



Министерство образования Республики Беларусь

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ

---

---

Кафедра «Таможенное дело»

Т.Р. Разорёнова

## БАЗЫ ДАННЫХ: РАЗРАБОТКА И УПРАВЛЕНИЕ

Методическое пособие  
для выполнения курсовой работы



Минск  
БНТУ  
2011

Кафедра «Таможенное дело»

Т.Р. Разорёнова

## БАЗЫ ДАННЫХ: РАЗРАБОТКА И УПРАВЛЕНИЕ

Методическое пособие  
для выполнения курсовой работы  
по дисциплинам  
«Информационные таможенные технологии»  
для студентов специальности 1–96 01 01 «Таможенное дело»  
и «Компьютерные информационные технологии»  
для студентов специальности 1–25 01 07  
«Экономика и управление на предприятии»

УДК 004.65:378.147.091.313 (075.8)

ББК 32.97 я7

Р 17

**Рецензенты:**

*Л.И. Дроздович*, доцент кафедры «Экономика и управление на предприятии»  
БНТУ, кандидат экономических наук;

*Н.Н. Гурский*, доцент кафедры «Программное обеспечение вычислительной  
техники и автоматизированных систем»  
БНТУ, кандидат технических наук

**Разорёнова, Т.Р.**

Р 17 Базы данных: разработка и управление: методическое пособие для выполнения курсовой работы по дисциплинам «Информационные таможенные технологии» для студентов специальности 1–96 01 01 «Таможенное дело» и «Компьютерные информационные технологии» для студентов специальности 1–25 01 07 «Экономика и управление на предприятии» / Т.Р. Разорёнова. – Минск: БНТУ, 2012. – 48 с.

ISBN 978-985-525-631-2.

Методическое пособие содержит общие положения по выполнению курсовой работы по разделу «Базы данных: разработка и управление» в курсах «Компьютерные информационные технологии» и «Информационные таможенные технологии», где целью работы является создание баз данных и управление ими в сфере деятельности таможенных органов или некоторой заданной предметной области. Изложены цели курсового проектирования, тематика работ. Приведены основные сведения из теории СУБД, описан процесс разработки базы данных, приводятся макет и правила оформления пояснительной записки.

УДК 004.65:378.147.091.313 (075.8)

ББК 32.97 я7

ISBN 978-985-525-631-2

© Разорёнова Т.Р., 2012

© БНТУ, 2012

## ВВЕДЕНИЕ

Во всем мире организации накапливают или уже накопили в процессе своей административно-хозяйственной деятельности большие объемы данных, в том числе и в электронном виде. Эти коллекции данных хранят в себе большие потенциальные возможности по извлечению новой аналитической информации, на основе которой можно и необходимо строить стратегию организации, выявлять тенденции развития рынка, находить новые решения, обуславливающие успешное развитие в условиях конкурентной борьбы. Для некоторых организаций такой анализ является неотъемлемой частью их повседневной деятельности, другие только начинают активно приступать к нему.

Для хранения, упорядочения и анализа больших объемов информации предназначены комплексы средств, именуемых информационными системами (ИС). Одними из видов таких, сегодня уже автоматизированных информационных систем (АИС), являются базы данных (БД), управляемые с помощью систем управления базами данных (СУБД), и информационно-аналитические системы, предназначенные как для хранения, так и для анализа хранимой информации, изучению которых и предназначен соответствующий раздел курса «Компьютерные информационные технологии».

Основным назначением информационных систем является оперативное обеспечение пользователя информацией о внешнем мире путем реализации отношений типа «вопрос-ответ».

Современные автоматизированные информационные системы:

- 1) ориентируются на конкретного пользователя, предоставляя ему удобный интерфейс и необходимые для работы функции;
- 2) должны обеспечивать надежность и продолжительность хранения данных, обладающих разными структурами;
- 3) должны обладать функциями, обеспечивающими ввод, обновление и удаление данных;
- 4) должны предоставлять данные для коллективной работы, обеспечивая согласованность их действий.

Поэтому важным является понимание вопросов трудоемкости тех задач, которые стоят перед разработчиками АИС и теми, кто является их конечными пользователями или заказчиками.

## 1 ЦЕЛИ КУРСОВОГО ПРОЕКТИРОВАНИЯ

Курсовая работа – важный этап обучения студента, где проявляются навыки ведения самостоятельной работы и овладения методикой проектирования и реализации задач по созданию автоматизированных информационных систем (АИС), основанных на базах данных (БД). При выполнении работы закрепляются теоретические знания, полученные при изучении курса. В ходе выполнения курсовой работы развивается индивидуальное творческое мышление студента и проявляются его способности к самостоятельному выбору и применению систем управления базами данных (СУБД) для решения задач в области баз данных.

Целью выполнения курсовой работы является:

- закрепление получаемых теоретических знаний;
- привитие навыков по информационному поиску и использованию необходимых материалов по исследуемой теме из научно-технической и справочной литературы;
- развитие навыков практического применения теоретических знаний для решения конкретных задач на базе самостоятельно спроектированной базы данных;
- привитие навыков самостоятельного изучения программного продукта и умения инсталляции необходимого программного обеспечения на персональный компьютер;
- освоение правил оформления курсовой работы в соответствии с требованиями, установленными стандартом высшего учебного заведения (вуза).

Темы курсовых работ определяются кафедрой, которая ведет дисциплины «Информационные таможенные технологии» и «Компьютерные информационные технологии» и отражает один из разделов этих курсов – управление базами данных в деятельности таможенных органов и технологии организации, хранения и обработки данных.

Курсовая работа состоит из двух частей: теоретическая и практическая.

**Теоретическая часть** направлена на проведение обзора и изучение современных технологий, применяемых при проектировании, разработке и эксплуатации автоматизированных информационных систем. Студент кратко описывает изученные по литературным источникам компьютерные информационные технологии, применяемые при разработке современных информационных систем. Если имеется возможность применить изученные технологии в практической части курсовой работы, это должно быть отражено в задании.

**Практическая часть** направлена на закрепление изученного материала и разработку небольшого приложения по управлению базой данных, построенной на основе спецификаций задания. Студент выполняет индивидуальное задание (или задания для группы студентов, но с индивидуальными спецификациями

для каждого студента) по созданию базы данных в заданной предметной области и разработке приложения по управлению этими данными. Приложение должно демонстрировать разработанные таблицы и запросы (представления), иметь дружественный интерфейс и позволять создавать отчетные документы для возможности проведения дальнейшего анализа данных. Наличие элементов автоматизации и применения клиент-серверных технологий позволит разработать реальный прототип информационной системы, который может стать основой для дальнейшего его развития в дисциплинах специализации, курсовом и дипломном проектировании.

Выбор среды разработки остается за студентом: или изучаемая в курсе, или может быть выбрана самостоятельно. При создании базы данных в системах MS SQL Server необходима разработка хранимых процедур, курсоров, триггеров и демонстрация работы с ними, а также других возможностей по защите и управлению данными в клиент-серверных СУБД.

Примерная тематика практической части курсовых работ:

- 1) разработка информационно-справочных систем и автоматизированных систем учета данных в определенной предметной области;
- 2) автоматизация рабочего места (АРМ).

В соответствии с выбранной темой курсовой работы руководитель работы выдает студенту задание (Приложение В), в котором указывается тема, исходные данные для выполнения работы (спецификации), определяется содержание работы, сроки выполнения курсовой работы, а также согласовывается календарный график выполнения отдельных этапов и всей работы.

Консультации проводятся в соответствии с утвержденным на кафедре графиком.

## 2 КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И ТЕРМИНОЛОГИЯ

Необходимость изучения современных технологий работы с накопленной информацией обусловлена тем, что системы управления базами данных существуют уже много лет, многие из них обязаны своим происхождением системам с неструктурированными файлами на больших ЭВМ. Наряду с общепринятыми современными технологиями в области систем управления базами данных начинают появляться новые направления, что обусловлено требованиями растущего бизнеса, все увеличивающимися объемами корпоративных данных и, конечно же, влиянием технологий Internet.

Концепция баз данных (БД) как метод представления и накопления данных в электронном виде сформировалась к середине 60-х годов прошлого века в фирме IBM. В 1969 году была создана первая СУБД для управления и манипулирования данными как самостоятельными информационными объектами. В 1970 году была предложена реляционная модель данных для БД, и на ее основе начали создаваться популярные ныне реляционные СУБД. В рамках реляционной модели с единых позиций были решены многие проблемы операционной (транзакционной) обработки данных.

С середины 80-х годов прошлого столетия стали интенсивно накапливаться электронные информационные массивы данных организаций, корпораций, научно-исследовательских учреждений. Ведущие мировые корпорации создали огромные электронные массивы конструкторской документации и документации по управлению производством. В это же время возникло четкое понимание, что сбор данных в электронном виде – не самоцель, а накопленные информационные массивы могут быть полезны. Первыми осознали этот факт в области управления бизнесом и производством. В накопленных данных организации находится «информационный снимок» хронологии ее поведения на рынке. Анализ истории административно-хозяйственной деятельности организации позволил существенно увеличить эффективность ее управления, эффективно организовать взаимоотношения с клиентами, производство и сбыт.

К ключевым факторам, определяющим развитие СУБД, относятся следующие:

- увеличение объемов данных и появление новых стандартов представления и обмена данными;
- появление новых типов данных (как структурированных, так и неструктурированных) – XML, электронной почты, календарей, файлов, документов, географических данных и т. п. в дополнение к традиционным типам данных, поддерживаемым на уровне баз данных и файловой системы;
- существенное снижение стоимости хранения самих данных, появление новых устройств хранения данных и уменьшение размеров накопителей.

В процессе своего развития приложения все больше и больше включают поддержку различных типов данных. Для обеспечения работоспособности таких приложений нужна соответствующая платформа, которая позволяла бы безопасно хранить такие данные, выполнять по ним поиск, анализировать их, синхронизировать и т. д.

## 2.1 Понятие предметной области

Понятие *предметной области* базы данных является одним из базовых понятий информатики и не имеет точного определения. Предметная область определяет ту часть реального мира, которая будет моделироваться и реализовываться в системах оперативной обработки данных или справочно-аналитических системах. *Предметные области* в базах данных формируются в соответствии с направлениями деятельности организации и определяют наиболее общие вытекающие из ее семантики критерии и требования к системе. Чтобы определить список *предметных областей* для таких систем, необходимо определить основные виды деятельности организации – например, продажи, производство, клиенты и т. д.

Проектирование базы данных представляет собой процесс отображения исследуемых явлений реального мира, называемых *предметной областью*, в виде данных в памяти компьютера. Основная цель проектирования базы данных – это сокращение избыточности хранимых данных, а следовательно, экономия объема используемой памяти, уменьшение затрат на многократные операции обновления и устранение возможности возникновения противоречий из-за хранения в разных местах сведений об одном и том же объекте. Перед созданием БД следует располагать описанием предметной области, а также иметь информацию для удовлетворения запросов пользователя и потребности в обработке данных. На основе такого описания на этапе проектирования определяется состав и структура данных предметной области.

Анализ *предметной области* позволяет выделить ее сущности, определить первоначальные требования к функциональности и границы проекта.

## 2.2 Этапы проектирования базы данных

Проектирование БД разбивают на два основных этапа: *инфологическое* (концептуальное) и *даталогическое*, которое, в свою очередь, подразделяется на *логическое* проектирование и *физическое*.

На этапе *инфологического* проектирования имеет место восприятие реальной действительности, абстрагирование, изучение и описание предметной области. Целью концептуального проектирования является разработка БД на основе описания предметной области. Это описание должно содержать совокупность документов и данных, необходимых для загрузки в БД, а также сведения об объектах и процессах, характеризующих предметную область. Это может быть за-



фиксировано с помощью диаграмм потоков данных (Data Flow Diagrams – DFD-диаграмм). Разработка БД начинается с определения состава данных, подлежащих хранению в базе для обеспечения выполнения запросов пользователя. Далее производится их анализ и структурирование.

