object style = "Сетка таблицы
3"; Object applystyle = true;
wordtable1.set_Style(ref style);
Word.Range wordcellrange;
    wordcellrange = wordtable1.Cell(1,1).Range;
    wordcellrange.Text = "№";
    wordcellrange = wordtable1.Cell(1, 2).Range;
    wordcellrange.Text = "Наименование товара";
    wordcellrange = wordtable1.Cell(1, 3).Range;
    wordcellrange.Text = "Цена";
    wordcellrange = wordtable1.Cell(1, 4).Range;
    wordcellrange.Text = "Количество";
    wordcellrange = wordtable1.Cell(1, 5).Range;
    wordcellrange.Text = "Сумма";
    for (int m = 0; m < wordtable1.Rows.Count-1; m++)
        for (int n = 0; n < wordtable1.Columns.Count; n++)
        {
            wordcellrange = wordtable1.Cell(m+2,
n+1).Range; if (dataGridView1[n, m].Value !=
null) {
                wordcellrange.Text = dataGridView1[n,
m].Value.ToString();
            }
        }
    wordparagraph =
worddocument.Paragraphs[worddocument.Paragraphs.Count];
    wordparagraph.Range.Text = "Исполнитель _____
Заказчик _____ ";

```

Результат выполнения описанного кода можно увидеть на рисунке 7.8. Аналогично выполняется вывод расходной накладной и кассовых ор-деров. Для вывода используется технология взаимодействия с MS Office, в частности с сервером Word. Компонентная модель объектов или технология COM (Component Object Model) предоставляет возможность одной программе (клиенту) работать с объектом другой программы (сервером). Для работы необходимо в проект добавить Microsoft Word 11.0 Object Library.

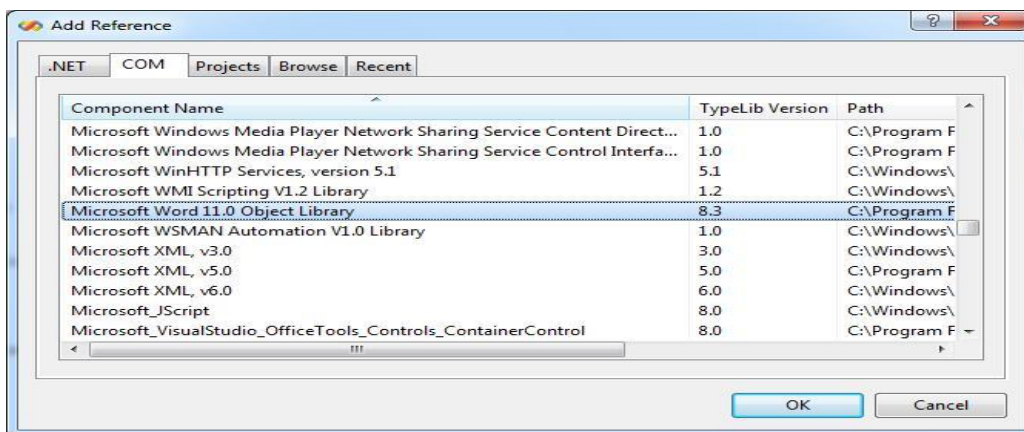


Рисунок 7.7 – Добавление COM объекта

Это осуществляется в «Solution Explorer», «References», где правой кнопкой мыши вызывается «Add Reference...». Найдя нужный объект и добавив его в приложение (рис 7.7), необходимо подключить его пространство имен в коде программы:

```
using Word = Microsoft.Office.Interop.Word;
```

и объявить в пределах класса основные используемые переменные:

```
private Word.Application wordapp;
private Word.Documents worddocuments;
private Word.Document worddocument;
private Word.Paragraphs wordparagraphs;
private Word.Paragraph wordparagraph;
```

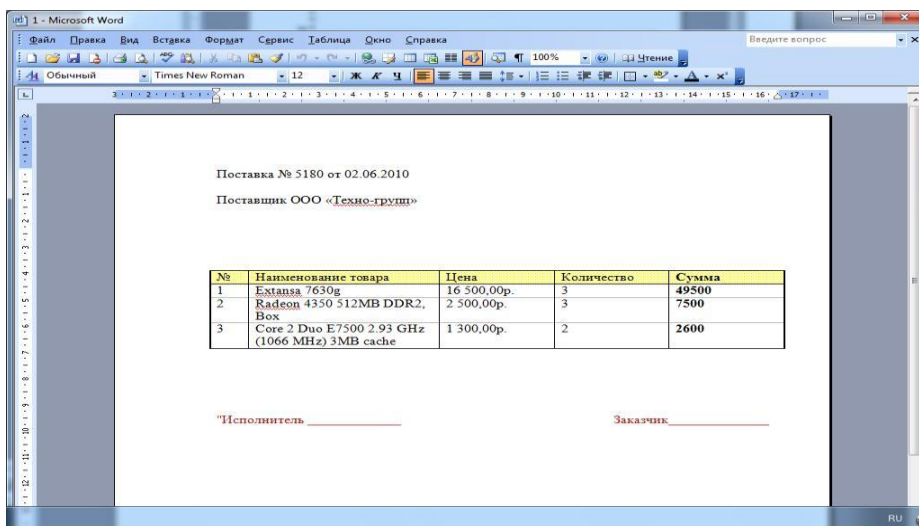


Рисунок 7.8 – Приходная накладная

В заключение демонстрируется реализация очень часто возникающей потребности вывода информации из БД на примере печати сведений о контрагентах фирмы (рис. 7.9). Вывод производится после фильтрации списка контрагентов, которая позволяет, например, найти:

- поставщиков или покупателей;
- конкретного поставщика или покупателя;
- всех клиентов, проживающих по определенной улице.

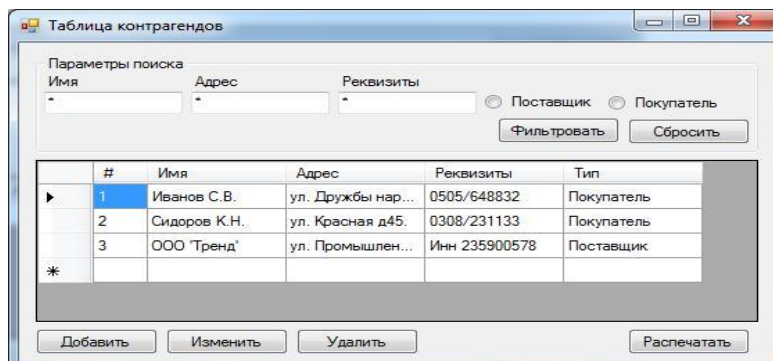