На этапе *логического* проектирования производится организация выделенных данных в форму, принятую в выбранной СУБД. Целью логического проектирования является преобразование концептуальной модели в логическую. Для реляционной БД этот этап состоит в разработке структуры объектов, определении связей между ними и выявлении ключевых реквизитов. Результат – построение диаграмм «сущность–связь» (Entity Relationship – ER-диаграмм). Далее для построения оптимальной структуры базы данных проводится процесс *нормализации*.

Этап *физического* проектирования дополняет логическую модель характеристиками, которые необходимы для определения способов физического хранения и использования БД, объема памяти и типа устройства для хранения. Результат – построение структуры таблиц с описанием типов данных и их декларативных ограничений.

### 2.3 Основные понятия ER-диаграмм

Основным средством разработки логической модели данных в настоящий момент являются различные варианты ER-диаграмм (Entity Relationship, диаграммы «сущность–связь»). ER-диаграмма позволяет графически представить все элементы логической модели согласно простым, интуитивно понятным, но строго определенным правилам – нотациям.

Термины, которыми оперирует реляционная модель данных, имеют соответствующие «табличные» синонимы, представленные в таблице 1.

Таблица 1 – Соответствие реляционных и табличных терминов

Реляционный термин	Соответствующий «табличный» термин
База данных	Набор таблиц
Схема базы данных	Набор заголовков таблиц
Отношение	Таблица
Заголовок отношения	Заголовок таблицы
Тело отношения	Тело таблицы
Атрибут отношения	Наименование столбца таблицы
Кортеж отношения	Строка таблицы
Степень (арность) отношения	Количество столбцов таблицы
Мощность отношения	Количество строк таблицы
Домены и типы данных	Типы данных в ячейках таблицы

**Определение 1. Сущность** – это класс однотипных объектов, информация о которых должна быть учтена в модели.

Каждая сущность должна иметь наименование, выраженное существительным в единственном числе. Примерами сущностей могут быть такие классы объектов как *Преподаватель*, *Студент*, *Отдел*.

**Определение 2. Экземпляр сущности** – это конкретный представитель данной сущности. Например, представителем сущности *Преподаватель* может быть *Преподаватель Егоров*.

Экземпляры сущностей должны быть различимы, т. е. сущности должны иметь некоторые свойства, уникальные для каждого экземпляра этой сущности.

**Определение 3. Атрибут сущности** – это именованная характеристика, являющаяся некоторым свойством сущности. Наименование атрибута должно быть выражено существительным в единственном числе (возможно, с характеризующими прилагательными). Примерами атрибутов сущности *Преподаватель* могут быть такие атрибуты как *Табельный номер*, *Фамилия*, *Имя*, *Отчество*, *Должность*, *Зарплата* и т. п.

**Определение 4. Ключ сущности** – это избыточный набор атрибутов, значения которых в совокупности являются уникальными для каждого экземпляра сущности. Избыточность заключается в том, что удаление любого атрибута из ключа нарушает его уникальность. Сущность может иметь несколько различных ключей.

**Определение 5. Связь** – это некоторая ассоциация между двумя сущностями. Одна сущность может быть связана с другой сущностью или сама с собою. Связи позволяют по одной сущности находить другие сущности, связанные с ней. Например, связи между сущностями могут выражаться следующими фразами – «ПРЕПОДАВАТЕЛЬ может вести несколько ПРЕДМЕТОВ», «каждый СТУДЕНТ обязан числиться ровно в одной ГРУППЕ».

Каждая связь может иметь один из следующих типов связи: «один-к-одному», «один-ко-многим» или «многие-к-одному», «многие-ко-многим».

Связи могут быть обязательными или необязательными. При выделении связей акцент делается на выявление их характеристик. При создании связи в одной из сущностей, называемой дочерней сущностью, создается новый атрибут, называемый *внешним ключом*. Связь должна именоваться глаголом или глагольной фразой, где имя связи выражает некоторое ограничение или бизнес-правило и облегчает чтение диаграммы.

## 2.4 Способы документирования ER-модели

База данных создается в несколько этапов, на каждом из которых необходимо согласовывать структуру данных с заказчиком и, что самое важное, подвергать созданную структуру данных экспертизе внутри команды, которая создает систему. Поэтому представление данных должно быть простым и понят-

ным всем заинтересованным лицам. Именно по этой причине, наибольшее распространение получило представление базы данных под названием «сущность–связь» (Entity Relationship), которое также известно как *ER-диаграмма*.

Метод моделирования «сущность–связь» был предложен С. Ченом в 1976 году. В диаграммах Чена информационные объекты (сущности) изображаются прямоугольниками, связи – ромбами или шестиугольниками, атрибуты – овалами. Сущности соединяются линиями, над которыми могут проставляться степени связи (1 или буква M) и необходимые пояснения.

Для представления элементов модели были разработаны несколько других графических нотаций: нотация Мартина, нотация IDEF1X, нотация Баркера и др. Проектировщик может выбрать графическую нотацию по своему вкусу.

Построение ER-диаграмм, как правило, ведется с использованием CASE-средств, например в MS Office Visio 2007 или в ERwin.

В таблице 2 приведены обозначения для двух способов документирования ER-модели.

Таблица 2 – Способы документирования ER-модели

Определение	Обозначение в нотации Баркера	Обозначение в нотации Чена
Сущность	Каждая сущность в модели изображается в виде прямоугольника с наименованием: 	Каждая сущность в модели изображается в виде прямоугольника с наименованием: 
Атрибут	Атрибуты изображаются в пределах прямоугольника, определяющего сущность: 	Атрибут – овал с названием атрибута: 
Ключевые атрибуты	Ключевые атрибуты изображаются на диаграмме подчеркиванием: 	Ключевой атрибут – овал с подчеркнутым названием ключевого атрибута: 

Продолжение таблицы 2

Определение	Обозначение в нотации Баркера	Обозначение в нотации Чена
Связи	Графически связь изображается линией, соединяющей две сущности: 	Связь – ромб с названием связи-отношения: 
Типы связей	<p>Один-к-одному</p> <p>Один-ко-многим</p> <p>Многие-ко-многим</p>	<p>1 — 1</p> <p>Один-к-одному</p> <p>1 — M</p> <p>Один-ко-многим</p> <p>M — N</p> <p>Многие-ко-многим</p>
Виды связей	<p>Может</p> <p>Должен</p>	<p>Может</p> <p>Должен</p>
Примеры	См. рисунок 1 и рисунок 2	См. рисунок 3

Как правило, ER-диаграммы используются для презентаций и обсуждения структуры данных с экспертами предметной области. Ниже приведены примеры разработки простой концептуальной ER-модели в различных нотациях.

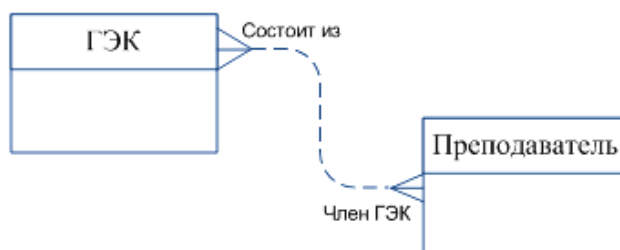


Рисунок 1 – Пример связи «многие-ко-многим»

Связь «многие-ко-многим» может быть создана только на уровне логической модели. Ее необходимо преобразовать к связи «один-ко-многим», изменив глагольную форму в существительное для связующей сущности, которая может иметь дополнительные атрибуты и иметь ключ.

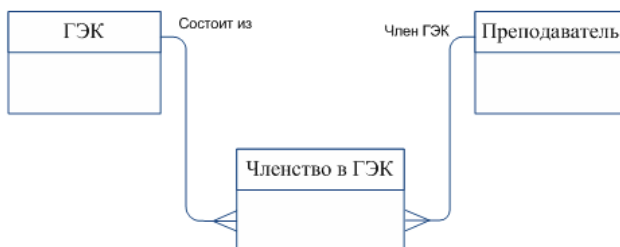


Рисунок 2 – Пример преобразования связи M:N к виду «один-ко-многим»

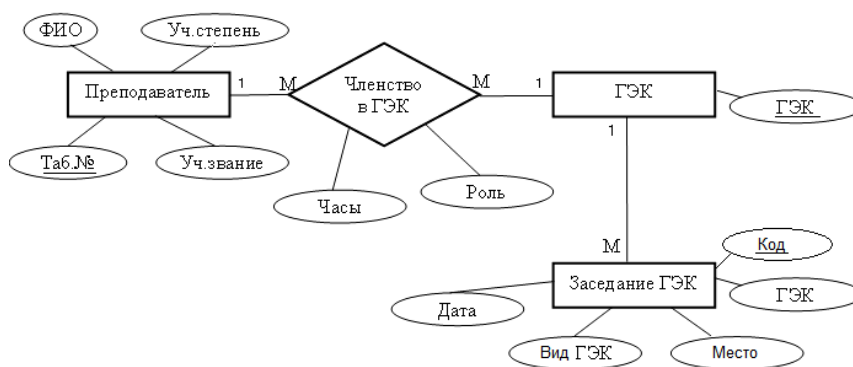


Рисунок 3 – Пример диаграммы Чена

Независимо от выбранной нотации, действия проектировщика базы данных при ER-моделировании сводятся к следующему алгоритму.

Для каждой сущности предметной области базы данных необходимо:

- получить список атрибутов сущности;
- определить функциональные зависимости;
- определить уникальный идентификатор сущности;
- выполнить нормализацию сущности;
- назначить первичные ключи новых, полученных в результате нормализации сущностей;
- сформировать бизнес-правила поддержки целостности сущности.

Для каждой связи между сущностями необходимо:

- определить степень связи;
- определить обязательность вхождения сущности в связь;
- разрешить связи «многие-ко-многим»;
- сформировать бизнес-правила поддержки целостности связей;
- документировать логическую модель реляционной базы данных;
- проверить логическую модель реляционной базы данных.

## 2.5 Нормализация модели «сущность–связь»

**Нормализация** – процесс проверки и реорганизации сущностей и атрибутов с целью удовлетворения требований к реляционной модели данных: каждый атрибут должен быть определен для своей сущности. Это достигается путем разбиения таблиц на две или более, обладающих лучшими свойствами при включении, изменении и удалении данных, и делается не столько с целью экономии памяти, сколько для исключения возможной противоречивости хранимых данных. Окончательная цель нормализации сводится к получению такого проекта базы данных, в котором каждый факт появляется лишь в одном месте, т. е. исключена избыточность информации. Процесс нормализации сводится к последовательному приведению структуры данных к нормальным формам – формализованным требованиям к организации данных. Известно 6 нормальных форм:

- первая нормальная форма (1NF);
- вторая нормальная форма (2NF);
- третья нормальная форма (3NF);
- нормальная форма Бойса–Кодда (усиленная 3NF);
- четвертая нормальная форма (4NF);
- пятая нормальная форма (5NF).

На практике обычно ограничиваются приведением данных к третьей нормальной форме. Отношения в 3NF являются самыми «хорошими» с точки зрения того, что устранены аномалии обновления, удаления и вставки и требуются только стандартные триггеры для поддержания ссылочной целостности. Для углубленного изучения нормализации можно рекомендовать книгу К. Дж. Дейта «Введение в системы баз данных» [1].

Нормальные формы основаны на понятии функциональной зависимости.

**Первая нормальная форма (1NF).** Сущность находится в первой нормальной форме тогда и только тогда, когда все атрибуты содержат атомарные значения.

Для приведения сущности к *первой нормальной форме* следует:

- разделить сложные атрибуты на атомарные;
- создать новую сущность;
- перенести в нее все «повторяющиеся» атрибуты;
- выбрать возможный ключ для нового первичного ключа или создать его;
- установить идентифицирующую связь от прежней сущности к новой, первичный ключ прежней сущности станет внешним ключом для новой сущности.

**Вторая нормальная форма (2NF).** Сущность находится во второй нормальной форме, если она находится в первой нормальной форме и каждый неключевой атрибут полностью зависит от первичного ключа (не должно быть зависимости от части ключа). Вторая нормальная форма имеет смысл только для сущностей, имеющих сложный первичный ключ.

Для приведения сущности ко *второй нормальной форме* следует:

- выделить атрибуты, которые зависят только от части первичного ключа, создать новую сущность;
- поместить атрибуты, зависящие от части ключа, в их собственную (новую) сущность;
- установить идентифицирующую связь от прежней сущности к новой.

**Третья нормальная форма (3NF).** Сущность находится в третьей нормальной форме, если она находится во второй нормальной форме и никакой неключевой атрибут не зависит от другого неключевого атрибута (не должно быть транзитивной зависимости между неключевыми атрибутами).

Для приведения сущности к *третьей нормальной форме* следует:

- создать новую сущность и перенести в нее атрибуты с одной и той же зависимостью от неключевого атрибута;

– использовать атрибут (атрибуты), определяющий (определяющие) эту зависимость, в качестве первичного ключа новой сущности;

– установить неидентифицирующую связь от новой сущности к старой.

Таким образом, переход от ненормализованных отношений к отношениям в 3NF может быть выполнен при помощи алгоритма нормализации. Алгоритм нормализации заключается в последовательной декомпозиции отношений для устранения функциональных зависимостей атрибутов от части сложного ключа (приведение к 2NF) и устранения функциональных (транзитивных) зависимостей неключевых атрибутов друг от друга (приведение к 3NF).

## 2.6 Основные объекты реляционных БД

К числу основных объектов реляционных БД относятся таблица, представление и пользователь.

**Таблица** (Table) является базовой структурой реляционной БД. Она представляет собой единицу хранения данных – отношение. Таблица представляет собой двумерный массив данных, в котором колонка определяет значение, а строки содержат данные. Таблица идентифицируется в БД своим уникальным именем.

**Представление** (View) – это поименованная, динамически поддерживаемая СУБД выборка из одной или нескольких таблиц базы данных. В MS Access это **запрос**. Оператор выборки ограничивает видимые пользователю данные. Обычно СУБД гарантирует актуальность представления: его формирование производится каждый раз, когда представление используется. Иногда представления или запросы называют виртуальными таблицами.

**Пользователь** (User) – это объект, обладающий возможностью создавать или использовать другие объекты базы данных и запрашивать выполнение функций СУБД, таких, как организация сеанса работы, изменение состояния базы данных и т. д.

При использовании клиент-серверных баз данных MS SQL Server имеется ряд дополнительных объектов, таких, как определенные пользователем типы данных и правила.

**Определенные пользователем типы данных** (User-defined data types) представляют собой определенные пользователем типы атрибутов (домены), которые отличаются от поддерживаемых СУБД типов. Они определяются на основе встроенных типов.

**Правила** (Rules) – это декларативные выражения, ограничивающие возможные значения данных. Для формулировки правила используются допустимые предикатные выражения SQL.

Для обеспечения эффективного доступа к данным в реляционных СУБД поддерживается ряд других объектов, в частности индекс.

**Индекс (Index)** – это объект базы данных, создаваемый для повышения производительности выборки данных и контроля уникальности первичного ключа (если он задан для таблицы).

Для обработки данных специальным образом или для реализации поддержки ссылочной целостности базы данных в MS SQL Server используются объекты: хранимая процедура, функция, триггер. С помощью этих объектов базы данных можно выполнять так называемую построчную обработку данных, для чего используются курсоры. С точки зрения приложений баз данных построчная обработка – это последовательная выборка данных по одной строке, ее обработка и переход к обработке следующей строки.

Данные объекты реляционной базы данных представляют собой программы, т. е. исполняемый код. Этот код обычно называют серверным кодом, поскольку он выполняется компьютером, на котором установлено ядро реляционной СУБД. Планирование и разработка такого кода является одной из задач проектировщика реляционной базы данных.

**Хранимая процедура (Stored procedure)** – это объект базы данных, представляющий поименованный набор команд SQL и/или операторов специализированных языков обработки программирования базы данных.

**Функция (Function)** – это объект базы данных, представляющий поименованный набор команд SQL и/или операторов специализированных языков обработки программирования базы данных, который при выполнении возвращает значение – результат вычислений.

**Триггер (Trigger)** – это объект базы данных, который представляет собой специальную хранимую процедуру. Эта процедура запускается автоматически, когда происходит связанное с триггером событие (например, вставка, изменение или удаление строки в таблице).

**Курсор (Cursor)** – это механизм, предоставляющий пользователю возможность доступа к любой строке выборки по ее номеру. Курсор представляет своего рода окно, накладываемое на результат выборки. Пользователь может работать в каждый момент времени только с одной строкой, но, перемещая окно, он способен получить доступ к любой строке выборки.

Для эффективного управления разграничением доступа к данным поддерживается объект роль.

**Роль (Role)** – объект базы данных, представляющий собой поименованную совокупность привилегий, которые могут назначаться пользователям, категориям пользователей или другим ролям.

## 2.7 Структура таблиц и целостность данных

Когда приступают к проектированию таблиц базы данных, требуется принять некоторые решения, относящиеся к их структуре. К этим решениям относится определение того, какие элементы данных должны храниться в этих



таблицах и как таблицы будут связаны друг с другом. Эта работа поможет представить общую картину базы данных, прежде чем вы углубитесь в создание таблиц. Ниже дан перечень вопросов для принятия решений:

– Какие данные будут храниться в каждой из таблиц?

– Какие колонки будут созданы для хранения данных и какие у них будут имена?

– Какие будут требования к диапазону данных, хранящихся в колонках (какие данные разрешено в них хранить), и какие типы данных должны применяться для каждой из колонок?

– Будут ли иметься колонки, которые могут содержать null-значения, или же вместо этого могут применяться значения по умолчанию?

– Какие колонки станут первичными ключами, а какие – внешними ключами?

– Какие типы ограничений должны использоваться?

– Для какой колонки или колонок должны быть определены индексы?

– Какие пользователи должны иметь доступ к тем или иным таблицам?

Необходимо найти ответы на как можно большее количество этих вопросов о проектировании системы и записать их на листок бумаги или в компьютерной программе для рисования схем, чтобы осознать общую конструкцию таблиц вашей базы данных, прежде чем начать создавать их. Нужно определить, каким образом пользователи будут осуществлять доступ к данным. Например, можно определить, что некоторая таблица будет предназначена только для чтения или же в нее будут производиться вставки, удаления или обновления данных. Необходимо предусмотреть, какие запросы будут выполняться чаще всего и из каких колонок будут извлекаться данные. Надо определить, какая информация действительно необходима для базы данных, а какую хранить не надо: она будет получена в результате вычислений.

При задании типа данных у колонки задаются следующие атрибуты:

– тип данных, которые могут содержаться в колонке (например, символьные данные, целые числа или изображения);

– размер (длина) данных, хранимых в колонке;

– точность чисел (этот атрибут применяется только для числовых типов данных), т.е. количество цифр, содержащихся в числах;

– масштаб чисел (этот атрибут применяется только для числовых типов данных), т.е. количество цифр, способных помещаться справа от десятичной точки;

– null-значение (null value) – это неизвестное значение, для которого применяется обозначение NULL. Колонки, способные хранить null-значения, могут оказаться полезными в случаях, когда необходимая информация пока что недоступна для вас. Как правило, не следует применять null-значения. Из-за них запросы и обновления становятся более сложными, кроме того, к колонкам, способным хранить null-значения, нельзя применять некоторые настройки, такие, как первичные ключи и свойство IDENTITY. Вместо применения null-значений в колонке можно задавать значения по умолчанию (default value).

**Декларативные ограничения.** Ограничения автоматически обеспечивают целостность данных. Ограничения задают правила, которые определяют значения данных, допустимые для какой-либо колонки. Они позволяют ограничивать значения данных, которые вводятся в колонку, чтобы в этой колонке не оказались неверные значения. Например, с помощью ограничения можно ограничить значения колонки для экзаменационной оценки диапазоном значений целого типа от 1 до 10, а для выбора пола – принадлежностью набору символьных значений *М* или *Ж*. В результате любые значения вне этого диапазона нельзя будет ввести в данную колонку.

Ограничение NOT NULL задается в описании колонки, чтобы воспрепятствовать вставке null-значений в эту колонку (в противоположность ограничению NULL, которое разрешает присваивать null-значения).

Ограничение UNIQUE устанавливает, что в колонке или наборе колонок не будут допускаться дублированные значения; иными словами, обеспечивается уникальность значений в этой колонке или наборе колонок.

Ограничение PRIMARY KEY используется, чтобы задать первичный ключ таблицы, представляемый колонкой или набором колонок, уникальным образом идентифицирующих строку таблицы. Поскольку первичный ключ идентифицирует строку, соответствующая колонка никогда не содержит значения NULL. В этом состоит отличие ограничения PRIMARY KEY от ограничения UNIQUE, которое допускает null-значения. Таблица может иметь только одно ограничение PRIMARY KEY.

Ограничение FOREIGN KEY определяет внешний ключ, который задает связь между двумя таблицами. Колонка или колонки внешнего ключа одной таблицы ссылаются на потенциальный ключ (одна или несколько колонок) в другой таблице. При вставке строки в таблицу с ограничением FOREIGN KEY значения, которые должны быть внесены в колонку или колонки, определенные как внешний ключ, сравниваются со значениями в потенциальном ключе ссылочной таблицы.

Декларативная поддержка ограничений целостности заключается в определении ограничений средствами языка определения данных (DDL – Data Definition Language).

Например, следующий оператор создает таблицу Куратор и определяет для нее некоторые ограничения целостности:

```
CREATE TABLE Куратор
(Группа_Id INTEGER,
ФИО CHAR(30) NOT NULL,
FOREIGN KEY (Группа_Id) REFERENCES Группа(Группа_Id)
ON UPDATE CASCADE ON DELETE CASCADE);
```

**Ссылочная целостность.** При работе БД должна обеспечиваться целостность данных. Под целостностью данных понимают обеспечения целостности связей между записями в таблицах при удалении записей из первичных таблиц. То есть при удалении записей из первичных таблиц, автоматически должны удаляться связанные с ними записи из вторичных таблиц.

В случае несоблюдения целостности данных со временем в БД накопится большое количество записей во вторичных таблицах, связанных с несуществующими записями в первичных таблицах, что приведет к сбоям в работе БД и ее засорению неиспользуемыми данными.

Для обеспечения целостности данных используют диаграммы и триггеры.

**Диаграммы** – это компоненты БД, которые блокируют удаление записей из первичных таблиц, если существуют связанные с ними записи во вторичных таблицах. Следовательно, диаграммы предотвращают нарушение целостности данных. Для поддержания ссылочной целостности используется каскадирование: разрешить выполнение требуемой операции, но внести каскадные изменения в другие отношения так, чтобы не допустить нарушения ссылочной целостности.

**Триггеры** – это аналог процедур обработчиков событий в Visual Basic. То есть они выполняют команды SQL, если происходят какие-либо действия с таблицей (например, добавление, изменение или удаление записей). При помощи триггеров можно организовать автоматическое удаление записей из вторичной таблицы при удалении связанной с ними записи из первичной таблицы. Если в базе данных определен некоторый триггер, то он запускается автоматически всегда при возникновении события, с которым этот триггер связан. Очень важным является то, что пользователь не может обойти триггер. Триггер срабатывает независимо от того, кто из пользователей и каким способом инициировал событие, вызвавшее запуск триггера. Таким образом, основное назначение триггеров – автоматическая (процедурная) поддержка целостности базы данных.

Структуры таблиц определяют физическую модель данных: отношения, разработанные на стадии формирования логической модели данных, преобразуются в таблицы, атрибуты становятся столбцами таблиц, для ключевых атрибутов создаются уникальные индексы, домены преобразуются в типы данных, принятые в конкретной СУБД.