Рисунок 7.9 – Таблица контрагентов

8 Загрузка данных

На этом этапе созданные в соответствии со схемой БД пустые фай-лы, предназначенные для хранения информации, должны быть заполнены данными. Наполнение базы данных может протекать по-разному.

Если БД создается вновь, то для ввода информации используются созданные в процессе проектирования БД удобные специальные формы, которые позволят занести в базу заранее подготовленные данные.

Если же новая база данных предназначена для замены старой, то возникает проблема переноса данных в новую систему, причем очень часто с преобразованием формата их представления. В настоящее время любая СУБД имеет утилиту загрузки уже существующих файлов в новую БД. Для автоматического преобразования данных в нужный формат обычно требуется указать тип файла-источника и целевой БД.

9 Тестирование

Тщательное тестирование должен проходить любой программный продукт, тем более такой, как прикладные программы ИС. Стратегия тестирования должна предполагать использование реальных данных и должна быть построена таким образом, чтобы весь процесс выполнялся строго последовательно и методически правильно. Помимо обнаружения имеющихся в прикладных программах и, возможно, в структурах базы данных ошибок, сбор статистических данных на стадии тестирования позволяет установить показатели надежности и качества созданного программного обеспечения.

Пользователи новой системы должны принимать в процессе ее тестирования самое активное участие с целью выяснения, какие их информационные потребности остались неучтенными. Если такие обнаружатся, в случае необходимости осуществляется откат назад в процессе проектирования на те стадии, на которых возможно внести необходимые изменения.

10 Задания на курсовое проектирование

Темы курсовых проектов, как правило, связаны с одним из объектов автоматизации целого ряда предметных областей. Тема курсового проекта выдается каждому студенту индивидуально. Для определения задания студенту на курсовое проектирование используется приведенный ниже обширный перечень тем.

В общем случае тема курсового проекта студента определяется в соответствии с номером студента по журналу группы. При этом желательно, чтобы студент имел предварительное представление о ПРО и объекте автоматизации выбранного задания, которое в процессе курсового проектирования он должен углубить и расширить путем изучения специальной литературы и/или консультаций у специалистов. В связи с этим по согласованию с преподавателем, может производиться обмен заданиями при условии утверждения этих изменений в течение первой недели курсового проектирования.