Средствами документирования разработки физической модели будем считать описания таблиц и их атрибутов. В таблице 3 приведено описание двух связанных отношений (объектов БД), представленных во фрагменте ER-диаграммы на рисунке 4, определены таблицы и названия полей, заданы ограничения. В колонке «Признак ключа» определяется декларативное определение первичного ключа и внешнего ключа – поля связи.

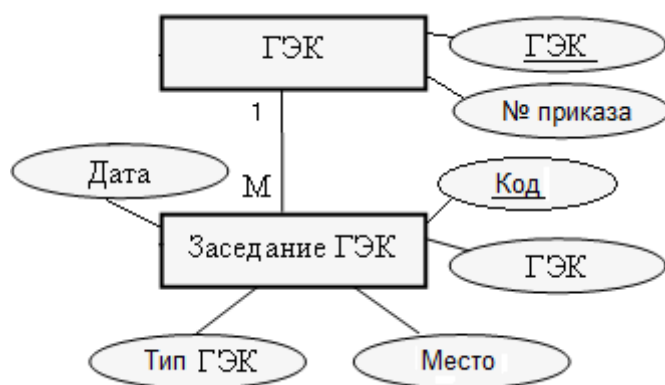


Рисунок 4 – Фрагмент ER-диаграммы

Таблица 3 – Описание таблиц и их атрибутов

Информационный объект (таблица)	Название реквизита	Обозначение реквизита	Признак ключа	Тип данных (размер)	Ограничения
ГЭК	№ ГЭК	ГЭК	Уникальный ключ	Счетчик	Уникальное
	№ приказа	Приказ		Текст	
Заседание_ГЭК	Код заседания	Код	Уникальный ключ	Счетчик	Уникальное
	№ ГЭК	ГЭК	Поле связи	Числовой (подстановка поля ГЭК из таблицы ГЭК)	Обязательное поле, по умолчанию – 1
	Дата и время заседания	Дата		Date	Формат «дата и время»
	Тип ГЭК	Вид		Текст (подстановка из значений «ГОС» или «ГЭК»)	3 символа, обязательное поле
	Место заседания	Место		Текст	30 символов

В соответствии с приведенным выше описанием необходимо представить каждую таблицу проектируемой базы данных.

Если база данных реализуется в MS Access, а затем преобразовывается в MS SQL Server, возможно определение процедурной поддержки целостности базы данных, т. е. разрабатываются триггеры.

### 3 ПОДХОДЫ К ПРОЕКТИРОВАНИЮ БАЗ ДАННЫХ

#### **1 способ: Проектирование на основе бумажных документов.**

Основой проектирования являются бланки и другие виды документации, с которыми работают конечные пользователи. По формальным правилам [11] из документов надо выделить информационные объекты (ИО), количество которых зависит от класса документов, представленных в таблице 4.

Таблица 4 – Описание классов документов

Класс	Описание
1. Список	Формируется один ИО из изменяемых реквизитов содержательной части документа.
2. Папка из списков	Формируется два ИО: 1) из изменяемых реквизитов заголовка папки; 2) из изменяемых реквизитов содержательной части документа.
3. Архив из папок	Формируется три ИО: 1) из изменяемых реквизитов заголовка папки; 2) из изменяемых реквизитов заголовочной части документа; 3) из изменяемых реквизитов содержательной части документа.

Вначале необходимо сравнить несколько отдельных экземпляров документа (и их папок, если они имеются) и проанализировать, какие атрибуты могут изменяться. Сразу следует исключить из дальнейшего рассмотрения постоянные и вычисляемые реквизиты. Затем для выделения ИО из документов необходимо разделить их реквизиты на группы в соответствии с ИО, которые они описывают.

Например, при разработке базы данных для учета преподавателей вуза за основу можно взять представленный на рисунке 5 документ с бланком информации о составе кафедры, откуда выбираются и группируются реквизиты, образующие два объекта предметной области – Кафедра и Преподаватель, поскольку в бланке можно выделить изменяемые части: заголовочную (информация о кафедре) и содержательную (список и информация о преподавателях кафедры).

Кафедра _____			
Зав.кафедрой _____			
тел. _____			
Список преподавателей:			
№	ФИО	должность	уч.степень
21	...	...	...
22	...	...	...
28	...	...	...

Рисунок 5 – Пример бланка, на основе которого проектируется база данных

Определяя предметную область, предполагается, что база данных проектируется для штатного состава, т. е. преподаватели работают на одной из кафедр вуза. Тогда инфологическая модель выглядит так, как представлено на рисунке 6 (на одной кафедре может работать один или много преподавателей или не работать ни одного, но каждый преподаватель может работать только на одной из кафедр – совместительство не предусмотрено).

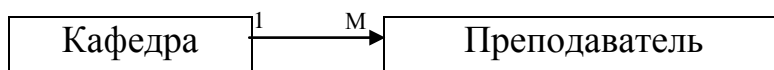


Рисунок 6 – ER-диаграмма проектируемой базы данных

Выписываем все реквизиты, даем им имена и определяем функциональные зависимости (таблица 5):

Таблица 5 – Имена и функциональные зависимости реквизитов

Документ	Наименование реквизита	Имя реквизита	Функциональная зависимость
Список преподавателей кафедр	<b>Код кафедры</b>	ККАФ	
	Название кафедры	НКАФ	
	Телефон	ТЕЛ	
	Фото зав.кафедрой	ФОТО	
	<b>Таб.номер</b>	ТАБН	
	Фамилия, имя, отчество	ФИО	
	Должность	ДОЛЖН	
	Ученая степень	СТ	
	<b>Код кафедры</b>	ККАФ	

Эти поля вводятся для идентификации объектов

Это поле вводится для связи объектов

Полученная информационно-логическая модель (ИЛМ) представлена на рисунке 7.

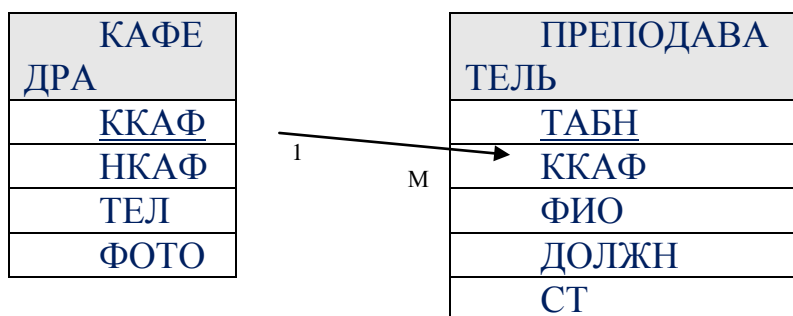


Рисунок 7 – ИЛМ базы данных «Кафедра»

**2 способ: Проектирование на основе имеющейся структуры или внешних данных.**

Иногда проектирование базы данных начинается с предлагаемых объектов будущей базы данных. Тогда необходимо определить границы предметной области, цели проектирования, самостоятельно продумать, какими атрибутами будут описываться объекты, выявить их взаимосвязи на основе взаимодействия экземпляров этих объектов. Для этого можно разработать бланк учетной информации и далее действовать по описанию 1-го способа.

Например, требуется разработать базу данных «Успеваемость» с таблицами Студент, Предмет, Экзамен. Исходя из названия, представляем предметную область следующим образом: в вузе необходимо вести учет сдачи экзаменов в сессию. Исходные данные могут быть выбраны из ведомости, которая выдается на экзамен. В ней фиксируют название предмета, фамилию преподавателя, семестр, группу и список студентов группы. Фраза «Студент сдает Экзамен по Предмету» позволяет выявить множественную зависимость между сущностями Студент и Предмет, которые свяжет связующая сущность Экзамен с описательным атрибутом – оценкой. На рисунке 8 представлена информационно-логическая модель (ИЛМ), которая требует дальнейшей нормализации, поскольку заметно излишнее дублирование данных в таблицах.



Рисунок 8 – ИЛМ базы данных «Успеваемость»

Если в основе разработки имеются данные, расположенные в других источниках, например, в электронных таблицах, то необходимо провести анализ наименований полей списка электронной таблицы, выявить объекты, построить их взаимосвязи, определиться с целями разработки базы данных. Ниже на рисунке 9 представлена электронная таблица с данными об оплате коммунальных услуг.

Из списка полей можно выявить следующие объекты и их атрибуты: Услуга (*Наименование, Тариф*), Платательщик (*Фамилия, Адрес*) и Оплата (*Код, Дата, Количество*: метраж – для квартплаты, объем – за газ и электроэнергию, минуты – за телефон и т. д.). Вычисляемое поле *Всего* в базу данных не заносится.

Дата	Наименование	Код	ФИО	Адрес	Кол-во	Тариф	Всего
2004-12-01	за газ	248	Петров А.Л.	ул. Ванеева, 15-26	25,6	2100	53760
2004-12-01	за электроэнергию	249	Петров А.Л.	ул. Ванеева, 15-26	220	110	24200
2004-12-01	за телефон	250	Петров А.Л.	ул. Ванеева, 15-26	27	1000	27000
2004-12-01	квартплата	251	Петров А.Л.	ул. Ванеева, 15-26	36,7	550	20185
2004-12-03	за электроэнергию	272	Сидоров Г.Д.	ул. Ванеева, 15-68	102	110	11220
2004-12-04	за газ	284	Авдеев Р.Р.	ул. Ванеева, 2-14	12	2100	25200
2004-12-05	за телефон	307	Уточкина Д.О.	ул. Солтыса, 62-14	39	1000	39000
2004-12-06	за электроэнергию	326	Уточкина Д.О.	ул. Солтыса, 62-14	156	110	17160
2004-12-10	квартплата	330	Сидоров Г.Д.	ул. Ванеева, 15-68	17,8	550	9790
2004-12-12	квартплата	352	Авакян Э.Ж.	ул. Ванеева, 19-1	54,9	550	30195
2004-12-15	за электроэнергию	378	Авакян Э.Ж.	ул. Ванеева, 19-1	145	110	15950
2004-12-15	за телефон	379	Авакян Э.Ж.	ул. Ванеева, 19-1	48	1000	48000
2004-12-15	за газ	380	Авакян Э.Ж.	ул. Ванеева, 19-1	27,9	2100	58590
2004-12-17	за телефон	394	Авдеев Р.Р.	ул. Ванеева, 2-14	13	1000	13000
2004-12-17	квартплата	395	Авдеев Р.Р.	ул. Ванеева, 2-14	26,6	550	14630
2004-12-19	за телефон	416	Сидоров Г.Д.	ул. Ванеева, 15-68	56	1000	56000
2004-12-20	квартплата	428	Уточкина Д.О.	ул. Солтыса, 62-14	47,8	550	26290
2004-12-20	за газ	429	Уточкина Д.О.	ул. Солтыса, 62-14	10	2100	21000

Рисунок 9 – Excel-таблица с данными «Коммунальные услуги»

Для проектирования базы данных информационно-логическая модель (ИЛМ) может выглядеть так, как показано на рисунке 10.





Рисунок 10 – ИЛМ базы данных «Коммунальные услуги»

Таблицы MS Access могут строиться на основе импорта данных, в частности из электронных таблиц. Используя эту возможность, вся таблица переносится в базу данных, затем, используя язык SQL, делаются проекции данных в новые таблицы, структура которых продумывается заранее. Затем эти таблицы в Конструкторе описываются ключевыми реквизитами и полями связи. Уточняются (корректируются) типы данных. Важно не потерять исходную информацию, которая была в электронной таблице.

При корректной работе в базе данных получатся таблицы-справочники и таблицы с учетной информацией, что разрешает аномалии, присущие ненормализованным отношениям импортированной исходной таблицы.

### **3 способ: Проектирование на основе описания предметной области – общей постановки.**

Например, требуется разработать базу данных для ведения учета денежных затрат на покупки членами семьи.

Процесс проектирования состоит в следующем:

1. Требуется определить, чем конкретно должна будет заниматься база данных. Для нашего примера это решение следующих задач:

- учет покупок;
- учет ежедневных денежных затрат на покупки;
- учет расходов каждого члена семьи и т. д.

Эти требования и пожелания к разрабатываемой БД записываются в столбик на большом листе бумаги.

2. Следует просмотреть все требования, удалить все повторяющиеся или явно взаимоисключающие условия и свести все в одну таблицу (таблица 6).

Таблица 6 – Выделение реквизитов

Требования	Описание и способ использования информации	Реквизиты	Принадлежность к объекту
Учет покупок	Ввод и вычисление – что куплено, в каком количестве, на какую сумму	Продукт, → Цена, → Количество, → Сумма денег →	ПРОДУКТ, ПОКУПКА
Расходы членов семьи	Ввод и вычисление – кто и когда совершил покупку, сколько денег потратил	ЧленСемьи, → Дата, → Сумма денег. →	ПЕРСОНА, ПОКУПКА
Учет денежных затрат	Вычисление – когда и какая сумма денег потрачены.	Дата, → Сумма денег →	ПОКУПКА

3. Необходимо конкретизировать требования к каждому объекту и выявить связи между ними, что показано на рисунке 11.

Анализируя исходные данные, можно выявить связи между объектами *Персона* и *Продукт* как «многие-ко-многим», что требуется разбить при помощи дополнительной сущности (глагола), который следует превратить в имя существительное.

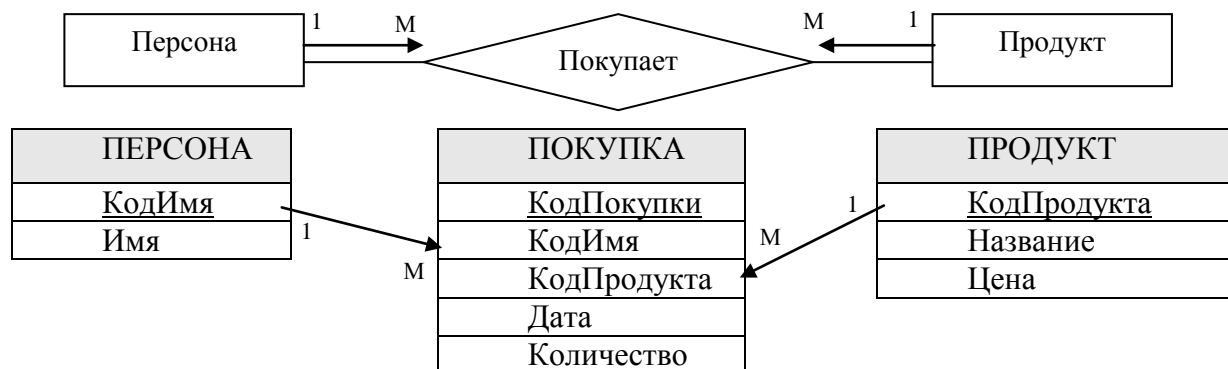


Рисунок 11 – ER-модель и ИЛМ базы данных «Затраты на покупки»

4. Далее проектируются таблицы, определяются данные, хранящиеся в них, прикидывается домен для данных, описывается каждое данное.

5. Затем следует провести нормализацию таблиц, в результате чего должны быть получены следующие итоги: все пожелания воплощены в конкретные типы и виды данных, все данные взаимосвязаны между собой.

Эти связи позволяют составить схему функционирования всей БД. В проектируемой базе данных можно делать выборки по временным интервалам (сум-

мы за покупки за день, неделю, месяц и т. д.), оценить на что уходит семейный бюджет.

#### 4 ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОГО РАЗДЕЛА КУРСОВОЙ РАБОТЫ

В процессе создания базы данных в MS Access (рисунок 12) вначале осуществляется конструирование таблиц, создаются схемы данных, которые осуществляют связи между таблицами. Внешний вид схемы совпадает с графическим представлением информационно-логической модели. В схеме данных могут быть заданы параметры обеспечения целостности БД, если модель разработана в соответствии с требованиями нормализации.

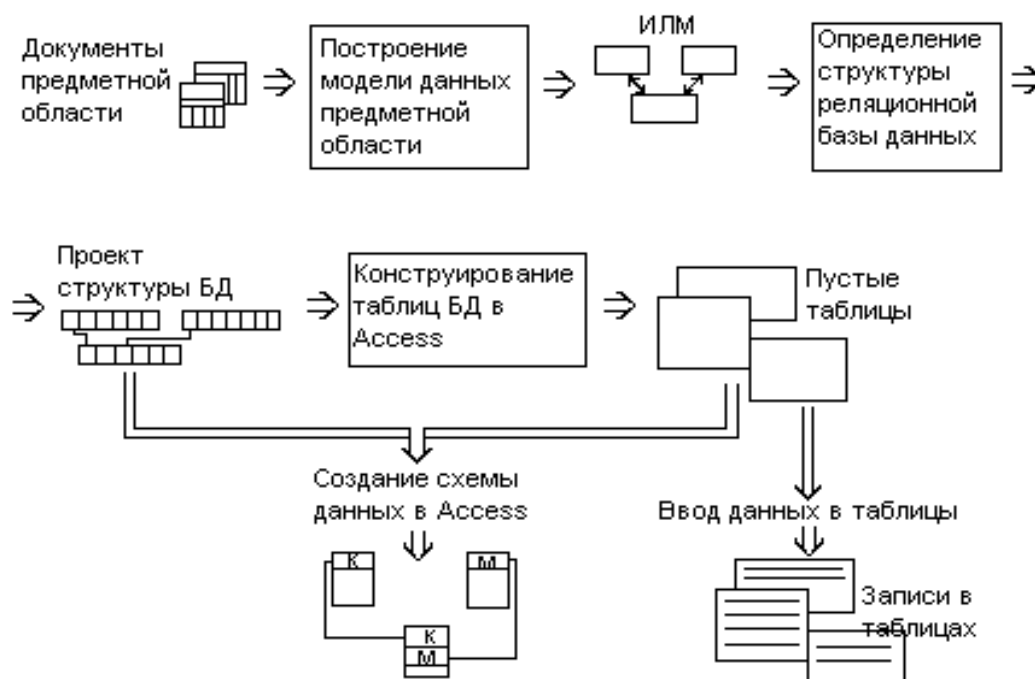


Рисунок 12 – Процесс проектирования структуры базы данных в MS Access

Для создания оптимальной структуры разрабатываемой базы данных и проектирования интерфейса по работе с ней следует применить следующие шаги:

1. Исследуйте моделируемую информационную среду:

- 1.1. Определите, откуда и в каком виде поступает информация.
- 1.2. Как часто информация меняется, кто и как работает с данными.
- 1.3. Какие бумажные носители, формы или файлы используются при обработке информации.

2. Создайте список объектов с их свойствами и атрибутами:

- 2.1. Сначала создайте список всех возможных атрибутов, а затем сгруппируйте их в объекты.
- 2.2. Проанализируйте объекты на соответствия их атрибутов (возможно, появятся дополнительные объекты или другие атрибуты).

2.3. Проведите процедуру нормализации: приведите разрабатываемую базу данных к 3-й нормальной форме.

2.4. Задокументируйте свои конструктивные решения в виде диаграммы «сущность–связь» (ER-диаграмма) или информационно-логической модели (ИЛМ).

2.5. Создайте макет таблиц и связей между ними.

3. Обоснуйте выбор СУБД и среды разработки клиентского приложения:

3.1. Разработайте физическую модель данных (объекты, поля и их типы, ограничения и значения по умолчанию, null-значения и прочие правила, обеспечивающие декларативную целостность данных).

3.2. Предусмотрите обеспечение целостности данных по ссылкам на уровне каскадных связей таблиц или в виде триггеров и обоснуйте свои решения.

3.3. Разработайте интерфейс клиентского приложения.

4. Создайте базу данных, поместите в нее прототипы данных:

4.1. Разработайте ряд запросов, задокументируйте их в виде SQL-скриптов и опишите полученные результаты.

5. Разработайте оконный интерфейс для демонстрации работы с базой данных:

5.1. Обеспечьте эффективность функционирования, простоту и удобство эксплуатации.

5.2. Разработайте управляющую кнопочную форму.

6. Разработайте отчетные документы для демонстрации работы с базой данных:

6.1. Обеспечьте эффективность представления информации, простоту и удобство просмотра.

6.2. Разработайте отчеты с группировками данных.

7. Проведите тестирование и опишите руководство пользователя по работе с базой данных.

Если база данных преобразована в среду MS SQL Server, разработайте хранимые процедуры, триггеры, курсоры, которые необходимы для обработки и извлечения данных, предусмотрите и обоснуйте способы обеспечения безопасности работы с базой данных на уровне групп пользователей.

## 5 ОФОРМЛЕНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Пояснительная записка должна отражать ход выполнения курсовой работы и содержать все структурные элементы отчетного документа.

Содержательная часть состоит из двух разделов, названия которых должны соответствовать теме курсовой работы и разрабатываемой базе данных. Минимальная реализация системы управления базами данных подразумевает создание базы данных и запросов, осуществляющих выполнение тех функций, которые оговорены в задании. В том случае, если СУБД реализуется не полностью, например, отсутствуют некоторые ограничения целостности или функциональные возможности, а также отчеты и формы, это должно быть указано в пояснительной записке.

Макет отчетного документа (расчетно-пояснительная записка) по курсовой работе представлен на рисунке 13 и должен иметь следующую структуру.

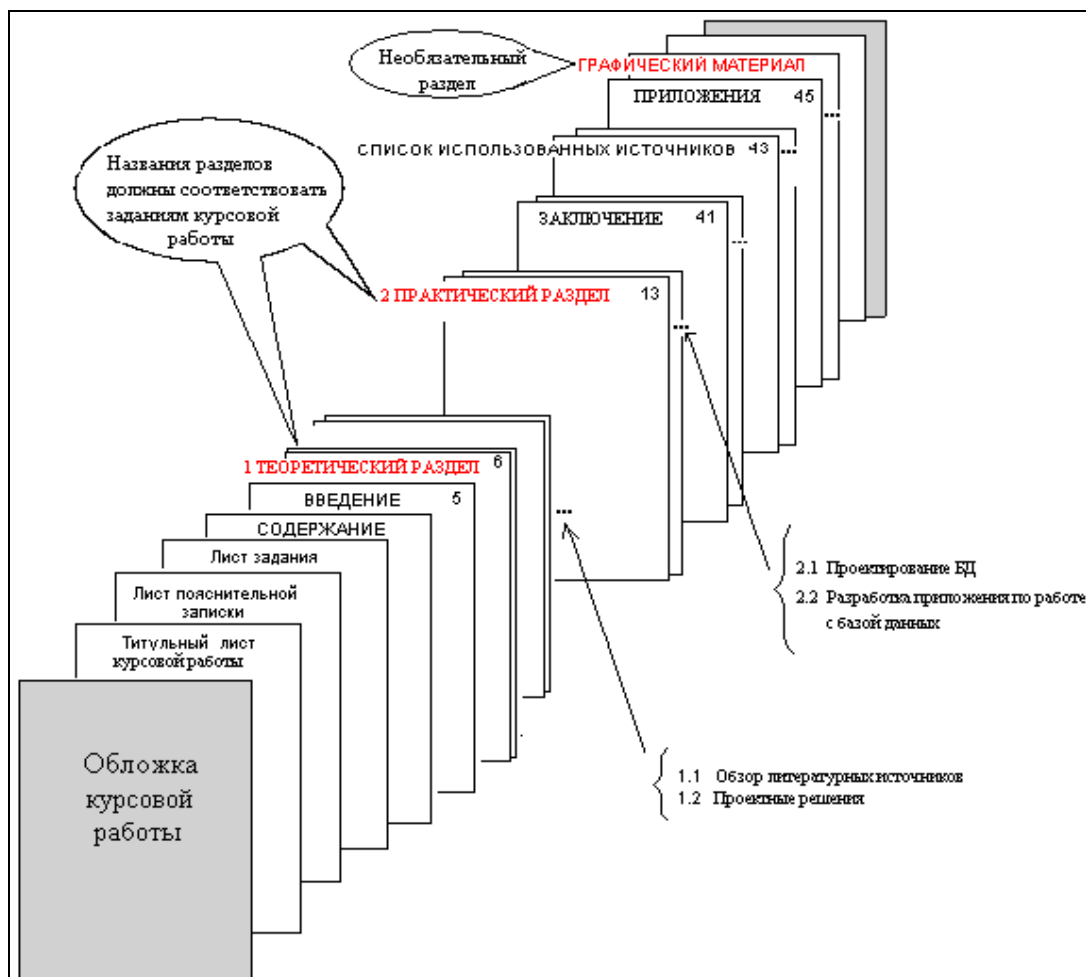


Рисунок 13 – Структура расчетно-пояснительной записки

**Титульный лист** является первой страницей работы и служит источником информации, подтверждающей допуск работы к защите. Образец титульного листа приведен в Приложении А\*.