Студент может (по согласованию с преподавателем) сам предложить задание на курсовое проектирование, аргументировано обосновав свое предложение. В качестве аргументов могут быть рассмотрены следующие ситуации:

- студент имеет доступ на предприятие, где ведётся деятельность по выбираемой им тематике;
- студент имеет перспективы трудоустройства на такое предприятие;
- студент имеет практические (выраженные в публикациях или действующих программных пакетах) наработки по выбираемой тематике;
- студент имеет возможность консультироваться с экспертом в выбранной предметной области.

Перечень возможных тем курсовых проектов

1. Торгово-развлекательный центр. Электронное хранилище документации.
2. Туристическое агентство. Клиенты, продажи.
3. Ресторанный бизнес / общепит. Управление меню.
4. Медицинские услуги. Врач – истории болезней.
5. Поставка специализированной техники. Торговля и склад.
6. Поставка специализированной техники. Контрагенты (клиенты / поставщики / субподрядчики на сервисе / кредитные организации).
7. Гостиничный бизнес. Фонд недвижимого имущества (здания / сооружения, номера помещения).
8. Строительство. Поставщики и подрядчики.
9. Предприятие оптовой торговли. Обработка заявок на поставку.
10. Медицинские услуги. Регистратура. Карточки пациентов, запись на приём по кабинетам.

11. Поставка специализированной техники. Сервис.
12. Ресторанный бизнес / общепит. Бар – торговля и склад.
13. Строительство. Отдел продаж.
14. Торгово-развлекательный / офисный центр. Служба работы с арендаторами: обработка заявок.
15. Проектная организация. Управление проектами.
16. Услуги связи. Техническая поддержка абонентов.
17. Производство мебели. Торговля и склад, управление индивидуальными заказами и оптовыми поставками.
18. Медицинские услуги. Инвентаризация оборудования.
19. Промышленное производство. Электронное хранилище проектной документации.
20. Строительство. Хранилище проектной документации.
21. Туристическое агентство. Туры, ценовые предложения.
22. Автомобильные перевозки. Тарификация и маршруты.
23. Услуги связи. Подрядчики и поставщики, цепочки поставок.
24. Санаторий. Фонд недвижимого имущества (здания / сооружения / помещения).
25. Офисный центр. Служба технического обслуживания.
26. Промышленное производство. Инвентаризация оборудования технологических линий.
27. Строительство. Парк техники.
28. Консалтинговая компания. Электронные консультации.
29. Промышленное производство. Ценовые предложения – формирование прайс-листов.
30. Ресторанный бизнес / общепит. Обработка заказов: зал / кухня.
31. Услуги связи. Инвентаризация инфраструктуры.
32. Предприятие оптовой торговли. Торговля и склад, управление запасами, цепочки поставки.
33. Недвижимость. Электронное хранилище документации.
34. Промышленное производство. Заказчики и статусы, график поставок, платежи / задолженности.
35. Автомобильные перевозки. Отдел продаж.
36. Строительство. Объекты – управление проектами.
37. Производство мебели. Контрагенты (поставщики / заказчики), цепочки поставки.
38. Туристический оператор. Туры, ценовые предложения.
39. Торгово-развлекательный / офисный центр. Отдел рекламы компании / календарь, поставщики.
40. Медицинские услуги. Регистратура – управление прайс-листом, занятость манипуляционных / операционных.
41. Рекламное агентство. Медиа-планирование (загрузка рекламных площадок, очереди контрагентов).

42. Медицинские услуги. Стационар - палаты, пациенты, курсы лечения, занятость манипуляционных / операционных.
43. Туристический оператор. Контрагенты (агентства, места размещения, перевозчики).
44. Торгово-развлекательный / офисный центр. Служба работы с арендаторами: карточки, платежи / взносы, история.
45. Торгово-развлекательный / офисный центр. Инвентаризация оборудования.
46. Ресторанный бизнес. Поставщики и управление запасами.
47. Проектная организация. Электронное хранилище проектной документации.
48. Автомобильные перевозки. Парк техники.
49. Санаторий. Регистратура - карточки отдыхающих, платные услуги, запись на приём.
50. Торгово-развлекательный / офисный центр. Служба безопасности: инциденты, вход / выход в административные помещения.
51. Торгово-развлекательный. Служба ИТ – обработка заявок.