**Лист пояснительной записки** является второй страницей работы и служит источником информации, необходимой для обработки и поиска документа. Образец листа пояснительной записки приведен в Приложении Б\*.

**Задание** на курсовую работу выдается студенту руководителем работы. Содержание работы отражается в спецификациях к исходным данным задания, является индивидуальным. Образец листа задания на курсовое проектирование приведен в Приложении В\*.

**Содержание** следует собирать по основным заголовкам курсовой работы: введению, номерам и заголовкам разделов, подразделов, заключению, списку использованных источников и приложений с указанием номера страницы, на которой помещен каждый заголовок. Сборку содержания следует выполнять средствами текстового редактора на основе применения заголовочных стилей 1-го и 2-го уровней или с помощью полей, управляющих сборкой оглавления.

**Введение** характеризует современное состояние проблемы, которой посвящена работа, должно содержать цель курсовой работы и задачи, на решение которых направлено данное исследование. Во введении дается обоснование актуальности темы с теоретической и практической точки зрения. Во введении следует четко формулировать, в чем конкретно заключается смысл описываемой работы, называется предметная область, на базе которой выполняется практическая часть работы.

**Основная часть** является наиболее важной и результативной частью работы. По структуре она зависит от темы курсовой работы и составляющих ее частей (теоретической и практической). Материал основной части отражает сущность, методику и основные результаты выполненной работы. Здесь же излагаются вопросы по применению программного обеспечения на этапах решения задач по теме практической разработки.

Таким образом, основная часть\*\* должна содержать следующие разделы и их наполнение:

1 Указать название теоретического раздела работы (подразумевает раскрытие тематики теоретической части курсовой работы с возможной увязкой ее с практической разработкой):

1.1 Обзор и анализ источников

1.2 Проектные решения

---

\* Шаблон электронного документа доступен в сети факультета и на сайте кафедры «Таможенное дело».

\*\* В зависимости от особенностей выполненной работы основную часть можно представить в виде текста, содержащего иллюстрации, схемы и таблицы.

2 *Проектирование БД* «указать название» (анализ предметной области, инфологическое и физическое проектирование разрабатываемой базы данных):

2.1 Анализ предметной области и составление спецификаций к данным

2.2 Построение концептуальной и логической модели данных

2.3 Физическая модель данных (описание таблиц с атрибутами и их свойствами, хранимых процедур, пользовательских функций, курсоров и триггеров)

3 *Разработка приложения по работе с базой данных* (обоснование выбора СУБД, проектирование запросов, форм, отчетов, управляющей формы приложения, элементов автоматизации):

3.1 Назначение объектов

3.2 Структура приложения

3.3 Руководство пользователя

**Заключение** является одной из важнейших частей курсовой работы, которое содержит оценку в виде выводов основных, наиболее важных полученных результатов. Заключение должно содержать: основные выводы по результатам выполнения работы или отдельных её этапов; оценку полноты решений поставленных задач; рекомендации и исходные данные по конкретному использованию результатов работы; оценку научно-технического уровня выполненной работы. В процессе работы могут выявиться новые (в известном смысле неожиданные) закономерности, новые данные. Все эти сведения также должны быть оценены в заключении. Помимо оценки результатов работы, заключение может содержать информацию о путях и целях дальнейшей работы или мотивированный вывод о нецелесообразности продолжения работы.

**Список использованных источников** должен содержать сведения об источниках, использованных при выполнении работы, в соответствии с требованиями ГОСТ 7.1. Источники следует располагать в порядке появления ссылок в тексте записки. В таблице 7 приведены примеры оформления описания изданий.

**Приложения** оформляют как продолжение работы в виде отдельной части, располагая их в порядке появления ссылок в тексте. Приложения должны иметь названия. В приложениях можно размещать таблицы с альбомной ориентацией.

В приложения могут быть включены SQL-скрипты для создания базы данных и ее таблиц, разработанных SQL-запросов (представлений), функций, хранимых процедур, курсоров, триггеров, основные фрагменты текстов программ клиентской части приложения и другая дополнительная информация (на усмотрение студента).

Таблица 7 – Примеры библиографического описания изданий

Характеристика источника	Пример оформления
Один, два или три автора	Сенов, А.С. Access 2007 / А.С. Сенов. – СПб и др.: Питер, 2008. – 266 с. Астахова, И.Ф. SQL в примерах и задачах: учеб. пособие / И.Ф. Астахова, А.П. Толстобров, В.М. Мельников. – Минск: Новое знание, 2002. – 176 с.
Более трех авторов	Культурология: учеб. пособие для вузов / С.В. Лапина [и др.]; под общ. ред. С.В. Лапиной. – 2-е изд. – Минск: Системс, 2004. – 470 с.
Учебник, учебное пособие, словарь, справочник	Эксплуатация и техническое обслуживание дорожных машин, автомобилей и тракторов: учеб. / С.Ф. Головин, В.М. Коншин, А.В. Рубайлов и др.; под ред. Е.С. Локшина. – М.: Мастерство, 2002. – 462 с.: ил. Разоренова, Т.Р. Управление базами данных: учебно-методич. пособие для студентов специальностей: 1–96 01 01 «Таможенное дело», 1–26 02 02 «Менеджмент», 1–25 01 08 «Бухгалтерский учет, анализ и аудит», 1–25 01 07 «Экономика и управление на предприятии», «Финансовое обеспечение и экономика боевой и хозяйственной деятельности войск (сил)» / Т.Р. Разоренова, О.В. Альшевская. – 2-е изд., испр. и доп. – Минск: Изд-во БНТУ, 2011. – 120 с. Иллюстрированный словарь по искусству и архитектуре / сост. Р.П. Андреева. – СПб.: Издат. Дом «Литера», 2003. – 447 с.: ил.
Методические указания	Методические указания к выполнению курсовой работы по дисциплине «Технология и оборудование восстановления деталей машин и приборов» для студентов специальности 1–36 01 04 «Оборудование и технологии высокоэффективных процессов обработки материалов» / сост. Е.Н. Сташевская. – Минск.: БНТУ, 2003. – 20 с.
Многотомное издание	История Беларуси: в 6 т. / редкол.: М.А. Костюкевич (гл. ред.) [и др.]. – Минск: Экоспектива, 2000–2005. – 6 т.
Сборник статей, трудов	Совершенствование методов гидравлических расчетов водопропускных и очистных сооружений: межвуз. науч. сб. / Саратов. гос. технич. ун-т; отв. ред. Л.И. Высоцкий. – Саратов: СГТУ, 2002. – 98 с.: ил.
Тезисы докладов и материалы конференций	Современные методы проектирования машин. Расчет, конструирование и технология изготовления: сб. трудов 1-й Междунар. конференции, Минск, 11–13 декабря 2002 г.: в 3 т. / под общ. ред. П.А. Витязя. – Минск: Технопринт, 2002.
Статья из газеты	Белый, С. Электроэнергетика Беларуси: настоящее и будущее /С. Белый // Рэспубліка. – 2003. – 20 снежня. – С.12.
Статья из журнала	Кравец, Ф. К. Динамика системы подготовки сжатого воздуха пневмопривода технологических машин /Ф.К. Кравец // Вестник Белорусского национального технического университета. – 2003. – № 4. – С. 44–49.
Стандарт, законодательные материалы	Единая система стандартизации БНТУ. Курсовое проектирование / В.М. Копко [и др.]. – Утв. приказом ректора БНТУ от 18.01.2003 г. – Минск.: БНТУ, 2003. – 15 с. Конституция Республики Беларусь 1994 года (с изменениями и дополнениями): принята на респ. референдуме 24 нояб. 1996 г. в ред. решения респ. референдума 17 нояб. 2004 г. – Минск: Акад. МВД Респ. Беларусь, 2005. – 39 с. ГОСТ 8.420–2002. Государственная система обеспечения единства измерений. Государственная поверочная схема для средств измерений отклонений от прямолинейности и плоскостности. – Взамен ГОСТ 8.420–81; вед. 01.09.03. – Минск: БелГИСС: Межгос. совет по стандартизации, метрологии и сертификации, 2003. – 6 с.



## Продолжение таблицы 7

Электронные ресурсы локального доступа	<p>Российская академия наук. Отделение геологии, геофизики, геохимии и горных наук. Вестник ОГГГН РАН [Электронный ресурс] / Объед. ин-т физики Земли им. О.Ю. Шмидта Рос. Акад. наук. – Электрон. журн. – М.: ОГГГН РАН, 1997. – 4 дискеты. – Систем. требования: от 386; Windows; Internet-браузер кл. Netscape Navigator 3.0 и выше. – Загл. с экрана. – 4 раза в год.</p> <p>Internet шаг за шагом [Электронный ресурс]: [интерактив. учеб.]. – Электрон. дан. и прогр. – СПб.: ПитерКом, 1997. – 1 электрон. опт. диск (CD-ROM) + прил. (127 с.). – Систем. требования: ПК от 486 DX 66 МГц; RAM 16 Мб; Windows 95; зв. плата; динамики или наушники. – Загл. с экрана.</p> <p>Oxford interactive encyclopedia [Электронный ресурс]. – Электрон. дан. и прогр. – [Б. м.]: The Learning Company, 1997. – 1 электрон. опт. диск (CD-ROM): зв., цв.; 12 см. – Систем. требования: ПК с процессором 486+; Windows 95 или Windows 3.1; дисковод CD-ROM; зв. карта. – Загл. с этикетки диска.</p>
Электронные ресурсы удаленного доступа	<p>Российская государственная библиотека [Электронный ресурс] / Центр информ. технологий РГБ; ред. Т.В. Власенко; Web-мастер Н.В. Козлова. – Электрон. дан. – М.: Рос. гос. б-ка, 1997. – Режим доступа: <a href="http://www.rsl.ru">http://www.rsl.ru</a>, свободный. – Загл. с экрана. – Яз. рус., англ.</p> <p>Проектирование баз данных (25 курсов) [Электронный ресурс] / Интернет-Ун-т Информ. Технологий – дистанцион. образование. – Россия, 2003–2011. – Режим доступа: <a href="http://www.INTUIT.ru">http://www.INTUIT.ru</a>, свободный. – Загл. с экрана.</p>

**Графический материал**<sup>\*</sup> представляет собой распечатанные на листах формата А4 (по одному на лист) плакаты, иллюстрирующие ход курсового проектирования. Объем графического материала не должен превышать пяти листов, быть «читабельным», не перегруженным текстовой информацией. Графический материал отражает проектные решения в виде DFD- и ER-диаграмм, информационно-логические модели, схему приложения, последовательность ввода данных и т. д. На каждом листе размещается рамка и штамп, выполненный по правилам инженерной графики, с подписями студента и руководителя (см. приложение Г). Графический материал нумеруется отдельно.

### 5.1 Правила оформления отчетного документа

1. Общий объем курсовой работы должен составлять не менее 20 и не более 50 листов формата А4, включая приложения.