52. Туристический оператор. Сделки, бронирование, загруженность направлений.
53. Производство мебели. Модельный ряд и индивидуальная комплектация.
54. Консалтинговая компания. Управление проектами.
55. Автосервис. Наряды и оперативное управление.
56. Промышленное производство. Управление номенклатурой - ценовые предложения, планирование производства.
57. Рекламное агентство. Карточки клиентов и контрагентов.
58. Санаторий. Управление запасами и цепочками поставок.
59. Гостиничный бизнес. Обслуживание номеров, платные услуги, работа с заявками.
60. Торгово-развлекательный / офисный центр. Фонд недвижимого имущества (здания / сооружения / помещения).
61. Промышленное производство. Управление запасами и цепочками поставок, поставщики / субподрядчики.
62. Гостиничный бизнес. Ресепшн, номерной фонд, клиенты.
63. Проектная организация. Служба поддержки заказчиков.
64. Недвижимость. Клиенты / агенты / сделки.
65. Медицинские услуги. Регистратура – расписание приёма врачей, работы лабораторий, запись на приём.
66. Инвестиционные компании. Управление инвестиционными портфелями.

Список литературы

1. Карпова, И.П. Базы данных. Учебное пособие. Курс лекций и материалы для практических занятий. – СПб.: Питер, 2013. — 240 с.
2. Голицына О.Л. Информационные системы: Учебное пособие / О.Л. Голицына, Н.В. Максимов, И.И. Попов. - 2-е изд. - М.: Форум: НИЦ ИНФРА-М, 2014. - 448 с.
3. Голицына О.Л. Основы проектирования баз данных: Учебное пособие / Голицына О.Л., Партыка Т.Л., Попов И.И., - 2-е изд. - М.:Форум, НИЦ ИНФРА-М, 2016. - 416 с.
4. Базы данных.Практическое применение СУБД SQL и NoSQL-типа для применения проектирования информационных систем: Учебное пособие / Мартишин С.А., Симонов В.Л., Храпченко М.В. - М.:ИД ФОРУМ, НИЦ ИНФРА-М, 2017. - 368 с.
<http://znanium.com/catalog.php?bookinfo=556449>

Приложение А (справочное)

Требования к оформлению пояснительной записки

Пояснительная записка к курсовому проекту должна быть оформлена на одной стороне сброшюрованных листов формата А4. Общий объем пояснительной записки в среднем должен быть около 30 листов, в том числе введение – не более 2-х листов.

Текст пояснительной записки набирается в формате WORD с использованием следующих установок: автоматические переносы; шрифт - Times New Roman; межстрочный интервал - 1,5; размер шрифта - 14 пт.

Поля страницы: левое - 30 мм; правое - 15 мм; верхнее - 15 мм; нижнее - 25 мм.

Страницы следует отформатировать с выравниванием по ширине. Текст программы в пояснительной записке и на электронном носителе в обязательном порядке должен быть в достаточной степени комментирован.

Листы нумеруют арабскими цифрами, соблюдая сквозную нумерацию по всему тексту, включая приложения. Титульный лист, задание на выполнение курсовой работы, реферат и содержание включают в общую нумерацию, но на них номера не ставятся. На последующих листах номера проставляются в центре нижней части листа без точки.

Иллюстрации, входящие в состав работы, следует именовать как «рисунки» и помещать после их первого упоминания. Наименование рисунка производится шрифтом основного текста работы. Внутри рисунка допускается использование различных шрифтов и уменьшение их размера. Нумерация рисунков в основном тексте и в приложениях – сквозная, например: «Рисунок 1 – ER-диаграмма базы данных». Подрисуночную надпись располагают по центру под иллюстрацией. Пояснения к рисунку располагают между иллюстрацией и наименованием. На каждый рисунок в основном тексте работы должна даваться ссылка (... на рисунке 1). Рисунок с его наименованием отделяется от текста пустыми строками.

Текст должен быть разбит на разделы. В разделе могут быть выделены подразделы, например, «2.3». Заголовки разделов и подразделов должны быть краткими и без переноса слов, точки после номера раздела (подраздела), а также в конце заголовка не ставятся. Тексты заголовков раздела и подраздела начинаются с абзачного отступа. Не допускается помещать заголовки отдельно от последующего текста. При подготовке параметров страницы и форматов текста в текстовой редакторе компьютера устанавливается «запрет висячих строк». Если заголовок состоит из двух предложений, их разделяют точкой. Все заголовки разделов и подразделов по тексту и в структурном элементе «Содержание» пишутся строчными буквами, кроме первой прописной. Каждый раздел и каждое приложение текстовой части должны начинаться с нового листа.