2. Шрифт – Times New Roman, 12 пт, через полтора интервала. Параметры страницы: левое поле – 30 мм, правое поле – 10 мм, верхнее и нижнее поля – 20 мм. Абзацы – 15–17 мм, одинаковые по всему тексту. Выравнивание текста – по ширине.

3. Страницы пояснительной записки следует нумеровать арабскими цифрами без точки в правом верхнем углу листа, соблюдая сквозную нумерацию по всему тексту записки (кроме листов графической части).

4. Номера страниц на титульных листах, на задании по курсовому проектированию и первой странице содержания не ставятся, но включаются в общую нумерацию страниц.

<sup>\*</sup> В курсовой работе может отсутствовать.

5. Наименования разделов основной части, «СОДЕРЖАНИЕ», «ВВЕДЕНИЕ», «ЗАКЛЮЧЕНИЕ», «СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ» следует располагать по центру строки без точки в конце и печатать прописными буквами, не подчеркивая.

6. Наименования, включенные в содержание, начинаются от левого края без абзацного отступа, записываются таким же шрифтом, как в тексте записки. Пример оформления листа содержания представлен на рисунке Д.1 в Приложении Д.

7. Каждый раздел следует начинать с нового листа. Разделы основной части нумеруются арабскими цифрами (например, «1 НОВЫЕ ВОЗМОЖНОСТИ SQL SERVER 2008» или «2 ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ «АБИТУРИЕНТЫ») и начинаются с абзацного отступа.

8. Заголовки подразделов, пунктов и подпунктов следует начинать с абзацного отступа и печатать с прописной буквы, не подчеркивая, без точки в конце. Они должны иметь порядковую нумерацию в пределах раздела, подраздела или пункта (например, «1.1 Подходы к проектированию баз данных» или «2.1 Анализ предметной области»).

9. Если заголовок включает несколько предложений, их разделяют точками. Переносы слов в заголовках не допускаются.

10. Расстояние между текстом и следующим за ним заголовком должно составлять 2 свободные строки. Расстояние между заголовками раздела и подраздела – 1 свободная строка. Расстояние между заголовком и следующим за ним текстом – 1 свободная строка. Пример оформления представлен на рисунке Д.2 в Приложении Д.

11. Иллюстрации (схемы, диаграммы, фрагменты программных кодов, копии экранов программы) следует располагать в записке непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. На все иллюстрации должны быть даны ссылки в записке.

12. Иллюстрации следует располагать по центру листа и нумеровать арабскими цифрами нумерацией в пределах раздела, например, «Рисунок 2.1 – ER-диаграмма» (тоже по центру, указывается под рисунком без точки). Расстояние между текстом и иллюстрацией, иллюстрацией и подписью, подписью и текстом – 1 свободная строка). При ссылках на иллюстрации следует писать «...приводится на рисунке 2.1». Пример оформления иллюстраций представлен на рисунке Д.3 в Приложении Д.

13. Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например, «Рисунок А.3».

14. Перечисления следует печатать строчными буквами с абзацного отступа, разделяя их точкой с запятой, оформив их в виде нумерованного списка, либо маркированного знаком дефиса, или многоуровневого, как показано ниже:

- |          |              |     |         |              |
|----------|--------------|-----|---------|--------------|
| 1. ... ; | или 1) ... ; | или | – ... ; | или 1. ... : |
| 2. ....  | 2) ....      |     | – ....  | 1.1. ... ;   |
|          |              |     |         | 1.2. ... .   |

15. Каждый пункт перечисления оформляется отдельным абзацем. Примером оформления списка является представленный на этой странице текст.

16. Формулы и вычисления следует выделять из текста в отдельную строку по центру. Выше и ниже их должно быть оставлено по одной свободной строке.

17. Каждая формула должна заканчиваться знаком препинания: запятой в случае наличия пояснений к ней, точкой с запятой в случае перечисления формул либо точкой. Вычисления заканчиваются точкой.

18. Формулы в пояснительной записке следует нумеровать в пределах раздела арабскими цифрами в круглых скобках в крайнем правом положении на строке.

19. Пояснение значений формул следует приводить под формулой в той же последовательности, в которой они даны в формуле. Каждое новое значение следует давать с новой строки и заканчивать точкой с запятой (кроме последнего значения). Первую строку начинают со слова «где» без двоеточия с абзаца, остальные строки – обычное выравнивание. Пример оформления формул представлен на рисунке Д.4 в Приложении Д.

20. Таблицы следует располагать в записке по центру страницы непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. На все таблицы должны быть ссылки в записке. При ссылках следует писать «... по таблице 2.1».

21. Таблицы следует нумеровать арабскими цифрами нумерацией в пределах раздела. Таблицы должны иметь подпись, которая указывается над таблицей и оформляется с абзаца: «Таблица 2.1 – Объекты проектирования». После названия таблицы точка не ставится. Расстояние между текстом и подписью, подписью и таблицей, таблицей и текстом – 1 свободная строка.

22. При переносе части таблицы на следующую страницу подпись, которая указывается над таблицей без абзацного отступа, имеет следующий вид: «Продолжение таблицы 2.1». Далее – шапка таблицы и ее продолжение. Пример оформления таблицы с переносом на следующую страницу представлен на рисунке Д.5 в Приложении Д.

23. Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например, «Таблица А.1 – Перечень атрибутов».

24. Список использованных источников может включать перечень стандартов, монографий, учебников, методических пособий, статей, тезисов докладов и материалов конференций, электронных ресурсов локального и удаленного доступов. Каждый пункт списка выравнивается по ширине окна без абзацного отступа.

25. На все использованные источники должны быть ссылки по тексту в следующем виде: <цитируемый текст> [х], где х – № источника в списке используемых источников.

26. Каждое приложение должно начинаться с новой страницы и иметь заголовки, напечатанный с прописной буквы симметрично тексту. Над заголовком в правом верхнем углу прописными буквами должно быть напечатано слово «ПРИЛОЖЕНИЕ». Если приложений несколько, их обозначают заглавными буквами русского (белорусского) алфавита, начиная с А, за исключением букв Е, З, Й, О, Ч, Ь, Ы, Ъ, или латинского алфавита за исключением букв I и O. Название приложения записывается строчными буквами, начиная с прописной.

## 6 ЗАЩИТА КУРСОВОЙ РАБОТЫ

К защите допускаются курсовые работы с визой руководителя на допуск к защите, которая ставится на титульном листе работы в срок, не позднее указанной на листе задания даты окончания выполненной работы. К защите допускаются только курсовые работы, оформленные в строгом соответствии с изложенными выше требованиями. За содержание и оформление курсовой работы, принятые в них решения, правильность всех данных и сделанные выводы отвечает студент – автор выполненной работы.

Дата защиты определяется на заседании кафедры и проводится в соответствии с расписанием, которое доводится до сведения студентов.

Курсовая работа защищается на рабочей комиссии (создаваемой на кафедре), на которую является студент с отчетным документом. На доклад по курсовой работе отводится до 10 минут. Доклад должен сопровождаться иллюстративным материалом (демонстрация разработки на компьютере). Для студентов специальности 1–96 01 01 «Таможенное дело» защита сопровождается разработанной презентацией, где в виде тезисов излагается теоретический вопрос, процесс и результаты разработки базы данных. Презентация должна быть выдержана в строгом стиле, не быть переполненной текстовой информацией, содержать формулировку цели, структурированную и графическую информацию о проделанной работе в ходе курсового проектирования и выводы.

После доклада студенту необходимо ответить на вопросы членов рабочей комиссии, выслушать рекомендации и замечания, высказанные членами рабочей комиссии и руководителем работы.

Работа оценивается в 10-балльной системе с учетом качественных характеристик творческой деятельности. При оценке учитываются:

- актуальность, значимость и практическая направленность разработки;
- самостоятельность выполнения работы, уровень творчества, оригинальность решения;
- аргументированность предложенных решений, наличие выводов;
- объем, полнота и завершенность разработки;
- использование технических и программных средств;
- правильность оформления пояснительной записки;
- активность в процессе консультаций и обсуждений (умение задавать вопросы и отвечать на них);
- балл промежуточного контроля.

## ЛИТЕРАТУРА

1. Дейт, К.Дж. Введение в системы баз данных / К.Дж. Дейт; пер. с англ. – 7-е изд. – М.: Издат. дом «Вильямс», 2001. – 1072 с.
2. Малыхина, М.П. Базы данных: основы, проектирование, использование / М.П. Малыхина. – СПб.: БХВ-Петербург, 2004. – 512 с.
3. Разоренова, Т.Р. Управление базами данных: учебно-методич. пособие для студентов специальностей: 1–96 01 01 «Таможенное дело», 1–26 02 02 «Менеджмент», 1–25 01 08 «Бухгалтерский учет, анализ и аудит», 1–25 01 07 «Экономика и управление на предприятии», «Финансовое обеспечение и экономика боевой и хозяйственной деятельности войск (сил)» / Т.Р. Разоренова, О.В. Альшевская. – 2-е изд., испр. и доп. – Минск: БНТУ, 2011. – 120 с.
4. Харрингтон, Джен Л. Проектирование реляционных баз данных. Просто и доступно / Джен Л. Харрингтон; пер. с англ. – М.: Лори, 2000. – 230 с.: ил.
5. Проектирование баз данных (25 курсов) [Электронный ресурс] / Интернет-Ун-т Информ. Технологий – дистанцион. образование. – Россия, 2003 – 2011. – Режим доступа: <http://www.INTUIT.ru>, свободный. – Загл. с экрана.
6. Сенов, А.С. Access 2007 / А.С. Сенов. – СПб. и др.: Питер, 2008. – 266 с.
7. Проектирование и реализация баз данных Microsoft SQL Server 2000 / Официальное пособие Microsoft для самостоятельной подготовки; пер. с англ. – М.: Русская редакция, 2001. – 652 с.
8. Хернандес, Майкл Дж. SQL запросы для простых смертных: практическое руководство по манипулированию данными в SQL / Майкл Дж. Хернандес, Джон Л. Вьескас. – М.: Лори, 2003. – 473 с.
9. Единая система стандартизации БНТУ. Курсовое проектирование / В.М. Копко [и др.]. – Утв. приказом ректора БНТУ от 18.01.2003 г. – Минск: БНТУ, 2003. – 15 с.
10. Астахова, И.Ф. SQL в примерах и задачах: учеб. пособие / И.Ф. Астахова, А.П. Толстобров, В.М. Мельников. – Минск: Новое знание, 2002. – 176 с.
11. Левчук, Е.А. Технологии организации, хранения и обработки данных: учеб. пособие / Е.А. Левчук. – 2-е изд. – Минск: Выш. шк., 2005. – 239 с.: ил.

# ПРИЛОЖЕНИЯ

## ПРИЛОЖЕНИЕ А

Титульный лист курсовой работы

**Белорусский национальный технический университет**  
**Факультет технологий управления и гуманитаризации**  
**Кафедра "Таможенное дело"**

### **КУРСОВАЯ РАБОТА**

по дисциплине

**"Компьютерные информационные технологии"**

Тема: \_\_\_\_\_

---

---

---

---

Исполнитель: студент \_\_ курса ФТУГ, группа \_\_\_\_\_  
\_\_\_\_\_  
(фамилия, имя, отчество)

Руководитель: \_\_\_\_\_  
(ученое звание, ученая степень, должность)  
\_\_\_\_\_  
(фамилия, имя, отчество)

**Минск 2011**





Лист задания

**БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Факультет ФТУГ  
 «Утверждено»  
 Заведующий кафедрой \_\_\_\_\_  
 (подпись)  
 « \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

**ЗАДАНИЕ**  
**по курсовому проектированию**

Студенту \_\_\_\_\_

1. Тема проекта \_\_\_\_\_

2. Сроки сдачи студентом законченного проекта \_\_\_\_\_

3. Исходные данные к проекту

4. Содержание расчетно-пояснительной записки (перечень вопросов, которые подлежат рассмотрению)

Титульный лист

Задание

Содержание

Введение

1 Теоретический раздел

1.1 Обзор и анализ источников

## Обратная сторона листа задания

- 1.2 Проектные решения
- 2 Проектирование БД
  - 2.1 Анализ предметной области и составление спецификаций к данным
  - 2.2 Построение концептуальной и логической модели данных
  - 2.3 Физическая модель данных
- 3 Разработка приложения по работе с базой данных
  - 3.1 Кнопочная форма
  - 3.2 Руководство пользователя

Выводы

Литература

Приложения

5. Перечисление графического материала (с подробным указанием обязательных чертежей и графиков)

- ER-диаграмма

6. Консультант по проекту (с указанием разделов проекта)

7. Дата выдачи задания \_\_\_\_\_

8. Календарный график работы над проектом на весь период проектирования

(с указанием сроков выполнения и трудоемкости отдельных этапов)

Анализ предметной области и составление спецификаций (до _____	10%
Проектирование и разработка БД (до _____	30%
Разработка приложения (до _____	30%
Отладка и тестирование приложения (до _____	10%
Оформление расчетно-пояснительной записки (до _____	10%

Руководитель \_\_\_\_\_

(подпись)

Задание принял к исполнению \_\_\_\_\_

(дата и подпись студента)

Рамка для оформления графического материала

<div style="text-align: right; margin-bottom: 5px;">                 номер группы, зачетки    год                  КР - 108410/18-ДО-2011             </div>										
Изм.	Лист	№ докум.	Подп.	Дата	Введите название темы курсовой работы			Лист	Лист	Листов
Разработ.	Введите слово фамилию									
Руководит.	Введите фам-ю				Введите название темы курсовой работы			1-25 01 07 БНТ У,г. МИНСК		
Зав. каф.	Введите фам-ю									

Примеры оформления элементов пояснительной записки

СОДЕРЖАНИЕ	
ВВЕДЕНИЕ .....	4
1 ЦЕЛИ КУРСОВОГО ПРОЕКТИРОВАНИЯ.....	6
2 КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И ТЕРМИНОЛОГИЯ.....	7
2.1 Понятие предметной области.....	8
2.2 Этапы проектирования базы данных .....	8
2.3 Основные понятия ER-диаграмм .....	10
3 ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ .....	14
3.1 Документирование ER-модели .....	18
3.2 Нормализация модели .....	21
3.3 Структуры таблиц.....	26
4 ПРАКТИЧЕСКИЙ РАЗДЕЛ КУРСОВОЙ РАБОТЫ.....	31
4.1 Описание предметной области .....	36
4.2 ER-модель и информационно-логическая модель .....	39
4.3 Таблицы, запросы и вычисления.....	43
4.4 Интерфейс приложения .....	46
4.5 Тестирование приложения.....	48
ЛИТЕРАТУРА .....	49

Рисунок Д.1 – Пример оформления содержания пояснительной записки

## 1 МЕТОДОЛОГИЯ И ИСТОЧНИКИ ИССЛЕДОВАНИЯ

¶

¶

### 1.1 Описание объекта и методологии исследования

¶

Специфика тематики курсовой работы, вызванная объектом исследования (общественные отношения, складывающиеся между государствами и Всемирной таможенной организацией в процессе осуществления между ними сотрудничества в области регулирования международной торговли) и предметом исследования (деятельность Всемирной таможенной организации по сближению национальных правовых систем в области таможенного дела) вызвала необходимость использования при написании работы различных методов научного познания.

¶

¶

### 1.2 Обзор литературы

¶

Основным зарубежным не русскоязычным источником использованным для написания курсовой работы является официальный сайт ВТаО [1]. Информация на сайте публикуется на английском и французском языках. На сайте можно найти практически любую необходимую информацию о ВТаО и ее деятельности, в частности:

Рисунок Д.2 – Пример расстановки разделительных строк между заголовками и текстом в пояснительной записки

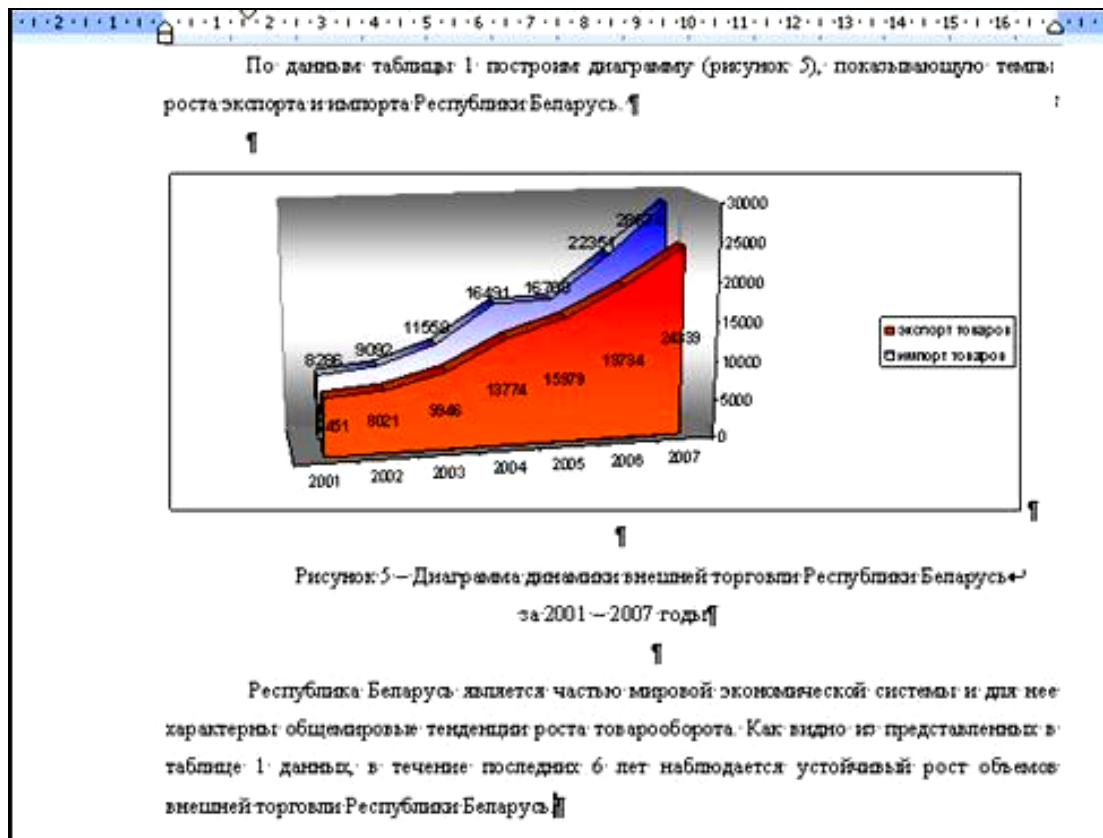


Рисунок Д.3 – Пример размещения рисунка с подписью в тексте пояснительной записки

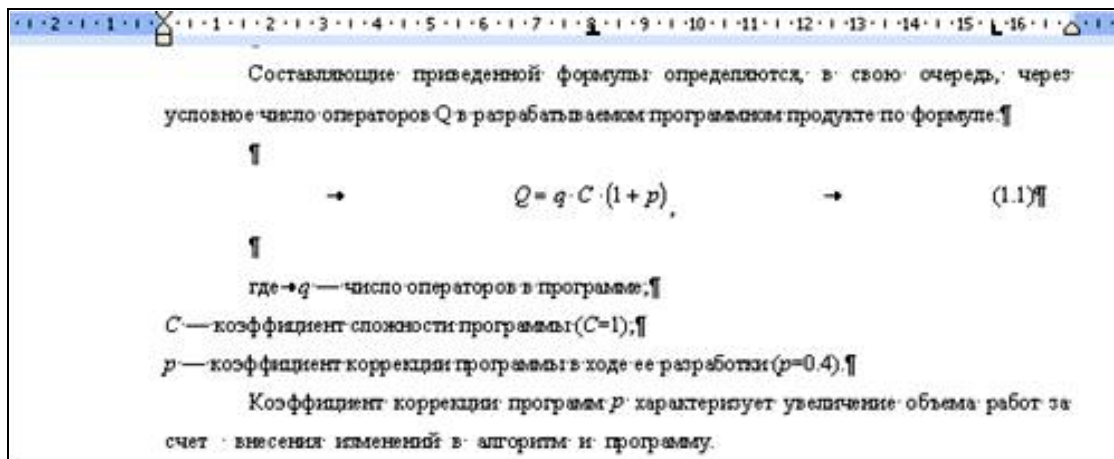


Рисунок Д.4 – Пример размещения формулы в тексте пояснительной записки

Таблица Д.1 – Показатели внешней торговли за 2001 – 2007 год (млн. долларов США)

	<b>2001</b>	<b>2002</b>	<b>2003</b>	<b>2004</b>	<b>2005</b>	<b>2006</b>	<b>2007</b>
со странами СНГ	10291	10679	13484	19201	18202	23121	30228
экспорт товаров	4494	4384	5435	7318	7060	8609	11231
импорт товаров	5797	6295	8049	11883	11142	14512	18997
сальдо	-1303	-1911	-2614	-4565	-4082	-5903	-7766
из них с Россией	9401	9899	12482	17704	15834	19944	26074
экспорт	3963	3977	4880	6485	5716	6845	8887
импорт	5438	5922	7602	11219	10118	13099	17187
сальдо	-1475	-1945	-2722	-4734	-4402	-6254	-8300

Продолжение таблицы Д.1

	<b>2001</b>	<b>2002</b>	<b>2003</b>	<b>2004</b>	<b>2005</b>	<b>2006</b>	<b>2007</b>
со странами вне СНГ	5446	6434	8020	11064	14485	18964	22785
экспорт товаров	2957	3637	4511	6456	8919	11125	13108
импорт товаров	2489	2797	3509	4608	5566	7839	9677

Рисунок Д.5 – Пример оформления таблиц при разрыве страниц  
в тексте пояснительной записки

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
1 ЦЕЛИ КУРСОВОГО ПРОЕКТИРОВАНИЯ.....	4
2 КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И ТЕРМИНОЛОГИЯ.....	6
2.1 Понятие предметной области .....	7
2.2 Этапы проектирования базы данных.....	7
2.3 Основные понятия ER-диаграмм.....	8
2.4 Способы документирования ER-модели.....	9
2.5 Нормализация модели «сущность–связь».....	12
2.6 Основные объекты реляционных БД.....	14
2.7 Структура таблиц и целостность данных.....	15
3 ПОДХОДЫ К ПРОЕКТИРОВАНИЮ БАЗ ДАННЫХ.....	20
4 ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОГО РАЗДЕЛА КУРСОВОЙ РАБОТЫ.....	26
5 ОФОРМЛЕНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ.....	28
5.1 Правила оформления отчетного документа.....	32
6 ЗАЩИТА КУРСОВОЙ РАБОТЫ.....	36
ЛИТЕРАТУРА.....	37
ПРИЛОЖЕНИЯ.....	38



Учебное издание

РАЗОРЁНОВА Татьяна Ромуальдовна

БАЗЫ ДАННЫХ: РАЗРАБОТКА И УПРАВЛЕНИЕ

Методическое пособие  
для выполнения курсовой работы  
по дисциплинам  
«Информационные таможенные технологии»  
для студентов специальности 1–96 01 01 «Таможенное дело»  
и «Компьютерные информационные технологии»  
для студентов специальности 1–25 01 07  
«Экономика и управление на предприятии»

Редактор В.О. Кутас  
Компьютерная верстка А.Г. Занкевич

---

Подписано в печать 01.09.2011.  
Формат 60×84<sup>1</sup>/<sub>8</sub>. Бумага офсетная.  
Отпечатано на ризографе. Гарнитура Таймс.  
Усл. печ. л. 5,58. Уч.-изд. л. 2,18. Тираж 100. Заказ 346.

---

Издатель и полиграфическое исполнение:  
Белорусский национальный технический университет.  
ЛИ № 02330/0494349 от 16.03.2009.  
Проспект Независимости, 65. 220013, Минск.