При оформлении работ с использованием компьютера применяются следующие способы выделения разделов:

- наименования разделов и подразделов выделяются п/ж шрифтом;
- для наименования раздела допускается размер шрифта до 16 пт.

Цифровой материал, как правило, оформляют в виде таблиц. Каждая таблица должна иметь заголовок, начинающийся со слова «Таблица». Начало заголовка таблицы соответствует левой границы ее сетки. Если таблица не помещается на одном листе, то ее переносят с надписью, например, «Продолжение таблицы 1» или «Окончание таблицы 1». При этом в месте разрыва нижняя линия сетки таблицы не отображается. Все таблицы имеют сквозную нумерацию. На все таблицы должны быть ссылки в тексте, например. Заголовки граф таблицы следует писать с прописной буквы в единственном числе, а подзаголовки - со строчной буквы, если они являются продолжением заголовка. Если подзаголовок имеет самостоятельное значение, то название подзаголовка пишется с прописной буквы.

Формулы нумеруются арабскими цифрами в пределах раздела. Номер формулы состоит из номера раздела и порядкового номера формулы (например, «(3.5)») и помещается в круглых скобках у правого поля листа.

Размеры знаков для формул: прописные - 5 мм, строчные 2,5 мм, показатели степени и индексы - 2 мм.

Ссылки на источник информации даются в виде заключенного в квадратные скобки номера источника и страниц, например, [25, с. 233-235]. В «Списке» источники могут располагаться в порядке их упоминания в тексте или в алфавитном порядке. Наименования источников удаленного доступа указываются в конце «Списка». В тексте курсового проекта допускаются общепринятые сокращения.

Правила оформления «Списка использованных источников»:

Для статей:

Лысов А.С. Разграничение доступа в реляционных БД на уровне строк и столбцов.//КубГТУ: 2008 г.

Для отдельных изданий:

Малыхина М.П. Базы данных: основы, проектирование, использование. Учебное пособие. – Спб.: БХВ - Петербург, 2007.- 512 с.

Для ресурсов удаленного доступа (локальных и интернет-сайтов):

Russian Software Developers Network [Электронный ресурс]: Online сообщество-библиотека, посвященное вопросам разработки программного обеспечения и баз данных. - Электрон, дан. (55 518 файлов). - М., [2009]. – Адрес доступа: <http://www.rsdn.ru/> - Главная страница.

Академия маркетинга и социально-информационных технологий
(ИМСИТ)

Кафедра Математика и вычислительная техника
Факультет Информатика и вычислительная техника

УТВЕРЖДАЮ

Зав. кафедрой Н.С.Нестерова

(дата, подпись, расшифровка подписи)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

по дисциплине Базы данных
(наименование дисциплины)

на тему _____
(тема курсового проекта)

Выполнил студент группы _____
(Ф.И.О.)

Допущен к защите _____

Руководитель (нормоконтролер) проекта _____
(подпись, дата, расшифровка подписи)

Защищен _____ Оценка _____
(дата)

Члены комиссии _____

(подпись, дата, расшифровка подписи)

Краснодар
20 г

Приложение В

(обязательное)

Задание на курсовое проектирование

Академия маркетинга и социально-информационных технологий (ИМСИТ)

Кафедра Математика и вычислительная техника

Факультет Информатика и вычислительная техника

УТВЕРЖДАЮ

Зав. кафедрой Н.С.Нестерова

(дата, подпись, расшифровка подписи)

ЗАДАНИЕ

на курсовое проектирование

Студенту _____ группы _____
факультета _____
направления _____
(код и наименование направления)

Тема проекта _____

Содержание задания: _____

Объем работы:

а) пояснительная записка _____ (электрон. носитель)

б) база данных (электрон. носитель)

в) программное приложение _____ (электрон. носитель) _____ г)
иллюстративная часть _____

Рекомендуемая литература _____

Срок выполнения: с _____ по _____ 20 г

Срок защиты: _____ 20 г

Дата выдачи задания: _____ 20 г

Дата сдачи проекта на кафедру: _____ 20 г

Руководитель проекта _____
(подпись)

Задание принял студент _____
(подпись)